Improving OOD Detection, Recognition, and Understanding via Multi-Modal Feature Alignment

by

Yimu Wang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2026

© Yimu Wang 2026

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Nicholas Rhinehart
Assistant Professor
The Institute for Aerospace Studies,
University of Toronto

Supervisor(s): Krzysztof Czarnecki
Professor
Department of Electrical and Computer Engineering,
University of Waterloo

Internal Member: Olga Veksler
Professor
David R. Cheriton School of Computer Science,
University of Waterloo

Freda Shi
Assistant Professor
David R. Cheriton School of Computer Science,
University of Waterloo

Internal-External Member: Sirisha Rambhatla
Assistant Professor
Department of Management Science and Engineering,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

The following publications are included in parts in this thesis:

[1] **Yimu Wang**, Evelien Riddell, Adrian Chow, Sean Sedwards, and Krzysztof Czarnecki. 2026. Mitigating the Modality Gap: Few-Shot Out-of-Distribution Detection with Multi-modal Prototypes and Image Bias Estimation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pages 2741-2751, 2026.

[2] **Yimu Wang**, and Krzysztof Czarnecki. 2025. AiDe: Improving 3D Open-Vocabulary Semantic Segmentation by Aligned Vision-Language Learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pages 2674-2685, 2025.

[3] **Yimu Wang**, Mozhgan Nasr Azadani, Sean Sedwards, and Krzysztof Czarnecki. 2025. LEO-MINI: An Efficient Multimodal Large Language Model using Conditional Token Reduction and Mixture of Multi-Modal Experts. In Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 7246–7261, 2025.

All three studies were conducted under the supervision of Prof. Krzysztof Czarnecki. Prof. Krzysztof Czarnecki provided overarching research guidance, supervised the projects, offered extensive feedback during weekly meetings, and thoroughly reviewed and edited all the manuscripts.

Regarding the individual contributions of other co-authors: In [1], Adrian Chow provided valuable discussions and feedback on the experimental design and implementation. Evelien Riddell and Sean Sedwards contributed to the discussions and assisted with the writing of the manuscript. In [3], Mozhgan Nasr Azadani and Sean Sedwards contributed to the discussions and assisted with each stage of the research, including the early design, implementation, writing, and analysis.

Abstract

The deployment of artificial intelligence (AI) systems in real-world scenarios, such as autonomous vehicles encountering novel road conditions and medical AI analyzing rare pathologies, requires robust handling of out-of-distribution (OOD) data—inputs that differ from the training distribution due to semantic shift (*e.g.*, novel object categories) or covariate shift (*e.g.*, changes in lighting or sensor noise). Achieving this robustness requires that AI systems progress from detecting OOD data, to recognizing novel categories within OOD data, and ultimately understanding OOD scenarios to answer user queries—an essential capability for safety-critical applications where understanding unfamiliar situations is required. This motivates the use of increasingly advanced vision-language capabilities, yet current models face technical barriers in multi-modal feature alignment that limit practical deployment in OOD detection, recognition, and understanding. The challenges and applications of multi-modal feature alignment for these tasks have not been fully explored. This thesis makes three key contributions to advance the understanding and application of multi-modal feature alignment in vision-language models (VLMs).

First, we address the limitations of VLMs in OOD detection. We observe that the modality gap between image and text features causes high false positive rates, as OOD samples can exhibit high similarity to in-distribution (ID) text prototypes. To overcome this limitation, we propose a novel few-shot OOD detection method that incorporates ID image prototypes alongside ID text prototypes. Our method introduces the Bias Prompt Generation module to enhance image-text fusion and the Image-Text Contrastive module to reduce the modality gap. This multi-modal prototype approach significantly improves OOD detection accuracy across multiple benchmarks.

Next, we tackle OOD recognition through 3D open-vocabulary semantic segmentation, which leverages VLMs to generate point-wise recognition results for novel object categories. Due to the lack of large-scale 3D-language data, current methods distill knowledge from pre-trained 2D VLMs into 3D models. However, this distillation is supervised by misaligned 3D-scene-image-to-text data pairs, leading to suboptimal performance. To address this issue, we propose an aligned 3D open-vocabulary semantic segmentation framework with two novel modules: a CLIP-Rewarded Alignment Module that generates high-quality, well-aligned 3D-scene-image-to-text pairs through temperature-based generation and CLIP-rewarded sampling, and an Adaptive Segmentation Module that introduces trainable tokens within the text encoder to adapt it to 3D contexts. This approach significantly outperforms previous methods on representative benchmarks.

Finally, we explore efficient multi-modal feature alignment for OOD understanding. In real-world applications such as autonomous driving and medical diagnosis, AI systems must not only detect and recognize OOD data but also generate appropriate responses by understanding the

scenario—for instance, determining safe actions when encountering an unexpected obstacle or providing diagnostic insights for rare pathologies. Multi-modal large language models (MLLMs, as a class of generative VLMs) offer strong generalization capabilities for such understanding tasks, but their substantial computational requirements limit practical deployment. To address this, we propose an efficient MLLM that incorporates a novel conditional token reduction module to consolidate visual tokens based on their similarity to text tokens and learnable queries, and a novel mixture of multi-modal experts module with a router that takes both text and visual tokens as input for better switching between different low-rank adaptation (LoRA) experts. The proposed method achieves competitive performance while using significantly fewer visual tokens, enabling efficient OOD understanding without sacrificing effectiveness.

This thesis demonstrates that systematic improvements in multi-modal feature alignment can address multiple complex OOD challenges, from detection through recognition to understanding. These contributions establish a foundation for the deployment of AI systems in open-world environments, enabling more reliable and scalable AI systems that can robustly handle novel scenarios.

Acknowledgements

I would like to express my deepest gratitude to everyone who has supported me throughout my doctoral studies. In a world full of uncertainty, I have been fortunate to count on my family, friends, mentors, and colleagues.

First, I thank my family for their unconditional love and support. My parents, Mr. Yong Wang and Mrs. Aixia Yang, have been both my strongest supporters and my most trusted guides. They taught me how to love, how to grow, and how to live a meaningful life. I am deeply grateful to my partner, Dr. Shen, who has stood by me even in the hardest moments. I am also grateful to my pets, Yangyang, Buding, and Nature, whose companionship brought me comfort and emotional support throughout this journey.

It has been a privilege to work with my supervisor, Prof. Krzysztof Czarnecki. I am deeply grateful for his exceptional open-mindedness, understanding, and crucial support at pivotal moments of my doctoral journey. His expertise and dedication have inspired me and sustained me through this degree. Beyond academic guidance, his invaluable patience, support, and encouragement have shaped not only the way I conduct research but also how I hope to mentor and support others in the future.

I am profoundly grateful to the members of my doctoral examination committee for their exceptional guidance and dedication. My sincere thanks go to Professors Olga Veksler, Freda Shi, and Sirisha Rambhatla from the University of Waterloo for their invaluable insights and continuous support. I also extend my deepest appreciation to my external examiner, Professor Nicholas Rhinehart from the University of Toronto, for his thorough review, stimulating discussion, and constructive comments on this thesis.

I also thank Sean Sedwards and Mozhgan Nasr Azadani for their thoughtful feedback and suggestions. My thanks extend to Michal Antkiewicz for expertly maintaining the WISE Lab's computational resources and for his help with technical issues. I am grateful to my colleagues in the WISE Lab for their friendship and support, including Nick, Evelien, Adrian, Kath, Songcheng, and many others who shared this journey with me.

I would also like to thank my mentors, collaborators, and friends during my internships, including Lingjuan Lyu, He Zhao, Ruizhi Deng, Frederick Tung, Greg Mori, Markus Asta, Chenge Li, Shaonan Zhang, and Hasan Tosun, for providing inspiring research environments, broadening my perspective, and making these experiences both intellectually rewarding and personally meaningful.

Finally, my friends deserve special mention for sharing the ups and downs of these years—especially on the climbing walls and through the Vancouver summers.

Thank you all. Your support has been invaluable and has made this journey possible.

Table of Contents

Examining Committee Membership ii

Author's Declaration iii

Statement of Contributions iv

Abstract v

Acknowledgements vii

List of Figures xii

List of Tables xxi

1 Introduction 1

1.1 Motivation 2

1.2 The Promise and Limitations of Vision-Language Models 3

1.3 Thesis Overview and Contributions 5

1.3.1 Contribution 1: Multi-Modal Prototype Learning for OOD Detection . . 7

1.3.2 Contribution 2: Improving Multi-modal Feature Alignment for Recognizing Novel 3D Objects 7

1.3.3 Contribution 3: Context-Aware Multi-modal Feature Alignment for Efficiently Understanding OOD Scenarios 8

1.4 Thesis Organization 8

2 Background **10**
2.1 Vision-Language Models . . . 10
2.1.1 Pretrained Language Models . . . 10
2.1.2 Discriminative Vision-Language Models . . . 12
2.1.3 Generative Vision-Language Models . . . 15
2.1.4 Efficient Adaptation and Scaling of VLMs . . . 20
2.2 Out-of-Distribution Detection . . . 26
2.2.1 Problem Formulation . . . 26
2.2.2 Traditional OOD Detection Methods . . . 29
2.2.3 VLM-based OOD Detection . . . 29
2.3 3D Semantic Segmentation . . . 33
2.3.1 Problem Formulation . . . 33
2.3.2 Deep Learning Methods for Point Clouds . . . 35
2.3.3 3D Open-Vocabulary Recognition . . . 35
2.3.4 Open-Vocabulary 3D Semantic Segmentation . . . 36
2.4 Efficient Generative VLMs . . . 37
2.4.1 Problem Formulation and Evaluation . . . 38
2.4.2 Visual Token Reduction . . . 38
2.5 Summary . . . 40

3 Mitigating the Modality Gap: Few-Shot Out-of-Distribution Detection with Multi-modal Prototypes and Image Bias Estimation **42**
3.1 Introduction . . . 42
3.2 Preliminaries . . . 49
3.2.1 Problem Setup . . . 49
3.2.2 Revisiting CLIP, MCM, and Prompt Tuning . . . 50
3.3 Methodology . . . 51
3.3.1 RQ1: Multi-modal Prototypes . . . 52

3.3.2 RQ2: SUPREME . . . 57
3.4 Experiments . . . 60
3.4.1 Experimental Details . . . 60
3.4.2 Main Results . . . 61
3.4.3 Ablation Study . . . 64
3.4.4 Visualizing OOD Examples . . . 73
3.4.5 Ablations . . . 75
3.4.6 Details of Benchmarking Datasets . . . 75
3.5 Conclusion and Limitations . . . 77

4 AIDE: Improving 3D Open-Vocabulary Semantic Segmentation by Aligned Vision-Language Learning 79
4.1 Introduction . . . 79
4.2 AIDE . . . 81
4.2.1 Problem Definition . . . 81
4.2.2 Preliminaries, Problems, and Solutions . . . 83
4.2.3 CLIP-Rewarded Alignment — Data Generation . . . 85
4.2.4 Adaptive Segmentation — Text Modeling . . . 87
4.2.5 Training Objective . . . 88
4.3 Experiments . . . 88
4.3.1 Benchmarks, Baselines, and Implementation . . . 88
4.3.2 Quantitative Results . . . 90
4.3.3 Ablation Studies . . . 91
4.3.4 Qualitative Results—Generalization . . . 96
4.3.5 Qualitative Results . . . 99
4.3.6 Efficiency . . . 102
4.4 Prompt Templates . . . 102
4.5 Conclusion and Limitations . . . 105

5 LEO-MINI: An Efficient Multimodal Large Language Model using Conditional Token Reduction and Mixture of Multi-Modal Experts **107**

5.1 Introduction . . . 107

5.2 Preliminaries . . . 110

5.2.1 Problem Definition . . . 110

5.2.2 Standard MLLM Pipeline . . . 111

5.3 Methodology . . . 111

5.3.1 Architecture . . . 112

5.3.2 Conditional Token Reduction (**CoTR**) . . . 113

5.3.3 Mixture of Multi-modal Experts (**MMoE**) . . . 115

5.3.4 Training stages . . . 116

5.4 Experiments . . . 117

5.4.1 Implementations . . . 117

5.4.2 Benchmark Datasets . . . 117

5.4.3 Evaluation Metrics . . . 119

5.4.4 Main Results . . . 120

5.4.5 Ablation Studies . . . 121

5.4.6 Efficiency Analysis . . . 126

5.4.7 Qualitative Analysis . . . 127

5.5 Conclusion and Limitations . . . 128

6 Conclusion and Future Work **129**

References **133**

List of Figures

1.1 Real-world AI deployment requires models to detect, recognize, and understand OOD data across diverse applications. Consider an autonomous vehicle encountering a bear on a rural road—an object never seen during training. The system must first reliably *detect* the bear as OOD (left), then *recognize* this novel object in 3D space (middle), and finally *understand* the scenario to answer user queries (right). This thesis addresses these interconnected OOD problems through systematic improvements in multi-modal feature alignment, enabling safer and more robust AI deployment. 2

2.1 BERT architecture and training paradigm. The same bidirectional Transformer encoder is first pretrained with self-supervised language objectives and then fine-tuned for downstream tasks by adding task-specific output layers. The `[CLS]` token is inserted at the beginning of each input sequence and its final representation is commonly used for sentence-level prediction, while `[SEP]` marks the boundary between text segments such as a question and an answer. The figure is taken from [40]. 11

2.2 CLIP architecture and contrastive training paradigm. A batch of matched image-text pairs is encoded by separate image and text encoders into a shared embedding space. Training increases the similarity of matched image-text pairs along the diagonal of the similarity matrix while decreasing the similarity of mismatched pairs, enabling zero-shot recognition by comparing an image embedding with text embeddings of class-name prompts. This figure is taken from [174]. 13

2.3 The pervasive modality gap in multi-modal contrastive representation learning. (a) Overview of multi-modal contrastive learning. Paired inputs from two modalities (*e.g.*, image-caption) are sampled from the dataset and embedded into the hypersphere using two different encoders. The loss function is to maximize the cosine similarity between matched pairs given all the pairs within the same batch. (b) UMAP visualization of generated embeddings from pre-trained models. Paired inputs are fed into the pre-trained models and the embeddings are visualized in 2D using UMAP (lines indicate pairs). The visualization shows a clear modality gap for various models trained on different modalities. (c) UMAP visualization of generated embeddings from the same architectures with random weights. Modality gap exists in the initialization stage without any training. This figure is taken from [128]. 14

2.4 BLIP-2 architecture and first-stage vision-language pretraining objectives. The left panel shows how a frozen image encoder produces visual features that are queried by a Q-Former through learnable query embeddings. The Q-Former is trained with image-text contrastive learning, image-text matching, and image-grounded text generation so that the query tokens extract visual information relevant to the paired text. The right panel shows the attention masks used by the different objectives to control whether query tokens and text tokens can attend to each other. The figure is taken from [113]. 17

2.5 LLaVA architecture for visual instruction following. An input image is encoded by a vision encoder, and a trainable projection module maps the resulting visual features into the token embedding space of a language model. The projected visual tokens are concatenated with the language instruction, allowing the language model to generate a natural-language response conditioned on both the image and the instruction. The figure is taken from [135]. 18

2.6 Comparison between full model tuning and prompt tuning for adapting a pretrained language model to multiple tasks. Full model tuning creates and stores a separate copy of all pretrained parameters for each downstream task, so different tasks must be served by different model instances. Prompt tuning keeps the pretrained model fixed and stores only a small learned prompt for each task, allowing multiple tasks to share the same backbone during inference. For a T5 "XXL" model, this reduces task-specific storage from 11 billion parameters per tuned model to 20,480 prompt parameters per task when using a prompt length of five tokens. The figure is taken from [105]. 19

2.7 AUTOPROMPT procedure for converting a classification task into masked language model inference. Each input text x_{inp} is inserted into a prompt template λ containing learned trigger tokens x_{trig} and one `[MASK]` position. The trigger tokens are shared across examples and selected by a gradient-based search. At inference time, the masked language model predicts candidate words at the `[MASK]` position, and class probabilities are computed by aggregating the probabilities of label words associated with each class, such as positive or negative sentiment. The figure is taken from [181]. 20

2.8 Context Optimization (CoOp) for adapting CLIP with learnable text prompts. Instead of using a manually written prompt such as "a photo of a [CLASS]," CoOp replaces the context words with continuous vectors optimized from labeled training examples. The unified-context design shares one set of learnable context vectors across all classes, while the class-specific design learns a separate context for each class before encoding the completed prompts with CLIP's text encoder. The figure is taken from [254]. 21

2.9 Conditional Context Optimization (CoCoOp) for image-conditioned prompt learning. CoCoOp extends CoOp by combining learnable context vectors with a lightweight Meta-Net that maps each image feature to an input-conditional token. The resulting prompt changes with the image being classified, which improves generalization to unseen classes compared with fixed learned prompts. The figure is taken from [253]. 22

2.10 Low-Rank Adaptation (LoRA) reparameterization of a pretrained weight matrix. The original weight W_0 is frozen, and the task-specific update ΔW is represented as the product of two low-rank matrices, BA. Only A and B are trained, so adaptation changes the model through a small number of additional parameters while preserving the pretrained backbone. The figure is taken from [48]. 23

2.11 Mini-Gemini framework for improving fine-grained visual understanding in multimodal large language models. The model supplements a standard low-resolution visual stream with higher-resolution image inputs and additional visual encoders so that small objects and local details can be preserved before visual features are passed to the language model. The design improves perception but also increases the number of visual tokens that later stages must process. The figure is taken from [123]. 24

2.12 MoE-LLAVA framework for parameter-efficient multimodal instruction tuning. The model builds on LLaVA by inserting sparse mixture-of-experts modules, such as LoRA-style experts, into the language model. A routing mechanism selects which experts should process each hidden representation, increasing task capacity while activating only a subset of parameters for a given input. The figure is taken from [129]. 25

2.13 LEO framework for combining multiple vision encoders in a multimodal large language model. LEO uses a dual-vision architecture to extract complementary visual features, fuses visual tokens after tile-level adaptation, and applies pixel unshuffle to reduce the number of visual tokens before they are consumed by the language model. The figure illustrates how the architecture balances richer visual perception with token-efficiency constraints. The figure is taken from [6]. 26

2.14 Maximum Concept Matching (MCM) framework for zero-shot OOD detection with CLIP [160]. The in-distribution task is specified by a label set $\mathcal{Y}_{\text{ID}}$, whose class names are wrapped in text templates and encoded as text prototypes. A test image is encoded by CLIP's image encoder and compared with all ID text prototypes. If the image has a high scaled similarity to one ID prototype, it is treated as likely in-distribution; if its closest similarity is low, it is treated as uncertain and more likely OOD. The figure is taken from [160]. 30

2.15 Envisioning Outlier Exposure (EOE) framework for zero-shot OOD detection [16]. Starting from the ID label set $\mathcal{Y}_{\text{ID}}$, an LLM is prompted to generate plausible outlier labels $\mathcal{Y}_{\text{OOD}}$ without using real OOD images. Both ID and generated OOD labels are encoded by CLIP's text encoder to form a textual classifier. At test time, the image encoder extracts the visual feature, similarities to ID and generated OOD text prototypes are computed, and the detection score S_{EOE} uses the contrast between these similarities to separate ID from OOD inputs. The figure is taken from [16]. 31

2.16 Effect of adding OOD text prototypes on zero-shot OOD score distributions. Panel (a) uses only closed-set ID class labels, which can produce overlapping ID and OOD score distributions. Panel (b) adds actual OOD class labels and improves separation, showing the value of explicit negative concepts. Panel (c) replaces actual OOD labels with EOE-generated outlier labels [16], demonstrating that LLM-generated negative concepts can improve OOD detection without access to real OOD labels. The figure is taken from [16]. 32

2.17 LSN framework for few-shot VLM-based OOD detection. The CLIP image encoder and text encoder remain frozen while the method learns positive prompts for ID concepts and negative prompts for OOD concepts from limited ID training data. During inference, a test image is compared with both sets of prompt embeddings, and the relative similarity to positive versus negative prompts is used to determine whether the image should be accepted as ID or rejected as OOD. The figure is taken from [164]. 33

2.18 PLA framework for open-vocabulary 3D semantic segmentation. PLA transfers semantic knowledge from vision-language models to 3D point clouds by generating image captions for 3D scenes and using them as language supervision. Compared with a closed-set segmentation network, PLA replaces the fixed learnable semantic classifier with text-encoder category embeddings derived from class names, adds a binary head to calibrate base and novel class probabilities, and uses an instance head for instance-level segmentation. Point-language association further supervises point embeddings with caption embeddings so that the 3D model can recognize categories beyond the closed training label set. The figure is adapted from [43]. 36

2.19 LLaVA-Mini framework for reducing visual tokens in multimodal large language models. The left panel shows the high-level idea of representing each image with a compact visual token sequence before sending it to the language model. The right panel details the query-based compression module and modality pre-fusion module, which summarize visual features and fuse them with text-side information so that the model can reduce computation while preserving visual information needed for response generation. The figure is taken from [247]. . . . 39

3.1 Comparison between text-only [160] and our multi-modal prototype-based OOD detection. Circles denote ID image embeddings, stars denote OOD image embeddings, diamonds denote ID text prototypes, and squares denote ID image prototypes. (a) Text-only scoring can assign high similarity to an OOD image that lies near an ID text prototype in embedding space, producing an OOD false negative (accepted as ID). (b) Our proposed SUPREME adds ID image prototypes so ID samples are supported by both text and visual anchors, sharpening the ID/OOD boundary. 44

3.2 Four arbitrary classes in ImageNet-1k are selected to visualize the similarity distribution between image embeddings and text prototypes within and across each class. We also compare Kolmogorov–Smirnov (KS) statistics between inter- and intra-class similarity distributions. "A - A" refers to the similarity distribution within class A, and "A - B" refers to the similarity distribution between class A and class B. Higher KS statistic values indicate a greater difference between two distributions. Best viewed in color. 47

3.3 t-SNE visualization of CLIP (ViT-B/16) embeddings showing failure cases for text-prototype OOD detection. The plot compares ID image embeddings from the ImageNet-1k test set, OOD image embeddings from iNaturalist, and ID text prototypes generated from class-name prompts. Several OOD image embeddings are located closer to ID text prototypes than genuine ID image embeddings are, illustrating how spatial proximity caused by the image-text modality gap can produce false positives. Figure 3.5 shows that SUPREME reduces this gap after alignment. 48

3.4 Overview of SUPREME for few-shot VLM-based OOD detection. The two novel modules, *i.e.*, BPG (biased prompt generation) and ITC (image-text consistency), are designed to reduce the excessive image-text modality gap in this OOD setting. (a) During the few-shot fine-tuning stage, (b) BPG generates image domain-biased prompts, conditioned on the estimated image domain bias and the prompt-conditioning image feature $m(I)$ for better image-text fusion. (c) ITC reduces the modality gap directly by the intra- and inter-modal losses (ℓ_{intra} and ℓ_{inter}) with the text-to-image mapping f_{t2i} and the image-to-text mapping f_{i2t}. (d) During inference, ID image prototypes are obtained by averaging each class's base ID image embeddings from the training data (Equation (3.3)). The proposed S_{GMP} (Equation (3.20)) is calculated from the maximum similarities between the test image embedding I, its image-to-text mapped embedding I', and ID multi-modal prototypes (*i.e.*, text prototypes $\{P_{txt,c}\}_{c=1}^{C}$ and image prototypes $\{P_{img,c}\}_{c=1}^{C}$). S_{MCM} refers to the MCM score [160]. 51

3.5 t-SNE visualization of embeddings from two arbitrary ImageNet-1k validation classes after few-shot training with SUPREME. The plot shows original image embeddings, mapped image embeddings, ID text prototypes, and ID image prototypes in the same two-dimensional projection. ID image prototypes are computed from the corresponding ImageNet-1k training images, while the displayed image embeddings are from held-out test images; thus their proximity reflects generalization of the training-derived class prototypes to unseen test samples, rather than comparing an image to a prototype computed from itself. After training, the mapped image embeddings move toward the corresponding text-prototype region, while the original test image embeddings remain close to training-derived image prototypes and the two classes remain separated. 67

3.6 Impact of the number of few-shot ID training images per class on OOD detection performance. The plot reports average FPR95 and AUROC as the available tuning data increases, showing how sensitive SUPREME is to the size of the labeled ID subset. 69

3.7 Effect of learnable prompt length on OOD detection performance. The plot compares average FPR95 and AUROC for different numbers of learnable context tokens, showing how prompt capacity affects the balance between adaptation and overfitting. 69

3.8 Effect of the number of base ID training images used to construct image prototypes. The plot reports average FPR95 and AUROC as more training images are included in the class-wise prototype average. "1281" corresponds to using all available training images per class as the base image set. 70

3.9 Hyperparameter sensitivity of the loss weights α and β. The two plots report average FPR95 and AUROC while varying α, which controls the ITC losses, and β, which controls the image-domain-bias loss. 70

3.10 Comparison between MCM and GMP score distributions on iNaturalist and SUN 71

3.11 Comparison between MCM and GMP score distributions on Places and Texture . 72

3.12 t-SNE visualizations of CLIP (ViT-B/16) embeddings showing OOD false-positive patterns across four OOD datasets. Each panel compares ImageNet-1k ID image embeddings from the test set, OOD image embeddings, and ID text prototypes generated from class-name prompts. The examples highlight cases where OOD images are positioned closer to an ID text prototype than genuine ID images are, motivating the use of image prototypes and image-text alignment in SUPREME. . 74

4.1 Previous methods use misaligned paired data (*e.g.*, image/point cloud 1 is closest to text 2) and freeze the text encoder trained on 2D benchmark datasets, leading to misalignment between text encoders and 3D models and suboptimal performance. Our proposed AIDE aligns text encoders with 3D models by (i) generating better aligned 3D-image-to-text training data (CLIP-Rewarded Alignment Module) and (ii) optimizing deep trainable prompt tokens (Adaptive Segmentation Module). . 80

4.2 The illustration of our AIDE with two proposed modules, *i.e.*, CLIP-rewarded alignment module (Section 4.2.3) for enhancing the quality of 3D-text data pairs and adaptive segmentation module (Section 4.2.4) for adapting text encoders. **CLIP-Rewarded Alignment Module.** We first generate numerous captions using the temperature-based generation strategy, then sample captions based on their similarity to the images (CLIP-rewarded sampling), and finally align 3D point clouds and captions hierarchically. **Adaptive Segmentation Module.** To automatically find the most suitable prompt for adapting text encoders into 3D scenarios, AIDE extends prompt tuning [89] by incorporating learnable tokens in the input space and each layer of the text encoder f_{text}. A confidence calibration head is employed for balancing the confidence on base and novel categories, as models tend to overestimate their certainty in base categories [43, 121, 150]. . . 83

4.3 The number of most-aligned captions generated by different temperatures for each image. We use temperatures of 0.5, 1.0 (standard temperature), 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, and 100.0 for generating diverse captions. . 94

4.4 Examples of captions generated at different sampling temperatures for the same 3D-scene view, together with their CLIP cosine similarity to the paired image. Higher temperatures increase caption diversity but also introduce noisier phrases, whereas lower temperatures are more conservative. During training, CLIP-rewarded sampling preferentially selects captions with higher image–text similarity, so the caption distribution used for supervision shifts toward better-aligned descriptions after model training. 100

4.5 Qualitative open-vocabulary segmentation results on ScanNet validation scenes. For each scene, we compare the baseline (PLA) and AIDE on both base categories seen during training and novel categories held out at training time. 102

5.1 Performance overview of LEO-MINI-Llama3-8B. a: Comparison between LEO-MINI (64 tokens) and existing token reduction MLLMs [78, 118, 178, 232, 247], where LEO-MINI outperforms all models. b: Comparison with other MLLMs [123, 134, 180, 193] on 11 vision-language tasks [55, 84, 95, 107, 139, 140, 154, 155, 184, 234] using Llama3-8B [145] as the LLM. LEO-MINI achieves competitive performance while using only 64 visual tokens 108

5.2 The overview of the proposed LEO-MINI. LEO-MINI is designed with MMOE (Section 5.3.3) to enhance visual comprehension and COTR (Section 5.3.2) to reduce the number of visual tokens for efficiency. 112

5.3 The overview of the proposed COTR. The COTR module takes a group of visual tokens, a learnable query, and text tokens as input, and outputs a consolidated group of visual tokens to reduce the number of visual tokens. 114

5.4 The overview of the proposed MMOE-LLM for LLM fine-tuning. MMOE (language) consists of a set of LoRA experts and a routing network that selects the appropriate expert based on the input visual tokens and textual instructions. Moreover, a general expert is employed to learn the general knowledge and improve the robustness of the model. 115

5.5 Visualization of the expert choice using LEO-MINI-Llama-8B on TextVQA, ScienceQA, and GQA. Best viewed in color. 123

5.6 Visualization of the expert choice using LEO-MINI-Llama-8B (right) on TextVQA, ScienceQA, and GQA compared to the vanilla MoE (left) on selected layers. Best viewed in color. 124

5.7 Visualization of the expert choice using the vanilla MoE on TextVQA, ScienceQA, and GQA. Best viewed in color. 125

5.8 Qualitative results across four vision-language tasks 127

List of Tables

2.1 Summary of the three research problems addressed in this thesis and how they relate to multi-modal feature alignment. OOD detection is framed as binary detection, where image and text representations must be calibrated for reliable ID/OOD separation. Open-vocabulary 3D semantic segmentation is framed as multi-class recognition, where 3D point-cloud features must be aligned with text embeddings for seen and unseen categories. Efficient MLLMs are framed as generative visual understanding, where visual tokens must be selected and routed according to instruction semantics. The table also indicates the modality pair, main technical challenge, alignment goal, and thesis chapter associated with each problem. . . . 41

3.1 Overall pre-alignment OOD detection errors under different prototype settings. This analysis is conducted in the original CLIP embedding space (CLIP-B/16) before applying explicit alignment modules, in order to isolate the effect of prototype modality. OOD FN denotes OOD samples incorrectly predicted as ID, while OOD FP denotes ID samples incorrectly predicted as OOD. Lower values are better. For each prototype setting, the threshold is selected at 95% ID TPR. Therefore, the aggregate OOD FP rate is fixed at 5%, but the identities of rejected ID samples differ across prototype settings; Table 3.2 reports these sample-level error transitions. Specifically, the threshold of text, image, and multimodal prototypes are 0.010747829, 0.011646281, and 0.011249677, respectively. 45

3.2 Error transitions between prototype settings before alignment using CLIP-B/16. We compare each target setting against a reference setting. "Fixed" denotes samples incorrectly classified by the reference setting but correctly classified by the target setting. "Introduced" denotes samples correctly classified by the reference setting but incorrectly classified by the target setting. This table shows how changing the prototype modality can both correct and introduce OOD false negatives and false positives, making explicit the non-monotonic effect of adding image prototypes. 45

3.3 Representative prototype-level error cases before alignment using CLIP-B/16. Each row shows a case where one prototype modality fails and the complementary modality or the combined multi-modal prototype corrects the prediction. $S_{MCM,T}$, $S_{MCM,I}$, and S_{MMP} denote the ID confidence scores obtained using text prototypes, image prototypes, and multimodal prototypes, respectively. We use the same threshold as in Table 3.1 for each setting. Higher scores indicate stronger ID confidence. We multiply the scores by 100 for better illustration. 46

3.4 Comparison of OOD detection results on ImageNet-100 as the ID dataset and four OOD datasets. The table compares three scoring choices: MCM with ID text prototypes, MCM with ID image prototypes, and the proposed S_{MMP} that averages both prototype types. ID image prototypes are computed from ImageNet-100 training images, while evaluation uses held-out ID test images and OOD test images. We use CLIP-B/16 for S_{MCM} and S_{MMP} (ours, Equation (3.4)). FPR95 is the false positive rate of OOD images when the true positive rate of ID images is fixed at 95%, and AUROC is the area under the receiver operating characteristic curve. Best results are shown in **bold**. 53

3.5 OOD detection results on ImageNet-1k 61

3.6 OOD detection results using ImageNet-100 as the ID dataset and four OOD datasets: iNaturalist, SUN, Places, and Texture. The table compares baseline post-hoc scores, prompt-tuning methods, and SUPREME using CLIP (ViT-B/16). FPR95 and AUROC are reported for each OOD dataset and averaged across datasets; lower FPR95 and higher AUROC indicate better detection. Best results are shown in **bold**, and second-best results are underlined. 62

3.7 OOD detection results on the OpenOOD benchmark using ImageNet-1k as the ID dataset. The table separates NearOOD and FarOOD settings, where NearOOD contains semantically closer OOD classes and FarOOD contains more distinct OOD classes. FPR95 and AUROC are reported for each setting; lower FPR95 and higher AUROC indicate better detection. Best results are shown in **bold**. . . 62

3.8 OOD detection results on the OpenOOD covariate-shifted ID (CSID) setting using ImageNet-1k as the ID dataset. CSID evaluates whether the detector remains reliable when test samples still belong to ID classes but undergo distribution shifts. FPR95 and AUROC are reported, with best results shown in **bold**. 62

3.9 Near-OOD comparison between ImageNet-10 and ImageNet-20. The left block treats ImageNet-10 as ID and ImageNet-20 as OOD, while the right block reverses the split. This setting evaluates fine-grained OOD detection because the ID and OOD categories are semantically related subsets of ImageNet. We use CLIP-B/16 for all experiments and report FPR95 and AUROC. Best results are shown in **bold**. "*" represents our reproduction. 63

3.10 Ablation study of the proposed modules and OOD score on ImageNet-1k → four OOD datasets. The first rows compare combinations of ITC, BPG, and S_{GMP} using CLIP (ViT-B/16), and the last rows evaluate the full method with ResNet-50 and ResNet-101 backbones. The table reports FPR95 and AUROC for each OOD dataset and the average across datasets. When S_{GMP} is not combined with BPG, it degenerates to S_{MCM}. 64

3.11 Effect of replacing the linear ITC mappings with a two-layer MLP. The table reports average FPR95 and AUROC on ImageNet-1k → four OOD datasets. The comparison tests whether a more expressive mapping improves image-text alignment or overfits in the few-shot setting. 64

3.12 Comparison of in-distribution classification accuracy on the ImageNet-1k validation set. The table reports Top-1 accuracy for several OOD detection methods to verify that improved OOD detection does not come at the cost of degraded ID classification performance. 65

3.13 Performance comparison using different visual backbones. The table reports average FPR95 and AUROC when SUPREME and baseline methods use ResNet-50 or ResNet-101 as the base image encoder, showing whether the proposed method remains effective beyond CLIP ViT-B/16. 65

3.14 Performance of SUPREME combined with post-hoc methods 66

3.15 Quantifying the modality gap with different methods 68

3.16 Comparison of parameter count and inference complexity between MCM and SUPREME using CLIP-B/16 as the base model. The table separates total parameters, trainable parameters, the offline cost of constructing ID image prototypes, and per-sample inference complexity. Image prototypes are computed once from the ID training data and reused for all test samples, so SUPREME keeps constant-time inference with respect to the number of test inputs. 68

4.1 Semantic segmentation performance on ScanNet (B15/N4) 82

4.2 Category partitions for open-vocabulary semantic segmentation on ScanNet and S3DIS. 89

4.3 Results on ScanNet. † and ‡ refer to numbers copied from He *et al.* [67] and Ding *et al.* [43]. Best in **Bold**. 90

4.4 Results on S3DIS and nuScenes. ‡ refers to numbers copied from Ding *et al.* [43]. Best in **Bold**. 90

4.5 Ablation studies on different modules of AIDE. "Adaptive Segmentation" refers to the adaptive segmentation module. "Caption Selection" and "Caption Sampling" refer to using temperature-based generation and then selecting the best-aligned captions or sampling captions based on their similarity to the images (CLIP-rewarded alignment module). 91

4.6 Ablation studies on different numbers of learnable tokens of AIDE on ScanNet (B15/N4). 92

4.7 Ablation studies on different numbers of captions of AIDE for each temperature. 92

4.8 Ablation studies on using different text encoders of AIDE on ScanNet (B15/N4). 93

4.9 Ablation studies of different temperatures used for caption generation of AIDE. . 95

4.10 Ablation studies on using the image alignment and comparisons with image-based methods. † refers to numbers copied from He *et al.* [67]. 96

4.11 Synonyms and hypernyms of class names on ScanNet and S3DIS used for generating class embeddings. For classes without suitable synonyms or hypernyms, we use the original class name as their synonyms and hypernyms and mark "-" in the table. 96

4.12 Open-vocabulary semantic segmentation results on ScanNet (B15/N4) using the original class names, their synonyms, and hypernyms. The synonyms and hypernyms of class names are presented in the Appendix. 97

4.13 Zero-shot transfer ability of baseline and AIDE. We train models on ScanNet (B15/N4) or S3DIS (B8/N4) and test them on another. Best in **bold**. 97

4.14 Generalization of AIDE to instance segmentation on S3DIS. 99

4.15 Similarity between learned prompts (tokens) in the input space and "human vocabulary" under use the cosine similarity. "Word" refers to subwords in the CLIP's vocabulary. 101

4.16 Comparison of parameters and throughput between the baseline and AIDE. . . . 102

5.1 The training stages of LEO-MINI. COTR and MMOE-LLM are added and fine-tuned in the third stage, while the LLM and vision experts (MMOE-Vision) remain frozen. 116

5.2 Comparison to token reduction methods on general, OCR, and knowledge-based tasks. Best in **Bold**. Second best in Underline. 119

5.3 Comparison to general and MoVE (mixture of vision experts)-based MLLMs. Best in **Bold**. Second best in Underline. 120

5.4 Ablation studies of LEO-MINI. We explore the impact of the number of visual tokens, the amount of training data, and the MMOE. Numbers in green indicate the performance is improved compared to LEO-MINI (64 tokens). 122

5.5 Ablation study on the effectiveness of COTR and MMOE-LLM. We use Llama3-8B as the backbone. 122

5.6 Efficiency analysis of LEO-MINI and other MLLMs using Llama3-8B. "VT" represents visual tokens. Δ indicates the difference between LEO-MINI and LLaVA-Next. 126

Chapter 1

Introduction

This thesis investigates the fundamental challenge of handling out-of-distribution (OOD) data in artificial intelligence (AI) systems [66,134,142,174], with multi-modal feature alignment serving as ‘ technical approach. As AI systems transition from controlled laboratory settings to real-world deployments, they inevitably encounter novel situations that differ from their training data, such as autonomous vehicles encountering novel road conditions [235] and medical AI analyzing rare pathologies [50]. These novel situations [72, 79, 138, 227, 243] are often referred to as OOD, which encompasses both semantic shift (*e.g.*, encountering object categories absent from training data) and covariate shift (*e.g.*, changes in environmental conditions, sensor characteristics, or data acquisition protocols). The ability of AI systems to handle OOD data is crucial for reliable deployment in real-world scenarios, as these systems are expected to operate in open-world environments stably and robustly.

In this thesis, we present a unified framework that addresses this challenge through three interconnected research problems of increasing complexity: detecting OOD inputs (a binary decision of whether data is OOD), recognizing novel object categories in 3D environments (assigning semantic labels to OOD samples), and understanding OOD scenarios to answer user queries (understanding and responding to unfamiliar situations). These three problems represent different levels of OOD handling that real-world AI systems must address. The core insight behind this thesis is that multi-modal feature alignment, which calibrates cross-modal gaps so information can be effectively integrated when the downstream task requires it, offers a principled solution across all three settings. For discriminative VLMs [120, 174] (*e.g.*, CLIP), alignment is realized by contrastive similarity in a shared embedding space and supports OOD detection and open-vocabulary recognition. For generative VLMs [113, 114] (*e.g.*, LLaVA [135]), alignment is realized by instruction-conditioned fusion of visual and textual tokens and supports OOD understanding through natural-language responses.

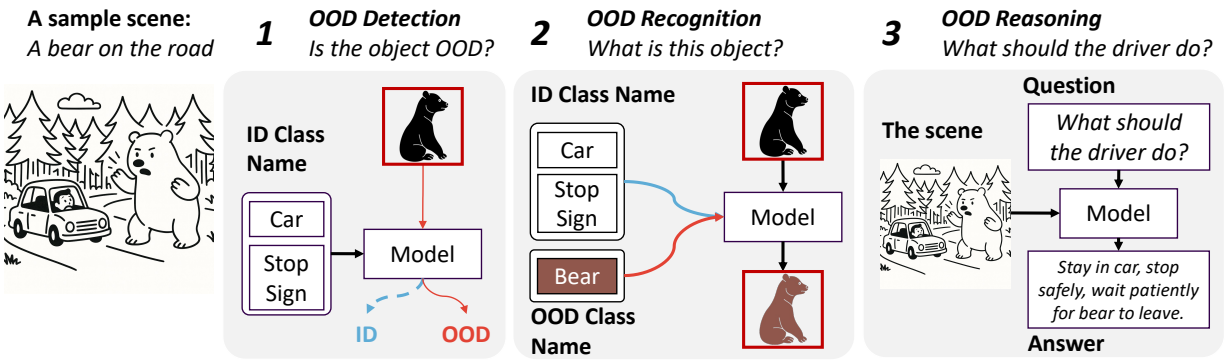


Figure 1.1: Real-world AI deployment requires models to detect, recognize, and understand OOD data across diverse applications. Consider an autonomous vehicle encountering a bear on a rural road—an object never seen during training. The system must first reliably *detect* the bear as OOD (left), then *recognize* this novel object in 3D space (middle), and finally *understand* the scenario to answer user queries (right). This thesis addresses these interconnected OOD problems through systematic improvements in multi-modal feature alignment, enabling safer and more robust AI deployment.

1.1 Motivation

Since the emergence of deep learning, AI has achieved remarkable success across a wide range of applications, from medical diagnosis [50] to autonomous driving [235], and robotic manipulation [152]. However, deploying AI systems in real-world scenarios remains fundamentally challenging. Different from controlled laboratory environments where training and test data follow similar distributions [14, 39], real-world deployments constantly encounter novel situations: medical AI systems face rare pathologies absent from training data [98,201], autonomous vehicles navigate unprecedented weather conditions, and robotic systems must manipulate previously unseen objects. For example, in Figure 1.1, an autonomous vehicle is navigating a rural environment and encounters a novel object, such as a bear it has never seen. Without proper OOD handling capabilities, the AI system may fail to respond appropriately to this unexpected scenario, potentially compromising safety, which is a critical safety concern in real-world deployments. Therefore, it becomes more and more important to ensure that AI systems can handle OOD data effectively and stably in real-world scenarios.

The consequences of failing to handle OOD data can be severe. In 2022, an estimated 5.93 million motor vehicle accidents occurred in the United States alone, with studies suggesting that 94% of crashes trace back to driver-related recognition and decision-making errors [185]. This

vulnerability is mirrored in AI systems. For instance, medical diagnostic models have shown performance drops of nearly 30% when encountering data from new clinical environments [237], and autonomous systems have historically failed to recognize overturned vehicles on highways—a classic "out-of-distribution" perception failure. These empirical instances underscore that OOD robustness is not just a theoretical challenge, but a prerequisite for the safe integration of AI into society.

In autonomous driving, while researchers promise to reduce such accidents through more advanced perception and decision-making systems, this potential can only be realized if AI systems can reliably, robustly, and safely handle OOD data. In our example shown in Figure 1.1, when an autonomous vehicle encounters a bear on a rural road, it must navigate a progression of increasingly complex challenges: first *detecting* that the object is unfamiliar, then *recognizing* what it might be in 3D space, and finally *understanding* the scenario to answer user queries.

This progression from detection through recognition to understanding represents a fundamental challenge that extends far beyond autonomous driving. In digital pathology, whole slide image (WSI) scanners must identify when tissue samples contain artifacts or rare pathologies that could mislead diagnostic systems [87], recognizing whether the tissue corresponds to a known pathology/artifact or should be flagged as OOD for further inspection, and finally understanding the scenario to guide the pathologist to the correct diagnosis. In robotics, manipulation systems need to recognize when encountering novel scenarios that differ from training data, such as a novel object or a novel environment, and understand the scenario to navigate the environment or manipulate the object. Across all these domains, the core challenge remains the same: AI systems must move beyond the closed-world assumption that test data will resemble training data, and instead develop robust capabilities for open-world operation.

To address this fundamental challenge, we propose a unified framework consisting of three related research problems of increasing complexity: *detecting OOD inputs* [160], *recognizing novel objects in 3D environments* [111], and *understanding unfamiliar scenarios* [110]. While each problem targets a different aspect of OOD handling, they share a common technical foundation: improved multi-modal feature alignment between visual and semantic representations.

1.2 The Promise and Limitations of Vision-Language Models

Effectively handling OOD data requires moving beyond traditional vision-only approaches [142, 182]. While convolutional neural networks (CNNs) [66] and vision transformers [142] excel on academic benchmarks [39, 131] where training and test distributions align, they fundamentally lack the semantic understanding capability necessary for open-world scenarios [12, 222]. For

example, a CNN trained to classify the species of a bird [199] cannot meaningfully respond when presented with a car, as it can only force the input into one of its predetermined classes, often with dangerously high confidence [201].

To address this limitation, instead of processing visual information independently of text instructions, inspired by the remarkable success of pretrained large language models (LLMs) [166, 194] in capturing rich semantic knowledge from massive text corpora, vision-language models (VLMs) [134, 135, 174] have emerged as a promising paradigm. By aligning visual representations with the rich semantic space of natural language [135, 174], VLMs gain capabilities that were previously impossible: zero-shot recognition of novel categories [174], open-vocabulary understanding [108], and visual understanding guided by natural language instructions [110].

For discriminative VLMs [120, 174], the key insight is that natural-language supervision can define a visual recognition space without requiring a fixed task-specific classifier. Discriminative VLMs, such as CLIP [174], learn to associate images and text by maximizing similarity between matched pairs and pushing unmatched pairs apart in a shared embedding space using contrastive learning [203]. This contrastive objective, trained on hundreds of millions of image-text pairs from the web, produces representations where semantic similarity in language space can be compared directly with visual similarity in image space. The result is a model that can recognize objects it has never explicitly been trained to classify, simply by computing similarity to textual descriptions.

Relying on a different mechanism, generative VLMs, such as LLaVA [113, 114, 135], learn to generate text given images and text instructions for visual understanding tasks, enabling more flexible and context-aware understanding. These models usually map input images into a sequence of visual tokens using pretrained vision encoders, such as CLIP-L [174], SAM [100], and EVA [52], and then feed these visual tokens into a LLM [28, 145] pretrained on large-scale text corpora to utilize the rich semantic knowledge and generate the answer given the text instructions.

However, despite their promise, current VLMs still face significant limitations when processing OOD data. For both discriminative and generative VLMs, a critical challenge is the *modality gap* [128], where image and text embeddings occupy clearly separated regions of the embedding space, even for semantically matched image-text pairs.

This gap arises from model initialization and the contrastive learning optimization process, and has profound implications for OOD detection. Specifically, OOD images may exhibit high similarity to in-distribution text prototypes due to spatial proximity rather than true semantic alignment, leading to unreliable OOD detection. In fact, because the region (or "cone") of text embeddings is very narrow [128], there is a high likelihood for such mismatches: as shown in Table 3.4, using text prototypes for OOD detection results in a high FPR95 of 32.58%, indicating that many OOD samples are incorrectly identified as ID, which strongly demonstrates the practical

limitations of relying solely on text embeddings and validates the necessity of better multi-modal alignment for effective OOD detection. Furthermore, extending VLM capabilities to 3D open-vocabulary semantic segmentation introduces additional alignment challenges. The scarcity of large-scale 3D-text paired data makes direct training infeasible. Thus, transferring knowledge from 2D VLMs to 3D models is a promising approach. However, this requires more precise alignment between 3D point cloud features and the vision-language embedding space while 3D point clouds are inherently sparse and unstructured. Last but not least, current generative VLMs, such as LLaVA, struggle with text-image alignment for understanding tasks, processing visual information independently of instruction semantics. The generated image features are redundant and noisy, making it difficult to focus on the query-relevant visual information.

To address these limitations, different from prior work that addresses these challenges separately, this thesis investigates how a core technique, *i.e.*, improved multi-modal feature alignment, can address OOD detection, recognition, and understanding challenges, with each building upon insights from the previous.

1.3 Thesis Overview and Contributions

This thesis investigates how *multi-modal feature alignment* can systematically address the progression from OOD detection through recognition to understanding. The methodological core of this thesis consists of three chapters based on our prior peer-reviewed publications: few-shot OOD detection (Chapter 3, [207]), 3D open-vocabulary semantic segmentation (Chapter 4, [206]), and efficient multimodal large language models (Chapter 5, [205]). Co-authorship for these works is detailed in the Statement of Contributions. Rather than treating these as separate challenges requiring distinct solutions, we identify a unifying principle: each problem fundamentally requires establishing precise feature alignment between visual content and text semantics, but at progressively more sophisticated levels. This alignment-centered view leads to three research questions:

RQ1 (Detection): *Can we improve OOD detection by incorporating multi-modal prototypes to mitigate excessive or poorly calibrated modality gaps between image and text embeddings?*

RQ2 (Recognition): *Can we enable open-vocabulary 3D semantic segmentation through improved alignment between 3D point cloud features and vision-language representations?*

RQ3 (Understanding): *Can context-aware alignment between visual tokens and text instructions reduce MLLM computation while maintaining the ability to accurately and efficiently answer benchmark vision-language understanding queries, such as VQA [65, 84], OCR-oriented question answering [184], and science/document understanding tasks [154, 155]?*

In response to these research questions, we investigate the following problems.

Detecting OOD objects (RQ1): As the foundation of processing OOD data, reliably detecting inputs outside the training distribution is critical. OOD detection is fundamentally a binary classification problem: determining whether an input belongs to the in-distribution (ID) data or should be flagged as OOD for further processing. For the example shown in Figure 1.1, this means detecting the bear and flagging it as an OOD object. Current VLM-based OOD detection methods [10, 160, 162] rely heavily on the alignment between image and text embeddings. However, a recent study shows that image and text embeddings are clearly separated [128], leading to overconfident predictions on unfamiliar data. To this end, improving the alignment between image and text spaces becomes crucial for OOD detection.

Recognizing OOD data (RQ2): Once OOD data are reliably detected, the next challenge is to recognize these novel samples by assigning them to meaningful categories—going beyond binary detection to provide semantic understanding of what the OOD object is. In our setting, this is not unconstrained novel class discovery: at test time, we provide a candidate label set that can include categories unseen during training, and the system assigns objects to the best-matching label in this set. In our example (Figure 1.1), the system needs to recognize the detected OOD object in the 3D environment from a provided candidate set, even though "bear" was never a category in the training data. However, existing 3D semantic segmentation methods [23, 102] focus on recognizing known categories rather than handling novel objects in 3D space. This creates a critical gap: AI systems can detect that something is OOD, but lack the capability to categorize it in 3D spatial context. As directly training a 3D open-vocabulary semantic segmentation model is infeasible due to the lack of high-quality 3D-text data, aligning 3D models with existing 2D VLMs [174] for recognizing novel 3D objects becomes a promising approach.

Understanding OOD data (RQ3): With OOD objects detected and recognized, the final and most complex problem is understanding these OOD scenarios to answer user queries and guide decision-making. In this thesis, we define "understanding" as the ability to generate visually grounded, instruction-following natural-language responses, and evaluate it using standard vision-language benchmarks such as VQA [65, 84], OCR-oriented question answering [184], and science/document understanding tasks [154, 155]. This capability is essential in safety-critical applications: in autonomous driving, the system must determine appropriate actions when encountering unexpected obstacles; in medical diagnosis, AI must provide insights about rare pathologies

to guide clinical decisions. In our example (Figure 1.1), AI systems are expected to guide the user to navigate through the rural scene with a bear on the road by answering the user's question ("What should the driver do?") while understanding the semantic relationship between the query and the visual elements in the scene. Generative VLMs offer strong generalization capabilities that make them well-suited for understanding OOD scenarios. However, current models, such as LLaVA, process visual information independently of instruction semantics, leading to redundant and noisy visual features that obscure query-relevant information. To address this, rather than processing visual information independent of text instructions, we hypothesize that establishing fine-grained correspondence between understanding instructions and visual features can enable more effective and efficient understanding OOD scenarios.

While these three problems increase in complexity, they reveal a deeper connection: each fundamentally requires establishing precise feature alignment between visual content and text semantics, but at progressively more sophisticated levels. In response to these research problems, this thesis makes the following contributions.

1.3.1 Contribution 1: Multi-Modal Prototype Learning for OOD Detection

In the first part of this thesis (Chapter 3), we investigate the fundamental challenge of OOD detection using VLMs. We analyze how the modality gap between image and text embeddings leads to unreliable detection when relying solely on text prototypes, as is common in existing methods [10, 160, 162]. Based on our analysis, we propose a novel approach that incorporates *multi-modal prototypes* by combining both image and text representations to calibrate the modality gap and enable more reliable OOD detection.

Our method introduces learnable image prototypes that complement text prototypes, establishing cross-modal anchors that better capture the true decision boundary between in-distribution and OOD samples. We conduct extensive experiments demonstrating that our approach effectively addresses the modality gap limitation and achieves state-of-the-art performance across multiple OOD detection benchmarks.

1.3.2 Contribution 2: Improving Multi-modal Feature Alignment for Recognizing Novel 3D Objects

In the second part of this thesis (Chapter 4), we extend multi-modal feature alignment to 3D point-wise recognition. Building on the improved image-text alignment from the previous chapter, we address the challenge of open-vocabulary 3D semantic segmentation, enabling models to recognize novel object categories in 3D point clouds without explicit training on those categories.

We identify two key limitations in existing approaches: (i) generated 3D-scene-image-to-caption data often exhibits semantic misalignment [43, 226], and (ii) text encoders trained on 2D data lack 3D spatial semantics [67, 171]. To address these issues, we propose a novel framework that establishes more precise alignment between 3D point cloud features and the vision-language embedding space. Our approach leverages carefully designed training strategies and novel model architecture to transfer knowledge from 2D VLMs to 3D models effectively. Experiments on standard benchmarks demonstrate significant improvements in recognizing both seen and unseen/novel categories.

1.3.3 Contribution 3: Context-Aware Multi-modal Feature Alignment for Efficiently Understanding OOD Scenarios

In the final part of this thesis (Chapter 5), we address the most sophisticated alignment challenge: enabling VLMs to effectively understand OOD scenarios guided by natural language instructions. Multi-modal large language models (MLLMs, as a class of generative VLMs) offer strong generalization capabilities that make them particularly suitable for understanding unfamiliar situations, but their substantial computational requirements limit deployment in resource-constrained environments where OOD scenarios are most likely to occur (*e.g.*, edge devices in autonomous vehicles, portable medical diagnostic systems). A major obstacle is that high-resolution images generate numerous visual tokens that overwhelm the LLM backbone's context capacity.

We observe that existing token reduction methods [19, 118, 122, 178, 247] discard visual information without considering instruction relevance, while mixture-of-experts (MoE) methods [45, 48, 192, 216] select experts based solely on hidden states, ignoring rich multi-modal context. To address these limitations, we propose a context-aware framework that establishes fine-grained correspondence between understanding instructions and visual features. Our approach enables adaptive visual token selection based on instruction semantics and context-aware expert routing that leverages both visual and textual information. This results in more efficient VLMs that maintain strong understanding performance while significantly reducing computational costs, enabling practical deployment for OOD understanding in real-world applications.

1.4 Thesis Organization

The remainder of this thesis is organized as follows:

- **Chapter 2** provides comprehensive background on VLMs and reviews related work for

each of the three research problems: OOD detection, 3D open-vocabulary semantic segmentation, and visual understanding.

- **Chapter 3** presents our multi-modal prototype learning approach for OOD detection, including detailed methodology, implementation, and experimental evaluation.
- **Chapter 4** describes our improved multi-modal feature alignment framework for 3D open-vocabulary semantic segmentation, enabling recognition of novel 3D object categories.
- **Chapter 5** details our context-aware approach for efficiently understanding OOD scenarios, showing how instruction-guided visual alignment enables effective understanding with reduced computational overhead.
- **Chapter 6** concludes the thesis with a summary of contributions, discussion of limitations, and directions for future research.

Chapter 2

Background

This chapter provides the necessary background and reviews related work for the three research problems addressed in this thesis: out-of-distribution (OOD) detection, 3D open-vocabulary semantic segmentation, and efficient MLLMs (a subclass of generative VLMs). As discussed in Chapter 1, these three problems represent different levels of OOD handling—from binary detection, through semantic recognition, to understanding—and share a common technical foundation in multi-modal feature alignment.

We begin with an introduction to vision-language models (Section 2.1), covering pretrained language models, discriminative VLMs, generative VLMs, and efficient adaptation and scaling techniques, which serve as the foundation for all three projects. We then review the background and related work for each research problem in Section 2.2, Section 2.3, and Section 2.4, respectively. Finally, we summarize the key limitations of existing methods and preview how our proposed approaches address these challenges (Section 2.5).

2.1 Vision-Language Models

2.1.1 Pretrained Language Models

Early language models, such as Word2Vec [157], BERT [40], and GPT [166], have achieved remarkable success across a wide range of natural language processing tasks [126]. These models are typically trained on massive text corpora using self-supervised objectives. For example, Word2Vec uses skip-gram or continuous bag of words (CBOW) to predict context words given a target word, while GPT uses next-token prediction to predict the next word in a sequence.

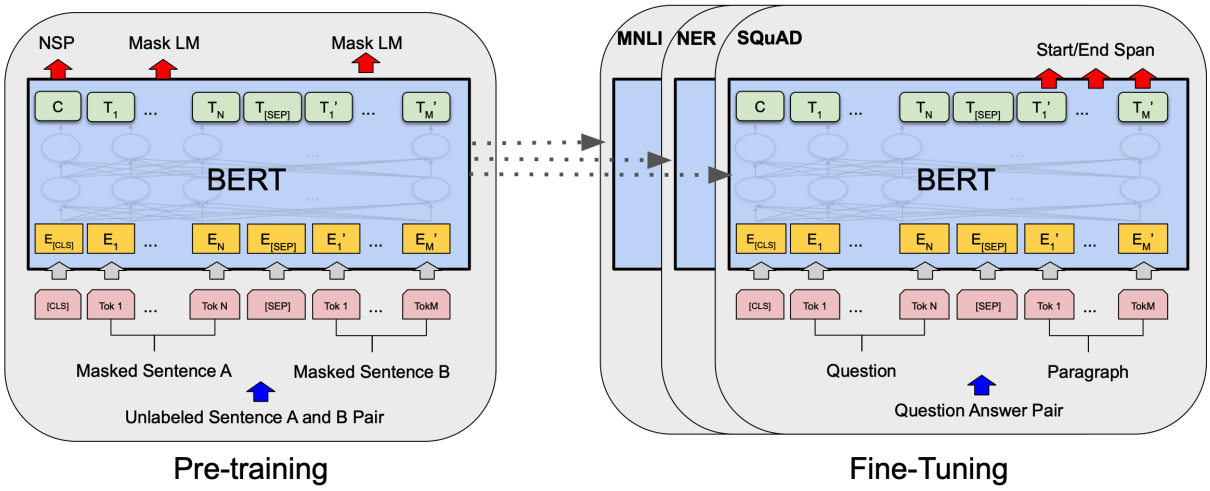


Figure 2.1: BERT architecture and training paradigm. The same bidirectional Transformer encoder is first pretrained with self-supervised language objectives and then fine-tuned for downstream tasks by adding task-specific output layers. The `[CLS]` token is inserted at the beginning of each input sequence and its final representation is commonly used for sentence-level prediction, while `[SEP]` marks the boundary between text segments such as a question and an answer. The figure is taken from [40].

A representative work, BERT [40], is shown in Figure 2.1, where the same architecture is used in both pre-training and fine-tuning, and the same pre-trained model parameters are used to initialize models for different downstream tasks. BERT employs a bidirectional transformer architecture [197] to capture the context of words in a sentence and generate a contextualized representation for each word. During pretraining, BERT uses masked language modeling to predict masked words in a sentence.

Following the success of early language models, pretrained large language models (LLMs), such as LLaMA [194] and GPT-4 [166], have achieved remarkable success by scaling up the model size and training data. Different from early language models, which mainly use bidirectional transformers (encoder-only transformers), LLMs are typically decoder-only or encoder-decoder architectures, where the encoder is used to encode the input text into a sequence of tokens, and the decoder is used to generate the output text. For example, GPT-4 and LLaMA are decoder-only transformer models. Researchers have shown that by simply scaling up the model size and training data, LLMs can learn from predicting the next token in a sequence, achieving strong capabilities in various NLP tasks with strong generalization ability to new tasks and domains with minimal efforts such as prompt engineering [96, 119, 181].

Despite their success, LLMs face two fundamental limitations: (i) the limited availability of high-quality text data [198], and (ii) the inherent limitations of single-modality architectures that restrict their ability to capture and process real-world information requiring understanding of complex relationships between different modalities, such as vision and language [55, 84, 139]. This motivates the development of vision-language models (VLMs) that combine visual and textual inputs to provide a more comprehensive understanding of visual spatial relationships, objects, scenes, and abstract concepts.

2.1.2 Discriminative Vision-Language Models

To address the limitations of unimodal models, researchers have developed VLMs [120, 174, 208] that combine visual and textual inputs to provide a more comprehensive understanding of visual spatial relationships, objects, scenes, and abstract concepts.

Early VLMs, such as CLIP [174] and ALIGN [88], following the trace of language models, focus on learning to associate images and text by maximizing the similarity between their representations in a shared embedding space using contrastive learning [22]. Specifically, given a batch of N image-text pairs $\{(I_i, T_i)\}_{i=1}^{N}$, contrastive learning encourages matched pairs to have high similarity while pushing unmatched pairs apart, as shown in Figure 2.2. The loss of contrastive learning is typically defined as:

$$\mathcal{L}_{\text{contrastive}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s(\mathbf{x}_i, \mathbf{t}_i)/\tau)}{\sum_{j=1}^{N}\exp(s(\mathbf{x}_i, \mathbf{t}_j)/\tau)}, \tag{2.1}$$

where $\mathbf{x}_i$ and $\mathbf{t}_i$ are the image and text embeddings of the i-th pair, $s(\cdot,\cdot)$ is the similarity function, τ is the temperature parameter, and N is the number of pairs in the batch. The similarity function is typically defined as the cosine similarity:

$$s(\mathbf{x}, \mathbf{t}) = \frac{\mathbf{x}\cdot\mathbf{t}}{\|\mathbf{x}\|\|\mathbf{t}\|}. \tag{2.2}$$

In this thesis, we use *feature alignment* to refer to the objective of learning representations in which semantically matched inputs are close in the shared feature space. Following the formalization of alignment in contrastive representation learning by Wang and Isola [203], feature alignment can be defined as the expected distance between positive pairs:

$$\mathcal{L}_{\text{align}}(f;\alpha) \triangleq \mathbb{E}_{(\mathbf{u},\mathbf{v})\sim p_{\text{pos}}}\left[\|f(\mathbf{u}) - f(\mathbf{v})\|_2^{\alpha}\right], \tag{2.3}$$

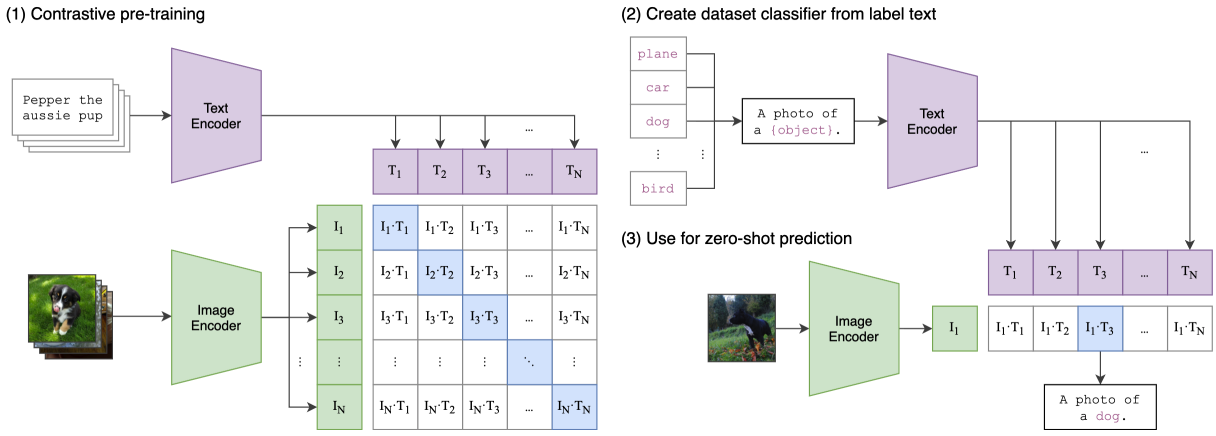


Figure 2.2: CLIP architecture and contrastive training paradigm. A batch of matched image-text pairs is encoded by separate image and text encoders into a shared embedding space. Training increases the similarity of matched image-text pairs along the diagonal of the similarity matrix while decreasing the similarity of mismatched pairs, enabling zero-shot recognition by comparing an image embedding with text embeddings of class-name prompts. This figure is taken from [174].

where p_{pos} denotes the distribution of positive pairs and $\alpha > 0$. For multi-modal feature alignment, this definition can be instantiated with modality-specific encoders as

$$\mathcal{L}_{\text{align}}^{\mathcal{I}\text{-}\mathcal{T}} = \mathbb{E}_{(I,T)\sim p_{\text{pos}}} \left[\| f_{\mathcal{I}}(I) - f_{\mathcal{T}}(T) \|_2^2 \right], \tag{2.4}$$

where $f_{\mathcal{I}}$ and $f_{\mathcal{T}}$ encode image and text inputs, respectively. This simple yet effective approach has enabled strong performance in various tasks, including zero-shot image classification [39] and image-text retrieval [131].

A pitfall: modality gap in discriminative VLMs. Despite the success of discriminative VLMs, Liang *et al.* [128] demonstrated that CLIP's image and text embeddings form two systematically separated distributions in the shared embedding space, a phenomenon known as the *modality gap*, as shown in Figure 2.3. That is, even semantically paired images and texts may remain separated by modality-dependent structure in the embedding space. Following this observation, this thesis uses the term *modality gap* to denote the separation between embedding distributions produced by different modality encoders. Operationally, given image embeddings $\{\mathbf{x}_i\}_{i=1}^N$ and text embeddings $\{\mathbf{t}_i\}_{i=1}^N$, we quantify the magnitude of this gap by the distance between modality centroids:

$$\Delta_{\text{mod}} = \left\| \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i - \frac{1}{N} \sum_{i=1}^{N} \mathbf{t}_i \right\|_2. \tag{2.5}$$

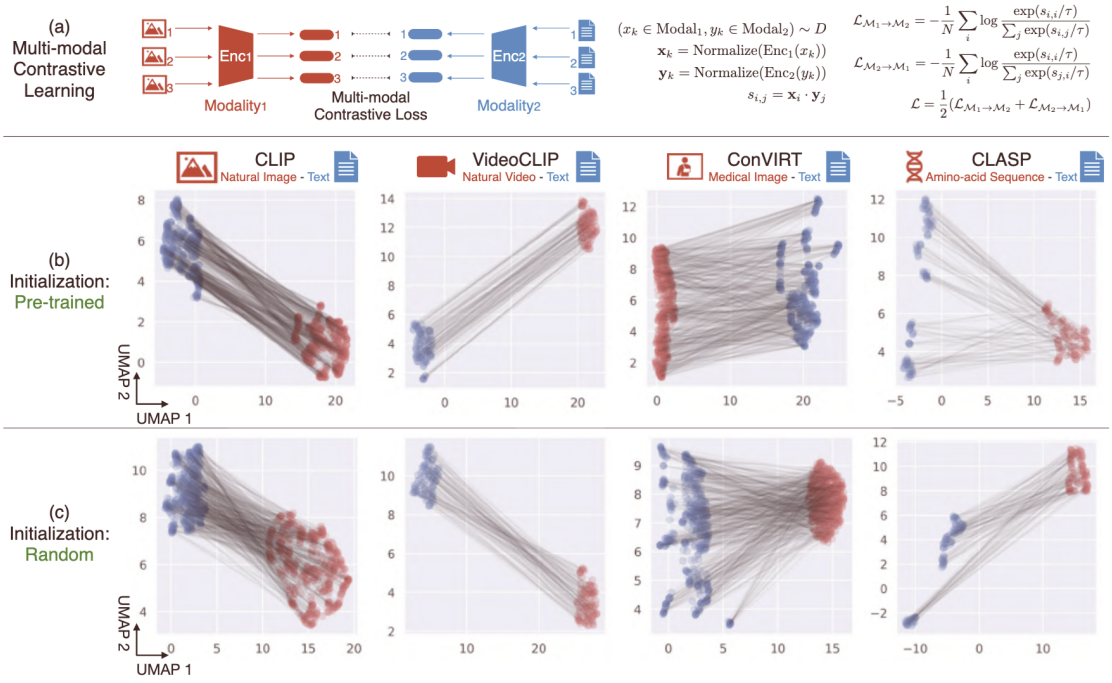


Figure 2.3: The pervasive modality gap in multi-modal contrastive representation learning. (a) Overview of multi-modal contrastive learning. Paired inputs from two modalities (*e.g.*, image-caption) are sampled from the dataset and embedded into the hypersphere using two different encoders. The loss function is to maximize the cosine similarity between matched pairs given all the pairs within the same batch. (b) UMAP visualization of generated embeddings from pre-trained models. Paired inputs are fed into the pre-trained models and the embeddings are visualized in 2D using UMAP (lines indicate pairs). The visualization shows a clear modality gap for various models trained on different modalities. (c) UMAP visualization of generated embeddings from the same architectures with random weights. Modality gap exists in the initialization stage without any training. This figure is taken from [128].

This gap arises from two primary factors: (i) model initialization, where the random initialization of separate encoders places embeddings in distinct regions, and (ii) the contrastive learning optimization process, which only requires matched pairs to be closer than unmatched pairs within a batch, without enforcing that image and text embeddings of the same concept occupy the same location.

Importantly, a smaller modality gap is not universally preferable. Liang *et al.* [128] show that the desirable gap size depends on the downstream objective: in their embedding-shift exper-

iments, increasing the gap improves zero-shot performance on some datasets, while decreasing it improves performance on others. Larger gaps can be useful when modality-specific structure or calibration should be preserved, whereas smaller gaps are useful when the task relies on direct cross-modal comparability. Therefore, this thesis does not aim to eliminate the modality gap. Instead, for the problems studied here, reducing an excessive or poorly calibrated gap is desirable because the tasks rely on comparing, transferring, or conditioning information across modalities. In OOD detection, a better-calibrated image-text gap makes image embeddings and text prototypes more semantically comparable. In 3D open-vocabulary segmentation, reducing excessive separation between 3D features and language features helps transfer language supervision to point-cloud representations. In efficient MLLMs, improving the alignment between visual tokens and textual instructions helps preserve instruction-relevant visual information during token reduction and expert routing. Thus, the desirable amount of gap reduction should be determined by downstream validation rather than by targeting a zero gap.

The modality gap has profound implications for downstream tasks addressed in this thesis:

- **OOD detection (Chapter 3):** OOD images may exhibit high similarity to in-distribution (ID) text prototypes due to spatial proximity rather than true semantic alignment, leading to unreliable detection and increased false positive rates.
- **3D semantic segmentation (Chapter 4):** When transferring knowledge from 2D VLMs to 3D models, excessive cross-modal separation compounds with the domain gap between 2D images and 3D point clouds, making precise alignment more challenging.
- **Visual understanding (Chapter 5):** Poorly calibrated separation between visual and textual representations limits the ability of generative VLMs to establish fine-grained correspondence between instruction semantics and visual features.

2.1.3 Generative Vision-Language Models

Following the success of discriminative VLMs, several works have explored the use of VLMs for more complex tasks requiring generation capabilities, such as visual question answering [84, 155], image captioning [131], and visual grounding [172]. These generative VLMs[1], also known as multimodal large language models (MLLMs), combine vision encoders [52, 100, 174] with

[1]In this thesis, while image/video generation models given text instructions are also referred to as generative VLMs, we only investigate MLLMs (a subclass of generative VLMs) that are specifically designed for multi-modal understanding tasks.

LLMs [28, 145] to enable cross-modal understanding through modality fusion and instruction tuning [38, 125, 130, 133, 134]. MLLMs typically consist of three main components:

Vision Encoder. The vision encoder (*e.g.*, CLIP [174], EVA [52]) extracts visual features from input images. High-resolution images are often divided into patches and encoded into visual tokens. For example, CLIP-L [174] encodes a single image into $24 \times 24 = 576$ visual tokens.

Vision-Language Projector. The projector aligns visual features with the LLM's input space. Common designs include linear projections, MLPs, or more sophisticated cross-attention mechanisms.

Large Language Model. The LLM backbone (*e.g.*, LLaMA [145], Vicuna [28]) takes the projected visual tokens along with text instruction tokens and generates responses autoregressively.

Early generative VLMs, such as BLIP [114] and BLIP-2 [113], employ an attention-based projector/fusion to inject visual features. For example, as shown in Figure 2.4, a Q-Former with learned queries is used to fuse visual features with text features. During pretraining, BLIP-2 introduces three training objectives. Given a batch of image-text pairs $\{(I_i, T_i)\}_{i=1}^{N}$, the Q-Former outputs query representations $\mathbf{q}_{i,m}$ for image I_i, where m indexes the learned queries, and the text encoder produces a text representation $\mathbf{t}_i$ for caption T_i. The image-text similarity used for contrastive learning can be written as the maximum query-text similarity:

$$s(I_i, T_j) = \max_m \frac{\mathbf{q}_{i,m} \cdot \mathbf{t}_j}{\|\mathbf{q}_{i,m}\| \|\mathbf{t}_j\|}. \tag{2.6}$$

The three objectives are then defined as follows:

1. Image-Text Matching (ITM). This objective encourages the queries to extract visual representations most relevant to the text by predicting whether an image-text pair is matched:

$$\mathcal{L}_{\text{ITM}} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log p_i^{\text{ITM}} + (1 - y_i) \log(1 - p_i^{\text{ITM}}) \right], \tag{2.7}$$

   where $y_i \in \{0, 1\}$ indicates whether the pair is matched and p_i^{ITM} is the predicted matching probability.

2. Image-Grounded Text Generation (ITG). This objective enforces the model to generate the target text autoregressively conditioned on the image:

$$\mathcal{L}_{\text{ITG}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{\ell=1}^{|T_i|} \log p(w_{i,\ell} \mid I_i, w_{i,<\ell}), \tag{2.8}$$

   where $w_{i,\ell}$ denotes the ℓ-th token in caption T_i.

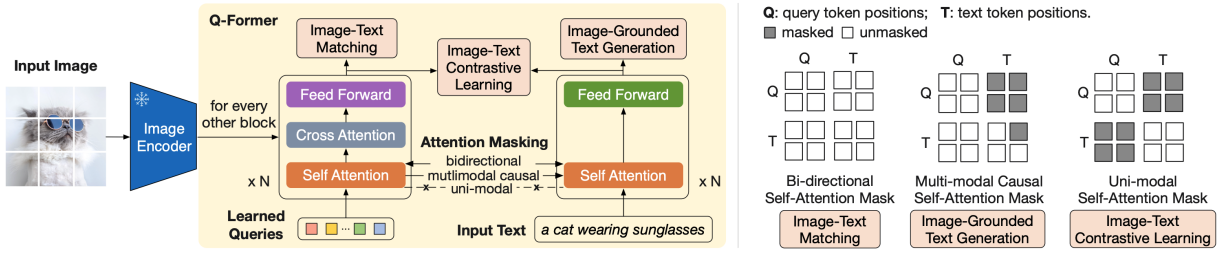


Figure 2.4: BLIP-2 architecture and first-stage vision-language pretraining objectives. The left panel shows how a frozen image encoder produces visual features that are queried by a Q-Former through learnable query embeddings. The Q-Former is trained with image-text contrastive learning, image-text matching, and image-grounded text generation so that the query tokens extract visual information relevant to the paired text. The right panel shows the attention masks used by the different objectives to control whether query tokens and text tokens can attend to each other. The figure is taken from [113].

3. Image-Text Contrastive Learning (ITC). This objective aligns matched image and text embeddings while separating mismatched pairs within the batch:

$$\mathcal{L}_{\text{ITC}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{\exp(s(I_i,T_i)/\tau)}{\sum_{j=1}^{N}\exp(s(I_i,T_j)/\tau)} + \log\frac{\exp(s(I_i,T_i)/\tau)}{\sum_{j=1}^{N}\exp(s(I_j,T_i)/\tau)}\right], \tag{2.9}$$

where τ is a temperature parameter.

The overall BLIP-2 pretraining objective is the sum of these losses:

$$\mathcal{L}_{\text{BLIP2}} = \mathcal{L}_{\text{ITM}} + \mathcal{L}_{\text{ITG}} + \mathcal{L}_{\text{ITC}}. \tag{2.10}$$

However, later researches, such as LLaVA [134], Qwen-VL [9], and InternVL [24], employ a more efficient projector/fusion to inject visual features. Instead of using a attention-based projector, they use a feed-forward network to translate visual features extracted by vision encoders into pseudo text tokens. As shown in Figure 2.5, the pretrained LLM is further finetuned to generate text given the text instructions and visual features mapped by the visual projector (the feed-forward network). This training objective can be formulated as a conditional language modeling loss. Given an image I_i, an instruction X_i, and a target response $Y_i = \{y_{i,1}, \ldots, y_{i,L_i}\}$, the vision encoder f_{v} first extracts visual features, and a feed-forward network g_ϕ maps them into the LLM token space:

$$\mathbf{z}_i = g_\phi(f_{\text{v}}(I_i)). \tag{2.11}$$

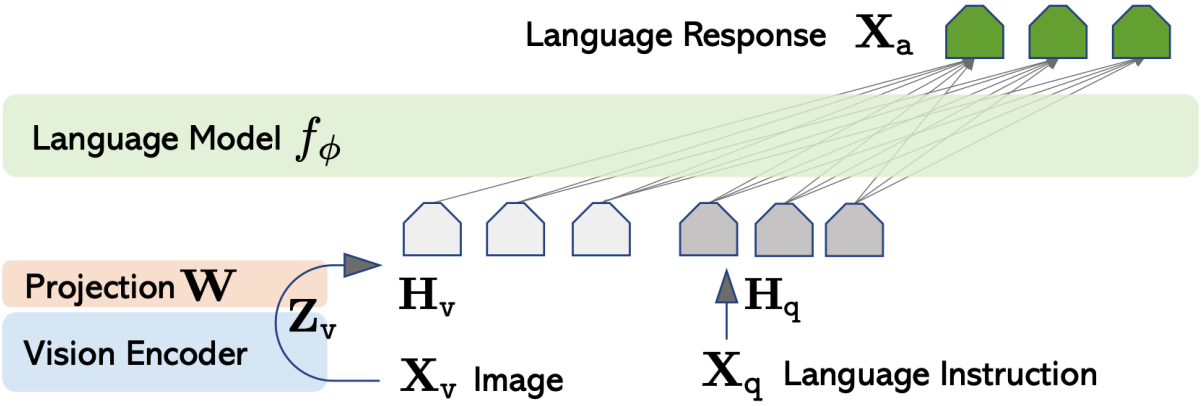


Figure 2.5: LLaVA architecture for visual instruction following. An input image is encoded by a vision encoder, and a trainable projection module maps the resulting visual features into the token embedding space of a language model. The projected visual tokens are concatenated with the language instruction, allowing the language model to generate a natural-language response conditioned on both the image and the instruction. The figure is taken from [135].

The model is then optimized to predict each response token autoregressively conditioned on the projected visual tokens and the text instruction:

$$\mathcal{L}_{\text{MLLM}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{\ell=1}^{L_i}\log p_\theta(y_{i,\ell} \mid \mathbf{z}_i, X_i, y_{i,<\ell}), \tag{2.12}$$

where p_θ is the language model's conditional probability distribution. For caption-style pretraining, X_i can be a simple image description prompt and Y_i is the image caption; for instruction tuning, X_i is the user instruction and Y_i is the desired response.

Empirically, the performance of using feed-forward network to inject visual features is better than the attention-based projector which might be due to the fact that the feed-forward network is more efficient and can capture the fine-grained visual details.

Comparison between discriminative and generative VLMs. While both discriminative and generative VLMs aim to bridge visual and textual modalities, they differ fundamentally in their objectives and capabilities. Discriminative VLMs (*e.g.*, CLIP) excel at computing similarity scores between images and text, enabling zero-shot classification and retrieval tasks. However, they are limited to predefined text templates and cannot generate free-form responses. Generative VLMs (*e.g.*, LLaVA), on the other hand, leverage the autoregressive generation capabilities of LLMs to produce natural language responses conditioned on visual inputs, enabling more

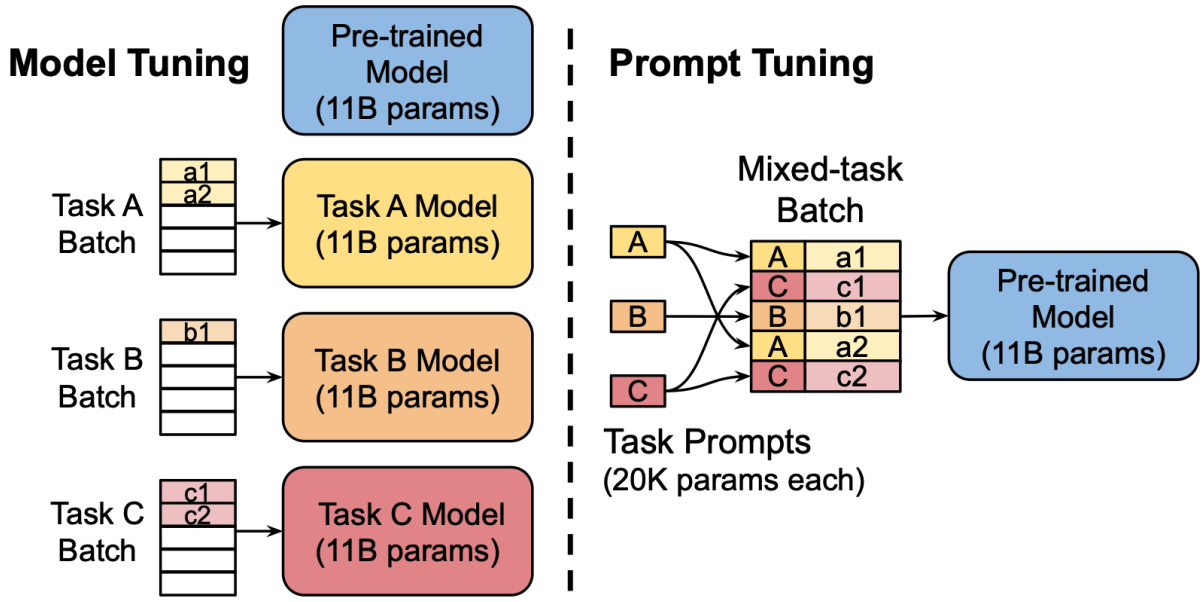


Figure 2.6: Comparison between full model tuning and prompt tuning for adapting a pretrained language model to multiple tasks. Full model tuning creates and stores a separate copy of all pretrained parameters for each downstream task, so different tasks must be served by different model instances. Prompt tuning keeps the pretrained model fixed and stores only a small learned prompt for each task, allowing multiple tasks to share the same backbone during inference. For a T5 "XXL" model, this reduces task-specific storage from 11 billion parameters per tuned model to 20,480 prompt parameters per task when using a prompt length of five tokens. The figure is taken from [105].

flexible tasks such as visual question answering, image captioning, and visual understanding. This flexibility makes generative VLMs particularly suitable for understanding OOD scenarios (Chapter 5), where the system must generate contextually appropriate responses to unfamiliar situations rather than simply classifying inputs into predefined categories. However, the substantial computational requirements of generative VLMs limit their deployment in resource-constrained environments where OOD scenarios are most likely to occur (*e.g.*, edge devices in autonomous vehicles, portable medical diagnostic systems).

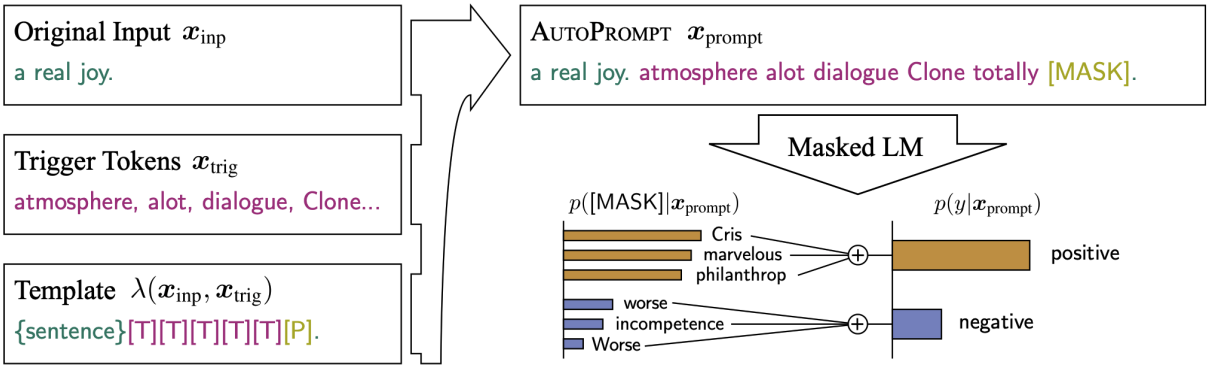


Figure 2.7: AUTOPROMPT procedure for converting a classification task into masked language model inference. Each input text x_{inp} is inserted into a prompt template λ containing learned trigger tokens x_{trig} and one `[MASK]` position. The trigger tokens are shared across examples and selected by a gradient-based search. At inference time, the masked language model predicts candidate words at the `[MASK]` position, and class probabilities are computed by aggregating the probabilities of label words associated with each class, such as positive or negative sentiment. The figure is taken from [181].

2.1.4 Efficient Adaptation and Scaling of VLMs

While VLMs have achieved remarkable success, two key challenges remain: (i) efficiently adapting pretrained VLMs to specific downstream tasks, and (ii) scaling model capacity without proportionally increasing computational costs. This section reviews two complementary lines of research addressing these challenges: parameter-efficient adaptation methods and mixture of experts architectures, which also serve as the foundation of three proposed methods in this thesis.

Parameter-Efficient Adaptation

Adapting VLMs to downstream tasks traditionally requires extensive fine-tuning, which is computationally expensive and may lead to catastrophic forgetting [97]. Parameter-efficient adaptation methods [33] address this by modifying only a small subset of parameters while keeping the pretrained weights frozen.

Prompt Tuning. Prompt tuning originates in natural language processing [105, 119, 181] as a method for automating template/prompt creation in models such as BERT [40] and GPT [166]. By learning continuous prompt embeddings in the input space based on downstream data, prompt

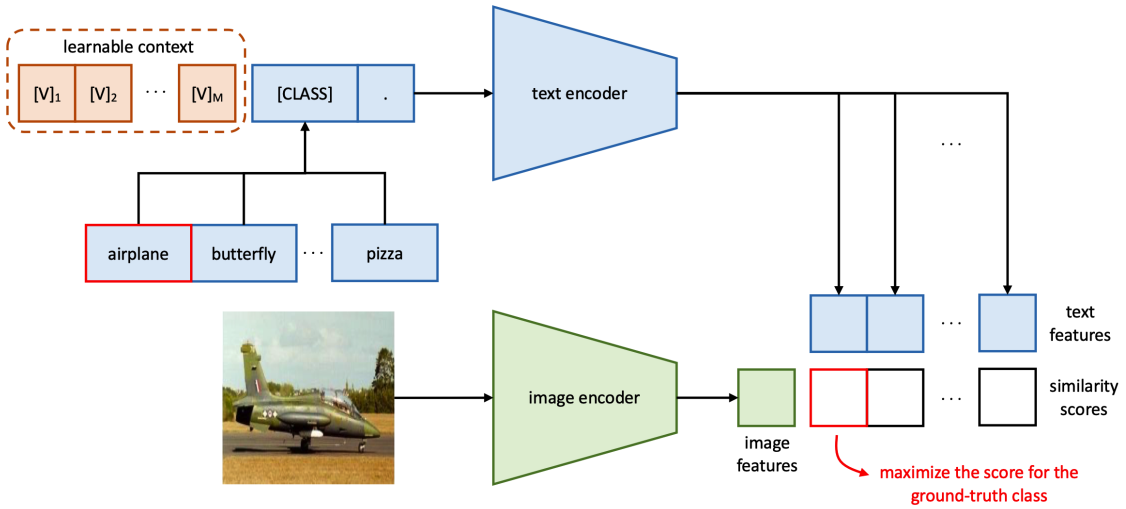


Figure 2.8: Context Optimization (CoOp) for adapting CLIP with learnable text prompts. Instead of using a manually written prompt such as "a photo of a [CLASS]," CoOp replaces the context words with continuous vectors optimized from labeled training examples. The unified-context design shares one set of learnable context vectors across all classes, while the class-specific design learns a separate context for each class before encoding the completed prompts with CLIP's text encoder. The figure is taken from [254].

tuning offers a parameter-efficient alternative to full fine-tuning as shown in Figure 2.6. For example, AutoPrompt [181] is a gradient-based approach for identifying optimal prompts, replacing manually designed templates as shown in Figure 2.7.

Recently, prompt tuning has been applied in computer vision models [97, 253, 254]. Notably, CoOp [254] and CoCoOp [253] employ learnable prompts optimized by minimizing classification loss to improve CLIP's performance. CoOp replaces manually designed prompt templates with learnable prompts optimized by minimizing classification loss as shown in Figure 2.8. However, due to the poor performance on unseen classes (OOD data), CoCoOp is proposed to condition prompts on the image embedding to improve the generalization ability as shown in Figure 2.9. Specifically, CoCoOp extends CoOp by further learning a lightweight neural network to generate for each image an input-conditional token (vector), and yields stronger domain generalization performance as well.

These methods serve as foundations for our prompt-based approaches in both OOD detection (Chapter 3) and 3D semantic segmentation (Chapter 4).

Low-Rank Adaptation (LoRA). Another line of parameter-efficient adaptation methods is low-

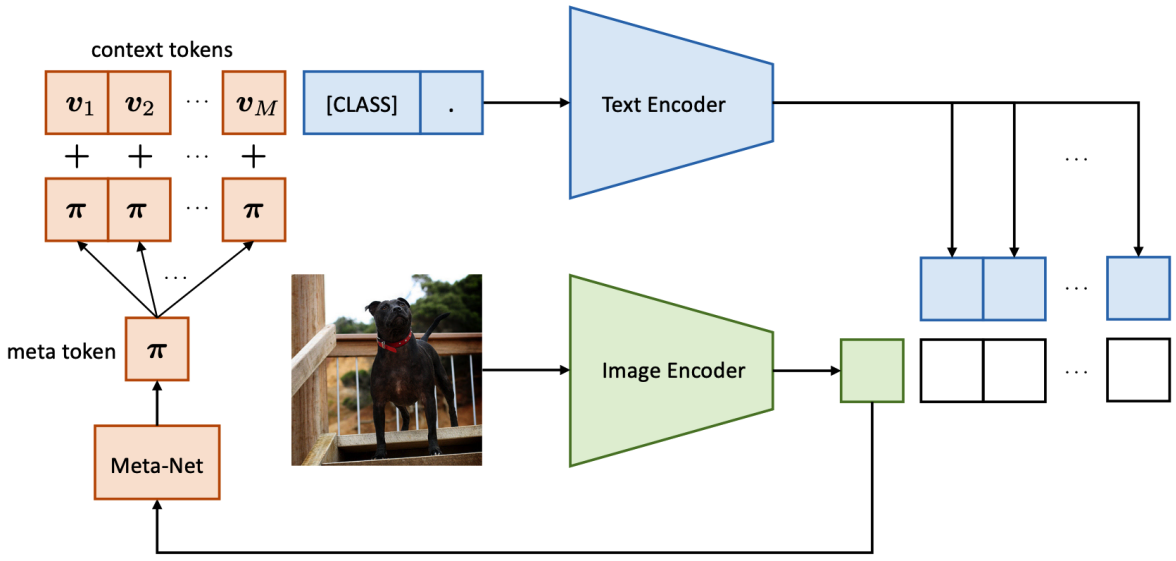


Figure 2.9: Conditional Context Optimization (CoCoOp) for image-conditioned prompt learning. CoCoOp extends CoOp by combining learnable context vectors with a lightweight Meta-Net that maps each image feature to an input-conditional token. The resulting prompt changes with the image being classified, which improves generalization to unseen classes compared with fixed learned prompts. The figure is taken from [253].

rank adaptation (LoRA) [48]. LoRA freezes pretrained model weights and injects trainable low-rank decomposition matrices into transformer layers. For a pretrained weight matrix $W_0 \in \mathbb{R}^{d\times k}$, LoRA represents the weight update as $\Delta W = BA$, where $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, and rank $r \ll \min(d, k)$. This significantly reduces trainable parameters while maintaining performance comparable to full fine-tuning, as shown in Figure 2.10.

Efficient Scaling with High-Resolution Inputs

Low-resolution inputs limit the model's ability to perceive fine-grained visual details [134]. Recent research has incorporated higher-resolution inputs [123, 149] to address this limitation.

For example, Mini-Gemini [123] incorporates higher-resolution inputs and visual encoders to address this limitation, as shown in Figure 2.11. However, the curse of high-resolution inputs is the increase in the number of visual tokens, which leads to higher computational costs. Specifically, in generative VLMs, the input images are divided into patches and encoded into visual

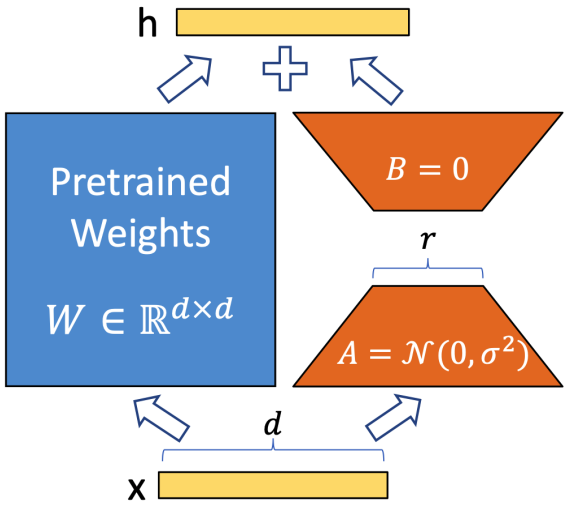


Figure 2.10: Low-Rank Adaptation (LoRA) reparameterization of a pretrained weight matrix. The original weight W_0 is frozen, and the task-specific update ΔW is represented as the product of two low-rank matrices, BA. Only A and B are trained, so adaptation changes the model through a small number of additional parameters while preserving the pretrained backbone. The figure is taken from [48].

tokens. Higher-resolution inputs result in more visual tokens, which leads to higher computational costs, as the LLM has to process more tokens.

This observation motivates the third project (chapter 5) of this thesis, which aims to reduce the number of visual tokens by looking at the patches that are most informative and most relevant to the input instructions.

However, employing multiple vision experts increases the number of visual tokens, leading to higher computational costs, which motivates efficient token reduction methods discussed in Section 2.4.

Efficient Scaling with Mixture of Experts

Mixture of Experts (MoE) is a model architecture that employs multiple specialized expert networks to process different parts of the input space [86]. Usually, a gating network (router) determines which experts to activate for each input, enabling efficient scaling by computing only a sparse subset of experts. This allows models to increase capacity without proportionally increasing computational costs during inference.

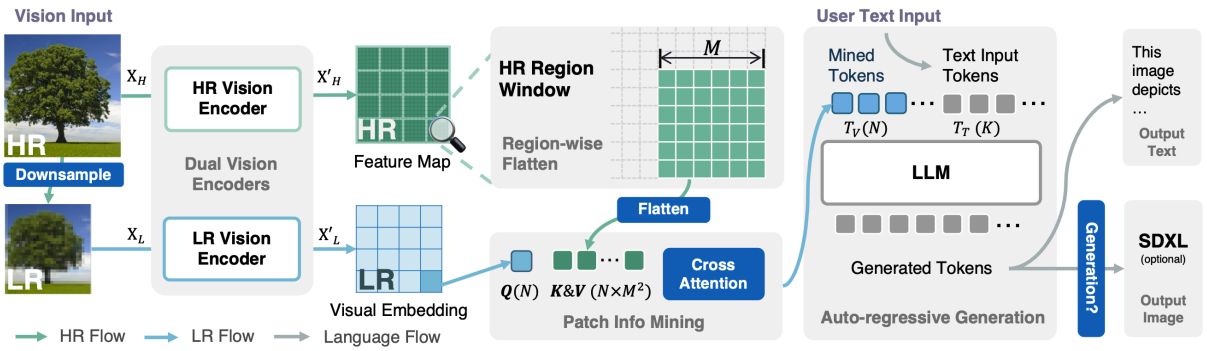


Figure 2.11: Mini-Gemini framework for improving fine-grained visual understanding in multi-modal large language models. The model supplements a standard low-resolution visual stream with higher-resolution image inputs and additional visual encoders so that small objects and local details can be preserved before visual features are passed to the language model. The design improves perception but also increases the number of visual tokens that later stages must process. The figure is taken from [123].

Formally, a MoE layer is defined as:

$$\text{MoE}(x) = \sum_{i=1}^{E} \alpha_i f_i(x), \tag{2.13}$$

where x is the input, E is the number of experts, α_i is the routing probability of the i-th expert, and $f_i(x)$ is the output of the i-th expert. The routing probability is computed by a gating network (router) that takes the input x as input.

MoE in LLMs. Early works [46, 53] demonstrated that sparse expert activation improves scalability and computational efficiency in language models. Modern MoE-based LLMs [37, 90] incorporate MoE by replacing standard feed-forward networks with MoE layers, where each token is routed to a small subset of experts.

LoRA-based MoE. Recent research [129,132,214,236] explores integrating MoE with LoRA [48] to combine the benefits of both approaches. These methods leverage LoRA's parameter efficiency to maintain multiple experts [21, 45, 216], enabling dynamic task adaptation without the computational cost of full-sized experts. A representative example is MoE-LLAVA [129], which incorporates MoE with LLaVA [134] to improve the efficiency of the model. Figure 2.12 shows the overview of the MoE-LLAVA framework. This paradigm serves as the foundation for our efficient MLLM architecture (Chapter 5).

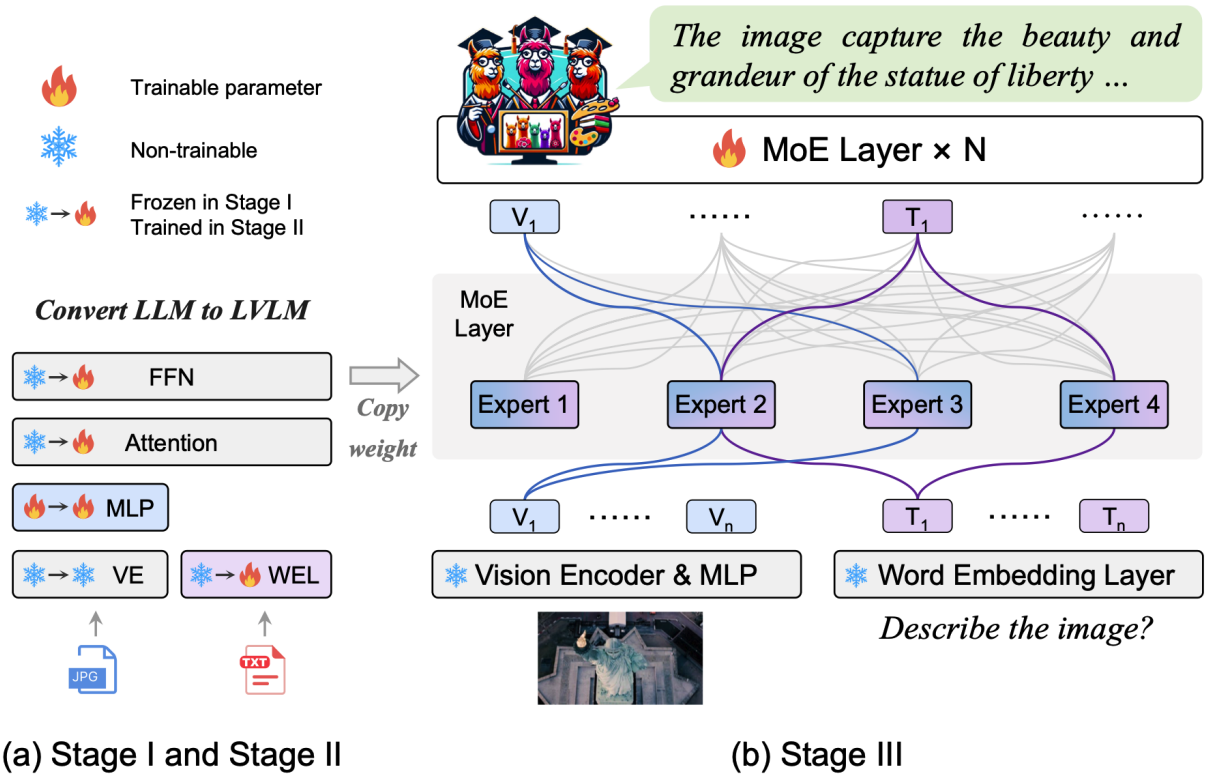


Figure 2.12: MoE-LLAVA framework for parameter-efficient multimodal instruction tuning. The model builds on LLaVA by inserting sparse mixture-of-experts modules, such as LoRA-style experts, into the language model. A routing mechanism selects which experts should process each hidden representation, increasing task capacity while activating only a subset of parameters for a given input. The figure is taken from [129].

Mixture of Vision Experts. In generative VLMs, a single vision encoder may not capture all aspects of visual information required for diverse tasks, such as autonomous driving [235], medical diagnosis [50], etc. One solution to this problem is to finetune generative VLMs on domain-specific data, which sometimes leads to better performance but at the cost of experiencing overfitting on the domain-specific data and forgetting the general knowledge [241]. Recent works have explored another solution, *i.e.*, mixtures of vision experts [6, 51, 94, 146, 180, 260], where multiple vision encoders [52, 100, 174], each trained on domain-specific data, extract diverse visual features, to address this limitation. These approaches enhance visual understanding by combining complementary representations from different experts [240, 256]. One representative method is LEO [6], which incorporates multiple vision encoders to extract diverse visual features with

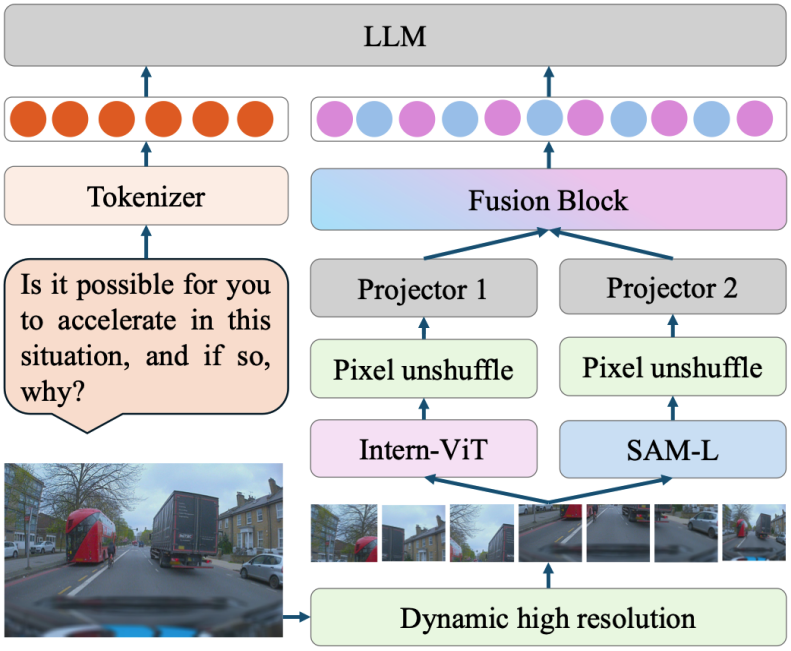


Figure 2.13: LEO framework for combining multiple vision encoders in a multimodal large language model. LEO uses a dual-vision architecture to extract complementary visual features, fuses visual tokens after tile-level adaptation, and applies pixel unshuffle to reduce the number of visual tokens before they are consumed by the language model. The figure illustrates how the architecture balances richer visual perception with token-efficiency constraints. The figure is taken from [6].

dynamic high resolution inputs, as shown in Figure 2.13. LEO also employs pixel unshuffle to decrease the visual token quantity to avoid the curse of high number of visual tokens.

2.2 Out-of-Distribution Detection

2.2.1 Problem Formulation

Out-of-distribution (OOD) detection [72, 79, 138, 227, 243] aims to discriminate in-distribution (ID) samples from OOD samples while maintaining classification accuracy on ID samples. In the real-world deployment of machine learning models, novel samples may emerge and should be flagged for careful consideration.

The literature formalizes OOD detection in several related ways [228]. A common distributional intuition treats the training data as samples from an ID distribution P_{ID} and views OOD detection as deciding whether a test input is sufficiently compatible with this ID distribution, with compatibility operationalized through an ID score [72, 151]. However, this intuition must be interpreted carefully. Strictly speaking, a single observed sample $\mathbf{x}$ does not itself have a distribution, and deciding that it is incompatible with P_{ID} is not well-posed without specifying a class of alternative distributions or an operational rejection rule. If one formulates OOD detection as a hypothesis testing problem, the alternative hypothesis should specify a set of possible non-ID distributions that differ from P_{ID}. For example, for a random test input $\mathbf{X}$ and a chosen class of alternative distributions $\mathcal{P}_{\text{alt}}$, one may write

$$H_0 : \mathbf{X} \sim P_{\text{ID}}, \quad H_1 : \mathbf{X} \sim Q, \quad \text{where } Q \in \mathcal{P}_{\text{alt}}. \tag{2.14}$$

In practice, however, most single-sample OOD detectors do not explicitly model such an alternative class. They instead assign each input an ID confidence, likelihood, typicality, or density-related score, with the convention here that lower scores indicate lower ID compatibility, and reject the input when this score falls below a threshold. When the ID density is available, this corresponds to a density level-set rule that declares $\mathbf{x}$ OOD when $p_{\text{ID}}(\mathbf{x})$ is below a threshold. A stronger support-based definition can be viewed as the limiting case, where a sample is regarded as OOD if it lies outside the support of the ID distribution, *i.e.*, $\mathbf{x} \notin \text{supp}(P_{\text{ID}})$. This definition is clean in principle but is rarely directly actionable in high-dimensional feature spaces, where the support of P_{ID} is typically inaccessible.

A complementary distinction is between semantic shift and covariate shift: semantic shift changes the object or class semantics, whereas covariate shift changes nuisance factors such as style, background, or acquisition conditions without necessarily changing the semantic label [229]. Purely distributional or support-based framings do not by themselves distinguish these two cases, even though they typically require different treatment: a covariate-shifted sample may still belong to the task and should be classified correctly, whereas a semantically novel sample should be flagged. Therefore, in vision classification benchmarks, OOD detection is commonly operationalized through label-space or semantic novelty: ID samples have labels from a predefined ID label set, while OOD samples come from semantic categories outside this label set [72, 160, 201].

This thesis adopts the label-space/semantic novelty setting rather than treating every covariate-shifted sample as OOD. Formally, let $\mathcal{X}$ denote the feature space and $\mathcal{Y}_{\text{ID}} = \{y_1, \dots, y_C\}$ represent the ID label space with C classes. The training data are sampled from $P_{\text{ID}}(\mathbf{x}, y)$ defined over $\mathcal{X} \times \mathcal{Y}_{\text{ID}}$. At test time, ID samples have labels $y \in \mathcal{Y}_{\text{ID}}$, whereas OOD samples have labels from an unseen label space $\mathcal{Y}_{\text{OOD}}$ such that $\mathcal{Y}_{\text{OOD}} \cap \mathcal{Y}_{\text{ID}} = \emptyset$. Equivalently, the test data are drawn from

either an ID test distribution over $\mathcal{X} \times \mathcal{Y}_{\text{ID}}$ or an OOD test distribution over $\mathcal{X} \times \mathcal{Y}_{\text{OOD}}$. The goal is not to identify the OOD class label, but to learn a binary detector

$$g_\gamma(\mathbf{x}) = \mathbf{1}\left[S(\mathbf{x}) \leq \gamma\right], \tag{2.15}$$

where $S(\cdot)$ is an ID confidence score, γ is a threshold, and $g_\gamma(\mathbf{x}) = 1$ indicates that $\mathbf{x}$ is predicted as OOD. OOD detection is evaluated using held-out ID and OOD test sets, denoted by $\mathcal{D}_{\text{ID}}^{\text{test}}$ and $\mathcal{D}_{\text{OOD}}^{\text{test}}$. In this thesis, we report three standard metrics:

- **AUROC** measures the threshold-independent separability between ID and OOD samples as the area under the receiver operating characteristic (ROC) curve obtained by varying the detection threshold γ.
- **FPR95** measures the false positive rate of OOD samples when the true positive rate on ID samples is fixed at 95%. Let γ_{95} satisfy

  $$\frac{1}{|\mathcal{D}_{\text{ID}}^{\text{test}}|} \sum_{\mathbf{x} \in \mathcal{D}_{\text{ID}}^{\text{test}}} \mathbf{1}\left[S(\mathbf{x}) > \gamma_{95}\right] = 0.95. \tag{2.16}$$

  Then,

  $$\text{FPR95} = \frac{1}{|\mathcal{D}_{\text{OOD}}^{\text{test}}|} \sum_{\mathbf{x} \in \mathcal{D}_{\text{OOD}}^{\text{test}}} \mathbf{1}\left[S(\mathbf{x}) > \gamma_{95}\right]. \tag{2.17}$$

- **ID classification accuracy** measures whether the model preserves its original classification performance on ID samples:

  $$\text{Acc}_{\text{ID}} = \frac{1}{|\mathcal{D}_{\text{ID}}^{\text{test}}|} \sum_{(\mathbf{x}, y) \in \mathcal{D}_{\text{ID}}^{\text{test}}} \mathbf{1}\left[h(\mathbf{x}) = y\right], \tag{2.18}$$

  where $h(\cdot)$ is the ID classifier.

Separately from OOD detection, recent work studies model-specific reliability and *out-of-model-scope* (OMS) detection as related but distinct runtime monitoring problems [5, 62]. Given a task oracle $\Omega(\cdot)$ and a deployed model $f(\cdot)$, the scope of the model is defined as the set of inputs on which the model is correct:

$$\mathcal{D}_f = \{\mathbf{x} \in \mathcal{X} \mid f(\mathbf{x}) = \Omega(\mathbf{x})\}. \tag{2.19}$$

OMS detection aims to reject inputs outside this model-specific scope, *i.e.*, inputs for which $f(\mathbf{x}) \neq \Omega(\mathbf{x})$. Therefore, OOD and OMS detection differ in what they regard as unsafe: OOD detection evaluates whether an input falls outside the adopted ID data or label-space setting, whereas

OMS detection evaluates whether an input will lead to an incorrect prediction by a particular deployed model. This distinction is important because an OOD sample may still be correctly handled by a model, while an ID sample may still cause a model error. In this thesis, we focus on the OOD detection problem in the computer vision domain.

2.2.2 Traditional OOD Detection Methods

Traditional OOD detection methods mainly employ pretrained image models to generate image embeddings and design scores based on the image embeddings. They can be broadly categorized into three groups:

Post-hoc Methods. Post-hoc methods [104, 167, 187, 188] design OOD scores based on the outputs of pre-trained models [66, 174] without additional training. One representative line of post-hoc methods derives OOD scores from the logits. For example, Maximum Softmax Probability (MSP) [72], which uses the maximum softmax probability as the OOD score, based on the observation that ID samples tend to have higher confidence. Another representative method is the Energy-based method [138], which uses the energy score derived from the logits, which provides better theoretical grounding. On the other hand, feature-based methods [104, 188] leverage intermediate feature representations, such as Mahalanobis distance to class-conditional Gaussians.

Confidence Enhancement Methods. These methods [41, 71, 163, 210, 223] aim to improve the model's ability to produce calibrated confidence scores by modifying the training objective or architecture.

Outlier Exposure Methods. Outlier exposure methods [47, 54, 74, 80, 91, 98, 141, 252] leverage auxiliary OOD data during training to improve detection performance. While effective, these methods require access to representative OOD samples, which may not always be available.

However, as traditional OOD methods are limited to the predefined ID classes due to the usage of pretrained image models, the models need to be retrained when meeting new categories/applications. To this end, inspired by the success of VLMs [59, 75, 174], researchers have explored using VLMs to enhance OOD detection.

2.2.3 VLM-based OOD Detection

Recently, the field has shifted towards exploiting large-scale pre-trained VLMs [59, 75, 174], particularly CLIP [174], to enhance OOD detection, transitioning from unimodal to multimodal approaches. One advantage of using VLMs is that they can be easily extended to new categories/applications without extensive fine-tuning.

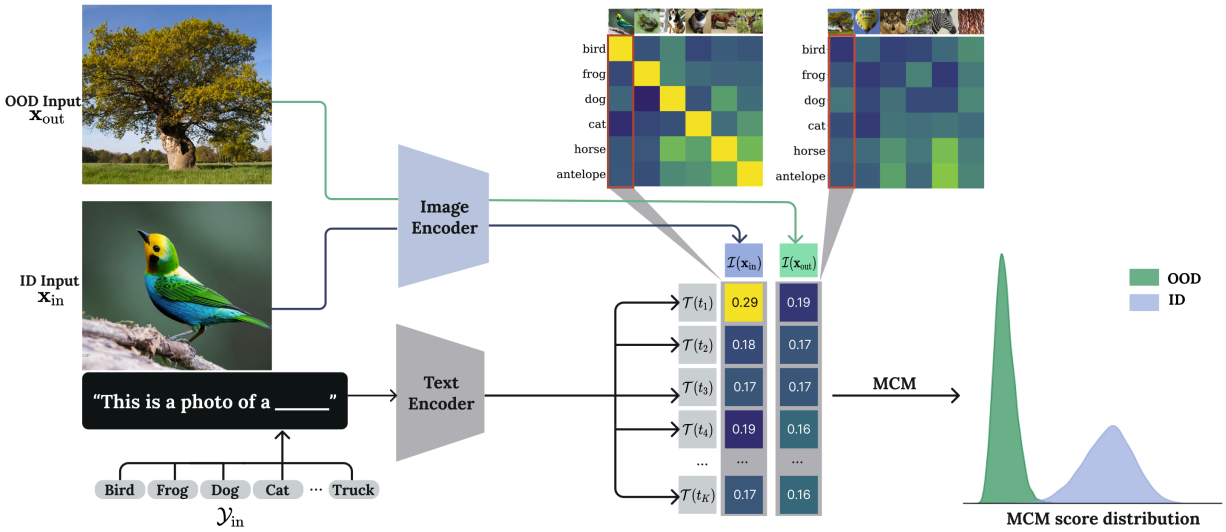


Figure 2.14: Maximum Concept Matching (MCM) framework for zero-shot OOD detection with CLIP [160]. The in-distribution task is specified by a label set $\mathcal{Y}_{\text{ID}}$, whose class names are wrapped in text templates and encoded as text prototypes. A test image is encoded by CLIP's image encoder and compared with all ID text prototypes. If the image has a high scaled similarity to one ID prototype, it is treated as likely in-distribution; if its closest similarity is low, it is treated as uncertain and more likely OOD. The figure is taken from [160].

Current VLM-based OOD detection methods can be broadly categorized into two groups:

Zero-shot Methods. Zero-shot methods mainly focus on leveraging the similarity between image and text embeddings to design OOD scores. Image embeddings are obtained by inputting images into the image encoder of the pretrained VLM. Text embeddings are generated using the ID class names and a set of predefined text prompts, such as "a photo of a *label*". A representative example of zero-shot methods is MCM [160], built upon MSP [72], which employs CLIP and explores the effects of softmax and temperature scaling in a zero-shot manner. Figure 2.14 shows the overview of the MCM framework. Specifically, MCM defines the OOD score as the maximum classification score of the input image,

$$S_{MCM}(I) = \max_{c \in \mathcal{Y}_{\text{ID}}} \frac{\exp(\cos(I, P_{txt,c})/\tau)}{\sum_{j \in \mathcal{Y}_{\text{ID}}} \exp(\cos(I, P_{txt,j})/\tau)}, \tag{2.20}$$

where $P_{txt,c}$ is the text embedding of the c-th ID class, τ is the temperature, and $\exp(\cdot)$ is the exponential function.

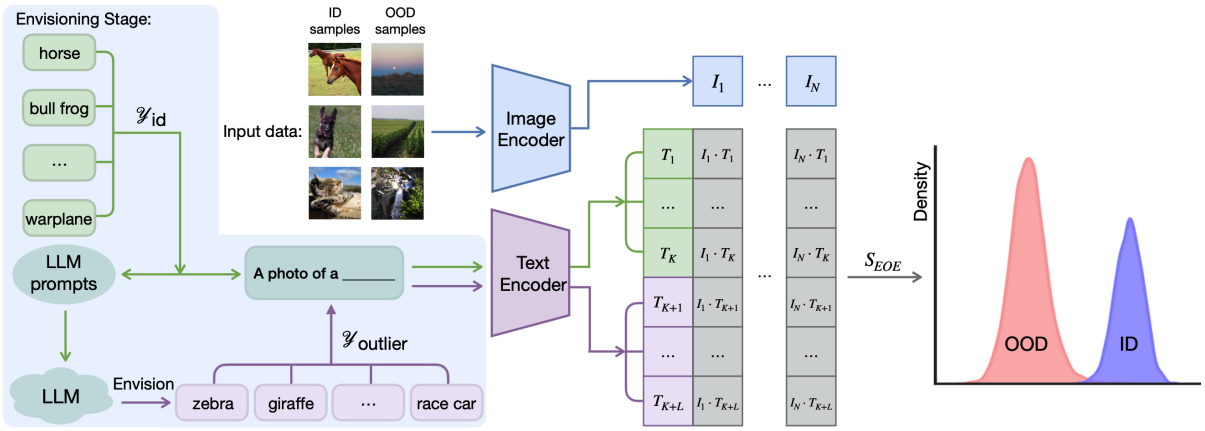


Figure 2.15: Envisioning Outlier Exposure (EOE) framework for zero-shot OOD detection [16]. Starting from the ID label set $\mathcal{Y}_{\text{ID}}$, an LLM is prompted to generate plausible outlier labels $\mathcal{Y}_{\text{OOD}}$ without using real OOD images. Both ID and generated OOD labels are encoded by CLIP's text encoder to form a textual classifier. At test time, the image encoder extracts the visual feature, similarities to ID and generated OOD text prototypes are computed, and the detection score S_{EOE} uses the contrast between these similarities to separate ID from OOD inputs. The figure is taken from [16].

Subsequent zero-shot methods [16, 92, 201] primarily focus on identifying negative (OOD) classes by querying LLMs [96, 166] or WordNet [159]. EOE [16] generates OOD text prototypes by querying LLMs, and then the OOD score is calculated by the similarity difference between ID and OOD text prototypes, as shown in Figure 2.15. EOE first calculates the label-wise matching scores between the input image embedding and the text embeddings of ID and OOD classes, respectively, as follows:

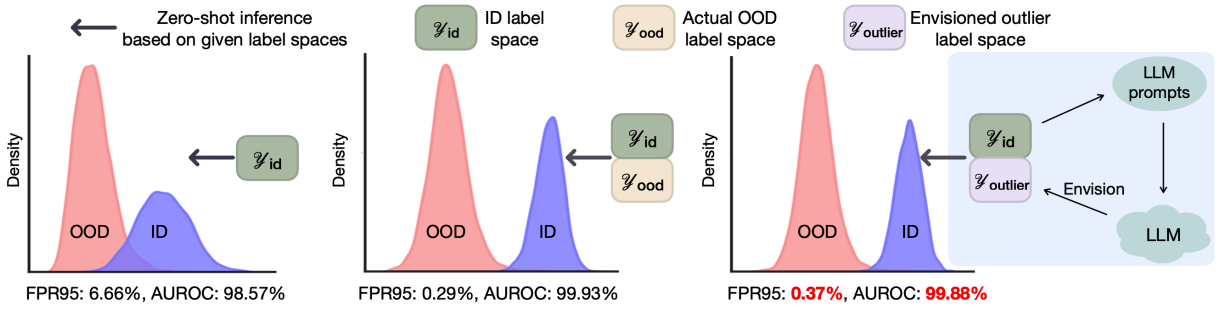
$$s(I, c, \mathcal{Y}) = \frac{\exp(f_{\mathcal{I}}(I) \cdot f_{\mathcal{T}}(c))}{\sum_{y \in \mathcal{Y}} \exp(f_{\mathcal{I}}(I) \cdot f_{\mathcal{T}}(y))}, \tag{2.21}$$

where $f_{\mathcal{I}}(\cdot)$ is the image encoder, $f_{\mathcal{T}}(\cdot)$ is the text encoder, I is the input image, c is a candidate class label, and $\mathcal{Y}$ is the set of all class labels $\mathcal{Y}_{\text{ID}} \cup \mathcal{Y}_{\text{OOD}}$. Then the EOE score is calculated as the difference between the ID and OOD matching scores, as follows:

$$S_{EOE}(I) = \max_{c \in \mathcal{Y}_{\text{ID}}} s(I, c, \mathcal{Y}_{\text{ID}}) - \beta \cdot \max_{c \in \mathcal{Y}_{\text{OOD}}} s(I, c, \mathcal{Y}_{\text{OOD}}), \tag{2.22}$$

where β is a trade-off parameter.

The comparison of zero-shot OOD detection score distribution is shown in Figure 2.16. With the introduction of LLM-generated OOD class labels, the OOD detection performance is signif-



(a) **Closed-Set:** Using only closed-set ID classes

(b) **Ground Truth:** Incorporating with actual OOD classes (unavailable)

(c) **EOE** : Incorporating with envisioned outlier classes

Figure 2.16: Effect of adding OOD text prototypes on zero-shot OOD score distributions. Panel (a) uses only closed-set ID class labels, which can produce overlapping ID and OOD score distributions. Panel (b) adds actual OOD class labels and improves separation, showing the value of explicit negative concepts. Panel (c) replaces actual OOD labels with EOE-generated outlier labels [16], demonstrating that LLM-generated negative concepts can improve OOD detection without access to real OOD labels. The figure is taken from [16].

icantly improved. However, as the vision encoder of VLMs is pretrained on a specific dataset, it may not be capable of capturing the true semantics of novel emerging categories. Thus, researchers started to explore the fine-tuning methods to improve the OOD detection performance.

Few-shot tuning methods. Few-shot tuning methods [10, 116, 136, 164, 248] fine-tune VLMs using a small amount of ID data. These methods mainly target learning negative text prompts (OOD text prototypes). For example, CATEX [136], LSN [164], and NegPrompt [116] learn negative prompts by minimizing the similarity between negative prompts and ID training data. LSN, as shown in Figure 2.17, learns negative prompts with CoCoOp [253] to improve the generalization ability on true OOD data.

Limitations. While these VLM-based methods have shown promise, they rely solely on text prototypes and are thus limited when the image-text modality gap is excessive or poorly calibrated [128]. As a result, OOD images may exhibit high similarity to ID text prototypes due to spatial closeness rather than true semantic similarity, causing increased false positives.

To illustrate this limitation, consider an autonomous driving scenario where the system is trained on common road objects (*e.g.*, cars, pedestrians, traffic signs). When encountering a novel object like a bear, the image embedding of the bear may fall spatially close to some ID text prototypes (*e.g.*, "dog" or "large animal") in the embedding space, not because of true semantic similarity, but due to a poorly calibrated image-text gap. This spatial proximity can cause the OOD detector to incorrectly classify the bear as an ID sample, potentially leading to dangerous

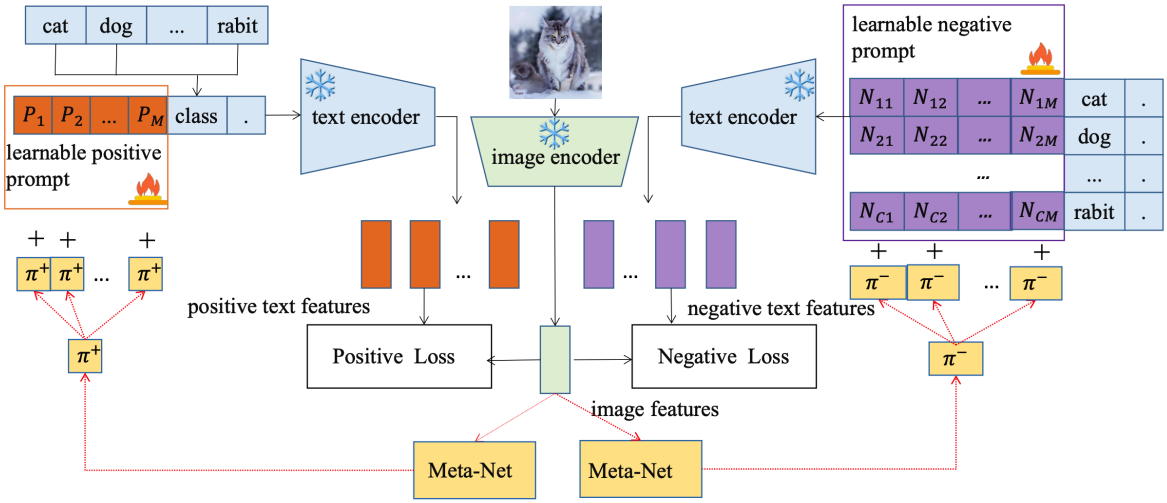


Figure 2.17: LSN framework for few-shot VLM-based OOD detection. The CLIP image encoder and text encoder remain frozen while the method learns positive prompts for ID concepts and negative prompts for OOD concepts from limited ID training data. During inference, a test image is compared with both sets of prompt embeddings, and the relative similarity to positive versus negative prompts is used to determine whether the image should be accepted as ID or rejected as OOD. The figure is taken from [164].

situations. This observation motivates our work on incorporating multi-modal prototypes for OOD detection (Chapter 3).

2.3 3D Semantic Segmentation

2.3.1 Problem Formulation

3D semantic segmentation [69] is a fundamental task in computer vision that involves assigning semantic labels to each point in a 3D point cloud. It plays a crucial role in various applications, such as autonomous driving, robotics, and augmented reality.

Formally, given a 3D scene represented as a point cloud $P = \{\mathbf{p}_i\}_{i=1}^{N}$ where each point $\mathbf{p}_i \in \mathbb{R}^3$ (or $\mathbb{R}^{3+d}$ with additional features), the goal is to predict a semantic label y_i for each point. In the conventional closed-set setting, the training and test label spaces are the same predefined category

set $\mathcal{Y}_{\text{seen}} = \{c_1, \dots, c_C\}$, and the model predicts

$$\hat{y}_i \in \mathcal{Y}_{\text{seen}}, \quad i = 1, \dots, N. \tag{2.23}$$

From closed-set to open-vocabulary segmentation. Traditional 3D semantic segmentation [35, 69, 158, 213, 218, 249] operates in a *closed-set* setting, where models can only recognize categories seen during training (*i.e.*, $\mathcal{Y}_{\text{seen}}$ is fixed and known). However, real-world deployments frequently encounter novel object categories absent from training data—a manifestation of the OOD challenge discussed in Section 1.1. For instance, an autonomous vehicle trained on urban datasets may encounter wildlife on rural roads, or a robotic system may need to manipulate previously unseen objects. The *open-vocabulary* setting addresses this limitation by enabling models to recognize arbitrary categories specified through natural language descriptions, leveraging the semantic knowledge encoded in VLMs. Specifically, at test time the model is given a candidate vocabulary $\mathcal{C}_{\text{test}} = \{c_1, \dots, c_K\}$, which may include both seen and unseen categories:

$$\mathcal{C}_{\text{test}} = \mathcal{Y}_{\text{seen}} \cup \mathcal{Y}_{\text{unseen}}, \quad \mathcal{Y}_{\text{seen}} \cap \mathcal{Y}_{\text{unseen}} = \emptyset. \tag{2.24}$$

Each category name or description c_k is encoded by a text encoder $E_{\text{text}}(\cdot)$ into a semantic embedding $\mathbf{t}_k = E_{\text{text}}(c_k)$. Given a 3D feature extractor F_θ that maps each point to a feature $\mathbf{f}_i = F_\theta(P)_i$, open-vocabulary segmentation predicts each point label by matching 3D point features to the provided text embeddings:

$$\hat{y}_i = \arg\max_{c_k \in \mathcal{C}_{\text{test}}} s(\mathbf{f}_i, \mathbf{t}_k), \tag{2.25}$$

where $s(\cdot, \cdot)$ denotes a similarity function, such as cosine similarity. Thus, open-vocabulary 3D semantic segmentation is not unconstrained class discovery; rather, it is recognition over a candidate label set provided at test time, where this set can contain categories unseen during training.

The standard evaluation metric for 3D semantic segmentation is mean Intersection over Union (mIoU). For a class c, let $\mathcal{P}_c$ be the set of points whose ground-truth label is c, and let $\hat{\mathcal{P}}_c$ be the set of points predicted as c. The class-wise IoU is defined as

$$\text{IoU}_c = \frac{|\mathcal{P}_c \cap \hat{\mathcal{P}}_c|}{|\mathcal{P}_c \cup \hat{\mathcal{P}}_c|}. \tag{2.26}$$

Given a class set $\mathcal{C}$, the mIoU is

$$\text{mIoU}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \text{IoU}_c \,. \tag{2.27}$$

For open-vocabulary evaluation, performance is usually reported separately on seen/base and unseen/novel classes:

$$\text{mIoU}^{\mathcal{B}} = \text{mIoU}(\mathcal{Y}_{\text{seen}}), \quad \text{mIoU}^{\mathcal{N}} = \text{mIoU}(\mathcal{Y}_{\text{unseen}}). \tag{2.28}$$

Because a model can overfit to seen classes while failing on novel classes, the harmonic IoU (hIoU) is commonly used to summarize the balance between seen and unseen performance:

$$\text{hIoU} = \frac{2\,\text{mIoU}^{\mathcal{B}}\,\text{mIoU}^{\mathcal{N}}}{\text{mIoU}^{\mathcal{B}} + \text{mIoU}^{\mathcal{N}}}. \tag{2.29}$$

2.3.2 Deep Learning Methods for Point Clouds

3D semantic segmentation has been developed rapidly through various deep neural network architectures:

Projection-based Methods. These methods [35, 158, 213, 218, 249] project 3D point clouds onto 2D representations (*e.g.*, range images, bird's-eye view) and apply standard 2D convolutional networks. While efficient, projection-based methods may lose 3D geometric information during the projection process.

Point-based Methods. PointNet [173] pioneered direct processing of raw point clouds using shared MLPs and symmetric aggregation functions. Subsequent works [77] extended this approach with hierarchical feature learning and local neighborhood aggregation.

Voxel-based Methods. These methods [29, 259] discretize point clouds into regular 3D voxel grids and apply 3D convolutions. Sparse convolutions [29] significantly improve efficiency by only computing on occupied voxels.

Hybrid Methods. Hybrid approaches [26, 143, 189, 220] combine multiple representations to leverage their complementary strengths.

2.3.3 3D Open-Vocabulary Recognition

While vision-language models (VLMs) [100, 114, 147, 174, 183, 208] have achieved remarkable results in zero-shot or few-shot learning of 2D images by utilizing web-scale image-text data [14, 39, 131], the availability of such large-scale data for 3D point clouds is limited. To extend this zero-shot ability with VLMs in 3D point clouds, pioneering work [246, 258] has been working on converting point clouds into CLIP-recognizable images and aligning the point cloud features with the language features. Following these works, CLIP2Point [82] proposes to align the projected depth map with CLIP's image space by a trainable depth encoder. Another line of research [43, 137, 225, 226, 239] focuses on collecting a large amount of point cloud-text-(image) data and training multimodal models on the collected data. This line of research focuses on aligning the 3D features with text and image features within CLIP's representation space, furthes with CLIP, and further enhancing both the zero-shot and standard 3D recognition capability.

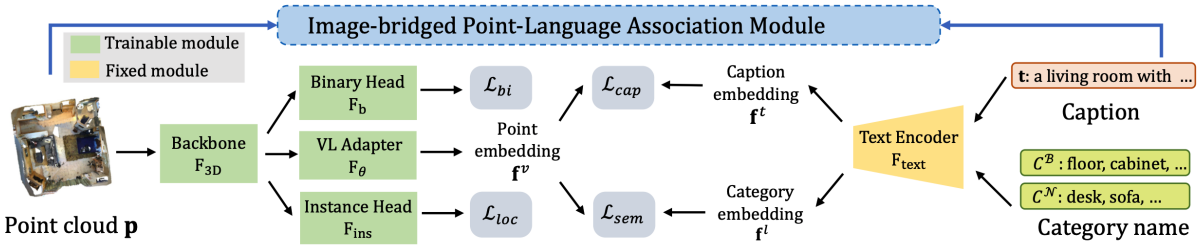


Figure 2.18: PLA framework for open-vocabulary 3D semantic segmentation. PLA transfers semantic knowledge from vision-language models to 3D point clouds by generating image captions for 3D scenes and using them as language supervision. Compared with a closed-set segmentation network, PLA replaces the fixed learnable semantic classifier with text-encoder category embeddings derived from class names, adds a binary head to calibrate base and novel class probabilities, and uses an instance head for instance-level segmentation. Point-language association further supervises point embeddings with caption embeddings so that the 3D model can recognize categories beyond the closed training label set. The figure is adapted from [43].

2.3.4 Open-Vocabulary 3D Semantic Segmentation

Traditional 3D semantic segmentation methods operate in a closed-set setting, where models can only recognize categories seen during training. To handle real-world scenarios where novel categories frequently emerge, researchers have explored the open-vocabulary setting [43,67,226], which requires models to recognize novel classes beyond the training label space.

Zero-shot Methods. Early attempts [1,27,68,156,209,230] extend effective 2D zero-shot methods [7,12] into 3D scenarios. 3DGenZ [156] shows that generated data can boost the performance of zero-shot segmentation. QGPA [68] employs a prototype-based method to enable efficient knowledge transfer between seen and unseen categories.

VLM-based Methods. Inspired by advances in 2D open-vocabulary segmentation [18, 44, 58, 165, 244, 257], recent works transfer knowledge from pretrained 2D VLMs to 3D models. A pioneer work, PLA [43], first proposed distilling knowledge encoded in VLMs using generated 3D-scene-image-to-caption data, allowing explicit associations between 3D point clouds and captions, as shown in Figure 2.18. Specifically, PLA employs a learnable semantic head to rectify the semantic scores with base and novel probability as conditions. Base class names will be leveraged to generate the corresponding class embeddings, which are then used to supervise the point embeddings with point-language association, while novel class names are used during inference to predict the novel categories.

Following PLA, RegionPLC [226] boosts performance by using dense object-level associ-

ations for better alignment. Most recent works [42, 43, 67, 171, 226] focus on improving the alignment between 3D and text representations using generated 3D-scene-image-to-text pair data from off-the-shelf image captioning methods [34, 117, 202].

Limitations. While VLM-based methods have shown promising results, they suffer from two major issues: (i) the generated captions do not accurately match the corresponding image data of 3D scenes semantically, leading to misalignment between 3D models and text encoders; and (ii) text encoders of 2D VLMs are trained on 2D datasets and lack 3D semantics, making them sensitive to prompt templates.

These limitations directly impact the ability to recognize OOD objects in 3D environments. When a 3D perception system encounters a novel object category (*e.g.*, a bear on a rural road, as illustrated in Figure 1.1), it must leverage the semantic knowledge in VLMs to provide meaningful recognition. However, if the 3D-to-text alignment is imprecise due to misaligned training data, or if the text encoder cannot properly interpret 3D spatial context, the system may fail to correctly identify the novel object. These observations motivate our work on aligned vision-language learning for 3D open-vocabulary segmentation (Chapter 4).

2.4 Efficient Generative VLMs

As introduced in Section 2.1, generative VLMs (MLLMs) [6, 24, 38, 133, 134, 146, 180, 247] combine vision encoders with LLMs to enable cross-modal understanding. While recent advances such as higher-resolution inputs [123, 149] and mixture of vision experts (Section 2.1) have improved visual understanding, they significantly increase the number of visual tokens, leading to efficiency challenges.

The efficiency of MLLMs is constrained by the LLM backbone's context length, as high-resolution images generate numerous vision tokens that quickly consume available space and increase computational cost. While vision encoders are relatively lightweight (*e.g.*, CLIP-L [174] has 0.3B parameters), the visual tokens they produce significantly increase the computational overhead of the LLM backbone (*e.g.*, LLaMA-8B [145]). This becomes more challenging in high-resolution image understanding [6, 24] or video understanding [70, 115, 221], which requires either more visual tokens per image or processing multiple images. Reducing the number of visual tokens is therefore a common strategy for improving efficiency, either with trainable compressors [118, 122, 178, 247] or training-free summarizers [19, 211, 245].

2.4.1 Problem Formulation and Evaluation

For visual understanding with MLLMs, an input consists of visual content I and a text instruction X. A vision encoder maps I into visual tokens, which are projected into the LLM input space and combined with the instruction tokens. The model then generates a response $\hat{Y} = \{\hat{y}_1, \ldots, \hat{y}_L\}$ autoregressively:

$$\hat{Y} = \arg\max_{Y} p_\theta(Y \mid I, X), \tag{2.30}$$

where p_θ is the language model's conditional probability distribution.

Unlike OOD detection and semantic segmentation, visual understanding is not evaluated by a single universal metric. Instead, MLLMs are evaluated on benchmark datasets consisting of image-question-answer triples $\mathcal{D} = \{(I_i, X_i, Y_i)\}_{i=1}^{N}$, where each benchmark defines an answer parser or scoring function $r(\hat{Y}_i, Y_i)$. Representative benchmarks include general perception and reasoning benchmarks [55,107,139,234], VQA benchmarks [65,84], science and diagram benchmarks [95, 148], hallucination evaluation [124], and text-, document-, chart-, and OCR-oriented benchmarks [140, 154, 155, 184]. For classification-style or multiple-choice benchmarks, this is typically accuracy:

$$\text{Acc} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\left[r(\hat{Y}_i, Y_i) = 1\right], \tag{2.31}$$

where $r(\hat{Y}_i, Y_i) = 1$ if the predicted response $\hat{Y}_i$ matches the ground-truth response Y_i, and 0 otherwise.

For benchmarks with task-specific scoring rules, such as MME [55] or OCR-oriented benchmarks [140], we report the official benchmark score. Because this thesis studies efficient MLLMs, evaluation must consider both effectiveness and efficiency. Effectiveness is measured by the benchmark accuracy or score, while efficiency is measured by the number of visual tokens sent to the LLM, denoted N^V. Given an original visual token count N^V_{full} and a reduced token count N^V_{red}, the token compression ratio is

$$\rho_{\text{token}} = \frac{N^V_{\text{red}}}{N^V_{\text{full}}}. \tag{2.32}$$

The desired outcome is to maintain or improve benchmark performance while reducing N^V, thereby lowering the computational cost of visual understanding.

2.4.2 Visual Token Reduction

To address efficiency challenges arisen from the large number of visual tokens, recent methods have focused on reducing the number of visual tokens through both training-free [19,83,211] and

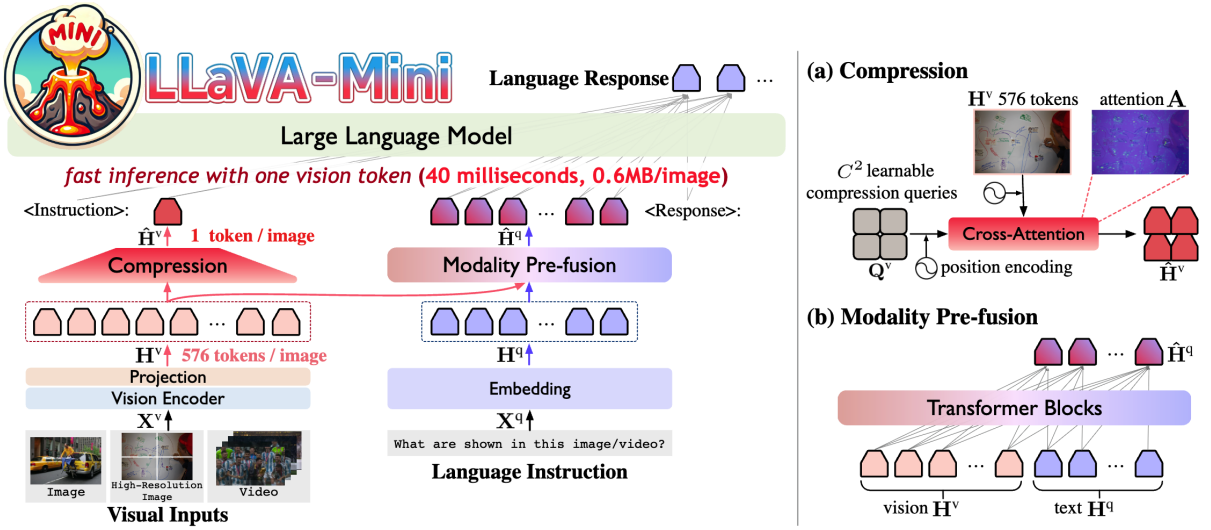


Figure 2.19: LLaVA-Mini framework for reducing visual tokens in multimodal large language models. The left panel shows the high-level idea of representing each image with a compact visual token sequence before sending it to the language model. The right panel details the query-based compression module and modality pre-fusion module, which summarize visual features and fuse them with text-side information so that the model can reduce computation while preserving visual information needed for response generation. The figure is taken from [247].

training-based [19, 122, 178, 247] strategies.

Training-free Methods. These methods [19, 83, 211] reduce tokens at inference time based on attention patterns or token importance scores without additional training.

Training-based Methods. While training-free methods focus on reducing tokens at inference time, training-based methods train a compressor or summarizer to reduce the number of visual tokens during training. Some models aggregate tokens based on visual feature similarities [178] or high-low resolution similarities [118], while others use attention distillation [232]. MQT-LLaVA [78] employs a query transformer to process a random subset of latent query tokens per step. LLaMA-VID [122] integrates text tokens as contextual information and applies average pooling for efficient token reduction. More recently, LLaVA-Mini [247], as shown in Figure 2.19, employs a attention-based token reduction module and a modality pre-fusion module to reduce the number of visual tokens. The attention-based module captures the most informative visual tokens based on the attention maps and the modality pre-fusion module fuses the visual features into text features to avoid the loss of important visual information.

Limitations. Existing methods face two key limitations: (i) token reduction approaches [78, 122, 247] tend to select tokens by learnable queries or saliency maps, potentially losing important visual information; and (ii) LoRA-based MoEs select experts based solely on the hidden state, ignoring the rich information in visual tokens and textual instructions.

These limitations are particularly problematic for OOD understanding scenarios. When an MLLM encounters an unfamiliar situation (*e.g.*, an autonomous vehicle facing an unexpected obstacle), the instruction context becomes crucial for determining which visual information is relevant. A query like "What should the driver do?" requires different visual attention than "Describe the scene." Existing methods that reduce tokens or route experts without considering instruction semantics may discard precisely the visual information needed to understand the OOD scenario. These observations motivate our work on conditional token reduction and context-aware mixture of multi-modal experts (Chapter 5).

2.5 Summary

In this chapter, we have provided the necessary background for understanding the three research problems addressed in this thesis. Vision-language models, particularly CLIP and its derivatives, serve as the foundation for all three projects, enabling the alignment of visual and textual representations. The key insight connecting these problems is that *multi-modal feature alignment* is the fundamental technique required to address OOD challenges at different levels of complexity.

Table 2.1 summarizes the three research problems, their core alignment challenges, and our proposed solutions. Despite their different application domains, all three problems share a common technical foundation: bridging gaps between different modalities to enable effective integration of multi-modal information.

However, existing methods face several limitations that motivate our research:

- **OOD detection (Chapter 3):** An excessive or poorly calibrated modality gap between image and text embeddings leads to unreliable detection when using only text prototypes. OOD images may exhibit high similarity to ID text prototypes due to spatial proximity rather than true semantic alignment, causing increased false positives.
- **3D open-vocabulary segmentation (Chapter 4):** Two key issues hinder precise alignment: (i) generated 3D-scene-image-to-caption data often exhibits semantic misalignment with the actual visual content, and (ii) text encoders trained on 2D data lack understanding of 3D spatial semantics, making them sensitive to prompt templates.

Table 2.1: Summary of the three research problems addressed in this thesis and how they relate to multi-modal feature alignment. OOD detection is framed as binary detection, where image and text representations must be calibrated for reliable ID/OOD separation. Open-vocabulary 3D semantic segmentation is framed as multi-class recognition, where 3D point-cloud features must be aligned with text embeddings for seen and unseen categories. Efficient MLLMs are framed as generative visual understanding, where visual tokens must be selected and routed according to instruction semantics. The table also indicates the modality pair, main technical challenge, alignment goal, and thesis chapter associated with each problem.

Aspect	**OOD Detection**	**3D Open-Vocab Segmentation**	**Efficient MLLMs**
OOD Level	Detection (binary)	Recognition (multi-class)	Understanding (generation)
Modalities	Image ↔ Text	3D Point Cloud ↔ Text	Visual Tokens ↔ Instructions
Core Challenge	Modality gap	3D-text misalignment	Token redundancy & expert routing
Alignment Goal	Calibrate image-text comparability	Transfer 2D VLM to 3D	Instruction-aware visual selection
Chapter	Chapter 3	Chapter 4	Chapter 5

- **Efficient MLLMs (Chapter 5):** Existing token reduction methods discard visual information without considering instruction relevance, potentially losing important visual details. Meanwhile, MoE approaches select experts based solely on hidden states, ignoring the rich context available in both visual tokens and textual instructions.

In the following chapters, we present our proposed methods to address these limitations through improved multi-modal alignment techniques. Each method builds upon the foundational concepts introduced in this chapter while contributing novel solutions to its specific alignment challenge.

Chapter 3

Mitigating the Modality Gap: Few-Shot Out-of-Distribution Detection with Multi-modal Prototypes and Image Bias Estimation

3.1 Introduction

Detecting out-of-distribution (OOD) samples [54, 72, 80, 127, 138, 188, 200] is essential for the real-world deployment of machine learning models [66, 174, 227], as novel samples may emerge and should be flagged for careful consideration. Recently, inspired by the power of vision-language foundation models (VLMs) [59, 75, 174], novel approaches to OOD detection using VLMs [2, 10, 16, 49, 54, 92, 116, 136, 160, 162, 168, 201, 248] have gained significant attention. As reviewed in Sections 2.1 and 2.2, this line of work changes the OOD detection problem from purely modeling visual confidence to asking whether a test image is sufficiently compatible with the semantic concepts specified by language. This distinction matters because VLM-based OOD detection inherits both the strengths and the limitations of contrastive vision-language pretraining: class names provide a flexible semantic interface, but the resulting decision score is computed across image and text embeddings that are not perfectly co-located in the shared space. Therefore, the central background issue for this chapter is not the general definition of OOD detection, which has already been introduced in Section 2.2, but how the geometry of the CLIP embedding space affects the reliability of OOD scores.

Early VLM-based OOD detection works [16, 92, 160, 201] mainly focus on using CLIP [174]

in a zero-shot setting, where only the in-distribution (ID) class names are utilized. For example, Maximum Concept Matching (MCM) [160] measures the similarity between input images and text embeddings of ID classes, also referred to as ID *text prototypes*, in the joint vision-language representation space of CLIP. MCM then uses this similarity score to differentiate OOD and ID samples, on the basis that ID samples should have higher similarity scores.

Advantages of VLMs in OOD Detection. Beyond empirical gains over unimodal baselines [71, 138, 187], VLMs offer four advantages:

1. **Open-vocabulary knowledge.** VLMs [174] encode arbitrary class names as text embeddings, enabling construction of both ID and OOD semantic prototypes [16] without retraining.

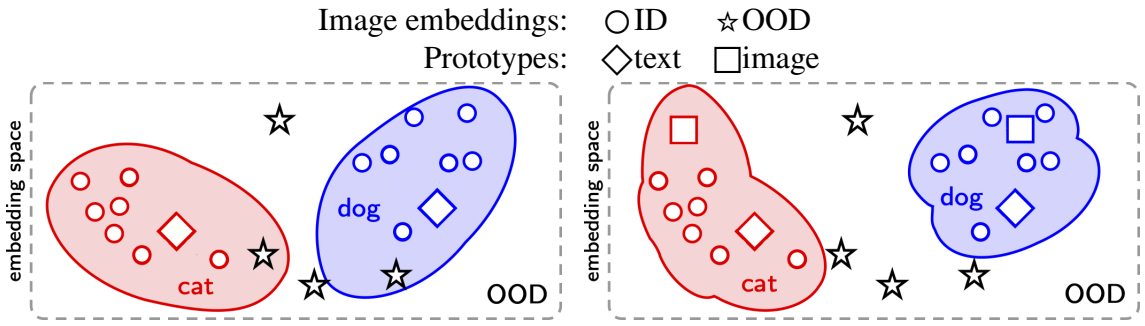
2. **Zero-shot generalization.** Pre-training on large image–text corpora [174] supports deployment across domains with limited labeled data [153, 199].

3. **Semantic grounding.** Compared with vision-only density anomalies [167], VLMs ground visual features in language, providing dual-modal verification.

4. **Hierarchical granularity.** Linguistic priors help separate near-OOD categories with similar appearance [212], such as distinguishing pickup trucks from heavy trucks.

Following MCM, other zero-shot methods [16, 92, 201] aim at generating OOD (negative) text prototypes by querying large language models (LLMs) [96, 166] or WordNet [159]. These methods obtain OOD scores by evaluating the similarity difference between ID and OOD text prototypes. To further improve performance, various few-shot tuning OOD detection methods [10, 116, 136, 164, 248] have been proposed. These methods mainly target learning OOD (negative) text prototypes. For example, NegPrompt [116] learns OOD text prototypes by minimizing the similarity between OOD text prototypes and ID data with contrastive learning.

Insights from prototype errors. Tables 3.1 to 3.3 provide a systematic view of what can happen before applying any explicit alignment module. Table 3.1 first shows that the two prototype modalities have different error profiles in the original CLIP space: text-only scoring produces 1813 OOD false negatives (FNs), image-only scoring produces 2606 FNs, and the combined text-image score reduces this number to 1476 FNs. Since the threshold is chosen at 95% ID true positive rate (TPR) for each setting, the aggregate OOD false positive rate (FPR) is fixed, but the underlying ID samples rejected by each setting differ. Table 3.2 therefore reports sample-level transitions rather than only aggregate rates. The transition counts show that the two modalities are complementary but not monotonically beneficial for every individual sample. For example, using image prototypes instead of text prototypes fixes 757 text-only OOD FNs, but also introduces



(a) Previous methods (b) Our method: SUPREME

Figure 3.1: Comparison between text-only [160] and our multi-modal prototype-based OOD detection. Circles denote ID image embeddings, stars denote OOD image embeddings, diamonds denote ID text prototypes, and squares denote ID image prototypes. (a) Text-only scoring can assign high similarity to an OOD image that lies near an ID text prototype in embedding space, producing an OOD false negative (accepted as ID). (b) Our proposed SUPREME adds ID image prototypes so ID samples are supported by both text and visual anchors, sharpening the ID/OOD boundary.

1550 new OOD FNs. The combined text-image score is more stable: compared with text-only scoring, it fixes 802 OOD FNs while introducing 465; compared with image-only scoring, it fixes 1151 while introducing only 21. This analysis directly exposes both sides of adding image prototypes: they can rescue text-prototype failures, but can also create new OOD FNs when an OOD image is visually close to an ID image prototype. Table 3.3 illustrates these cases with images. The first four rows correspond to the four pre-alignment failure modes: text-only OOD FNs, text-only OOD false positives (FPs), image-only OOD FNs, and image-only OOD FPs, each corrected by the other modality. The final row shows a case where both single-modality scores accept an OOD sample as ID, while the combined score rejects it correctly, indicating that both modalities are needed. These observations are consistent with the modality gap: text prototypes and image prototypes occupy different regions of the CLIP space and therefore fail for different reasons. Text prototypes can over-accept OOD samples due to cross-modal proximity or prompt semantics, whereas image prototypes can over-accept visually similar OOD samples. Multi-modal prototypes reduce the net error because these failure modes are not identical, motivating the later alignment design that further improves cross-modal compatibility.

While these approaches have shown promise using CLIP, they rely solely on text prototypes and are thus constrained by the modality gap [128] between the image and text domains. Liang *et al.* [128] demonstrate that CLIP's image and text embeddings are located in two clearly separate regions of the embedding space. Figure 3.2 illustrates this through three similarity distributions.

Prototype setting	OOD FN: OOD → ID ↓	OOD FP: ID → OOD ↓	Total error ↓	AUROC ↑
Text only	1813 (18.13%)	250 (5%)	2063	96.77
Image only	2606 (26.06%)	250 (5%)	2856	93.09
Text + Image	1476 (14.76%)	250 (5%)	1726	97.36

Table 3.1: Overall pre-alignment OOD detection errors under different prototype settings. This analysis is conducted in the original CLIP embedding space (CLIP-B/16) before applying explicit alignment modules, in order to isolate the effect of prototype modality. OOD FN denotes OOD samples incorrectly predicted as ID, while OOD FP denotes ID samples incorrectly predicted as OOD. Lower values are better. For each prototype setting, the threshold is selected at 95% ID TPR. Therefore, the aggregate OOD FP rate is fixed at 5%, but the identities of rejected ID samples differ across prototype settings; Table 3.2 reports these sample-level error transitions. Specifically, the threshold of text, image, and multimodal prototypes are 0.010747829, 0.011646281, and 0.011249677, respectively.

Target setting	Reference setting	OOD FN fixed ↑	OOD FN introduced ↓	OOD FP fixed ↑	OOD FP introduced ↓
Image only	Text only	757	1550	148	148
Text only	Image only	1550	757	148	148
Text + Image	Text only	802	465	110	110
Text + Image	Image only	1151	21	50	50

Table 3.2: Error transitions between prototype settings before alignment using CLIP-B/16. We compare each target setting against a reference setting. "Fixed" denotes samples incorrectly classified by the reference setting but correctly classified by the target setting. "Introduced" denotes samples correctly classified by the reference setting but incorrectly classified by the target setting. This table shows how changing the prototype modality can both correct and introduce OOD false negatives and false positives, making explicit the non-monotonic effect of adding image prototypes.

The first two plots compare within-class (solid lines) and between-class (dotted lines) similarities within the same modality, *i.e.*, text and image, respectively. In both cases, the two distributions are well separated, as quantified by a large Kolmogorov–Smirnov (KS) distance, indicating that same-modality embeddings reliably capture class-level semantics. The third plot shows cross-modal (image–text) similarities, where separability is markedly lower than in the text-only case, though still higher than in the image-only case. This reduced cross-modal discriminability means that OOD images can attain similarity scores to ID text prototypes that are comparable to those of genuine ID images (Figure 3.3; more examples in Section 3.4.4): spatial proximity in the joint embedding space can masquerade as semantic relatedness, inflating the false positive rate. This

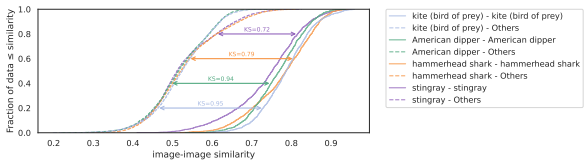
Case type	GT	Image	Dataset / class	Nearest text prototype	Nearest image prototype	Score / prediction
Text-only FN fixed by image	OOD		iNaturalist	cradle 	birdhouse 	$S_{MCM,T}$: 1.0766 / ID $S_{MCM,I}$: 1.0617 / OOD S_{MMP}: 1.0691 / OOD
Text-only FP fixed by image	ID		ImageNet-100 (ice_cream)	cornet 	ice_cream 	$S_{MCM,T}$: 1.0714 / OOD $S_{MCM,I}$: 1.3617 / ID $S_{MCM,MMP}$: 1.2166 / ID
Image-only FN fixed by text	OOD		iNaturalist	overskirt 	greenhouse 	$S_{MCM,T}$: 1.0460 / OOD $S_{MCM,I}$: 1.1922 / ID $S_{MCM,MMP}$: 1.1191 / OOD
Image-only FP fixed by text	ID		ImageNet-100 (colobus)	colobus 	colobus 	$S_{MCM,T}$: 1.1694 / ID $S_{MCM,I}$: 1.1467 / OOD $S_{MCM,MMP}$: 1.1581 / ID
Both modalities needed	OOD		iNaturalist	alligator lizard 	hen-of-the-woods 	$S_{MCM,T}$: 1.0807 / ID $S_{MCM,I}$: 1.1680 / ID $S_{MCM,MMP}$: 1.1244 / OOD

Table 3.3: Representative prototype-level error cases before alignment using CLIP-B/16. Each row shows a case where one prototype modality fails and the complementary modality or the combined multi-modal prototype corrects the prediction. $S_{MCM,T}$, $S_{MCM,I}$, and S_{MMP} denote the ID confidence scores obtained using text prototypes, image prototypes, and multimodal prototypes, respectively. We use the same threshold as in Table 3.1 for each setting. Higher scores indicate stronger ID confidence. We multiply the scores by 100 for better illustration.

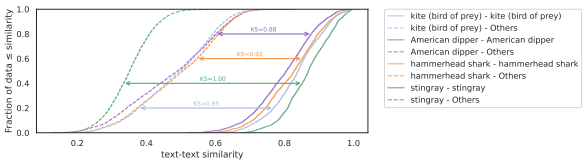
observation raises a critical question about the impact of the modality gap on VLM-based OOD detection, motivating our first research question:

RQ1: What is the impact of the modality gap on VLM-based OOD detection methods?

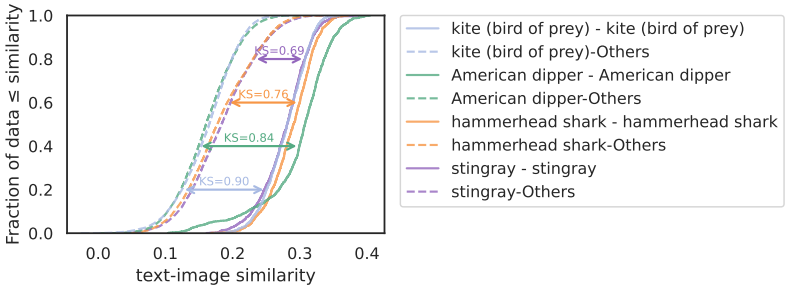
In response to RQ1, we hypothesize that the modality gap negatively impacts performance and that incorporating image and text (multi-modal) prototypes, as opposed to only text prototypes, could alleviate this effect. We illustrate this in Figure 3.1. To explore this, we compute class-specific ID image prototypes by averaging embeddings extracted from the ID data within each class. We extend the MCM score S_{MCM} [160] to incorporate the image prototypes alongside the



(a) Image–image similarity distributions computed between image embeddings of selected ID and OOD examples.



(b) Text–text similarity distributions computed among ID class text prototypes generated from class-name prompts.



(c) Image–text similarity distributions computed between input image embeddings and ID text prototypes.

Figure 3.2: Four arbitrary classes in ImageNet-1k are selected to visualize the similarity distribution between image embeddings and text prototypes within and across each class. We also compare Kolmogorov–Smirnov (KS) statistics between inter- and intra-class similarity distributions. "A - A" refers to the similarity distribution within class A, and "A - B" refers to the similarity distribution between class A and class B. Higher KS statistic values indicate a greater difference between two distributions. Best viewed in color.

text prototypes. Compared to S_{MCM}, our empirical analysis shows that, without any training, using the extension of S_{MCM} with multi-modal prototypes improves average FPR95 and AUROC from 32.4 and 94.5 to 24.2 and 95.8, respectively, on Imagenet-100 [39] (ID) across four OOD datasets (Table 3.4). For theoretical support, in Theorem 1, we demonstrate that incorporating image prototypes increases the score separation between ID and OOD data, leading to improved performance.

Given these findings, *i.e.*, that the modality gap significantly affects VLM-based OOD detection, we conclude that it is necessary to explore approaches mitigating the gap to improve reliability. This leads us to our second question:

RQ2: How can we mitigate the impact of modality gap to improve VLM-based OOD detection?

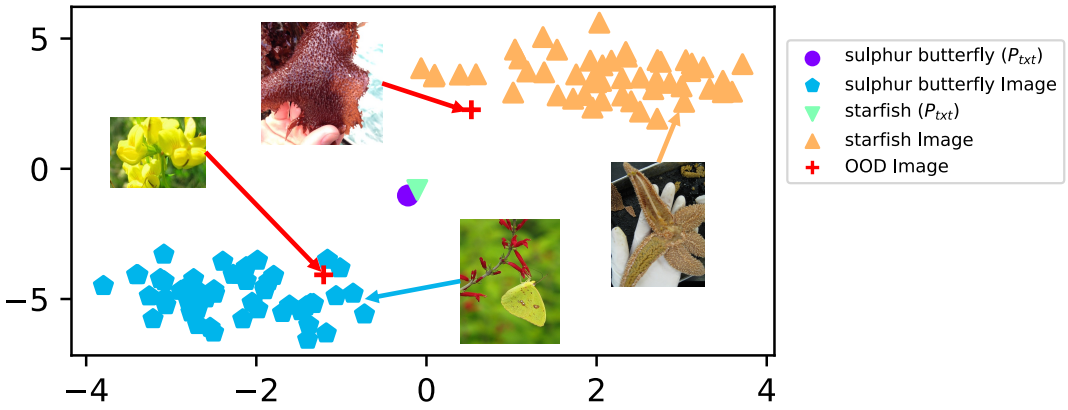


Figure 3.3: t-SNE visualization of CLIP (ViT-B/16) embeddings showing failure cases for text-prototype OOD detection. The plot compares ID image embeddings from the ImageNet-1k test set, OOD image embeddings from iNaturalist, and ID text prototypes generated from class-name prompts. Several OOD image embeddings are located closer to ID text prototypes than genuine ID image embeddings are, illustrating how spatial proximity caused by the image-text modality gap can produce false positives. Figure 3.5 shows that SUPREME reduces this gap after alignment.

To address RQ2, we propose a novel few-**S**hot tuning, m**U**lti-modal **PR**ototyp**E**-based **ME**thod for OOD detection with CLIP, termed SUPREME. SUPREME comprises biased prompt generation (BPG) and image-text consistency (ITC) modules. BPG introduces a Gaussian-based image domain bias to enhance image-text fusion and generalization on ID data unseen during fine-tuning by preventing overfitting on the limited training ID data. ITC reduces the modality gap by minimizing inter- and intra-modal distances as CLIP remains frozen. SUPREME also makes use of a new OOD score, which we call the **G**eneralized **M**ulti-modal **P**rototype OOD score, denoted S_{GMP}.

BPG. In contrast to previous OOD detection methods [10, 116, 162] that only utilize learnable contexts for text prototypes, BPG conditions text prototypes on three components: learnable contexts, the image embedding, and the ID image domain bias. The image embedding facilitates image-text fusion, enhancing cross-modal alignment. The third component, the Gaussian-based ID image domain bias, captures the distribution of ID image embeddings to improve generalization on unseen ID data and avoid overfitting on the few-shot training ID data.

ITC. As CLIP remains frozen during training, to reduce the excessive image-text modality gap in this OOD setting, we introduce intra- and inter-modal losses with two mappings. First, the image embedding I is mapped to the text domain as $I' = f_{i2t}(I)$ with the image-to-text mapping $f_{i2t}(\cdot)$. To ensure it aligns with ID text prototypes, we employ the inter-modal loss ℓ_{inter}. Additionally,

to avoid information loss, the intra-modal loss ℓ_{intra} is applied between the original embedding I and the reconstructed image embedding $\hat{I} = f_{t2i}(I')$ using the text-to-image mapping $f_{t2i}(\cdot)$.

S_{GMP}. Building on our findings from RQ1, we introduce a new OOD score, S_{GMP}, which integrates uni- and cross-modal similarities. While previous methods rely solely on the similarity between ID text prototypes and input image embedding, S_{GMP} incorporates the similarity between multi-modal input embeddings—the image embedding I and the mapped image embedding $f_{i2t}(I)$—and ID multi-modal (image and text) prototypes, respectively. This allows S_{GMP} to balance inter- and intra-modal similarities, enhancing robustness and performance.

We include extensive experiments across multiple benchmarks, including ImageNet-1k [39], ImageNet-100 [39], ImageNet-10 [39], ImageNet-20 [39], and OpenOOD [243], that demonstrate that SUPREME outperforms existing OOD detection methods and achieves new state-of-the-art performance.

Geometric analysis of the image and text embeddings with SUPREME. We perform a geometric analysis of the embeddings with SUPREME. We choose two arbitrary classes in ImageNet-1k and employ t-SNE to visualize the embeddings of the two classes in the joint embedding space as shown in Figure 3.5. We observe that the embeddings of the two classes are well separated in the joint embedding space.

Quantitative evaluation of the modality gap with SUPREME. We calculate the KS distance and modality gap between the two classes as shown in Table 3.15 and fig. 3.10. The results show that the modality gap is reduced by SUPREME compared to the baseline methods.

3.2 Preliminaries

3.2.1 Problem Setup

This section establishes the notation used in the chapter and specializes the general OOD formulation in Section 2.2 to the few-shot VLM setting studied here. The key difference from conventional image-only OOD detection is that the scoring function can depend on both visual evidence from a test image and semantic evidence from class-name prompts. We therefore distinguish between image embeddings, text prototypes, and the image prototypes introduced later in Section 3.3.1.

Let $\mathcal{X}$ and $\mathcal{Y} = \{y_1, \dots, y_C\}$ represent the feature space and ID label space, respectively, where C is the number of ID classes.

OOD detection. In real-world applications, AI models are trained on ID data and may misclassify OOD data into ID classes with high confidence [93]. To tackle this problem, OOD detection [161]

is proposed to identify OOD samples using a score function $S(\cdot)$ [112,250]. A sample is classified as OOD if $S(\mathbf{x}) \leq \gamma$, where γ is a predefined threshold.

Few-shot tuning. In contrast to existing works that either utilize the entire ID training dataset [104, 200] or avoid tuning entirely [16, 92, 160, 201], we study the scenario where the model is fine-tuned using only a subset of the ID training data (16 images per class) without access to OOD data or other additional data.

3.2.2 Revisiting CLIP, MCM, and Prompt Tuning

CLIP [174] is a foundational VLM pre-trained on a web-scale image-text dataset using self-supervised contrastive learning [22], which achieves strong zero-shot classification results on numerous benchmark datasets [11, 101, 109, 153, 170]. Specifically, for classification, CLIP first incorporates class labels $\mathcal{Y} = \{y_c\}_{c=1}^{C}$, *e.g.*, "cat", into fixed pre-designed, rather than learned, text prompts, *e.g.*, "`a photo of a` *label*". These prompts combined with class labels are processed by CLIP's text encoder $f_{text}(\cdot)$ to generate text embeddings $\{P_{txt,c}\}_{c=1}^{C}$, where $P_{txt,c} = f_{text}(\text{"}\texttt{a photo of a } \{y_c\}\text{"})$. Given an image $\mathbf{x}$, the image embedding I is obtained by the image encoder $f_{image}(\cdot)$ as $I = f_{image}(\mathbf{x})$. The index of the predicted class label is given by $\arg\max_{c\in\{1..C\}} \cos(I, P_{txt,c})$, where $\cos(\cdot,\cdot)$ denotes the cosine similarity.

CLIP for OOD Detection (MCM) [160]. Beyond its strong classification capabilities, MCM demonstrates that pre-trained CLIP models also exhibit robust zero-shot OOD detection capabilities. Specifically, MCM defines the maximum classification score as the OOD score S_{MCM},

$$S_{MCM}(I, \{P_{txt,c}\}_{c\in[C]}) = \max_{c\in[C]} \frac{\exp(\cos(I, P_{txt,c})/\tau)}{\sum_{j\in[C]} \exp(\cos(I, P_{txt,j})/\tau)}, \tag{3.1}$$

where τ is the temperature, and $\exp(\cdot)$ is the exponential function.

Visual prompt tuning [63, 231, 253, 254]. To improve CLIP's performance when target training data is accessible, CoOp [254] replaces manually designed prompt templates with learnable soft prompts (context tokens) $CoOp_c = [V_1, \ldots, V_L, y_c]$ for class index c. L is the length of learnable prompts and each V_i represents a learnable vector. The class-wise text embedding is then generated as $P_{txt,c}^{CoOp} = f_{text}(CoOp_c)$ for class index c. With CLIP's text and image encoder parameters frozen, the learnable prompts are optimized using the cross-entropy loss:

$$\ell_{ID}(I) = -\log \frac{\exp(\cos(I, P_{txt,c}^{CoOp})/\tau)}{\sum_{j=1}^{C} \exp(\cos(I, P_{txt,j}^{CoOp})/\tau)}, \tag{3.2}$$

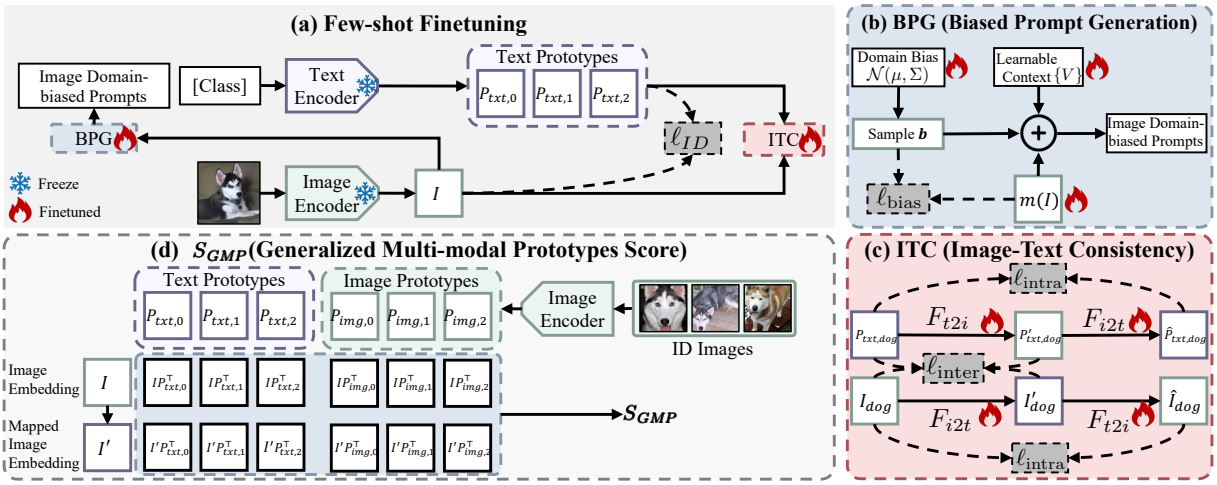


Figure 3.4: Overview of SUPREME for few-shot VLM-based OOD detection. The two novel modules, *i.e.*, BPG (biased prompt generation) and ITC (image-text consistency), are designed to reduce the excessive image-text modality gap in this OOD setting. (a) During the few-shot fine-tuning stage, (b) BPG generates image domain-biased prompts, conditioned on the estimated image domain bias and the prompt-conditioning image feature $m(I)$ for better image-text fusion. (c) ITC reduces the modality gap directly by the intra- and inter-modal losses (ℓ_{intra} and ℓ_{inter}) with the text-to-image mapping f_{t2i} and the image-to-text mapping f_{i2t}. (d) During inference, ID image prototypes are obtained by averaging each class's base ID image embeddings from the training data (Equation (3.3)). The proposed S_{GMP} (Equation (3.20)) is calculated from the maximum similarities between the test image embedding I, its image-to-text mapped embedding I', and ID multi-modal prototypes (*i.e.*, text prototypes $\{P_{txt,c}\}_{c=1}^{C}$ and image prototypes $\{P_{img,c}\}_{c=1}^{C}$). S_{MCM} refers to the MCM score [160].

where c is the index of the ground truth label for the input image. Following CoOp, to avoid potential overfitting on the training data, CoCoOp [253] conditions prompts on the image embedding I as $CoCoOp_c = [V_1 + m(I), \ldots, V_L + m(I), y_c]$, where $m(\cdot)$ is a two-layer multi-layer perception (MLP, Linear-ReLU-Linear).

3.3 Methodology

In this section, we first answer RQ1 by presenting our hypothesis and findings in Section 3.3.1, where we demonstrate that the modality gap negatively impacts performance and that incorporating image prototypes alleviates this effect. Then, to mitigate the modality gap (RQ2), we

introduce SUPREME in Section 3.3.2 with two modules and a novel OOD score, S_{GMP}, as shown in Figure 3.4. Note that in this section, we denote the input image embedding by I. The methodology follows directly from the background observation in Section 2.1: contrastive VLMs make image-text comparison possible, but they do not guarantee that image and text embeddings of the same class occupy the same local region. We therefore separate the chapter into two steps. First, we test whether adding same-modality visual anchors is already sufficient to improve OOD scoring without training. Second, we introduce a few-shot tuning strategy that improves cross-modal compatibility while keeping the pretrained CLIP encoders fixed.

3.3.1 RQ1: Multi-modal Prototypes

CLIP has been widely applied in different areas [57, 195]. However, a recent study [128] demonstrates that representations of images and texts are clearly separated, creating what is known as the modality gap. It remains unclear, however, whether this modality gap positively or negatively impacts OOD detection.

We hypothesize that the modality gap negatively impacts OOD detection performance by causing misalignment between image and text embeddings, and that this misalignment can lead to increased false positives, as OOD images may exhibit high similarity to ID text prototypes due to either semantic similarity or small spatial distance.

To test our hypothesis, we demonstrate that incorporating multi-modal (image and text) prototypes can mitigate the negative impact of the modality gap and reduce false positives. We obtain ID text prototypes $\{P_{txt,c}\}_{c=1}^{C}$ as in MCM [160]. For ID image prototypes $\{P_{img,c}\}_{c=1}^{C}$, we collect ID base images $\{\mathbf{x}_i\}_{i=1}^{N_{base}}$ with corresponding labels $\{y_i\}_{i=1}^{N_{base}}$, where N_{base} is the number of base images. ID image prototypes are then calculated by averaging the image embeddings for each class label y_c:

$$P_{img,c} = \frac{\sum_{i=1}^{N_{base}} \mathbb{I}(y_i = y_c) f_{image}(\mathbf{x}_i)}{\sum_{j=1}^{N_{base}} \mathbb{I}(y_j = y_c)}, \tag{3.3}$$

where $\mathbb{I}(\cdot)$ is the indicator function and $f_{image}(\cdot)$ is CLIP's image encoder. With these image prototypes, we extend S_{MCM} (Equation (3.1)) [160] to a new OOD score, which we term the Multi-Modal Prototypes score, S_{MMP}, defined as

$$S_{MMP} = \frac{S_{MCM}(I, \{P_{img,c}\}_{c=1}^{C}) + S_{MCM}(I, \{P_{txt,c}\}_{c=1}^{C})}{2}. \tag{3.4}$$

This approach balances contributions from both modalities, potentially mitigating the impact of the modality gap.

Methods	FPR95 (%) ↓	AUROC (%) ↑
ImageNet-100 → 4 datasets (Average)		
S_{MCM} (Text prototypes)	32.58	94.48
S_{MCM} (Image prototypes)	31.09	94.74
S_{MMP} (Text + Image prototypes)	**24.18**	**95.79**
ImageNet-100 → iNaturalist		
S_{MCM} (Text prototypes)	18.13	96.77
S_{MCM} (Image prototypes)	26.06	93.09
S_{MMP} (Text + Image prototypes)	**14.76**	**97.36**
ImageNet-100 → SUN		
S_{MCM} (Text prototypes)	36.45	**94.54**
S_{MCM} (Image prototypes)	39.25	93.17
S_{MMP} (Text + Image prototypes)	**30.28**	92.51
ImageNet-100 → Places		
S_{MCM} (Text prototypes)	34.52	**94.36**
S_{MCM} (Image prototypes)	40.59	96.43
S_{MMP} (Text + Image prototypes)	**34.04**	93.92
ImageNet-100 → Texture		
S_{MCM} (Text prototypes)	41.22	92.25
S_{MCM} (Image prototypes)	18.46	96.27
S_{MMP} (Text + Image prototypes)	**17.66**	**96.66**

Table 3.4: Comparison of OOD detection results on ImageNet-100 as the ID dataset and four OOD datasets. The table compares three scoring choices: MCM with ID text prototypes, MCM with ID image prototypes, and the proposed S_{MMP} that averages both prototype types. ID image prototypes are computed from ImageNet-100 training images, while evaluation uses held-out ID test images and OOD test images. We use CLIP-B/16 for S_{MCM} and S_{MMP} (ours, Equation (3.4)). FPR95 is the false positive rate of OOD images when the true positive rate of ID images is fixed at 95%, and AUROC is the area under the receiver operating characteristic curve. Best results are shown in **bold**.

Empirical evidence. We begin by empirically validating the effectiveness of image prototypes and S_{MMP}. We use ImageNet-100 [39] as the ID dataset, with four OOD datasets: iNaturalist [196], SUN [219], Places [251], and Texture [32]. We use CLIP-B/16 as the base model for S_{MCM} and S_{MMP}. All the training images from ImageNet-100 are used for generating image prototypes. We also include S_{MCM} with image prototypes as a comparison baseline.

With all training images of ImageNet-100 as the base set, we observe in Table 3.4 that, on average, S_{MMP} outperforms S_{MCM} in both FPR95 and AUROC, demonstrating the effectiveness of image prototypes and S_{MMP}. Our method, S_{MMP}, consistently demonstrates superior performance in terms of FPR95 across all four tasks, achieving the lowest values, with notable improvements on iNaturalist (14.76 vs. 18.13) and Texture (17.66 vs. 41.22), highlighting its robustness in reducing false positives. For AUROC, S_{MMP} achieves the highest score on iNaturalist (97.36) and Texture (96.66). However, in the SUN and Places datasets, while S_{MMP} maintains competitive performance, it slightly underperforms S_{MCM} in AUROC (92.51 vs. 94.54 on SUN and 93.92 vs. 94.36 on Places). Similarly, we observe that S_{MMP} outperforms S_{MCM} with image prototypes on most of the metrics, while S_{MCM} with image prototypes has slightly better performance compared to S_{MCM} (with text prototypes). This shows that image and text prototypes are complementary to each other and employing them at the same time improves performance.

Importantly, S_{MMP} does not directly reduce the image–text modality gap, since the CLIP encoders and prototypes remain in the original embedding space. Instead, it mitigates the effect of the gap by combining cross-modal text prototypes with same-modality image prototypes, whose errors are complementary.

Theoretical evidence. To further demonstrate why image prototypes improve performance, we present a theoretical justification in Theorem 1. The theorem shows that S_{MMP} increases the expected score separation between ID and OOD samples compared to using text prototypes alone as in S_{MCM}.

Theorem 1 (Multi-modal Prototypes Increase Score Separation between ID and OOD Data)**.** *Assuming that the OOD data is not drawn from any ID distribution, we have,*

$$\mathbb{E}\left[S_{MMP}(I_{ID}) - S_{MMP}(I_{OOD})\right] \geq \mathbb{E}\left[S_{MCM}(I_{ID}) - S_{MCM}(I_{OOD})\right] , \tag{3.5}$$

where I_{ID} and I_{OOD} are the image embeddings of ID and OOD samples. We omit multi-modal prototypes for clarity.

Proof. We first introduce three necessary assumptions before proceeding with the proof of our theorem.

Assumption 1 (ID image embeddings follow the distribution centered at ID image prototypes)**.** *Image embeddings $\mathbf{x}_c$ are drawn from the symmetric distribution $\mathcal{D}_c$ which is centered at the image prototypes $P_{img,c}$ in the embedding space.*

Assumption 2 (The image-text modality gap exists)**.** *The image embeddings are closer to the ground truth class's image prototypes than the text prototypes due to the modality gap.*

Remark 1. *These two assumptions are the basic assumptions about the multi-modal prototypes and the modality gap. Based on the findings from a recent study [128], these two hold true for most real-world scenarios.*

Assumption 3 (Models are well trained)**.** *Cosine similarity between embeddings from the same class distribution $\mathcal{D}_i$ is higher on average than between embeddings from different distributions. And the image embeddings from the c-th class are equally dissimilar to other classes.*

Remark 2. *Here we only consider the optimal representation extractors without any other assumptions on the models.*

Assumption 4 (OOD embeddings are not sampled from any ID distribution [56])**.** *OOD image embeddings $\mathbf{x}_{OOD}$ are drawn from a distribution $\mathcal{D}_{OOD}$ different from any $\mathcal{D}_c^*, \forall c \in \{1..C\}$ and $\forall * \in \{text, image\}$. In addition, the OOD image embeddings are equally similar to all ID text/image classes.*

Remark 3. *This assumption is a general basic assumption used by previous studies [56] on the OOD data, which provides us with an overall characteristic of the OOD data.*

Under the above assumptions, we first would like to prove the following inequality holds:

$$\begin{aligned} \Delta_{image} &= \mathbb{E}\left[\max_{i\in\{1..C\}} \sigma_i^{image}(\mathbf{x}_{ID}) - \max_{i\in\{1..C\}} \sigma_i^{image}(\mathbf{x}_{OOD})\right] \\ \geq \Delta_{text} &= \mathbb{E}\left[\max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{ID}) - \max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{OOD})\right] \\ &\text{s.t. } \sigma_i^* = \frac{\exp(s_{*,i})}{\sum_{k=1}^{N}\exp(s_k)} \text{ and } s_{*,i} = cos(\mathbf{x}, P_{*,i}) \\ &\forall * \in \{image, text\}, \end{aligned} \tag{3.6}$$

where $\mathbf{x}_{ID}$ and $\mathbf{x}_{OOD}$ are samples from any ID distributions and OOD samples.

First, for ID samples, with Assumption 3, we have

$$\begin{aligned} \mathbb{E}\left[\max_{c\in\{1..C\}} \sigma_c^*(\mathbf{x}_{ID})\right] &= \frac{\exp\left(\mu_{intra}^*\right)}{\exp\left(\mu_{intra}^*\right) + (C-1)\exp\left(\mu_{inter}^*\right)} \\ &\forall * \in \{text, image\}, \end{aligned} \tag{3.7}$$

where $\mu_{intra}^* = \max_{c\in\{1..C\}} \mathbb{E}[\cos(\mathbf{x}_{ID}, P_{*,c})]$, $\mu_{inter}^* = \mathbb{E}[cos(\mathbf{x}_{ID}, I_{c_j})], \forall c_j \in \{1..C\} \setminus \{c_{ID}\}$, and C_{ID} is the ground-truth class for the input $\mathbf{x}_{ID}$.

With assumption 4, for OOD samples, we have

$$\mathbb{E}\left[\max_{c\in\{1..C\}} \sigma_c^{image}(\mathbf{x}_{OOD})\right] = \mathbb{E}\left[\max_{c\in\{1..C\}} \sigma_c^{text}(\mathbf{x}_{OOD})\right] = \frac{1}{C}\,. \tag{3.8}$$

Then we have

$$\Delta_{image} = \mathbb{E}\left[\max_{c\in\{1..C\}} \sigma_c^{image}(\mathbf{x}_i)\right] - \frac{1}{C}\,. \tag{3.9}$$

Similarly, we have

$$\Delta_{text} = \mathbb{E}\left[\max_{c\in\{1..C\}} \sigma_c^{text}(\mathbf{x}_i)\right] - \frac{1}{C}\,. \tag{3.10}$$

Since $\mu_{intra}^{image} > \mu_{intra}^{text}$ (Assumption 2), we have

$$\Delta_{image} > \Delta_{text}\,. \tag{3.11}$$

By simple algebra, we have

$$\begin{aligned}
&\mathbb{E}\left[\frac{\max_{i\in\{1..C\}} \sigma_i^{image}(\mathbf{x}_{ID}) + \max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{ID})}{2}\right. \\
&\left. -\frac{\max_{i\in\{1..C\}} \sigma_i^{image}(\mathbf{x}_{OOD}) + \max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{OOD})}{2}\right] \\
\geq &\mathbb{E}\left[\max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{ID}) - \max_{i\in\{1..C\}} \sigma_i^{text}(\mathbf{x}_{OOD})\right]\,, \\
&\mathbb{E}\left[S_{MMP}(\mathbf{x}_{ID}) - S_{MMP}(\mathbf{x}_{OOD})\right] \\
\geq &\mathbb{E}\left[S_{MCM}(\mathbf{x}_{ID}) - S_{MCM}(\mathbf{x}_{OOD})\right]\,.
\end{aligned} \tag{3.12}$$

This completes the proof. □

Remark 4. *This greater score separation enhances the separation between ID and OOD data, improving detection performance, under general assumptions used by previous works [56, 128]. And it further improves performance.*

While incorporating image prototypes improves performance both empirically and theoretically, inspired by recent research [116, 162], we further fine-tune CLIP in a few-shot tuning manner to mitigate the impact of the modality gap (RQ2). The details are presented in the following section.

3.3.2 RQ2: SUPREME

To alleviate the negative impact of the image-text modality gap [128] in CLIP (RQ2), we propose SUPREME, which is summarized in Figure 3.4. SUPREME incorporates two novel modules, *i.e.*, the biased prompt generation module (BPG) and the image-text consistency module (ITC), along with a new OOD score. BPG enhances image-text fusion using image domain-biased prompts, while ITC minimizes both intra- and inter-modal distances. SUPREME's novel OOD score, S_{GMP}, leverages uni- and cross-modal similarities for improving robustness and performance.

Biased Prompt Generation (BPG). Drawing inspiration from CoCoOp [254], we employ a set of learnable context tokens $\{V_i \in \mathcal{R}^{1\times N_{\text{lm}}}\}_{i=1}^{L}$ and $m(I) \in \mathcal{R}^{1\times N_{\text{lm}}}$ to generate the text prototypes, where L is the length of learnable contexts, $m(\cdot)$ is linear projection that maps an image embedding into the text-token input dimension, and N_{lm} is the dimension of text encoder's input. The feature $m(I)$ is used only for prompt conditioning in BPG and is distinct from the image-to-text mapped embedding $f_{i2t}(I)$ used by ITC and the OOD score, which lies in the CLIP text-embedding space. Similar to previous methods [162, 253], learnable context tokens replace the manually designed prompt templates, *e.g.*, "`a photo of a` *class*", for generating text prototypes.

To improve generalization on unseen ID data and avoid overfitting on a limited few-shot training data, we introduce a third component, a Gaussian-based estimated image domain bias $\mathcal{N}(\mu, \Sigma)$, for generating the prompts and ID text prototypes. This bias counteracts the modality gap by pre-biasing text prototypes toward ID images and augments the training data to better approximate the distribution of prompt-conditioning image features. Using a Gaussian distribution is a reasonable a priori choice and is mathematically tractable. Our ablation study (Table 3.10) confirms that this choice is valid and is successful in reducing FPs. After that, we generate image domain-biased prompts (IDBP) conditioned on these three components. We next introduce the details of estimating the bias and generating IDBP.

Estimating image domain bias. In each training iteration, for each image I in a minibatch $\mathcal{B}$, we sample $\mathbf{b}_t$ from the Gaussian-based image domain bias $\mathcal{N}(\mu, \Sigma)$ as,

$$\mathbf{b}_t = \mu + \sigma n_t, \quad n_t \sim \mathcal{N}(0, \mathcal{I}), \tag{3.13}$$

where $\Sigma = \sigma\sigma^\top$ (based on the Cholesky decomposition) and $\mathcal{N}(0, \mathcal{I})$ represents the standard Gaussian distribution, given identity matrix $\mathcal{I}$. We use the Gaussian distribution due to its simplicity. To align the domain bias with the distribution of training ID image embeddings, we employ the following loss,

$$\ell_{bias} = \frac{1}{|\mathcal{B}|}\sum_{I\in\mathcal{B}}\left(\|\mu - m(I)\|_1 + \|\mathbf{b}_t - m(I)\|_1\right), \tag{3.14}$$

where $\|\cdot\|_1$ is the ℓ_1 distance. In the first term of the loss (Equation (3.14)), μ is updated. In the second term, as $\mathbf{b}_t$ is sampled using μ and Σ, μ and Σ will be updated during the few-shot training procedure. The first term pushes the Gaussian mean μ toward the prompt-conditioning image features $m(I)$ in the minibatch, while the second term ensures sampled bias vectors $\mathbf{b}_t$ remain close to these image-derived features.

Generating image domain-biased prompts (IDBP). We condition IDBPs on three components: the learnable contexts $\{V_i \in \mathcal{R}^{1\times N_{\text{lm}}}\}_{i=1}^{L}$, the prompt-conditioning image feature $m(I) \in \mathcal{R}^{1\times N_{\text{lm}}}$, and the estimated image domain bias. During training, this bias is the sampled vector $\mathbf{b}_t \in \mathcal{R}^{1\times N_{\text{lm}}}$ from $\mathcal{N}(\mu, \Sigma)$:

$$IDBP_c^{\text{train}} = [V_1 + m(I) + \mathbf{b}_t, \ldots, V_L + m(I) + \mathbf{b}_t, y_c]\,. \tag{3.15}$$

At inference time, we do not sample the bias; as described later, we use the deterministic mean of the learned Gaussian, $\mathbf{b}^{\text{inf}} = \mu$. The final ID text prototype $P_{txt,c}$ for the c-th class is obtained by encoding the corresponding IDBP with CLIP's text encoder, *i.e.*, $P_{txt,c}^{\text{train}} = f_{text}(IDBP_c^{\text{train}})$ during training and $P_{txt,c}^{\text{inf}} = f_{text}(IDBP_c^{\text{inf}})$ during inference.

We thus establish a foundation for cross-modal fusion by conditioning prompts on learned contexts, image embeddings, and the estimated image domain bias.

Image-Text Consistency (ITC). While BPG reduces the gap by image-text fusion, image-text consistency (ITC) directly reduces the modality gap by aligning image and text embeddings through inter- and intra-modal distances at a fine-grained level. It consists of two mappings, *i.e.*, the image-to-text mapping $f_{i2t}(\cdot)$ and text-to-image mapping $f_{t2i}(\cdot)$, which are used for mapping embeddings into another domain/modality.

The first component of ITC is the *inter-modal loss*. It aligns each modality with its cross-modal counterpart: image embeddings I are pulled toward text prototypes mapped into image space, $f_{t2i}(P_{txt,c})$, and image-to-text mapped embeddings $f_{i2t}(I)$ are pulled toward text prototypes $P_{txt,c}$. It is designed to minimize the cross-entropy loss,

$$\ell_{inter} = -\log \frac{\exp(\cos(I, f_{t2i}(P_{txt,y}))/\tau)}{\sum_{c\in[C]} \exp(\cos(I, f_{t2i}(P_{txt,c}))/\tau)} - \log \frac{\exp(\cos(f_{i2t}(I), P_{txt,y})/\tau)}{\sum_{c\in[C]} \exp(\cos(f_{i2t}(I), P_{txt,c})/\tau)}\,, \tag{3.16}$$

where y is the ground truth label for the input image, C is the number of classes, and τ is the temperature.

Next, to prevent information loss, we introduce *intra-modal loss*. Specifically, we reconstruct the original embedding by mapping the mapped embedding back to its initial modality. Then, we

minimize the ℓ_1 distance between the reconstructed and original embeddings as follows,

$$\ell_{intra} = \|I - f_{t2i}(f_{i2t}(I))\|_1 + \sum_{c\in[C]} \|P_{txt,c} - f_{i2t}(f_{t2i}(P_{txt,c}))\|_1 , \tag{3.17}$$

The inter- and intra-modal losses serve to reduce the modality gap between image and text, and to preserve all information during mapping.

Combining BPG with ITC, SUPREME mitigates the modality gap by enhancing image-text fusion (BPG) as well as reducing inter- and intra-modal distances (ITC).

Training, Inference, and S_{GMP}. Our overall training objectives are a linear combination of the four losses: $\mathcal{L} = \ell_{ID} + \alpha(\ell_{intra} + \ell_{inter}) + \beta\ell_{bias}$, where α and β are trade-off parameters. During inference, we do not sample from $\mathcal{N}(\mu, \Sigma)$. Instead, we use the expectation of the learned image-domain-bias distribution as the deterministic bias vector:

$$\mathbf{b}^{\text{inf}} = \mathbb{E}_{\mathbf{b}\sim\mathcal{N}(\mu,\Sigma)}[\mathbf{b}] = \mu. \tag{3.18}$$

Thus, for an inference image I, the image domain-biased prompt for class c is

$$IDBP_c^{\text{inf}}(I) = [V_1 + m(I) + \mu, \ldots, V_L + m(I) + \mu, y_c]. \tag{3.19}$$

Here, "mean of the domain bias" refers to the learned Gaussian mean μ, not to the empirical mean of the test images.

Generalized multi-modal prototypes OOD score S_{GMP}. Building on our empirical and theoretical findings in Section 3.3.1, with the image-to-text mapping $f_{i2t}(\cdot)$, we introduce the generalized multi-modal prototypes OOD score S_{GMP}. S_{GMP} uses the average of the maximum similarities between the multi-modal input embedding (the image embedding I and image-to-text mapped embedding $I' = f_{i2t}(I)$) and multi-modal prototypes:

$$\begin{aligned} &S_{GMP}(I, I', \{P_{txt,c}\}_{c\in[C]}, \{P_{img,c}\}_{c\in[C]}) \\ =&\frac{1}{4}(S_{MCM}(I, \{P_{txt,c}\}_{c\in[C]}) + S_{MCM}(I, \{P_{img,c}\}_{c\in[C]}) \\ &+S_{MCM}(I', \{P_{txt,c}\}_{c\in[C]}) + S_{MCM}(I', \{P_{img,c}\}_{c\in[C]})) . \end{aligned} \tag{3.20}$$

The four terms in Equation (3.20) compare two test representations, I and I', with two prototype types, $\{P_{txt,c}\}$ and $\{P_{img,c}\}$. The middle two terms are the most directly aligned pairs:

$S_{MCM}(I, \{P_{img,c}\})$ compares image embeddings with image prototypes, and $S_{MCM}(I', \{P_{txt,c}\})$ compares mapped image embeddings with text prototypes. The first term, $S_{MCM}(I, \{P_{txt,c}\})$, preserves the original CLIP image–text comparison used by MCM and retains zero-shot language knowledge. The fourth term, $S_{MCM}(I', \{P_{img,c}\})$, checks whether the mapped representation remains consistent with visual ID prototype structure instead of drifting toward text prototypes alone. Using only the two aligned terms would discard the original CLIP cross-modal semantic signal and would not test whether the mapped embedding still preserves visual ID structure; the additional two terms therefore act as complementary semantic and consistency checks. Together, these four terms form a symmetric multi-modal score.

Remark 5. *S_{GMP} extends S_{MCM} by incorporating both uni- and cross-modal similarities. In different scenarios, the importance of intra- and inter-modal similarity is different. However, we **average the contributions** from the image and text modalities as a starting point to ensure the generalization ability of S_{GMP} and reduce the gap. This approach enhances the separation between ID and OOD samples, improving the OOD detection performance of VLM-based methods as shown in Figures 3.10 and 3.11.*

3.4 Experiments

3.4.1 Experimental Details

Datasets and benchmarks. Our experiments primarily use the ImageNet-1k [39] and ImageNet-100 [39] dataset as ID data. Aligning with standards from prior works [160, 201], we evaluate on four diverse OOD datasets: iNaturalist [196], SUN [219], Places [251], and Texture [32]. OpenOOD benchmark [243] is also used for evaluation on ImageNet-1k. We test on ImageNet-10 [39] and ImageNet-20 [39] for hard OOD. Details are in the Section 3.4.6.

Baselines methods. We compare SUPREME with 20 baseline methods, including traditional OOD detection methods [54, 72, 80, 127, 138, 188, 200], VLM (CLIP)-based OOD methods [10, 16, 92, 116, 136, 160, 162, 164, 190, 201, 233, 238], and visual prompt tuning methods [253, 254].

Implementation details. For each class, 16 images are used for few-shot tuning and the calculation of ID image prototypes. We use the image and text encoders of VIT-B/16 pre-trained by CLIP [174] for all the experiments. To avoid overfitting, instead of using deep MLPs, we employ linear transformations without bias as the mappings ($f_{i2t}(\cdot)$ and $f_{t2i}(\cdot)$). The parameters of CLIP are frozen. We use SGD to optimize other parameters, *e.g.*, estimated image domain bias and learnable context, with a momentum of 0.9. Training epochs, learning rate, batch size, and learnable context length are set to $50, 0.002, 32$, and 16, respectively. We set $\alpha = 0.005$ and $\beta = 0.1$.

Methods	Venue	Shots	ImageNet-1k → OOD Datasets								Average	
			iNaturalist		SUN		Places		Texture			
			FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
MOS† [80] (BiT)	CVPR'21	-	9.28	98.15	40.63	92.01	49.54	89.06	60.43	81.23	39.97	90.11
Fort *et al.*† [54] (ViT-B)	NeurIPS'21	-	15.07	96.64	54.12	86.37	57.99	85.24	53.32	84.77	45.12	88.25
Energy† [138] (ViT-B)	NeurIPS'20	-	21.59	95.99	34.28	93.15	36.64	91.82	51.18	88.09	35.92	92.26
MSP† [72] (ViT-B)	ICLR'17	-	40.89	88.63	65.81	81.24	67.90	80.14	64.96	78.16	59.89	82.04
ODIN‡ [127]	ICLR'18	-	30.22	94.65	54.04	87.17	55.06	85.54	51.67	87.85	47.75	88.80
ViM‡ [200]	CVPR'22	-	32.19	93.16	54.01	87.19	60.67	83.75	53.94	87.18	50.20	87.82
KNN‡ [188]	ICML'22	-	29.17	94.52	35.62	92.67	39.61	91.02	64.35	85.67	42.19	90.97
MCM [160]	NeurIPS'22	0	30.91	94.61	37.59	92.57	44.69	89.77	57.77	86.11	42.74	90.77
CLIPN [201]	CVPR'23	0	23.94	95.27	26.17	93.93	33.45	92.28	**40.83**	**90.93**	31.10	93.10
EOE [16]	ICML'24	0	12.29	97.52	20.40	95.73	30.16	92.95	57.53	85.64	30.09	92.96
Training-based Models												
CoOp* [254]	IJCV	16	29.47	94.89	31.34	93.36	40.28	90.07	54.25	87.58	38.83	91.47
CoCoOp* [253]	CVPR'22	16	30.74	94.73	31.18	93.15	38.75	90.63	53.84	87.92	38.63	91.61
NPOS [190]	ICLR'23	-	16.58	96.19	43.77	90.44	45.27	89.44	46.12	88.80	37.93	91.22
LoCoOp [162]	NeurIPS'23	16	16.05	96.86	23.44	95.07	32.87	91.98	42.28	90.19	28.66	93.52
CATEX [136]	NeurIPS'23	-	10.18	97.88	33.87	92.83	41.43	90.48	33.17	92.73	29.66	93.48
LSN [164]	ICLR'24	-	21.56	95.83	26.32	94.35	34.48	91.25	38.54	90.42	30.22	92.96
ID-Like [10]	CVPR'24	4	8.98	98.19	42.03	91.64	44.00	90.57	**25.27**	94.32	30.07	93.68
NegPrompt [116]	CVPR'24	16	6.32	**98.73**	22.89	95.55	27.60	93.34	35.21	91.60	23.01	94.81
SCT [233]	NeurIPS'24	16	13.94	95.86	20.55	95.33	29.86	92.24	41.51	89.06	26.47	93.37
Local-Prompt [238]	ICLR'25	16	8.63	98.07	23.23	95.12	31.74	92.42	34.50	92.29	24.52	94.48
SUPREME		16	**5.91**	**98.73**	**15.85**	**96.08**	**26.18**	93.19	34.88	91.93	20.70	95.38
SUPREME†		16	8.27	98.29	19.40	95.84	26.69	**93.56**	26.77	**94.45**	**20.28**	**95.54**

Table 3.5: OOD detection results using ImageNet-1k as the ID dataset and four standard OOD datasets: iNaturalist, SUN, Places, and Texture. All VLM-based results use CLIP (ViT-B/16) unless otherwise noted. The table reports FPR95 and AUROC for each ID-OOD pair and the average across the four OOD datasets; lower FPR95 and higher AUROC indicate better OOD detection. Best results are shown in **bold**, and second-best results are underlined. "*" is cited from LSN [164]. "†" represents results from MCM [160]. "‡" represents results from NPOS [190]. SUPREME† employs 16 images per class for training and the whole training split for calculating ID image prototypes.

Experiments are conducted on a single Nvidia V100 with 32 GB memory. All the results for SUPREME are the average of three trials.

Metrics. We employ two widely accepted metrics: FPR95, the false positive rate of OOD images when the true positive rate of ID images is at 95%, and AUROC, the area under the receiver operating characteristic curve.

3.4.2 Main Results

ImageNet-1k and ImageNet-100 as the ID dataset. In Table 3.5, we present a comprehensive analysis on ImageNet-1k (ID) across four OOD datasets (iNaturalist, SUN, Places, Texture) using CLIP (VIT-B/16). SUPREME demonstrates superior performance, achieving the lowest FPR95

Methods	ImageNet-100 → OOD Datasets								Average	
	iNaturalist		SUN		Places		Texture			
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
MSP	23.55	95.92	37.02	92.45	40.76	91.23	24.40	94.90	31.43	93.63
ViM	20.11	96.22	38.56	93.12	44.01	87.33	33.12	93.24	33.95	92.48
MCM	18.13	96.77	36.45	94.54	34.52	94.36	41.22	92.25	32.58	94.48
CoOp	9.30	97.95	11.64	97.61	17.45	96.53	15.94	96.90	13.58	97.25
CoCoOp	11.76	97.84	14.28	97.13	15.16	96.73	18.27	96.54	14.86	97.06
SUPREME	**2.31**	**99.28**	8.26	**98.38**	12.20	97.12	11.10	97.53	8.47	**98.08**
SUPREME†	2.54	99.21	**8.08**	98.28	**12.17**	**97.15**	**9.98**	**97.69**	**8.19**	**98.08**

Table 3.6: OOD detection results using ImageNet-100 as the ID dataset and four OOD datasets: iNaturalist, SUN, Places, and Texture. The table compares baseline post-hoc scores, prompt-tuning methods, and SUPREME using CLIP (ViT-B/16). FPR95 and AUROC are reported for each OOD dataset and averaged across datasets; lower FPR95 and higher AUROC indicate better detection. Best results are shown in **bold**, and second-best results are underlined.

Methods	NearOOD		FarOOD	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
MCM	85.37	58.97	69.87	77.11
LoCoOp	87.95	64.97	59.88	83.23
SUPREME	**71.44**	**75.95**	**20.92**	**95.45**

Table 3.7: OOD detection results on the OpenOOD benchmark using ImageNet-1k as the ID dataset. The table separates NearOOD and FarOOD settings, where NearOOD contains semantically closer OOD classes and FarOOD contains more distinct OOD classes. FPR95 and AUROC are reported for each setting; lower FPR95 and higher AUROC indicate better detection. Best results are shown in **bold**.

Methods	FPR95 (%) ↓	AUROC (%) ↑
LoCoOp	85.60	60.73
SUPREME	**64.97**	**75.32**

Table 3.8: OOD detection results on the OpenOOD covariate-shifted ID (CSID) setting using ImageNet-1k as the ID dataset. CSID evaluates whether the detector remains reliable when test samples still belong to ID classes but undergo distribution shifts. FPR95 and AUROC are reported, with best results shown in **bold**.

Methods	ImageNet-10 → ImageNet-20		ImageNet-20 → ImageNet-10	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
Energy	10.30	97.94	16.40	97.37
CLIPN	7.80	98.07	13.67	97.47
MCM	5.00	98.71	17.40	98.87
LoCoOp*	5.60	98.47	5.40	98.92
SUPREME	**4.10**	**98.96**	**3.60**	**99.05**

Table 3.9: Near-OOD comparison between ImageNet-10 and ImageNet-20. The left block treats ImageNet-10 as ID and ImageNet-20 as OOD, while the right block reverses the split. This setting evaluates fine-grained OOD detection because the ID and OOD categories are semantically related subsets of ImageNet. We use CLIP-B/16 for all experiments and report FPR95 and AUROC. Best results are shown in **bold**. "*" represents our reproduction.

and highest AUROC scores across most datasets. Notably, SUPREME achieves an average FPR95 of 20.70 and an average AUROC of 95.38. With all the training data for calculating ID prototypes, SUPREME† is further boosted with an average FPR95 of 20.28 and an average AUROC of 95.54. In Table 3.6, we observe consistent trends using ImageNet-100 as the ID dataset. SUPREME achieves the best average results, with FPR95 of 8.47 and AUROC of 98.08, showing substantial improvements over other methods.

ImageNet-1k on OpenOOD. The results on the OpenOOD benchmark are shown in Table 3.7. Compared to previous methods, SUPREME achieves the best performance in both near and far OOD scenarios. Specifically, SUPREME achieves an FPR95 of 71.44 and AUROC of 75.95 in the near OOD scenario, and an FPR95 of 20.92 and AUROC of 95.45 in the far OOD scenario.

ImageNet-1k on OpenOOD (CSID). We evaluate the performance of SUPREME on the OpenOOD benchmark (CSID) [80] using the ImageNet-1k dataset as the ID data in Table 3.8. The results show that SUPREME outperforms LoCoOp, achieving the lowest FPR95 and highest AUROC, demonstrating its effectiveness in OOD detection across different benchmarks.

ImageNet-10 ↔ ImageNet-20. To evaluate SUPREME under challenging conditions where ID and OOD datasets share similar semantics, we follow previous work [92, 160] and conduct experiments on ImageNet-10 and ImageNet-20 as shown in Table 3.9. SUPREME achieves the best performance, with the lowest FPR95 and highest AUROC scores in both scenarios, demonstrating its robustness.

ITC	BPG	S_{GMP}	ImageNet-1k → OOD Datasets								Average	
			iNaturalist		SUN		Places		Texture			
			FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
			30.91	94.61	37.59	92.57	44.69	89.77	57.77	86.11	42.74	90.77
✓	✓		8.56	98.48	22.36	95.38	33.60	92.55	39.57	**92.75**	26.02	94.79
✓		✓	20.86	95.06	22.04	94.92	30.22	91.64	39.38	89.46	28.12	92.77
	✓	✓	16.10	96.50	23.81	94.53	32.63	91.94	42.41	90.11	28.74	93.27
✓	✓(w/o Bias)	✓	21.55	97.79	16.37	**98.00**	26.25	92.24	34.25	91.56	24.61	94.90
✓	✓	✓	**5.91**	**98.73**	**15.85**	96.08	**26.18**	**93.19**	**34.88**	91.93	**20.70**	**95.38**
w. ResNet-50			21.93	96.22	37.73	91.00	43.27	87.41	28.14	93.51	32.77	92.04
w. ResNet-101			6.13	98.58	20.61	95.03	33.16	90.94	41.31	90.70	25.30	93.81

Table 3.10: Ablation study of the proposed modules and OOD score on ImageNet-1k → four OOD datasets. The first rows compare combinations of ITC, BPG, and S_{GMP} using CLIP (ViT-B/16), and the last rows evaluate the full method with ResNet-50 and ResNet-101 backbones. The table reports FPR95 and AUROC for each OOD dataset and the average across datasets. When S_{GMP} is not combined with BPG, it degenerates to S_{MCM}.

Methods	FPR95 ↓	AUROC ↑
SUPREME	**20.70**	**95.38**
w. MLP	25.24	93.06

Table 3.11: Effect of replacing the linear ITC mappings with a two-layer MLP. The table reports average FPR95 and AUROC on ImageNet-1k → four OOD datasets. The comparison tests whether a more expressive mapping improves image-text alignment or overfits in the few-shot setting.

3.4.3 Ablation Study

We present an ablation study using CLIP (ViT-B/16) on ImageNet-1k → four OOD datasets.

Effectiveness of the proposed modules (BPG and ITC) and scores (S_{GMP}). In Table 3.10, we evaluate the effectiveness of BPG and ITC, as well as the proposed score S_{GMP}, for enhancing OOD detection. We see that excluding BPG (-BPG) raises FPR95 to 28.12 and reduces AUROC to 92.77, highlighting BPG's importance in reducing false positives. Similarly, removing ITC (-ITC) significantly degrades performance, increasing FPR95 to 28.74 and lowering AUROC to 93.27. Furthermore, when the model uses S_{MCM} instead of S_{GMP}, the performance also declines, with FPR95 at 26.02 and AUROC at 94.79. Our full model, incorporating ITC, BPG, and S_{GMP}, achieves the best performance.

Using MLP for image-text consistency. To understand whether treating $f_{i2t}(\cdot)$ and $f_{t2i}(\cdot)$ in ITC

Methods	Top-1 Acc. (%)
MCM	67.0
CLIPN	68.5
ID-like	68.3
SUPREME	**71.8**

Table 3.12: Comparison of in-distribution classification accuracy on the ImageNet-1k validation set. The table reports Top-1 accuracy for several OOD detection methods to verify that improved OOD detection does not come at the cost of degraded ID classification performance.

Methods	ResNet-50		ResNet-101	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
NegPrompt	57.19	85.70	50.35	88.04
LoCoOp	34.45	91.95	-	-
SUPREME	**30.55**	**92.90**	**25.30**	**93.81**

Table 3.13: Performance comparison using different visual backbones. The table reports average FPR95 and AUROC when SUPREME and baseline methods use ResNet-50 or ResNet-101 as the base image encoder, showing whether the proposed method remains effective beyond CLIP ViT-B/16.

as *linear* transformations sufficiently addresses the modality gap, we replace them with a two-layer MLP (Linear-ReLU-Linear) and present the experimental results in Table 3.11 (w. MLP). Results indicate that using MLP negatively impacts performance, which might be because, with limited training data, sophisticated modules such as MLP could overfit.

SUPREME improves the ID performance. OOD detection acts as a preliminary step before models make predictions, aiming to perform a binary classification to distinguish between ID and OOD samples. However, it is also important to understand how OOD methods work with ID classification. In Table 3.12, we compare the Top-1 accuracy of different methods on the ImageNet-1K validation set. SUPREME achieves the highest accuracy, reaching 71.8%. Compared to other OOD detection methods, *e.g.*, CLIPN (68.5%) and ID-like (68.3%), SUPREME shows a considerable gain, underscoring its effectiveness in maintaining high ID accuracy while also excelling in OOD detection.

Effectiveness of SUPREME with different backbones. The results in Table 3.13 show that SUPREME consistently outperforms other methods across different backbones, including ResNet-

ReAct	S_{EOE}	ImageNet-1k → OOD Datasets								Average	
		iNaturalist		SUN		Places		Texture			
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
		5.91	**98.73**	15.85	96.08	26.18	93.19	34.88	91.93	20.70	95.38
✓		8.03	97.93	17.39	95.51	24.66	93.32	**33.21**	**92.13**	20.82	94.72
	✓	**4.85**	98.71	14.90	96.13	23.82	93.68	34.96	91.22	19.63	94.93
✓	✓	7.27	98.32	**11.79**	**97.23**	**20.28**	**94.36**	38.87	90.97	**19.55**	**95.22**

Table 3.14: Performance of SUPREME combined with post-hoc OOD detection methods. The table evaluates whether applying ReAct [187], S_{EOE} [16], or both on top of SUPREME improves FPR95 or AUROC. Results are reported for ImageNet-1k as ID data and four OOD datasets, with averages shown in the last two columns.

50 [66] and ResNet-101 [66]. Specifically, SUPREME achieves 30.55 for FPR95 and 92.90 for AUROC with ResNet-50, and 25.30 for FPR95 and 93.81 for AUROC with ResNet-101, demonstrating its robustness and effectiveness across various model architectures.

Combining SUPREME with post-hoc methods. To understand how SUPREME combines with existing representative post-hoc methods, *i.e.*, ReAct [187] and S_{EOE} [16], we present a systematic study in Table 3.14. We directly apply ReAct and S_{EOE} on our best model as post-hoc methods, which means we do not use them during training. For ReAct, we set the threshold at 0.95, while for S_{EOE}, we follow the standard parameters. We see that using ReAct alone makes the performance worse with an average FPR95 of 20.82 and AUROC of 94.72. On the other hand, S_{EOE} boosts average FPR95 from 20.70 to 19.63. When combining ReAct and S_{EOE}, the FPR95 is boosted to 19.55 on average. However, using ReAct and/or S_{EOE} decreases the average AUROC compared to vanilla SUPREME, showing that these post-hoc methods might only help reduce false positives. The experiments show that SUPREME is compatible with these post-hoc methods without performance drops.

Reduced modality gap (t-SNE visualization, geometric assessment of modality gap). We choose two arbitrary classes from ImageNet-1k and use t-SNE to visualize embeddings in Figure 3.5. This figure shows the distribution of samples after few-shot training with SUPREME. After training, mapped image embeddings I' move toward the corresponding text prototypes P_{txt}, reducing the image–text modality gap compared with the distance between original image embeddings I and text prototypes P_{txt}. At the same time, the two classes remain separated in the projected space, suggesting that the alignment does not collapse class-discriminative structure. Similarly, we observe that test image embeddings I are located close to image prototypes P_{img} computed from the training images of the corresponding classes. This proximity is therefore not a consequence of computing P_{img} from the same test images; rather, it indicates that the training-derived image prototypes serve as stable visual anchors for unseen ID test samples.

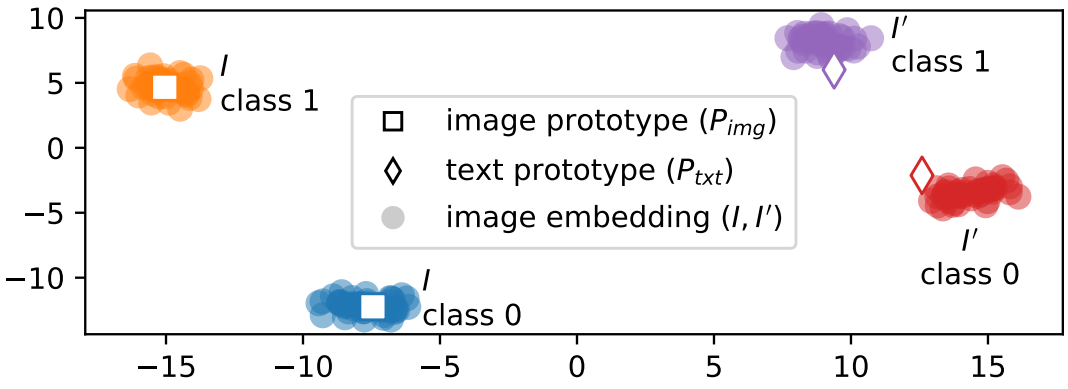


Figure 3.5: t-SNE visualization of embeddings from two arbitrary ImageNet-1k validation classes after few-shot training with SUPREME. The plot shows original image embeddings, mapped image embeddings, ID text prototypes, and ID image prototypes in the same two-dimensional projection. ID image prototypes are computed from the corresponding ImageNet-1k training images, while the displayed image embeddings are from held-out test images; thus their proximity reflects generalization of the training-derived class prototypes to unseen test samples, rather than comparing an image to a prototype computed from itself. After training, the mapped image embeddings move toward the corresponding text-prototype region, while the original test image embeddings remain close to training-derived image prototypes and the two classes remain separated.

Reduced modality gap (quantitative assessment of modality gap). To quantify the modality gap, following Liang et al. [128], we formulate the modality gap as the ℓ_2 distance between the center of image embeddings and (class) text embeddings, *i.e.*, $\left\|\vec{\Delta}_{\text{gap}}\right\| = \left\|\frac{1}{C}\sum_{c=1}^{C} P_{\text{txt},c} - \frac{1}{N_{\text{image}}}\sum_{i=1}^{N_{\text{image}}} I_i\right\|_2$, where $P_{\text{txt},c}$ is the embedding of c-th class, N_{image} is the number of images, and I_i is the image embedding of the i-th image. We use the test set of each dataset for calculating the modality gap. We also calculate the modality gap between the center of image-to-text mapped test image embeddings $f_{i2t}(I)$ and ID text prototypes P_{txt}, and between the center of held-out test image embeddings I and ID image prototypes P_{img}, as shown in the following table. Here, P_{img} is computed only from ID training images and not from the evaluated test images. The results in Table 3.15 demonstrate that SUPREME significantly reduces the modality gap across all datasets. Specifically, compared to the baseline MCM, SUPREME reduces the original modality gap by approximately 25-30%, which shows that the design of SUPREME could effectively reduce the gap. Moreover, when comparing text prototypes with our mapped image embeddings $f_{i2t}(I)$, we achieve a dramatic reduction of over 75% in the modality gap (row 3), indicating that

$\left\|\vec{\Delta}_{\text{gap}}\right\|$	Modality 1	Modality 2	ImageNet (ID) ↓	iNaturalist ↓	Places ↓	Texture ↓	SUN ↓
MCM	P_{txt}	I	0.8679	0.9839	0.9101	0.9373	0.9252
SUPREME	P_{txt}	I	0.6135	0.7159	0.8961	0.9273	0.9002
	P_{txt}	$f_{i2t}(I)$	0.1938	**0.2238**	**0.2376**	**0.2559**	**0.2454**
	P_{img}	I	**0.1348**	0.4531	0.3474	0.3333	0.3447

Table 3.15: Quantifying the modality gap with different methods using CLIP (ViT-B/16) as the base model. Each entry reports the ℓ_2 distance between the centroid of the embeddings listed under "Modality 1" and the centroid of the embeddings listed under "Modality 2"; lower values indicate a smaller modality gap. Text prototypes P_{txt} and image prototypes P_{img} are ID class prototypes constructed from the training data, while I and the mapped embeddings $f_{i2t}(I)$ are computed from the corresponding test split named in each column. The table shows that SUPREME reduces the gap between test image embeddings and ID text prototypes, further reduces the gap after image-to-text mapping, and keeps test image embeddings close to training-derived ID image prototypes.

Methods	Parameters	Trainable Parameters	Image Prototypes	Inference
MCM	124,323,841	0	-	$O(1)$
SUPREME	124,891,683 (↑567,842)	567,842	$O(N_{base})$	$O(1)$

Table 3.16: Comparison of parameter count and inference complexity between MCM and SUPREME using CLIP-B/16 as the base model. The table separates total parameters, trainable parameters, the offline cost of constructing ID image prototypes, and per-sample inference complexity. Image prototypes are computed once from the ID training data and reused for all test samples, so SUPREME keeps constant-time inference with respect to the number of test inputs.

our image-to-text mapping function effectively bridges the gap between modalities. Additionally, the comparison between image ID prototypes and original images (row 4) shows substantial improvement.

Computational cost of the training and inference. Table 3.16 compares parameter efficiency and inference scalability between the baseline MCM method and our proposed SUPREME, using CLIP-B/16 as the base model. While both methods exhibit comparable total parameter sizes, SUPREME introduces 567,842 trainable parameters to achieve its enhanced functionality, representing a negligible increase of 0.46% in total parameters. SUPREME obtains image prototypes with a complexity of $O(N_{base})$, while maintaining inference complexity at $O(1)$ given one test input. Also, the calculation of image prototypes can be conducted before inference, and image

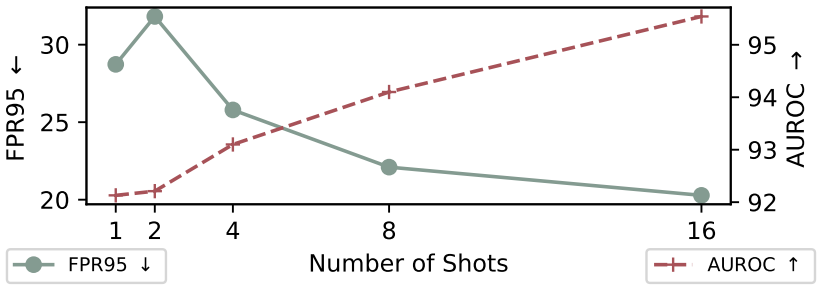


Figure 3.6: Impact of the number of few-shot ID training images per class on OOD detection performance. The plot reports average FPR95 and AUROC as the available tuning data increases, showing how sensitive SUPREME is to the size of the labeled ID subset.

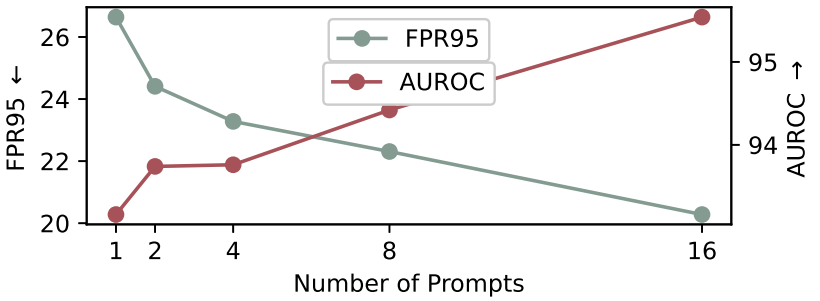


Figure 3.7: Effect of learnable prompt length on OOD detection performance. The plot compares average FPR95 and AUROC for different numbers of learnable context tokens, showing how prompt capacity affects the balance between adaptation and overfitting.

prototypes can be shared among all the test samples, which avoids introducing overhead during inference compared with MCM. It demonstrates the scalability of SUPREME without computational overhead during inference, effectively addressing challenges of learnable flexibility and inference efficiency.

How much data do we need for few-shot tuning? We evaluate the effect of the number of shots for few-shot tuning on the model's performance in Figure 3.6 with SUPREME †. As the number of shots increases from 1 to 16, we observe a consistent improvement in both metrics. Specifically, FPR95 decreases from 28.73 at 1-shot to 20.28 at 16-shot, indicating that more shots lead to better separability between ID and OOD data. Similarly, AUROC improves from 92.13 at 1-shot to a peak of 95.54 at 16-shot, demonstrating enhanced overall detection performance.

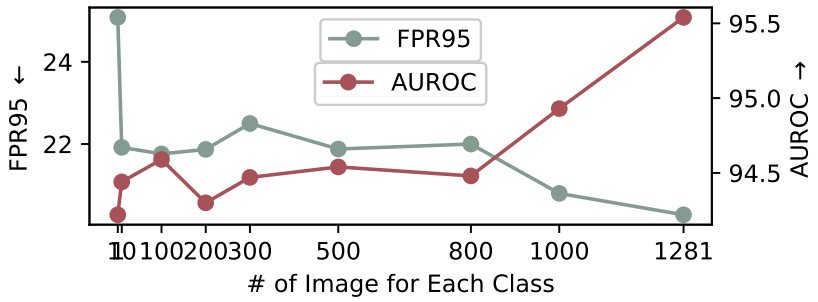


Figure 3.8: Effect of the number of base ID training images used to construct image prototypes. The plot reports average FPR95 and AUROC as more training images are included in the class-wise prototype average. "1281" corresponds to using all available training images per class as the base image set.

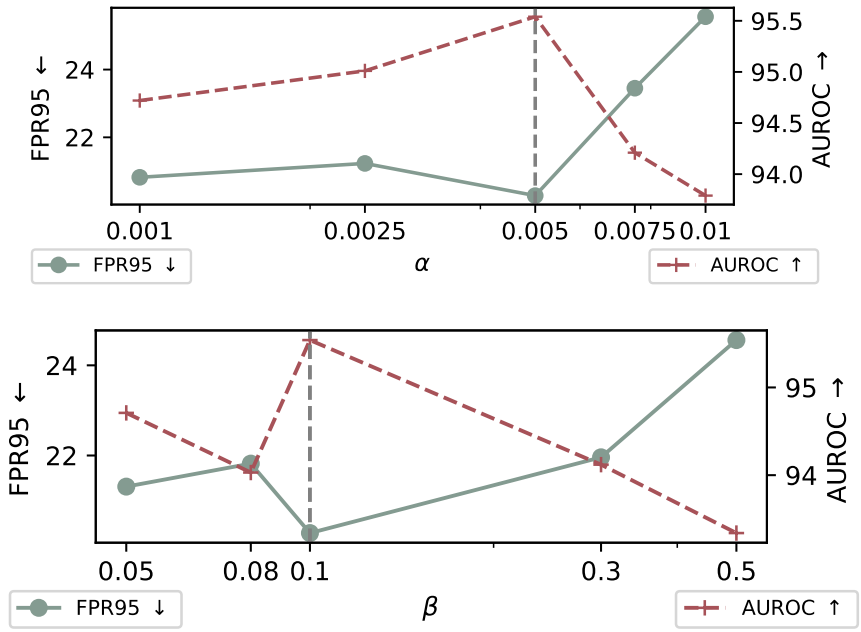
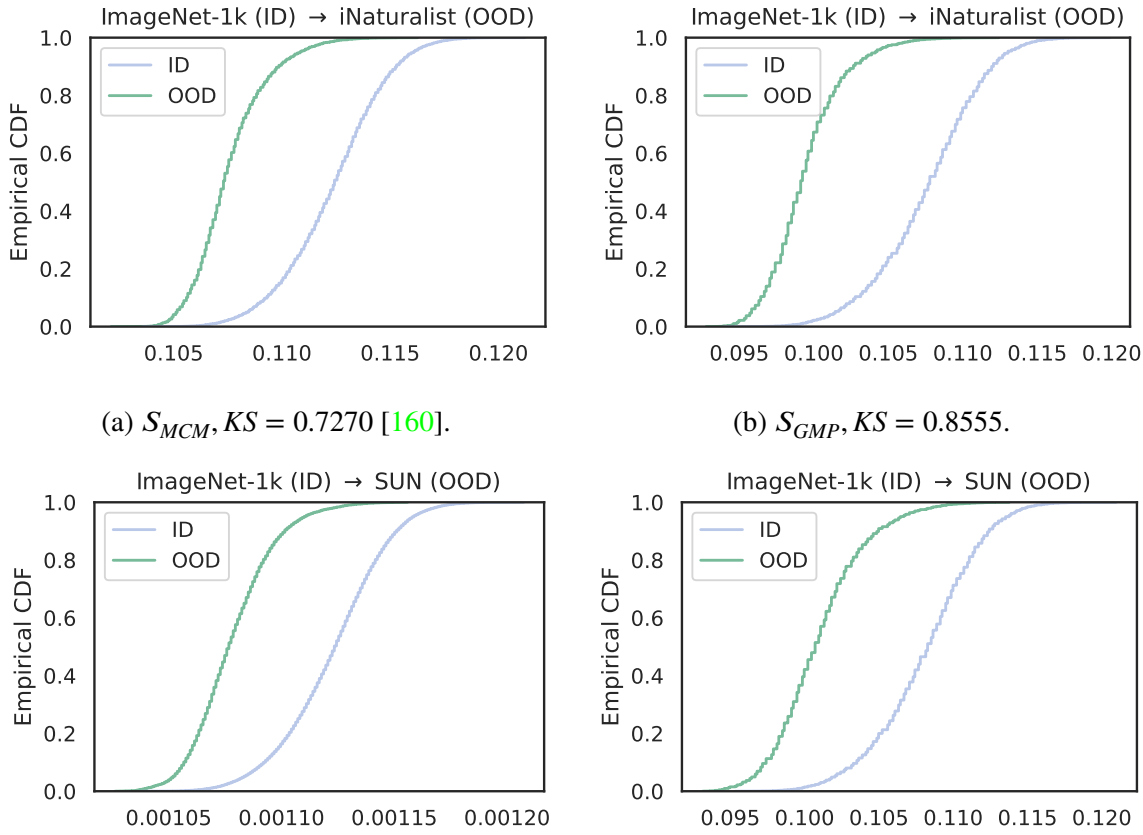


Figure 3.9: Hyperparameter sensitivity of the loss weights α and β. The two plots report average FPR95 and AUROC while varying α, which controls the ITC losses, and β, which controls the image-domain-bias loss.



(a) $S_{MCM}, KS = 0.7270$ [160].

(b) $S_{GMP}, KS = 0.8555$.

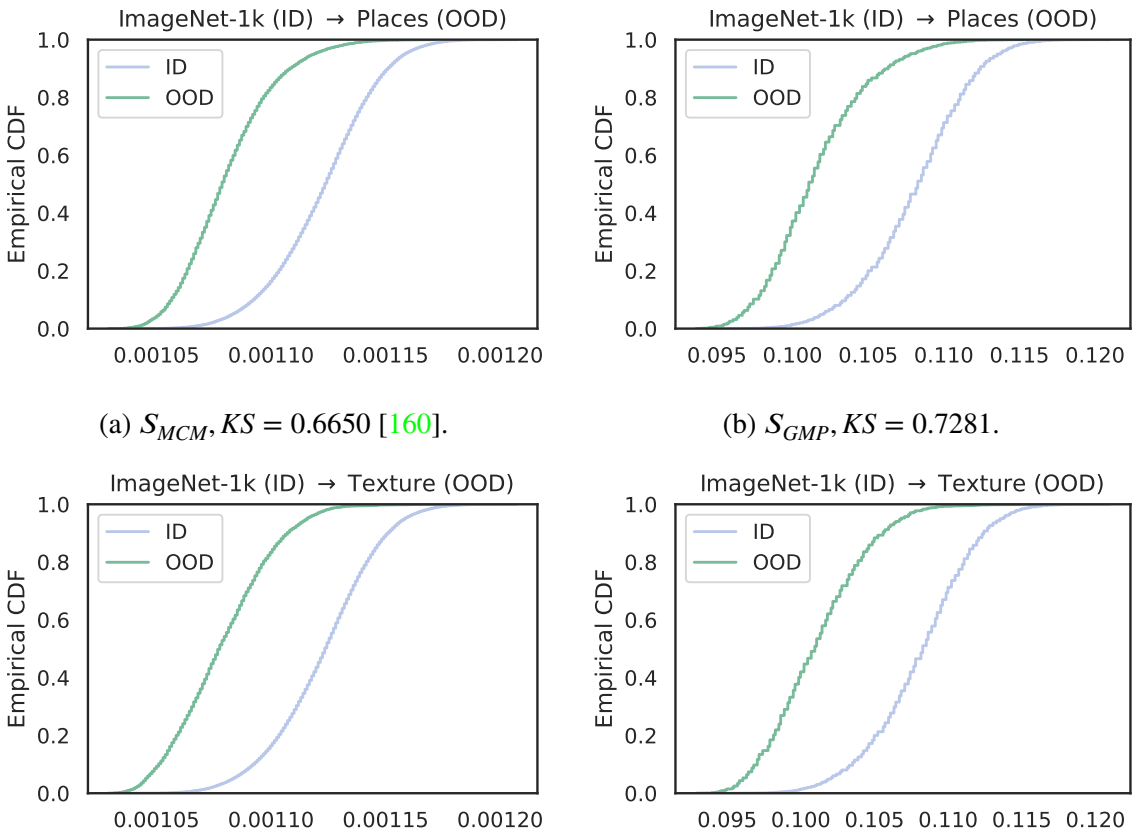
(c) $S_{MCM}, KS = 0.7182$ [160].

(d) $S_{GMP}, KS = 0.7732$.

Figure 3.10: Comparison between S_{MCM} [160] and our proposed S_{GMP} score (Equation (3.20)) on ImageNet-1k as ID data and two OOD datasets, iNaturalist and SUN. The scores are multiplied by 100 for better illustration. Each subfigure shows the empirical cumulative distributions of ID and OOD scores for one score function and reports the Kolmogorov–Smirnov (KS) statistic. Higher KS values indicate stronger separation between ID and OOD score distributions. Best viewed in color.

These results highlight the robustness of SUPREME in leveraging additional labeled examples, with diminishing returns on FPR95 as data size increases. While higher shot numbers provide the best performance, the model performs reasonably well even with a limited number of shots, showcasing its flexibility in low-data scenarios.

Impact of the length of prompts. In the main experiments, we follow previous works [162,



(a) $S_{MCM}, KS = 0.6650$ [160].

(b) $S_{GMP}, KS = 0.7281$.

(c) $S_{MCM}, KS = 0.6760$ [160].

(d) $S_{GMP}, KS = 0.6848$.

Figure 3.11: Comparison between S_{MCM} [160] and our proposed S_{GMP} score (Equation (3.20)) on ImageNet-1k as ID data and two OOD datasets, Places and Texture. The scores are multiplied by 100 for better illustration. Each subfigure shows the empirical cumulative distributions of ID and OOD scores for one score function and reports the Kolmogorov–Smirnov (KS) statistic. Higher KS values indicate stronger separation between ID and OOD score distributions. Best viewed in color.

253, 254] and set the length of prompts to $L = 16$. Here, we would like to validate that 16 is the best choice for SUPREME. We conducted experiments with SUPREME † and results shown in Figure 3.7 indicate that among the choices from 1 to 16, 16 obtains the lowest FPR95 and highest AUROC. Increasing the length of prompts improves the OOD detection performance.

Impact of base image size for constructing ID image prototypes. In S_{GMP}, we use a set of

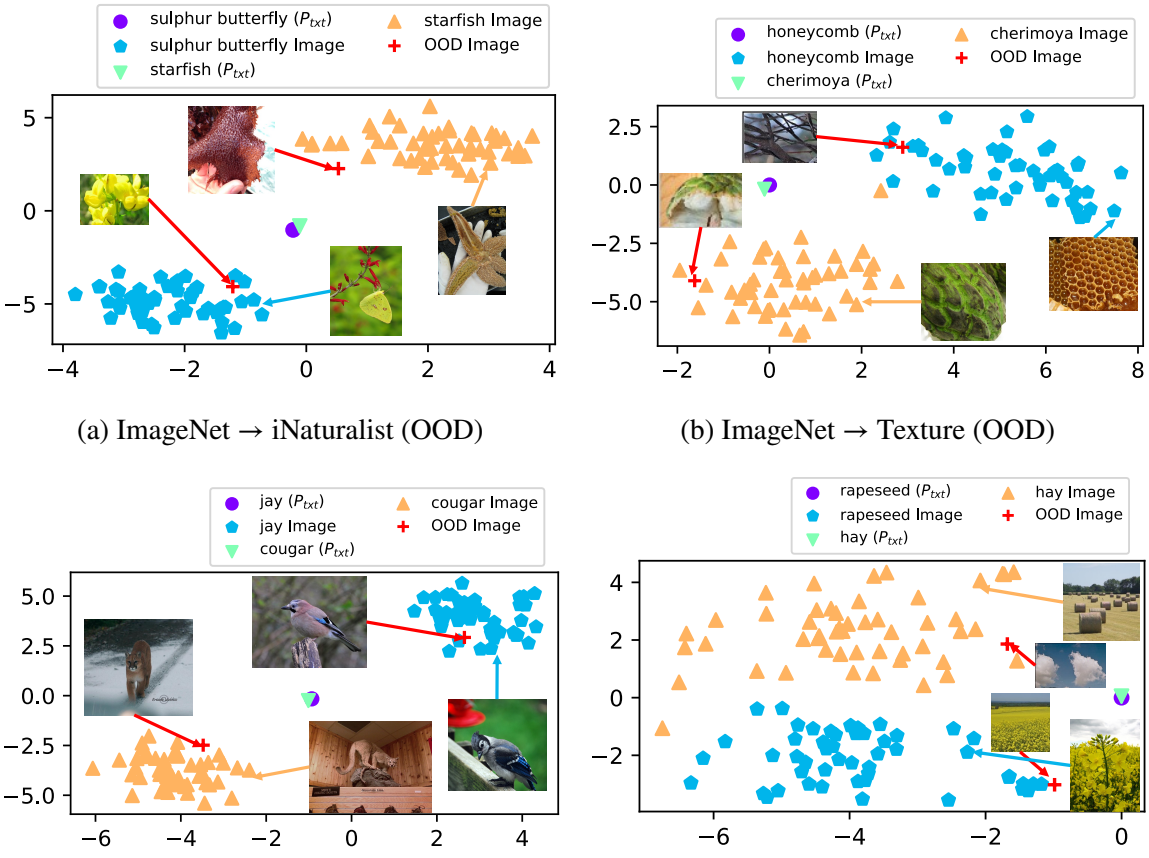
base images to construct image prototypes $\{P_{img,c}\}_{c=1}^{C}$. To understand how the size of base images affects performance, we conduct experiments varying its size by randomly selecting a subset of the training images. The results in Figure 3.8 show that by increasing the number of images for each class, the performance of SUPREME is improved, as image prototypes are closer to the real distribution. FPR95 and AUROC reach the best when using all the images. However, as shown in Table 3.5, the performance is already competitive when using the same set of few-shot data for fine-tuning, indicating that the model is robust to the size of base images.

Hyperparameter (α and β) sensitivity. We analyze the effect of the trade-off hyperparameters α and β (Section 3.3.2) on the model's performance, as shown in Figure 3.9 with SUPREME †. For α, we observe a trade-off between FPR95 and AUROC as its value increases. Specifically, as α grows from 0.001 to 0.01, FPR95 decreases initially from 20.82 to 20.28 at $\alpha = 0.005$, indicating improved separability, but then rises to 25.56 at $\alpha = 0.01$. Meanwhile, AUROC peaks at 95.54 for $\alpha = 0.005$ before declining, indicating that excessively large values of α may overemphasize certain features, leading to diminished overall performance. Similarly, β demonstrates a comparable trend. The FPR95 achieves its lowest value of 20.28 at $\beta = 0.1$, while AUROC peaks at 95.54. The results of both hyperparameters suggest that SUPREME is not sensitive to the parameters and that moderate values ($\alpha = 0.005, \beta = 0.1$) balance the trade-off between false positive rate and overall discriminative ability.

Effectiveness of S_{GMP}. We visualize the empirical cumulative distribution of S_{MCM} and our S_{GMP} in Figures 3.10 and 3.11 with the ID data of ImageNet-1k and the OOD data of iNaturalist using the same model (SUPREME). The evaluation uses our best model trained on the ImageNet-1k (ID) dataset, with iNaturalist, SUN, Places, and Texture serving as OOD datasets. The empirical CDF clearly shows that S_{GMP} improves the gap between ID and OOD data more effectively. Furthermore, the KS test statistics improve ceonsiderably from 0.7270, 0.7182, 0.6650, and 0.6760 (S_{MCM}) to 0.8724, 0.7798, 0.7219, and 0.6874 (S_{GMP}) across the four OOD datasets. These results indicate that S_{GMP} achieves better separation between ID and OOD scores compared to S_{MCM}.

3.4.4 Visualizing OOD Examples

In Figures 3.3 and 3.12, we visualize the embedding space using t-SNE projection across four different OOD datasets (iNaturalist, Texture, Places, and SUN) generated with CLIP (ViT-B/16). A pattern emerges: OOD images (red crosses) are consistently positioned closer to ID text prototypes than their corresponding ID images across all four scenarios. For instance, in Figure 3.12(a), OOD images (red cross) are positioned closer to the "sulphur butterfly" text prototype (purple circle) than the corresponding ID images (blue pentagon). This counterintuitive positioning is systematically observed across diverse semantic categories—from biological entities (sulphur



(a) ImageNet → iNaturalist (OOD)

(b) ImageNet → Texture (OOD)

(c) ImageNet → Places (OOD)

(d) ImageNet → SUN (OOD)

Figure 3.12: t-SNE visualizations of CLIP (ViT-B/16) embeddings showing OOD false-positive patterns across four OOD datasets. Each panel compares ImageNet-1k ID image embeddings from the test set, OOD image embeddings, and ID text prototypes generated from class-name prompts. The examples highlight cases where OOD images are positioned closer to an ID text prototype than genuine ID images are, motivating the use of image prototypes and image-text alignment in SUPREME.

butterfly, starfish) to textures (honeycomb, cherimoya) and environmental scenes (jay/cougar, rapeseed/hay).

This phenomenon reveals a fundamental limitation in current embedding spaces: *positional proximity does not guarantee semantic similarity*. The t-SNE visualization demonstrates that geometric distance in the learned representation space can be misleading, as semantically unre-

lated OOD samples may appear closer to class prototypes than genuine in-distribution examples. This misalignment between spatial proximity and semantic similarity undermines the reliability of distance-based OOD detection methods. It motivates our design of multimodal prototypes that better capture the true semantic boundaries of in-distribution data.

3.4.5 Ablations

3.4.6 Details of Benchmarking Datasets

ImageNet-100, ImageNet-10, and ImageNet-20. Following MCM [160], we choose 100/10/20 classes from ImageNet-1k [101] to form ImageNet-100, ImageNet-10, and ImageNet-20. The chosen classes for each dataset are as follows:

- **ImageNet-100**: n03877845, n03000684, n03110669, n03710721, n02825657, n02113186, n01817953, n04239074, n02002556, n04356056, n03187595, n03355925, n03125729, n02058221, n01580077, n03016953, n02843684, n04371430, n01944390, n03887697, n04037443, n02493793, n01518878, n03840681, n04179913, n01871265, n03866082, n03180011, n01910747, n03388549, n03908714, n01855032, n02134084, n03400231, n04483307, n03721384, n02033041, n01775062, n02808304, n13052670, n01601694, n04136333, n03272562, n03895866, n03995372, n06785654, n02111889, n03447721, n03666591, n04376876, n03929855, n02128757, n02326432, n07614500, n01695060, n02484975, n02105412, n04090263, n03127925, n04550184, n04606251, n02488702, n03404251, n03633091, n02091635, n03457902, n02233338, n02483362, n04461696, n02871525, n01689811, n01498041, n02107312, n01632458, n03394916, n04147183, n04418357, n03218198, n01917289, n02102318, n02088364, n09835506, n02095570, n03982430, n04041544, n04562935, n03933933, n01843065, n02128925, n02480495, n03425413, n03935335, n02971356, n02124075, n07714571, n03133878, n02097130, n02113799, n09399592, n03594945.
- **ImageNet-10**: n04552348, n04285008, n01530575, n02123597, n02422699, n02107574, n01641577, n03417042, n02389026, n03095699.

- **ImageNet-20**: n04147183, n02951358, n02782093, n04389033, n03773504, n02917067, n02317335, n01632458, n01630670, n01631663, n02391049, n01693334, n01697457, n02120079, n02114367, n02132136, n03785016, n04310018, n04266014, n04252077.

OOD datasets. Similarly, we use subsets from iNaturalist [196], SUN [219], Places [251], and Texture [32] as OOD datasets, which are created by Huang and Li [80].

- **iNaturalist** contains images from the natural world spanning 5089 classes in 13 supercategories, such as Plantae, Insecta, Aves, and Mammalia. Following Huang and Li [80], we use the subset of 110 plant classes not present in ImageNet-1k as the OOD test set. The classes are as follows: *Coprosma lucida, Cucurbita foetidissima, Mitella diphylla, Selaginella bigelovii, Toxicodendron vernix, Rumex obtusifolius, Ceratophyllum demersum, Streptopus amplexifolius, Portulaca oleracea, Cynodon dactylon, Agave lechuguilla, Pennantia corymbosa, Sapindus saponaria, Prunus serotina, Chondracanthus exasperatus, Sambucus racemosa, Polypodium vulgare, Rhus integrifolia, Woodwardia areolata, Epifagus virginiana, Rubus idaeus, Croton setiger, Mammillaria dioica, Opuntia littoralis, Cercis canadensis, Psidium guajava, Asclepias exaltata, Linaria purpurea, Ferocactus wislizeni, Briza minor, Arbutus menziesii, Corylus americana, Pleopeltis polypodioides, Myoporum laetum, Persea americana, Avena fatua, Blechnum discolor, Physocarpus capitatus, Ungnadia speciosa, Cercocarpus betuloides, Arisaema dracontium, Juniperus californica, Euphorbia prostrata, Leptopteris hymenophylloides, Arum italicum, Raphanus sativus, Myrsine australis, Lupinus stiversii, Pinus echinata, Geum macrophyllum, Ripogonum scandens, Echinocereus triglochidiatus, Cupressus macrocarpa, Ulmus crassifolia, Phormium tenax, Aptenia cordifolia, Osmunda claytoniana, Datura wrightii, Solanum rostratum, Viola adunca, Toxicodendron diversilobum, Viola sororia, Uropappus lindleyi, Veronica chamaedrys, Adenocaulon bicolor, Clintonia uniflora, Cirsium scariosum, Arum maculatum, Taraxacum officinale officinale, Orthilia secunda, Eryngium yuccifolium, Diodia virginiana, Cuscuta gronovii, Sisyrinchium montanum, Lotus corniculatus, Lamium purpureum, Ranunculus repens, Hirschfeldia incana, Phlox divaricata laphamii, Lilium martagon, Clarkia purpurea, Hibiscus moscheutos, Polanisia dodecandra, Fallugia paradoxa, Oenothera rosea, Proboscidea louisianica, Packera glabella, Impatiens parviflora, Glaucium flavum, Cirsium andersonii, Heliopsis helianthoides, Hesperis matronalis, Callirhoe pedata, Crocosmia × crocosmiiflora, Calochortus albus, Nuttallanthus canadensis, Argemone albiflora, Eriogonum fasciculatum, Pyrrhopappus pauciflorus, Zantedeschia aethiopica, Melilotus officinalis, Peritoma arborea, Sisyrinchium bellum, Lobelia siphilit-*

ica, Sorghastrum nutans, Typha domingensis, Rubus laciniatus, Dichelostemma congestum, Chimaphila maculata, Echinocactus texensis.

- **SUN** contains 899 classes that cover indoor, urban, and natural places. We use the subset that contains 50 natural objects that do not overlap with ImageNet-1k. The classes we use are *badlands, bamboo forest, bayou, botanical garden, canal (natural), canal (urban), catacomb, cavern (indoor), corn field, creek, crevasse, desert (sand), desert (vegetation), field (cultivated), field (wild), fishpond, forest (broadleaf), forest (needleleaf), forest path, forest road, hayfield, ice floe, ice shelf, iceberg, islet, marsh, ocean, orchard, pond, rainforest, rice paddy, river, rock arch, sky, snowfield, swamp, tree farm, trench, vineyard, waterfall (block), waterfall (fan), waterfall (plunge), wave, wheat field, herb garden, putting green, ski slope, topiary garden, vegetable garden, formal garden.*

- **Places** contains photos labeled with scene semantic categories from three macro-classes: Indoor, Nature, and Urban. We use a subset sampled from 50 categories that are not present in ImageNet-1k. The classes we use are *badlands, bamboo forest, canal (natural), canal (urban), corn field, creek, crevasse, desert (sand), desert (vegetation), desert road, field (cultivated), field (wild), field road, forest (broadleaf), forest path, forest road, formal garden, glacier, grotto, hayfield, ice floe, ice shelf, iceberg, igloo, islet, japanese garden, lagoon, lawn, marsh, ocean, orchard, pond, rainforest, rice paddy, river, rock arch, ski slope, sky, snowfield, swamp, swimming hole, topiary garden, tree farm, trench, tundra, underwater (ocean deep), vegetable garden, waterfall, wave, wheat field*.

- **Texture** contains images of textures and abstracted patterns. As no categories overlap with ImageNet1k, we use the entire dataset.

3.5 Conclusion and Limitations

In this chapter, we introduced SUPREME, a novel VLM-based few-shot tuning OOD detection framework. SUPREME incorporates two key modules: BPG, which enhances image-text fusion, and ITC, which minimizes the inter- and intra-modal distances. We also proposed the generalized multi-modal prototype OOD score, S_{GMP}, using multi-modal prototypes and embeddings to improve robustness and reduce false positives. Experimental results show that SUPREME consistently outperforms existing VLM-based OOD detection methods.

Limitations. As in some scenarios, the ID data may not be available, exploring methods such as image generation or retrieval from a web-scale dataset could help alleviate the dependency on the ID data. Moreover, as SUPREME uses the Gaussian distribution for its simplicity, exploring

more advanced distribution models may further improve the performance. Finally, we have only explored multi-modal positive/ID prototypes. Introducing negative/OOD prototypes may further enhance our framework's versatility.

Having established reliable OOD detection with calibrated image-text representations, the next step is to assign semantic meaning to detected novel objects. Chapter 4 addresses this recognition stage through open-vocabulary 3D semantic segmentation, where a test-time candidate label set can include categories unseen during training.

Chapter 4

AIDE: Improving 3D Open-Vocabulary Semantic Segmentation by Aligned Vision-Language Learning

The previous chapter studied OOD detection as a binary decision: whether an input should be accepted as in-distribution or flagged as unfamiliar. This chapter moves to the next level of OOD handling. Once a system detects that an object or scene element is unfamiliar, it must still assign a meaningful semantic label to that object so downstream modules can reason about it. We study this recognition problem in 3D point clouds through open-vocabulary 3D semantic segmentation, where the test-time label vocabulary can include categories not annotated during training.

4.1 Introduction

3D semantic segmentation has attracted extensive research attention [29, 102, 191, 224, 242]. To handle real-world scenarios where new categories frequently emerge [111, 169, 177], researchers explore the open-vocabulary setting [43, 67, 226], which requires models to recognize novel classes beyond the typical limited training label space. The background of closed-set and open-vocabulary 3D segmentation is introduced in Section 2.3; this chapter focuses on the specific alignment bottleneck that appears when the open-vocabulary label interface is provided by a 2D VLM. Unlike the OOD detection setting in Chapter 3, where the input is an image and the question is whether it should be rejected, open-vocabulary 3D segmentation requires assigning a label to every point or region in a sparse 3D scene. This makes the alignment problem denser and

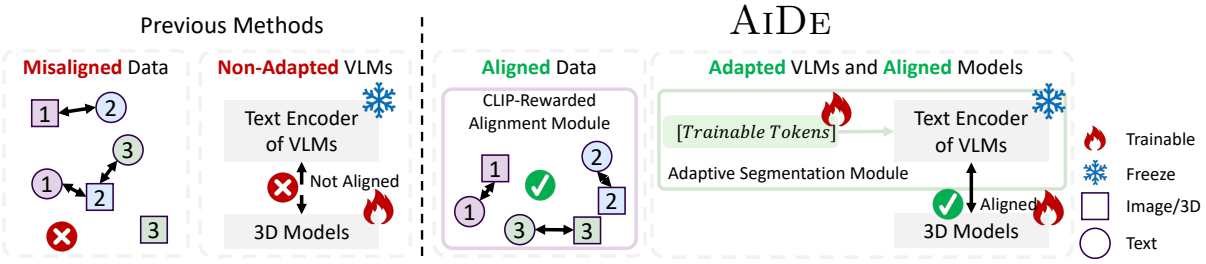


Figure 4.1: Previous methods use misaligned paired data (*e.g.*, image/point cloud 1 is closest to text 2) and freeze the text encoder trained on 2D benchmark datasets, leading to misalignment between text encoders and 3D models and suboptimal performance. Our proposed AIDE aligns text encoders with 3D models by (i) generating better aligned 3D-image-to-text training data (CLIP-Rewarded Alignment Module) and (ii) optimizing deep trainable prompt tokens (Adaptive Segmentation Module).

more structured: the model must transfer semantics from text to 3D geometry, usually through intermediate 2D views, while preserving local spatial detail.

Due to the lack of large-scale 3D-image-text pairs, instead of training a 3D-language model from scratch, recent works [20, 43, 171] propose to transfer the knowledge encoded in pretrained 2D vision-language foundation models (VLMs), *e.g.*, CLIP [174] and SAM [100], to 3D semantic segmentation models by aligning them with generated 3D-scene-image-to-text data. After the alignment, the 3D model with the VLM text encoder can handle unlimited categories. As a pioneering work, PLA [43] distills knowledge through captioning multi-view images [34, 202] of 3D scenes, allowing explicit association between 3D point clouds and captions. The reason this transfer route is attractive is practical: annotated 3D datasets are much smaller and less diverse than web-scale 2D image-text datasets, while VLM text encoders already provide a language-indexed semantic space. However, the transfer route also introduces an additional dependency that is absent in standard 2D open-vocabulary recognition. The captions, rendered views, and point clouds must describe the same scene content; if any part of this chain is weak, the 3D model can learn a noisy correspondence between geometry and language even when the underlying VLM is strong. Following PLA, RegionPLC [226] boosts the performance by using dense object-level associations for better alignment.

However, these methods suffer from several major issues, making them suboptimal. First, the generated low-quality captions do not accurately match the corresponding image data of the 3D scenes semantically. As shown in Table 4.1a, when view images are used to retrieve captions using CLIP, recall@10 is lower than 0.034, indicating that less than 3.4% of images have their corresponding generated captions ranked in the top 10. This misalignment between captions

and their corresponding images of 3D scenes—possibly stemming from the distribution shifts between 3D datasets [3,36] and 2D datasets [85,131] used for training captioners—hinders precise alignment between 3D models and text encoders. It further affects hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$ as they are proportional to recall@10 (*i.e.*, the quality of data). Second, as datasets [30,39,85,131] used for training VLMs differ from 3D datasets [3,13,36,186], the alignment between 3D models and text encoders of VLMs exhibits excessive sensitivity to the prompts used for adapting text encoders, further leading to unreliable performance as shown in Table 4.1b.

To address these issues, we propose a novel **A**l**I**gned 3**D** Open-Vocabulary S**E**mantic Segmentation framework, called AIDE, as shown in Figure 4.1. Our key idea is to collect high-quality well-aligned data and adapt the text encoder to 3D contexts for better alignment between 3D models and text encoders, further improving the performance. Specifically, to generate well-aligned 3D-scene-image-to-text data pairs, our CLIP-rewarded alignment module first generates a variety of rich captions using different temperatures for enhancing caption diversity and capturing rich details in the 2D images of the 3D scenes (temperature-based generation)[1]. Then, to encourage rich associations between 3D and text, we propose the CLIP-rewarded sampling method, which samples captions based on their similarity to the 3D-scene image in each training iteration. Next, drawing inspiration from previous methods [43, 226], we employ a hierarchical alignment strategy, enabling alignment from scene-level (coarse-grained) to entity-level (fine-grained). Furthermore, to adapt text encoders for 3D semantics, the adaptive segmentation module extends beyond the popular visual prompt tuning methods [89, 253] by incorporating trainable tokens not only in the input space but also across each transformer layer within text encoders, thereby enhancing its flexibility and effectiveness.

Extensive experiments on three representative benchmarks, *i.e.*, ScanNet [36], S3DIS [3], and one outdoor dataset (nuScenes [13]), demonstrate the superiority of AIDE compared to existing approaches across various metrics, highlighting its robustness and versatility.

4.2 AIDE

4.2.1 Problem Definition

3D open-vocabulary semantic segmentation targets recognizing unseen categories, *i.e.*, those unannotated during training. Following the general setting reviewed in Section 2.3, the training set contains annotations only for base categories, while evaluation includes both base and

[1]Here, deterministic refers to the caption decoder's word-sampling process during offline data generation, not to the final segmentation inference. Higher temperatures lead to more diverse captions, while lower temperatures produce less stochastic captions.

Captioner	R@10 ↑	hIoU ↑	mIoU$^{\mathcal{B}}$ ↑	mIoU$^{\mathcal{N}}$ ↑
OFA [202]	0.034	65.6	68.3	63.1
VIT-GPT2 [34]	0.004	65.3	68.3	62.4
Best	0.129	68.9	69.6	68.2
Ours-Sampling	**0.151**	**70.3**	**69.9**	**70.6**

(a) Impact of different captioning methods.

Prompt Templates	hIoU ↑	mIoU$^{\mathcal{B}}$ ↑	mIoU$^{\mathcal{N}}$ ↑
Identity	65.3	68.3	62.4
Simple	64.3	67.7	61.3
Full-ImageNet	64.6	68.1	61.5
Ours	**66.3**	**70.2**	**62.8**

(b) Impact of different prompt templates.

Table 4.1: Semantic segmentation performance on ScanNet (B15/N4) [36] using PLA [43]. Metrics include harmonic IoU (hIoU), base-category mIoU (mIoU$^{\mathcal{B}}$), and novel-category mIoU (mIoU$^{\mathcal{N}}$). Table 4.1a shows that segmentation performance correlates with recall at 10 (R@10) in view image-to-caption retrieval. "Best" indicates the selection of most-aligned captions generated by OFA and ViT-GPT2, based on cosine similarity. "Ours-Sampling" uses captions generated by our CLIP-rewarded alignment module. Table 4.1b highlights the importance of prompts for adapting 2D VLMs; "Ours" is our proposed adaptive segmentation module.

novel categories. In this chapter, the novel categories are not discovered without constraints; they are provided as candidate text labels at test time and encoded by the VLM text encoder. The main difficulty is therefore not only to learn discriminative 3D point features, but also to make those features comparable with language embeddings for labels that have no point-level supervision during training. Each 3D scene is represented by $(P, Y) = (\{\mathbf{p}_i\}_{i\in[N]}, \{\mathbf{y}_i\}_{i\in[N]})$, where N is the number of points, $\mathbf{p}$ is a point, $\mathbf{y} \in \mathcal{Y}$ represents the corresponding label, and $\mathcal{Y}$ contains all possible categories. Additionally, we have camera images $X = \{\mathbf{x}_i\}_{i\in[N_{\text{img}}]}$ for each 3D scene, where N_{img} is the number of images available for the scene P. $\mathcal{Y}$ is divided into base and novel classes, *i.e.*, $\mathcal{Y}_{\mathcal{B}}$ and $\mathcal{Y}_{\mathcal{N}}$, respectively. During training, all points P are available, while only the labels $Y_{\mathcal{B}}$ in the base classes $\mathcal{Y}_{\mathcal{B}}$ are accessible. Point annotations for novel classes and the names of these novel classes are not used during training. During inference, the model is provided with a candidate label set that includes both base and novel class names and is required to assign each point to the best-matching label.

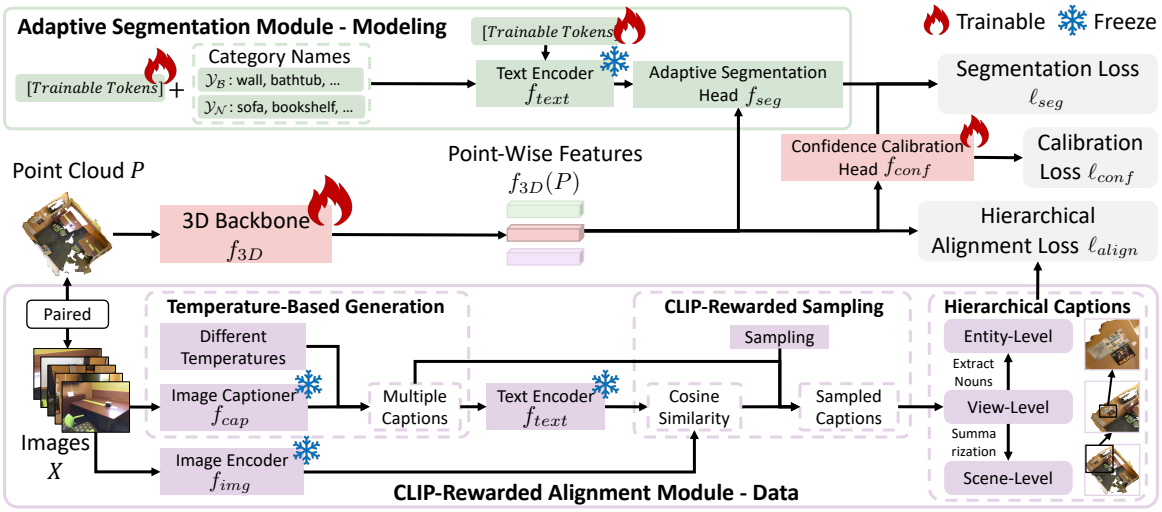


Figure 4.2: The illustration of our AIDE with two proposed modules, *i.e.*, CLIP-rewarded alignment module (Section 4.2.3) for enhancing the quality of 3D-text data pairs and adaptive segmentation module (Section 4.2.4) for adapting text encoders. **CLIP-Rewarded Alignment Module.** We first generate numerous captions using the temperature-based generation strategy, then sample captions based on their similarity to the images (CLIP-rewarded sampling), and finally align 3D point clouds and captions hierarchically. **Adaptive Segmentation Module.** To automatically find the most suitable prompt for adapting text encoders into 3D scenarios, AIDE extends prompt tuning [89] by incorporating learnable tokens in the input space and each layer of the text encoder f_{text}. A confidence calibration head is employed for balancing the confidence on base and novel categories, as models tend to overestimate their certainty in base categories [43, 121, 150].

4.2.2 Preliminaries, Problems, and Solutions

Following previous works [43, 226], in AIDE, point-wise features $f_{\text{3D}}(P) \in \mathcal{R}^{N \times D}$ are extracted by a 3D backbone $f_{\text{3D}}(\cdot)$, where D represents feature dimensions. Then, a semantic segmentation classifier $f_{\text{seg}}(\cdot)$ is employed, producing the point-wise segmentation results $f_{\text{seg}}(f_{\text{3D}}(P))$. AIDE incorporates a text encoder $f_{\text{text}}(\cdot)$, *i.e.*, the text encoder of CLIP [174], to generate embeddings for captions and classes. To align 3D and text encoders, paired 3D-scene-image-to-text data are generated by an off-the-shelf captioner $f_{\text{cap}}(\cdot)$, *e.g.*, OFA [202], detailed in Section 4.2.3. The model is illustrated in Figure 4.2.

Open-vocabulary segmentation [43, 67, 171]. By leveraging a classifier $f_{\text{seg}}(\cdot)$ with class-wise

embedding $C \in \mathcal{R}^{|\mathcal{Y}|\times D}$ as weights, we obtain point-wise semantic segmentation prediction $\hat{Y}$,

$$\hat{Y} = f_{\text{seg}}\left(f_{\text{3D}}(P)\right) = f_{\text{3D}}(P)C^{\top} \in \mathcal{R}^{N\times|\mathcal{Y}|}, \tag{4.1}$$

where C is generated by the text encoder $f_{\text{text}}(\cdot)$ with prompts, *e.g.*, "a photo of a [CLASS]", and class names, *e.g.*, table and chair. During training, the embedding C only contains the embeddings of the base categories $\mathcal{Y}_{\mathcal{B}}$. We employ the Cross-Entropy loss $CE(\cdot,\cdot)$ to train the model,

$$\ell_{\text{seg}} = \sum_{i\in[N]} \mathcal{I}(\mathbf{y}_i \in \mathbf{Y}_{\mathcal{B}}) CE\left(\hat{Y}_i, \mathbf{y}_i\right), \tag{4.2}$$

where $\mathcal{I}(cond)$ is the indicator function ($\mathcal{I}(cond) = 1$ when $cond$ is true) and $\hat{Y}_i$ is the predicted label for the i-th point.

Confidence-based calibration. To counteract the tendency of models to exhibit overconfidence in base categories while ignoring novel categories [25, 43, 226], a confidence-based calibration branch $f_{\text{conf}}(\cdot)$ is employed. It dynamically balances the confidence level by predicting whether a point falls in the base categories. During training, we employ the binary cross-entropy loss $\text{BCE}(\cdot,\cdot)$ as,

$$\ell_{\text{conf}} = \text{BCE}(f_{\text{conf}}(f_{\text{3D}}(P)), Y_{\text{conf}}), \tag{4.3}$$

where Y_{conf} is the binary label (1 represents the point belonging to the base categories and vice versa). At inference, confidence calibration is applied to balance predictions on base and novel categories as $\tilde{Y} = [f_{\text{conf}}(f_{\text{3D}}(P))\hat{Y}_{\mathcal{B}}, (1 - f_{\text{conf}}(f_{\text{3D}}(P)))\hat{Y}_{\mathcal{N}}]$, where $\hat{Y}_{\mathcal{B}}$ and $\hat{Y}_{\mathcal{N}}$ represent the predictions on base and novel categories, respectively.

In this chapter, we identify two problems in the current open-vocabulary segmentation pipeline and propose corresponding solutions [43, 67, 171].

Problem 1 (data). Our preliminary results (Table 4.1a) show that 3D-scene-image-text paired data for 3D scenes are mismatched, hurting alignment between text encoders and 3D models. **Our solution:** we propose the CLIP-rewarded alignment module in Section 4.2.3 with temperature-based caption generation and CLIP-rewarded sampling.

Problem 2 (modelling). Our preliminary results (Table 4.1b) show that alignment between text encoders and 3D models is highly sensitive to prompts for generating C, possibly because text encoders are trained on 2D datasets that differ from 3D benchmarks. **Our solution:** we propose the adaptive segmentation module in Section 4.2.4 to adapt the text encoder to 3D contexts with learnable tokens.

4.2.3 CLIP-Rewarded Alignment — Data Generation

Previous methods [20, 43, 171] align text encoders with 3D models using generated 3D-scene-image-to-text data, enabling 3D models to recognize novel categories. However, as shown in Table 4.1a, the images and the generated captions are mismatched, which leads to unsatisfying performance across both base and novel categories. A straightforward solution to this issue is to fine-tune captioners. However, the lack of existing large-scale 3D-vision-text data makes it impractical. To solve this issue and generate high-quality data, we introduce a novel CLIP-rewarded alignment module as shown in Figure 4.2. Specifically, we first generate several captions per image by the proposed **temperature-based generation**, which adjusts the temperature to encourage image captioners [17, 31, 34, 76, 202, 215] to produce diverse captions to capture details in the images. Next, to enable rich associations between 3D and text, we propose a novel **CLIP-rewarded sampling** method, which selects captions based on their similarity to the paired image data. Finally, inspired by previous methods [15,43,99,147,183,226,255], to enable coarse-to-fine alignment, we adopt hierarchical 3D-text alignment [43, 67].

Temperature-based caption generation. A widely accepted principle of text or caption generation [4, 194, 202] is that temperature controls the fidelity of the generated text. For image captioners, the i-th word w_i of generated captions are sampled from the probability $\sigma_{v \in \text{Vocab}}(P(v \mid I, w_{1:i-1})/\gamma)$, where γ is the temperature, w_i is the i-th word, I is the input image, $\sigma(\cdot)$ is the softmax function, $w_{1:i-1}$ represents the words before the i-th word, Vocab is the whole vocabulary, and $P(v \mid I, w_{1:i-1})$ is the prediction of v. As γ approaches 0, the caption decoder's sampling distribution becomes sharper and generation becomes less stochastic; as γ increases, generation becomes more diverse. By leveraging this principle, the temperature-based generation varies temperatures γ during captioning to foster caption diversity and reduce the mismatch in image-text data pairs. Specifically, we fix the captioner $f_{\text{cap}}(\cdot)$ and vary the temperature γ to get a set of captions $T_i = \{\text{caption}_j\}_{j \in [N_{\text{cap}}]}$ for the i-th view-image of the point cloud P, where N_{cap} is the number of captions.

CLIP-rewarded sampling. To avoid confusing models by matching one image with multiple captions, we need to obtain one caption. A simple way to get that caption is to select the best-aligned caption or summarize captions into one. However, previous studies [135, 202] suggest that different captions represent different view perspectives of the image. To fully utilize multiple perspective captions, instead of using one perspective, we propose the CLIP-rewarded sampling method. It samples one caption in each iteration based on the similarity to the image/3D data, to capture details in all captions and highlight the details that exhibit high similarity. Specifically, we first calculate the similarity $\mathbf{s}_{c2i,i}$ between the image $\mathbf{x}_i$ and the set of captions T_i as $\mathbf{s}_{c2i,i} = [cos\left(f_{\text{img}}(\mathbf{x}_i), f_{\text{text}}(\text{caption}_1)\right), \ldots, cos(f_{\text{img}}(\mathbf{x}_i), f_{\text{text}}\ (\text{caption}_{N_{\text{cap}}}))]$, where $cos(\cdot,\cdot)$ is the cosine similarity and $f_{\text{img}}(\cdot)$ is the image encoder of the utilized VLM. Next, in each iteration, we sample

one caption $T^{\text{view},i}$ from T_i based on the similarity $\mathbf{s}_{c2i,i}$ and obtain the image-caption data pair (P, X, T^{view}), where $T^{\text{view}} = \{T^{\text{view},i}\}_{i\in[N_{img}]}$ contains the sampled captions for all the view images in the point cloud P.

Hierarchical point cloud-text alignment. After obtaining the paired data, we are ready to align text encoders with 3D models. As suggested by previous 2D VLMs work [15, 43, 99, 147, 183, 255], hierarchical alignment, *e.g.*, region-level, image-level, and pixel-level alignment, is essential to cross-modal learning. To this end, we employ hierarchical 3D-text alignment [43, 67] to enable rich cross-modal association. Specifically, we have scene-, view-, and entity-level alignments. For **scene-level alignment** (coarse-grained), we first obtain a comprehensive scene-level caption $\mathbf{t}^{\text{scene}}$ with a text summarization model [106] to aggregate view-level captions T^{view}. Then, we directly maximize the similarity between the entire point clouds and the scene-level caption as,

$$\ell^{\text{scene}}_{\text{align}} = \ell_{\text{pdc}}(f_{\text{3D}}(P), f_{\text{text}}(\mathbf{t}^{\text{scene}}))\,, \tag{4.4}$$

where ℓ_{pdc} is the point-discriminative contrastive loss detailed below (Equation (4.9)). For **view-level alignment**, we maximize the similarity between the points $\hat{P}^{\text{view}}_i$ visible in an image (view) and its corresponding view-level caption T^{view}_i,

$$\ell^{\text{view}}_{\text{align}} = \sum_{i\in[|T^{\text{view}}|]} \ell_{\text{pdc}}\left(f_{\text{3D}}\left(\hat{P}^{\text{view}}_i\right), f_{\text{text}}\left(T^{\text{view}}_i\right)\right)\,. \tag{4.5}$$

For **entity-level alignment**, following PLA [43], we get nouns as the entity-level captions T^{entity} of each view by employing NLTK on the view-level captions T^{view}. Next, to associate a specific set of points (instead of the view-level set of points) with the entity-level captions, by the set differences and intersections of two adjacent views i and j, we obtain two entity-level pairs,

$$\begin{aligned} &(\hat{P}^{\text{entity}}_{i\setminus j}, \hat{T}^{\text{entity}}_{i\setminus j}), (\hat{P}^{\text{entity}}_{i\cap j}, \hat{T}^{\text{entity}}_{i\cap j}), \forall i, j \in [|T^{\text{view}}|]\,, \\ s.t.,\ &\sigma_{min} < |\hat{P}^{\text{entity}}_{i*j}| < \sigma_{max} \min(|\hat{P}^{\text{view}}_i|, |\hat{P}^{view}_j|), \\ &|\hat{T}^{\text{entity}}_{i*j}| > 0, \forall\, *\in \{\setminus, \cap\}\,, \end{aligned} \tag{4.6}$$

where $\setminus$ and $\cap$ represent the set difference and intersection, σ_{min} and σ_{max} are two hyperparameters, $\hat{P}^{\text{entity}}_{i*j} = \hat{P}^{\text{view}}_i * \hat{P}^{\text{view}}_j$, and $\hat{T}^{\text{entity}}_{i*j} = T^{\text{entity}}_i * T^{\text{entity}}_j$. In total, we have $2\binom{|T^{\text{view}}|}{2}$ entity-level associations, and now, we are ready to perform the entity-level alignment as,

$$\ell^{\text{entity}}_{\text{align}} = \sum_{i\in[2\binom{|T^{\text{view}}|}{2}]} \ell_{\text{pdc}}\left(f_{\text{3D}}\left(\hat{P}^{\text{entity}}_i\right), f_{\text{text}}\left(\hat{T}^{\text{entity}}_i\right)\right)\,. \tag{4.7}$$

Summarizing three losses, we have the hierarchical 3D-text alignment loss as,

$$\ell_{\text{align}} = \alpha^{\text{scene}}\ell^{\text{scene}}_{\text{align}} + \alpha^{\text{view}}\ell^{\text{view}}_{\text{align}} + \alpha^{\text{entity}}\ell^{\text{entity}}_{\text{align}}\,, \tag{4.8}$$

where α^{scene}, α^{view}, and α^{entity} are weighted parameters.

Point-discriminative contrastive learning. To maximize the similarity between the sets of point features and caption features, following previous studies [43, 226], we employ the point-discriminative contrastive loss [226] ℓ_{pdc} instead of the vanilla contrastive loss. Specifically, to encourage point-wise alignment, instead of grouping different-level point cloud features by mean pooling and applying contrastive loss, we first calculate point-wise contrastive-style activations and then group these activations,

$$\ell_{\text{pdc}}(F^P, F^T) = \frac{-\sum_{i \in [N_{F^P}]} \ln \frac{\exp(F_i^P F^{T\top})}{\exp(F_i^P F^{T\top}) + \sum_j \exp(F_i^P \hat{F}_j^{T\top})}}{N_{F^P}}, \tag{4.9}$$

where F_i^P represents the i-th point's feature, N_{F^P} is the number of points in $F^P \in \mathcal{R}^{N_{F^P} \times D}$, and $\hat{F}^T \in \mathcal{R}^{1 \times D}$ represents captions in the same training batch.

4.2.4 Adaptive Segmentation — Text Modeling

As shown in Table 4.1b, due to the domain shift between 3D datasets [3, 13, 36, 186] and the datasets [30, 39, 85, 131] 2D VLMs trained on, the performance of 3D open-vocabulary semantic segmentation models is highly sensitive to the prompt used for adapting the VLMs. One possible solution is fine-tuning VLMs to handle 3D data for better alignment between text encoders of VLMs and 3D models. However, due to catastrophic forgetting [81, 89], where the encoder loses its prior knowledge while attempting to adapt to the new data distribution, fine-tuning VLMs is infeasible.

Drawing inspiration from visual prompt tuning [89, 176, 231, 253], we focus on tailoring VLMs for 3D applications while freezing the model's parameters to avoid compromising the integrity of VLMs' knowledge, as shown in Figure 4.2. To better adapt the text encoder to 3D scenarios, we introduce a small number of learnable tokens *TOKENS* at the input and every transformer layer in the text encoder $f_{\text{text}}(\cdot)$. Specifically, at the input layer, we directly concatenate the trainable tokens with the text as the input of the text encoder. Sequentially, for each transformer layer, trainable tokens are merged with the output of the previous layer as the input of the current layer, *i.e.*, $\text{concat}([\textit{TOKENS}_i, F_{\text{text, i-1}}])$, where concat is the concatenation operation, $\textit{TOKENS}_i$ is the i-th layer's trainable token, and $F_{\text{text, i-1}}$ represents the output of the $(i-1)$-th layer. The trainable tokens are updated using the overall training objective (Equation (4.10)). During inference, we use the trainable tokens and the category names as the input of the text encoder $f_{\text{text}}(\cdot)$ to generate the category embedding C for the adaptive semantic segmentation head $f_{\text{seg}}(\cdot)$ as shown in Equation (4.1).

4.2.5 Training Objective

The training objective of our proposed AIDE is a weighted linear combination of segmentation loss (Equation (4.2)), confidence calibration loss (Equation (4.3)), and hierarchical alignment loss as follows:

$$\mathcal{L} = \beta^{\text{seg}}\ell_{\text{seg}} + \beta^{\text{align}}\ell_{\text{align}} + \beta^{\text{conf}}\ell_{\text{conf}}, \tag{4.10}$$

where β^{seg}, β^{align}, and β^{conf} are weight parameters.

4.3 Experiments

4.3.1 Benchmarks, Baselines, and Implementation

Benchmarks and category partitions. To validate the effectiveness of AIDE, we conducted extensive experiments on three popular 3D benchmarks: ScanNet [36], S3DIS [3], and one outdoor dataset (nuScenes [13]).

ScanNet [36] consists of 1,613 scenes (1,201 scenes for training, 312 scenes for validation, and 100 for testing) densely annotated in 20 classes. We discard the "otherfurniture" class and split the rest 19 classes into three partitions for semantic segmentation, as shown in Table 4.2.

S3DIS [3] contains 271 scans across 6 building areas and 13 categories. Following previous work [43, 173], we treat the 5-th area as the validation split and other areas as the training split. We discard the "clutter" class and partition the rest 12 classes into two partitions for both semantic segmentation and instance segmentation, as demonstrated in Table 4.2.

nuScenes [13] comprises 1000 driving scenes. 850 scenes of them form the training and validation set, and the other 150 scenes are for testing. It contains 16 semantic classes for the LiDAR semantic segmentation task. The scene is scanned by 32-line LiDAR, which is different from the previous two datasets.

For ScanNet, we split it into three base/novel partitions, *i.e.*, B15/N4, B12/N7, and B10/N9, where Bx/Ny refers to x base and y novel categories. As for S3DIS, we split it into 2 base/novel splits, *i.e.*, B8/N4, B6/N6. For nuScenes [13], we ignore the "otherflat" class and randomly divide the rest 15 classes into B12/N3 and B10/N5.

Evaluation metrics. Following previous methods [43, 217, 226], we evaluate semantic segmentation with Intersection over Union (IoU)-based metrics. For each category, IoU is computed as the ratio between the intersection and union of the predicted and ground-truth point sets. We report mean IoU over base categories (mIoU$^{\mathcal{B}}$) and mean IoU over novel categories (mIoU$^{\mathcal{N}}$),

Partition	Base Categories	Novel Categories
	ScanNet	
B15/N4	wall, floor, cabinet, bed, chair, table, door, window, picture, counter, curtain, refrigerator, showercurtain, sink, bathtub	sofa, bookshelf, desk, toilet
B12/N7	wall, floor, cabinet, sofa, door, window, counter, desk, curtain, refrigerator, showercurtain, toilet	bed, chair, table, bookshelf, picture, sink, bathtub
B10/N9	wall, floor, cabinet, bed, chair, sofa, table, door, window, curtain	bookshelf, picture, counter, desk, refrigerator, showercurtain, toilet, sink, bathtub
	S3DIS	
B8/N4	ceiling, floor, wall, beam, column, door, chair, board	window, table, sofa, bookcase
B6/N6	ceiling, wall, beam, column, chair, bookcase	floor, window, door, table, sofa, board

Table 4.2: Category partitions for open-vocabulary semantic segmentation on ScanNet and S3DIS.

where base categories are annotated during training and novel categories are held out from training annotations. Because open-vocabulary segmentation should perform well on both category groups, we also report the harmonic mean IoU:

$$\text{hIoU} = \frac{2 \cdot \text{mIoU}^{\mathcal{B}} \cdot \text{mIoU}^{\mathcal{N}}}{\text{mIoU}^{\mathcal{B}} + \text{mIoU}^{\mathcal{N}}}. \tag{4.11}$$

hIoU penalizes methods that perform well on base categories but poorly on novel categories, and all three metrics are higher-is-better.

Baselines. We compare AIDE with LSeg-3D [108], 3DGenZ [156], 3DTZSL [27], PLA [43], OpenScene [171], RegionPLC [226], and 3DPC-GZSL [230].

Implementation details. Following PLA [43], we use LSeg-3D [108] (UNet as the backbone, vision-language adapter implemented by MLP, and the CLIP [174] ViT-B/16 text encoder). For 3DGenZ [156] and 3DTZSL [27], we use the same setting with PLA [43]. We use 4 learnable tokens in the input space and each layer of the text encoder for enabling the adaptive segmentation module. For each temperature, we generate 30 captions enabling the sampling.

Implementation details. Following PLA [43], which is also our baseline, we employ the sparse-convolution-based UNet [61] with a base hidden dimension of 16 as our 3D backbone f_{3D}. We use CLIP (ViT-B16) [174] for the text and image encoders (f_{text} and f_{img}) and BART [106] for text summarization. For a fair comparison, we use ViT-GPT2 [34] as the image captioner f_{cap}. Four trainable tokens are used in adapting text encoders. The weight parameters are adapted from our baseline [43].

Methods	Venue	ScanNet B15/N4 hIoU	B15/N4 mIoU$^{\mathcal{B}}$	B15/N4 mIoU$^{\mathcal{N}}$	B12/N7 hIoU	B12/N7 mIoU$^{\mathcal{B}}$	B12/N7 mIoU$^{\mathcal{N}}$	B10/N9 hIoU	B10/N9 mIoU$^{\mathcal{B}}$	B10/N9 mIoU$^{\mathcal{N}}$
LSeg-3D [108]	ICLR'22	0.0	64.4	0.0	0.9	55.7	0.1	1.8	68.4	0.9
3DTZSL [27]	WACV'20	10.5	36.7	6.1	3.8	36.6	2.0	7.8	55.5	4.2
3DPC-GZSL [230]	ICCV'23	20.2	32.8	7.7	-	-	-	-	-	-
3DGenZ [156]	3DV'21	20.6	56.0	12.6	19.8	35.5	13.3	12.0	63.6	6.6
OpenScene† [171]	CVPR'23	67.1	68.8	62.8	56.8	61.5	51.7	55.7	71.8	43.6
RegionPLC [226]	CVPR'24	69.4	68.2	70.7	68.2	69.9	66.6	64.3	76.3	55.6
PLA (Baseline) [43]	CVPR'23	65.3	68.3	62.4	55.3	69.5	45.9	53.1	76.2	40.8
AIDE		**72.8**	**71.9**	**73.8**	**69.8**	**70.1**	**69.6**	**65.0**	**77.5**	**56.0**
Fully-Supervised‡		73.3	68.4	79.1	70.6	70.0	71.8	69.9	75.8	64.9

Table 4.3: Results on ScanNet. † and ‡ refer to numbers copied from He *et al.* [67] and Ding *et al.* [43]. Best in **Bold**.

Methods	Venue	S3DIS B8/N4 hIoU	S3DIS B8/N4 mIoU$^{\mathcal{B}}$	S3DIS B8/N4 mIoU$^{\mathcal{N}}$	S3DIS B6/N6 hIoU	S3DIS B6/N6 mIoU$^{\mathcal{B}}$	S3DIS B6/N6 mIoU$^{\mathcal{N}}$	nuScenes B12/N3 hIoU	nuScenes B12/N3 mIoU$^{\mathcal{B}}$	nuScenes B12/N3 mIoU$^{\mathcal{N}}$	nuScenes B10/N5 hIoU	nuScenes B10/N5 mIoU$^{\mathcal{B}}$	nuScenes B10/N5 mIoU$^{\mathcal{N}}$
LSeg-3D‡ [108]	ICLR'22	0.1	49.0	0.1	0	30.1	0	0.6	74.4	0.3	0	71.5	0
3DGenZ‡ [156]	3DV'21	8.8	50.3	4.8	9.4	20.3	6.1	1.6	53.3	0.8	1.9	44.6	1.0
3DTZSL‡ [27]	WACV'20	8.4	43.1	4.7	3.5	28.2	1.9	1.2	21.0	0.6	6.4	17.1	3.9
PLA (Baseline) [43]	CVPR'23	34.6	59.0	24.5	38.5	55.5	29.4	47.7	73.4	35.4	24.3	**73.1**	14.5
AIDE		**42.2**	**60.3**	**32.4**	**42.5**	**58.3**	**34.4**	**62.2**	**75.8**	**52.7**	**48.4**	66.4	**38.1**
Fully-Supervised‡		67.5	61.4	75.0	65.4	59.9	72.0	73.7	76.6	71.1	74.8	76.8	72.8

Table 4.4: Results on S3DIS and nuScenes. ‡ refers to numbers copied from Ding *et al.* [43]. Best in **Bold**.

4.3.2 Quantitative Results

In this part, we present the results of 3D open-vocabulary segmentation on ScanNet, S3DIS, and nuScenes in Tables 4.3 and 4.4, respectively.

ScanNet (Table 4.3). AIDE demonstrates superior performance across all metrics and splits, achieving hIoU of 72.8, 69.8, and 65.0 for B15/N4, B12/N7, and B10/N9 splits, respectively. This represents a significant improvement over PLA (Baseline), with increases of 7.5, 14.5, and 11.9 in hIoU. Notably, AIDE also narrows the gap with the fully supervised model, which represents the upper bound based on the backbone and training strategies.

S3DIS and nuScenes (Table 4.4). Notably, AIDE achieves the highest hIoU of 42.2 and 42.5 for B8/N4 and B6/N6 splits among various zero-shot learning-based and open-vocabulary segmen-

Adaptive Segmentation	Caption		ScanNet(B15/N4)		
	Best	Sampling	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
			65.3	68.3	62.4
✓			66.3	70.2	62.8
	✓		68.9	69.6	68.2
		✓	70.3	69.9	70.6
✓	✓		70.4	71.0	69.9
✓		✓	**72.8**	**71.9**	**73.8**
Fully-Supervised			73.3	68.4	79.1

Table 4.5: Ablation studies on different modules of AIDE. "Adaptive Segmentation" refers to the adaptive segmentation module. "Caption Selection" and "Caption Sampling" refer to using temperature-based generation and then selecting the best-aligned captions or sampling captions based on their similarity to the images (CLIP-rewarded alignment module).

tation methods, respectively, on S3DIS. Compared to our baseline, PLA, AIDE improves hIoU by 7.6 and 4.0 for each split. Improvements can also be observed on the outdoor dataset, nuScenes, as AIDE improves the hIoU from 47.7 and 24.3 to 62.2 and 48.4 on two different splits, showing the superiority of AIDE.

4.3.3 Ablation Studies

In this part, we present the ablation studies on the effects of two proposed modules (Table 4.5), hyperparameters (Tables 4.6 and 4.7), the choice of text encoders (Table 4.8), the choice of temperatures (Table 4.9 and fig. 4.3), and the importance of aligning with image space (Table 4.10). All experiments are conducted on ScanNet (B15/N4).

Impact of proposed modules (Table 4.5). Notably, the introduction of adaptive segmentation module alone improves hIoU from 65.3 to 66.3, and mIoU$^{\mathcal{B}}$ from 68.3 to 70.2, illustrating the efficacy in adapting the text encoders. "Caption Best" and "Caption Sampling" refer to using temperature-based generation and then selecting the best-aligned captions or captions sampled by the CLIP-rewarded sampling (the CLIP-rewarded alignment module). Specifically, while "Caption Best" improves hIoU to 68.9, "Caption Sampling" further boosts hIoU to 70.3 from 65.3. It underscores the importance of well-designed captioning techniques in improving alignment with the text encoder. Combining two modules together reaches the highest results. Specifically,

# of Learnable Prompts	ScanNet (B15/N4)		
	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
0 (Baseline)	65.3	68.3	62.4
2	71.9	70.9	72.9
4	**72.8**	71.9	**73.8**
8	71.8	72.2	71.0
16	71.2	**72.4**	70.0
Fully-Supervised	73.3	68.4	79.1

Table 4.6: Ablation studies on different numbers of learnable tokens of AIDE on ScanNet (B15/N4).

# of Generated Captions N_{cap}	ScanNet (B15/N4)		
	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
1 (Baseline)	65.3	68.3	62.4
10	71.9	71.3	72.5
20	72.7	**72.5**	73.0
30	**72.8**	71.9	**73.8**
50	70.2	69.6	70.8
Fully-Supervised	73.3	68.4	79.1

Table 4.7: Ablation studies on different numbers of captions of AIDE for each temperature.

combining the adaptive segmentation module with "Caption Best" achieves a hIoU of 70.4, while integrating it with "Caption Sampling" remarkably advances hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$ to 72.8, 71.9, and 73.8, respectively.

Numbers of learnable tokens. To understand how the length of (deep) learnable tokens impacts performance, we conduct a series of experiments as shown in Table 4.6. It is clear that increasing the number of learnable tokens from the baseline (zero prompts) significantly enhances the model's performance across all metrics (hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$). The optimal point is achieved with four learnable tokens, with the highest hIoU and mIoU$^{\mathcal{N}}$ of 72.8 and 73.8. However, further increasing the number of prompts to 8 or 16 leads to a slight performance decline, which might be due to potential overfitting on seen categories.

Number of captions N_{ca}. While the temperature used in caption generation affects the diversity

Text Encoder	ScanNet (B15/N4)		
	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
AIDE w. CLIP [174]	**72.8**	**71.9**	**73.8**
AIDE w. ImageBind [59]	71.0	70.4	71.7
AIDE w. PointBind [64]	70.0	70.7	69.4
Fully-Supervised	73.3	68.4	79.1

Table 4.8: Ablation studies on using different text encoders of AIDE on ScanNet (B15/N4).

of the generated captions, the number of captions (samples) generated for each temperature also matters. Thus, to understand the impact, we vary the number of samples for each temperature and present the results in Table 4.7. Notably, the peak performance is observed at 30 captions, with hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$ of 72.8, 71.9, and 73.8. Also, we observe consistent improvement when increasing the number of samples from 1 to 30, underscoring the value of leveraging more descriptive and diverse captions to enhance performance. However, it also illustrates a diminishing return beyond this optimal point, as evidenced by a decrease in all metrics when the number of captions is further increased to 50. Noisy caption generation might be the reason behind this phenomenon. As generating over 30 captions per temperature will result in a total of more than 120 captions and the training epochs are 128, the majority of these captions cannot be effectively utilized during training. It suggests that an optimal balance between the diversity and quality of captions is crucial.

Choice of text encoder. In this part, we compare the performance using three text encoders from multimodal foundational models, *e.g.*, CLIP [174], ImageBind [59], and PointBind [64], as shown in Table 4.8. The results indicate that the CLIP text encoder outperforms others across all metrics, with the highest hIoU (72.8), mIoU$^{\mathcal{B}}$ (71.9), and mIoU$^{\mathcal{N}}$ (73.8). This superiority likely stems from CLIP's adeptness at aligning visual and textual representations, a critical factor in open-vocabulary segmentation tasks.

Which temperature generates the captions that are closest to the corresponding 3D/image data? To better understand the impact of varying temperatures in generating paired 3D-(image)-text data, we evaluate the similarity between generated captions and their corresponding images. Specifically, we identify the temperature setting that produces the most aligned captions for each image, which we refer to as the "winner" temperature, and then count the time each temperature "wins". In our experiments, we use fourteen different temperatures, *i.e.*, $\{0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0\}$, and generate 30 captions for each temperature. The results are summarized in Figure 4.3, which indicates that temperatures of 4.0 and 5.0 frequently emerge as winners. This observation aligns with the fact that higher temperatures tend

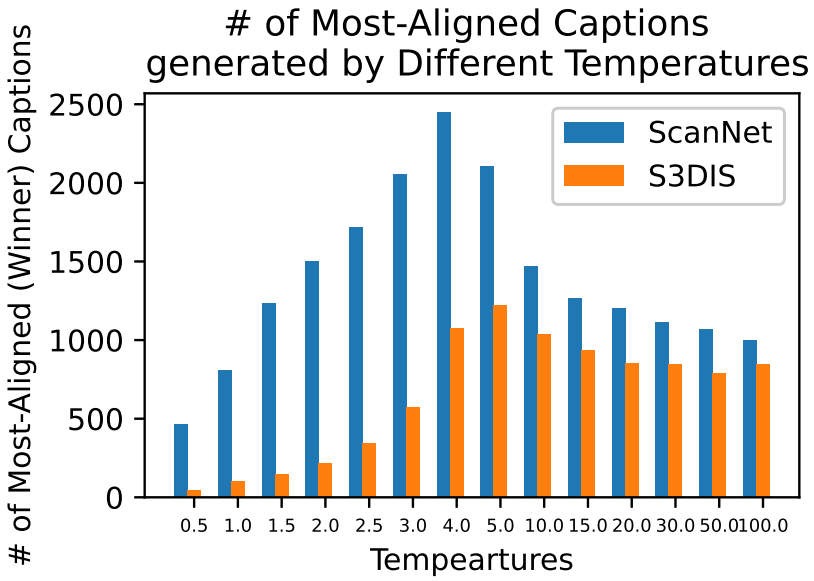


Figure 4.3: The number of most-aligned captions generated by different temperatures for each image. We use temperatures of 0.5, 1.0 (standard temperature), 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, and 100.0 for generating diverse captions.

to introduce more variability, resulting in noisier captions, whereas lower temperatures lead to less stochastic and potentially less diverse captions. Therefore, a moderate temperature setting is optimal for generating the best-aligned captions and aligned 3D-(image)-text data. The results also emphasize the importance of the CLIP-rewarded alignment module.

Impact of numbers of temperatures used for generation. To understand how the diversity of temperatures used for generating captions affects the performance, we employ four different temperature sets and evaluate the performance of AIDE on ScanNet (B15/N4) as shown in Table 4.9. A diverse set of temperatures significantly enhances AIDE's performance over the baseline, which only uses one temperature, *i.e.*, 1.0, with the optimal range being $\{0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0\}$, which slightly outperforms other settings with hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$ scores of 73.0, 71.7, and 74.3, respectively. This finding aligns perfectly with the observations in Figure 4.3 as temperatures 4.0 and 5.0 yield the most wins. Furthermore, increasing the temperature diversity further, even up to a wide range of $\{0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0\}$, results in diminishing returns, indicating a threshold beyond which additional temperature variance does not contribute to, and may even detract from performance. That might be because most

Temperatures used for Caption Generation	ScanNet (B15/N4)		
	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
{1.0} (Baseline)	65.3	68.3	62.4
{0.5, 1.0, 1.5, 2.0}	72.8	71.9	73.8
{0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0}	73.0	71.7	74.3
{0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0}	72.6	71.7	73.6
{0.5, 1.0, 1.5, 2.0, 3.0, 4.0, 5.0, 10.0, 15.0, 20.0, 30.0, 50.0, 100.0}	71.5	71.4	71.6
Fully-Supervised	73.3	68.4	79.1

Table 4.9: Ablation studies of different temperatures used for caption generation of AIDE.

captions will not be utilized even once throughout the training process. As the number of training epochs is 128, employing 14 temperatures results in a total of 420 captions, with most captions having a low probability of being sampled.

Do we also need aligning 3D with images? In this chapter, we focus on aligning the captions with 3D point cloud features, whereas another line of research [67, 171] aims at aligning 3D point cloud features with the corresponding images.

To have a fair comparison with these methods and to augment AIDE with image alignment capabilities, we implement view-level image alignment minimizing the PDC loss,

$$\ell_{\text{align}}^{\text{view, image}} = \sum_{i \in [|\boldsymbol{I}^{\text{view}}|]} \ell_{\text{pdc}} \left(f_{\text{3D}} \left(\hat{\boldsymbol{P}}_i^{\text{view}} \right), f_{\text{text}} \left(\boldsymbol{I}_i^{\text{view}} \right) \right) , \tag{4.12}$$

where $\boldsymbol{I}^{\text{view}}$ represents the images corresponding to a scene. For each image $\boldsymbol{I}_i^{\text{view}}$, we select the points visible in that image, mapped with poses and other parameters, as the paired point set $\hat{\boldsymbol{P}}_i^{\text{view}}$. This method is noted as "AIDE + View-Level 3D-Image Alignment" as shown in Table 4.10.

The table illustrates a compelling comparison between AIDE and other image-based methods [67, 171]. Results indicate that our proposed AIDE, even without additional view-level 3D-image alignment, achieves better results, specifically in mIoU$^{\mathcal{N}}$, demonstrating its robustness and effectiveness. With the introduction of view-level 3D-image alignment, AIDE outperforms UniM-OV3D (3D-Text) in all metrics, illustrating the effectiveness of precise image alignment strategies in enhancing segmentation accuracy. Furthermore, integrating view-level 3D-image alignment into our framework demonstrates not only the model's adaptability but also its ability to effectively utilize visual cues from aligned images for improved semantic alignment. These results suggest that while image alignment contributes positively to recognizing base categories (as

Method	ScanNet (B15/N4) hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
OpenScene† [171]	67.1	68.8	62.8
UniM-OV3D (3D-Text Only) [67]	62.1	66.2	56.4
AIDE	72.8	71.9	**73.8**
AIDE + View-Level 3D-Image Alignment	**73.0**	**72.7**	73.3
Fully-Supervised	73.3	68.4	79.1

Table 4.10: Ablation studies on using the image alignment and comparisons with image-based methods. † refers to numbers copied from He *et al.* [67].

					ScanNet					
Class Names	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf
Synonyms	palisade	deck	locker	bunk	bench	couch	board	exit	window	bookrack
Hypernyms	embankment	compartment	-	-	seat	-	-	entrance	-	shelf
Class Names	picture	counter	desk	curtain	refrigerator	showercurtain	toilet	sink	bathtub	
Synonyms	figure	-	escritoire	drapery	icebox	-	bathroom	-	-	
Hypernyms	image	-	-	-	-	-	room	-	-	
					S3DIS					
Class Names	ceiling	floor	wall	beam	column	window	door	table	chair	sofa
Synonyms	roof	deck	palisade	-	-	-	exit	board	bench	couch
Synonyms	cap	compartment	embankment	-	-	-	entrance	-	seat	-
Class Names	bookcase	board								
Synonyms	-	plank								
Hypernyms	-	-								

Table 4.11: Synonyms and hypernyms of class names on ScanNet and S3DIS used for generating class embeddings. For classes without suitable synonyms or hypernyms, we use the original class name as their synonyms and hypernyms and mark "-" in the table.

indicated by mIoU$^{\mathcal{B}}$), the approach might need refinement for equally impactful improvements in novel category recognition (mIoU$^{\mathcal{N}}$).

4.3.4 Qualitative Results—Generalization

To illustrate the effectiveness of our proposed AIDE, we present qualitative results on analyzing the open-vocabulary ability to segment point clouds with the synonyms and hypernyms of classes (Table 4.12) and the zero-shot domain transfer ability (Table 4.13).

Generalization on segmenting with synonyms and hypernyms of classes. VLMs have demon-

Methods	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$	IoU on Base Categories							
				wall	floor	cabinet	bed	chair	table	door	window
Original Class Names											
Baseline	0.645	0.679	0.615	0.839	0.950	0.638	0.808	0.880	0.705	0.595	0.617
AIDE	**0.728**	**0.719**	**0.738**	**0.874**	**0.979**	**0.684**	**0.818**	**0.916**	**0.744**	**0.637**	**0.658**
Synonyms											
Baseline	0.289	0.207	0.479	**0.262**	0.138	**0.029**	0.566	0.292	0.004	0.155	**0.346**
AIDE	**0.340**	**0.243**	**0.568**	0.166	**0.352**	0.012	**0.739**	**0.550**	**0.007**	**0.269**	0.326
Hypernyms											
Baseline	0.311	0.230	0.478	0.000	**0.000**	0.418	0.401	**0.528**	0.140	0.058	0.344
AIDE	**0.364**	**0.264**	**0.588**	**0.001**	**0.000**	**0.477**	**0.637**	0.326	**0.183**	**0.190**	**0.378**

Methods	IoU on Base Categories							IoU on Novel Categories			
	picture	counter	curtain	refrigerator	showercurtain	sink	bathtub	sofa	bookshelf	desk	toilet
Original Class Names											
Baseline	0.312	0.591	0.697	0.488	0.652	0.571	0.837	0.734	0.681	0.449	0.594
AIDE	**0.349**	**0.646**	**0.733**	**0.561**	**0.676**	**0.656**	**0.854**	**0.821**	**0.756**	**0.513**	**0.862**
Synonyms											
Baseline	**0.005**	**0.261**	**0.000**	0.079	**0.592**	**0.000**	**0.381**	**0.686**	0.524	0.242	0.464
AIDE	0.002	0.196	**0.000**	**0.249**	0.493	**0.000**	0.280	0.643	**0.577**	**0.304**	**0.746**
Hypernyms											
Baseline	**0.040**	**0.461**	0.506	**0.055**	0.339	**0.000**	0.165	0.724	0.621	0.425	0.143
AIDE	0.035	0.418	**0.537**	0.031	**0.489**	**0.000**	**0.255**	**0.750**	**0.666**	**0.474**	**0.464**

Table 4.12: Open-vocabulary semantic segmentation results on ScanNet (B15/N4) using the original class names, their synonyms, and hypernyms. The synonyms and hypernyms of class names are presented in the Appendix.

Train Dataset	Metrics (Baseline/AIDE)		
	hIoU	mIoU$^{\mathcal{B}}$	mIoU$^{\mathcal{N}}$
Test Dataset: S3DIS (B8/N4)			
ScanNet (B15/N4)	32.1/**35.9**	31.6/**39.9**	32.6/**33.8**
ScanNet (B12/N7)	22.2/**25.8**	**25.0**/23.3	19.9/**28.9**
ScanNet (B10/N9)	24.7/**31.0**	30.5/**38.9**	20.7/**25.7**
Test Dataset: ScanNet (B15/N4)			
S3DIS (B8/N4)	10.5/**11.9**	**15.0**/**15.0**	8.1/**9.9**
S3DIS (B6/N6)	5.9/**7.5**	7.1/**7.6**	5.1/**7.5**

Table 4.13: Zero-shot transfer ability of baseline and AIDE. We train models on ScanNet (B15/N4) or S3DIS (B8/N4) and test them on another. Best in **bold**.

strated a remarkable capacity for associating semantically similar words. To understand how well this ability is transferred to 3D models, we evaluate AIDE and baseline's performance in recognizing synonyms and hypernyms of original class names, as shown in Table 4.12. The synonyms and hypernyms of different classes are presented in Table 4.11. Results show that AIDE consistently surpasses the baseline across various metrics for most of the classes. We use WordNet [159] and Cambridge Dictionary[2] to select appropriate synonyms and hypernyms to avoid confusion. For classes without suitable synonyms or hypernyms, we use the original class name as their synonyms and hypernyms and mark "-" in the table.

When using synonyms and hypernyms, both methods experience a drop in performance compared to original class names, which is reasonable due to the increased difficulty in matching these broader or alternative terms with points precisely. On the other side, AIDE still maintains a lead over the baseline, demonstrating its robustness to variations in vocabulary. The drop is mostly in the synonyms category, especially with base categories, *e.g.*, "floor", "curtain", and "sink". It also highlights the challenge of segmentation with hypernyms, as the baseline and AIDE have (near-)zero IoU for several categories, indicating difficulty in generalizing to broader category terms.

Generalization on zero-shot domain transfer. To understand the zero-shot domain transfer ability of AIDE, we conduct experiments where models are trained on either ScanNet (B15/N4) or S3DIS (B8/N4) and tested on the other, as shown in Table 4.13. In every setting, AIDE consistently outperforms the baseline on hIoU, underscoring its superior generalization capability. Notably, when training on ScanNet (B15/N4) and testing on S3DIS (B8/N4), AIDE achieves significant improvements in all metrics, with hIoU, mIoU$^{\mathcal{B}}$, and mIoU$^{\mathcal{N}}$ increasing from 32.1, 31.6, and 32.6 to 35.9, 39.9, and 33.8, respectively. Similar patterns are observed with huge gaps between the baseline and AIDE in different settings. These results underscore the importance of the CLIP-rewarded alignment and adaptive segmentation modules in enhancing open-vocabulary segmentation models' transferability to novel categories and scenarios.

Generalization to instance segmentation. To understand the possibility of extending AIDE to instance segmentation, following previous methods [43], we introduce an instance localization head $f_{\text{loc}}(\cdot)$ for generating instance proposal and instance class prediction (details see S1.2 [43]). Specifically, instance proposals *pps* are first generated by grouping the segmentation prediction $f_{\text{seg}}(f_{3D}(P))$, offset head, and the point-wise features $f_{3\text{D}}(P)$. Next, with a TinyUNet and the class-agnostic seg and score heads, perclass confidence for instance proposals *pps* are generated with the pooled segmentation prediction $f_{\text{seg}}(f_{3\text{D}}(P))$.

The results of LSeg-3D, our baseline (PLA), and AIDE on instance segmentation on S3DIS (B8/N4) are shown in Table 4.14. We use average precision at an IoU threshold of 0.50 as the

[2] https://dictionary.cambridge.org/.

Methods	S3DIS (B8/N4)		
	hAP_{50}	$mAP^{\mathcal{B}}_{50}$	$mAP^{\mathcal{N}}_{50}$
LSeg-3D	0.5	58.3	0.3
Baseline	26.7	60.3	17.2
AIDE	**34.5**	**61.9**	**23.9**
Fully-Supervised	57.6	60.8	54.6

Table 4.14: Generalization of AIDE to instance segmentation on S3DIS.

instance segmentation metric. Specifically, $AP^{\mathcal{B}}_{50}$ is averaged over base categories and $AP^{\mathcal{N}}_{50}$ is averaged over novel categories. To measure the balance between base and novel instance recognition, we report their harmonic mean, hAP_{50}:

$$hAP_{50} = \frac{2 \cdot AP^{\mathcal{B}}_{50} \cdot AP^{\mathcal{N}}_{50}}{AP^{\mathcal{B}}_{50} + AP^{\mathcal{N}}_{50}}. \tag{4.13}$$

All AP-based metrics are higher-is-better. Notably, AIDE demonstrates superior performance across all metrics, achieving hAP_{50}, $mAP^{\mathcal{B}}_{50}$, and $mAP^{\mathcal{N}}_{50}$ of 34.5, 61.9, and 23.9, respectively. This demonstrates AIDE's effectiveness in handling instance segmentation tasks, significantly improving upon the baseline (26.7 in hAP_{50}, 60.3 in $mAP^{\mathcal{B}}_{50}$, and 17.2 in $mAP^{\mathcal{N}}_{50}$) and significantly closing the gap with the fully supervised model, which leads with 57.6 in hAP_{50} and 54.6 in $mAP^{\mathcal{N}}_{50}$. This underscores AIDE's potential to enhance open-vocabulary instance segmentation capabilities, while also indicating the need for further advancements to narrow the significant gap with fully supervised methods.

4.3.5 Qualitative Results

In this part, we extend our qualitative results on the interpretation of learnable prompts in the input space (Table 4.15), the quality of generated captions (Figure 4.4), and generalization on instance segmentation (Table 4.14).

Generated captions. To understand how the proposed CLIP-rewarded alignment module improves the quality of generated 3D-text data, we randomly choose some samples of captions presented in Figure 4.4. It is obvious that captions generated by lower temperatures (closer to the baseline of 1.0) might be more generic and closely tied to the most apparent elements in the

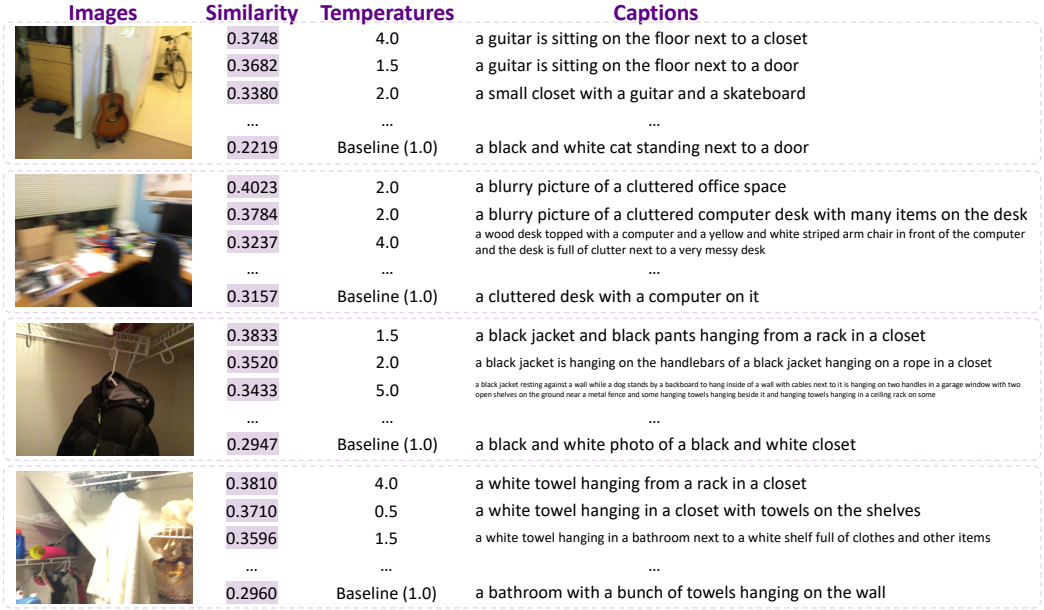


Figure 4.4: Examples of captions generated at different sampling temperatures for the same 3D-scene view, together with their CLIP cosine similarity to the paired image. Higher temperatures increase caption diversity but also introduce noisier phrases, whereas lower temperatures are more conservative. During training, CLIP-rewarded sampling preferentially selects captions with higher image–text similarity, so the caption distribution used for supervision shifts toward better-aligned descriptions after model training.

images. In contrast, higher temperatures could lead to more diverse and potentially creative interpretations of the same visual content. Moreover, high temperature always leads to informative generated captions, *e.g.*, "a wood desk topped with a computer and a yellow and white striped arm chair in front of the computer and the desk is full of clutter next to a very messy desk" for the second example. And high temperatures can capture key information too as 2.0 captures "blurry" in the second image while 1.0 can barely capture that. Meanwhile, as captions generated by higher temperatures are more informative, they exhibit higher similarity to the images than the baseline setting (1.0). This variability underscores the importance of temperature in controlling the trade-off between diversity and relevance of generated captions, a critical aspect in tasks requiring a nuanced understanding of visual content, such as semantic segmentation, object recognition, and image captioning in computer vision.

The meaning of learned prompts. To understand the meaning of learned prompts, we calculate the cosine similarity between the learned prompts in the input space and the human vocabulary

ScanNet										
					Token1					
Similarity	0.8044	0.5539	0.4234	0.3598	0.3563	0.3497	0.3366	0.3357	0.3108	0.3102
Word	a	an	the	his	my	some	this	yours	ans	s
					Token2					
Similarity	0.7393	0.4705	0.2389	0.2282	0.2209	0.213	0.208	0.1974	0.1966	0.1962
Word	scene	scenes	scen	scenery	moment	stage	sight	set	cameo	display
					Token3					
Similarity	0.7616	0.4286	0.4227	0.3786	0.3687	0.3614	0.3614	0.3415	0.3327	0.3264
Word	of	to	for	by	from	in	with	on	at	,
					Token4					
Similarity	0.6825	0.4426	0.2749	0.2704	0.2644	0.2624	0.2574	0.2538	0.2417	0.2387
Word	a	an	the	your	my	and	as	in	any	every
S3DIS										
					Token1					
Similarity	0.4468	0.3206	0.2098	0.2067	0.206	0.2036	0.2012	0.1966	0.1899	0.1892
Word	a	an	ah	his	need	sus	put	some	had	took
					Token2					
Similarity	0.3924	0.2307	0.2305	0.1852	0.1804	0.1771	0.1755	0.1755	0.1625	0.1621
Word	scene	scenes	sleet	steady	dumped	heavens	haz	smashes	pavement	sites
					Token3					
Similarity	0.3181	0.2087	0.1972	0.1919	0.1911	0.1865	0.1826	0.1778	0.1764	0.1748
Word	of	s	in	from	and	being	into	are	you	your
					Token4					
Similarity	0.3856	0.3002	0.2556	0.2413	0.2379	0.2343	0.23	0.2257	0.2217	0.2182
Word	a	an	his	my	your	you	he	us	her	me

Table 4.15: Similarity between learned prompts (tokens) in the input space and "human vocabulary" under use the cosine similarity. "Word" refers to subwords in the CLIP's vocabulary.

(subwords in CLIP's vocabulary) presented in Table 4.15. It shows that both models tend to align with basic English words such as articles ("a", "an", "the") and pronouns ("his", "my"). Notably, the learned prompts on both ScanNet and S3DIS exhibit high similarity with "scene". On the other side, learned prompts of different benchmarks tend to have different composed meanings as the top 2 - top 5 ranked words are different. It reflects the AIDE' capacity to approximate human-like linguistic structures and adaptation to domain-specific linguistic contexts.

Visualization. To better understand how our AIDE excels at segmenting seen and unseen objects, we visualize segmentation results in Figure 4.5. It is obvious that, compared with our baseline (PLA), AIDE has a better segmentation on both the seen and unseen classes. The desk on the bottom left, the toilet on the bottom right, the chair on the top right, and the bookshelf on the top

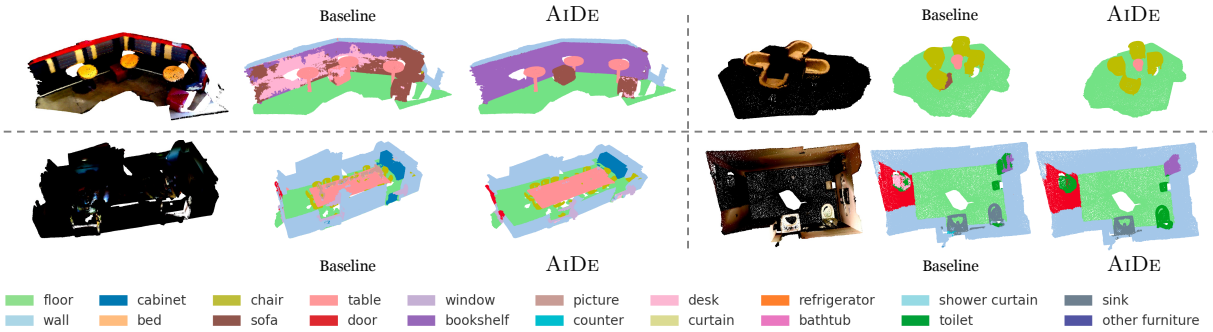


Figure 4.5: Qualitative open-vocabulary segmentation results on ScanNet validation scenes. For each scene, we compare the baseline (PLA) and AIDE on both base categories seen during training and novel categories held out at training time.

	# of Trainable Parameters	# of Parameters	Throughput (Scene/s)
Baseline	11,001,346	74,138,114	7.58
AIDE	11,027,970 (0.2% ↑)	74,164,738 (0.03% ↑)	7.58

Table 4.16: Comparison of parameters and throughput between the baseline and AIDE.

left are more clear than the baseline model.

4.3.6 Efficiency

We also include a parameter and throughput analysis in Table 4.16. The number of trainable parameters measures the additional learnable capacity optimized during training, while the total number of parameters measures the full model size. Throughput is reported in scenes per second (Scene/s), where higher throughput indicates faster inference. We notice that, as AIDE only introduces sets of trainable tokens to adapt the text encoder, the additional parameters are marginal and the throughput remains the same as the baseline model.

4.4 Prompt Templates

In this section, we present the prompt templates used for producing the results in Table 4.1b.

Identity. This prompt template is shown as follows,

{CLASS}

Simple. This prompt template is shown as follows,

a photo of a {CLASS}

LSeg. This prompt template is shown as follows,

a {CLASS} in a scene

Full-ImageNet. This prompt template uses the following 81 templates to generate text embedding for each class and then do an average of 81 embeddings.

a bad photo of a {CLASS}
a photo of many {CLASS}
a sculpture of a {CLASS}
a photo of the hard to see {CLASS}
a low resolution photo of the {CLASS}
a rendering of a {CLASS}
graffiti of a {CLASS}
a bad photo of the {CLASS}
a cropped photo of the {CLASS}
a tattoo of a {CLASS}
the embroidered {CLASS}
a photo of a hard to see {CLASS}
a bright photo of a {CLASS}
a photo of a clean {CLASS}
a photo of a dirty {CLASS}
a dark photo of the {CLASS}
a drawing of a {CLASS}
a photo of my {CLASS}

the plastic {CLASS}
a photo of the cool {CLASS}
a close-up photo of a {CLASS}
a black and white photo of the {CLASS}
a painting of the {CLASS}
a painting of a {CLASS}
a pixelated photo of the {CLASS}
a sculpture of the {CLASS}
a bright photo of the {CLASS}
a cropped photo of a {CLASS}
a plastic {CLASS}
a photo of the dirty {CLASS}
a jpeg corrupted photo of a {CLASS}
a blurry photo of the {CLASS} a photo of the {CLASS}
a good photo of the {CLASS}
a rendering of the {CLASS}
a {CLASS} in a video game
a photo of one {CLASS}
a doodle of a {CLASS}
a close-up photo of the {CLASS}
a photo of a {CLASS}
the origami {CLASS}
the {CLASS} in a video game
a sketch of a {CLASS}
a doodle of the {CLASS}
a origami {CLASS}
a low resolution photo of a {CLASS}
the toy {CLASS}
a rendition of the {CLASS}
a photo of the clean {CLASS}
a photo of a large {CLASS}
a rendition of a {CLASS}
a photo of a nice {CLASS}
a photo of a weird {CLASS}
a blurry photo of a {CLASS}
a cartoon {CLASS}
art of a {CLASS}
a sketch of the {CLASS}
a embroidered {CLASS}
a pixelated photo of a {CLASS}
itap of the {CLASS}
a jpeg corrupted photo of the {CLASS}
a good photo of a {CLASS}
a plushie {CLASS}
a photo of the nice {CLASS}

a photo of the small {CLASS}
a photo of the weird {CLASS}
the cartoon {CLASS}
art of the {CLASS}
a drawing of the {CLASS}
a photo of the large {CLASS}
a black and white photo of a {CLASS}
the plushie {CLASS}
a dark photo of a {CLASS}
itap of a {CLASS}
graffiti of the {CLASS}
a toy {CLASS}
itap of my {CLASS}
a photo of a cool {CLASS}
a photo of a small {CLASS}
a tattoo of the {CLASS}

4.5 Conclusion and Limitations

We introduced AIDE to collect well-aligned 3D-vision-and-text multimodal data and efficiently adapt 2D VLMs for 3D semantic segmentation, thereby enhancing the models' generalization capabilities. AIDE has two key components: (i) the CLIP-rewarded alignment module, using temperature-based caption generation combined with CLIP-rewarded sampling to generate well-aligned 3D-vision-and-text data, and (ii) adaptive segmentation module, adding a small set of learnable tokens in both the input space and each layer of text encoder to adapt the VLM text encoder to the 3D setting. Our experimental results demonstrated AIDE's superiority over previous methods, indicating the importance of high-quality data generation and the adaptation of text encoders.

Limitation. While we employ scene-level, view-level, and entity-level alignments, an even more fine-grained alignment could achieve better alignment [15, 99, 147, 183, 255]. Currently, the point-wise recognition method is limited to cosine similarity matching. In the future, an advanced text-3D fusion method should be applied to integrate multimodal information for segmentation.

Societal Impact. In this chapter, we propose a novel 3D open-vocabulary segmentation framework aimed at recognizing countless classes of objects to improve the generalization of segmentation models. We did not identify any obvious negative societal impacts. Instead, we hope our model can contribute to the development of reliable and generalizable machine learning models and advance progress in this area.

With open-vocabulary recognition in place, a deployed system must often go beyond assigning labels and respond to user queries about unfamiliar scenes. Chapter 5 studies this understanding stage with efficient MLLMs, focusing on instruction-conditioned alignment between visual tokens and text while reducing computational cost.

Chapter 5

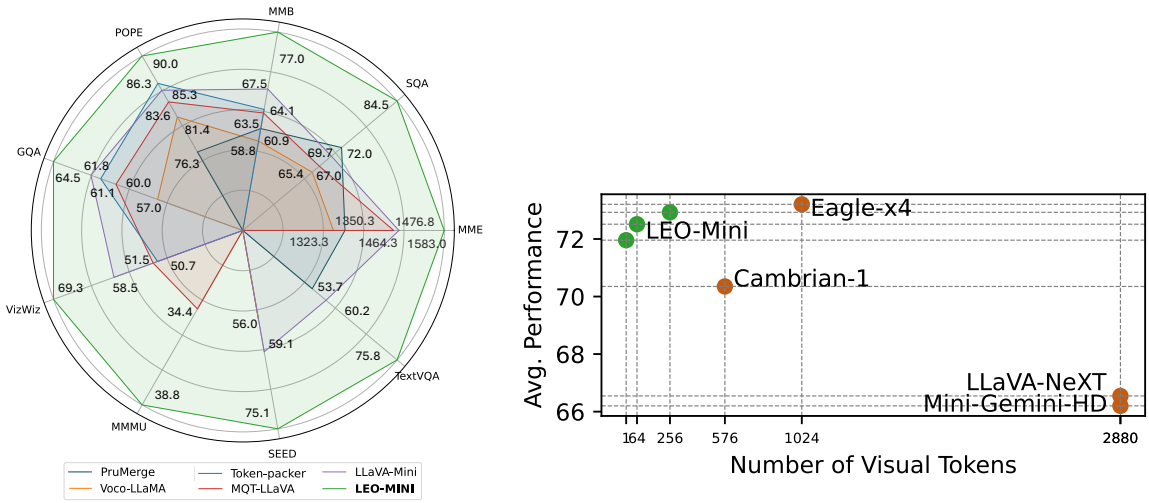
LEO-MINI: An Efficient Multimodal Large Language Model using Conditional Token Reduction and Mixture of Multi-Modal Experts

The previous chapter addressed OOD recognition by assigning semantic labels to novel 3D categories through open-vocabulary segmentation. Recognition, however, is not always sufficient for deployment: a system often needs to answer task-specific questions, explain an unfamiliar situation, or suggest an appropriate response. This chapter therefore considers OOD understanding with generative VLMs or multimodal LLMs (MLLMs)[1], focusing on how to preserve vision-language understanding performance while reducing the computational cost of processing many visual tokens.

5.1 Introduction

The development of multi-modal large language models (MLLMs) [6, 38, 134, 146, 180, 247] has been significantly advanced by aligning vision models [52, 103, 144] with large-scale pre-trained language models (LLMs) [28, 145]. MLLMs, such as the LLaVA [133, 134], BLIP [38], and InternVL [24], embed image patches into visual tokens through a vision expert [52, 100, 174]. Then, those visual tokens are input into the LLM for understanding. This has led to strong

[1] We interchange "generative VLMs" and "MLLMs" in this chapter.



(a) **Improved effectiveness** compared with token-reduction MLLMs.

(b) **Improved efficiency** of LEO-MINI with general and MoVE-based MLLMs.

Figure 5.1: Performance overview of LEO-MINI-Llama3-8B. a: Comparison between LEO-MINI (64 tokens) and existing token reduction MLLMs [78, 118, 178, 232, 247], where LEO-MINI outperforms all models. b: Comparison with other MLLMs [123, 134, 180, 193] on 11 vision-language tasks [55, 84, 95, 107, 139, 140, 154, 155, 184, 234] using Llama3-8B [145] as the LLM. LEO-MINI achieves competitive performance while using only 64 visual tokens

performance in image and video understanding tasks [55, 84, 95, 107, 124, 139, 234, 234], bridging the gap between vision and language models. As discussed in Section 2.4, generative VLMs differ from the discriminative VLMs used in Chapters 3 and 4: they do not only compare an image embedding with class-name prototypes, but condition a language model on visual tokens and textual instructions to generate free-form responses. This makes them suitable for the final level of OOD handling considered in this thesis, where the system may need to describe, explain, or answer questions about unfamiliar visual content rather than only reject or label it. At the same time, this generative formulation changes the alignment problem into a sequence-processing problem: the LLM must attend over a potentially large set of visual tokens while also following the user's textual instruction.

However, the substantial computational requirement of MLLMs presents a significant challenge to their efficiency. In MLLMs, the LLM predominantly drives computational costs, as the vision expert is smaller in comparison. For example, the commonly used vision expert, CLIP-L [174], has 0.3 billion parameters, whereas LLMs such as LLaMA [145] or Vicuna [28], have 7–8 billion and 13 billion parameters, respectively. While the vision expert is relatively lightweight,

its output, *i.e.*, the visual tokens, which is fed into the LLM along with text instruction tokens, significantly increases the computational overhead. For instance, CLIP-L [174] encodes a single image into $24 \times 24 = 576$ visual tokens, whereas textual instructions typically consist of fewer than 100 tokens. This becomes more challenging in high-resolution image understanding [6, 24] or video understanding [70, 115, 221], which requires either more visual tokens per image or processing multiple images. Thus, efficiency in this chapter is not merely a deployment concern; it is tied to how visual evidence is represented before it reaches the language model. If token reduction removes details that are needed by the instruction, the model may become faster but less reliable on fine-grained recognition, reasoning, text-rich images, or document-style tasks. The goal is therefore to reduce redundant visual tokens while preserving the information that is most relevant for instruction-conditioned understanding.

Reducing the number of visual tokens can thus be an effective strategy for enhancing the efficiency of MLLMs, either through training an efficient compressor [118, 247] or by using a training-free summarizer [19, 211, 245]. In this work, we focus on training-based methods. Recent training-based approaches [118, 122, 178, 247] reduce the number of visual tokens by selecting only the most informative ones [19], rather than using all visual tokens. Notably, LLaVA-Mini [247] achieves comparable performance to the full LLaVA-1.5 [133], while using only a single visual token. However, aggressively reducing visual tokens may result in the loss of essential visual information, potentially degrading the model's performance.

To extract informative visual features with improved efficiency, in this chapter, we propose LEO-MINI, a new MLLM that incorporates a novel conditional token reduction module (COTR) for increased efficiency and a novel mixture of multi-modal experts (MMOE) module for greater effectiveness.

Efficiency. As the number of visual tokens has become a major bottleneck for MLLM efficiency, LEO-MINI introduces COTR to reduce the number of tokens fed into the LLM. To focus on the most informative visual tokens based on the input instructions, COTR aggregates visual tokens into a smaller set of consolidated tokens by their similarity to both visual tokens from other vision experts and text tokens. Moreover, a learnable query is employed to control the length of consolidated visual tokens, which can be adjusted according to the task or computational requirements. This significantly reduces the number of visual tokens, leading to improved training and inference efficiency.

Effectiveness. For a better understanding of visual features and improved understanding ability, LEO-MINI incorporates MMOE, a novel mixture of multi-modal experts module consisting of MMOE-LLM and MMOE-Vision. Instead of conducting a full fine-tuning of the entire model after the pretraining, MMOE-LLM employs a mixture of LoRA experts [45, 48, 192, 216] with a novel router and a general expert. In contrast to previous work [45, 48, 192, 216] whose

routers only take the hidden state as input to switch between experts, our router takes the text tokens and the visual tokens as additional input. This facilitates more effective switching between different LoRA experts. The general expert is continuously activated to learn general knowledge. To extract more informative visual features, MMoE-Vision incorporates multiple vision experts [52, 100, 174], each trained on data from different domains. This boosts the model's ability to understand visual information, leading to improved performance on vision-language tasks while maintaining minimal computational overhead, as both the vision experts and LoRA experts are substantially smaller than the LLM.

We demonstrate the effectiveness and improved efficiency of LEO-Mini on various vision-language tasks [55, 60, 65, 84, 95, 107, 124, 139, 154, 155, 184, 234, 234], as illustrated in Figure 5.1.

5.2 Preliminaries

This section briefly introduces the standard MLLM pipeline [135] and problem definition used in this chapter. The general background of generative VLMs and efficient MLLMs is reviewed in Section 2.4; here, we focus on the notation and modeling assumptions needed to understand LEO-Mini.

5.2.1 Problem Definition

An MLLM takes visual content and a text instruction as input, and generates a natural-language response. Let $\mathbf{x}$ denote an input image and let $T = \{t_i\}_{i=1}^{N^T}$ denote the tokenized text instruction, where N^T is the number of text tokens. The target response is represented as $Y = \{y_i\}_{i=1}^{L}$, where L is the response length. The model is trained to maximize the conditional likelihood of the response:

$$p(Y|\mathbf{x}, T) = \prod_{i=1}^{L} p(y_i|\mathbf{x}, T, y_{<i}), \tag{5.1}$$

where $y_{<i}$ denotes the previously generated response tokens. This formulation covers common vision-language understanding tasks, including visual question answering, image captioning, OCR-oriented reasoning, and chart or document understanding. Unlike the previous two chapters, the output space is not a binary ID/OOD decision or a fixed set of semantic labels. Instead, the model must generate a response that is both visually grounded and instruction-following.

In this thesis, the key efficiency question is how much visual information needs to be passed to the LLM to preserve this response quality. Let N^V_{full} be the number of visual tokens produced by

the original vision encoder pipeline, and let N^V_{red} be the number of visual tokens after compression. The goal is to reduce N^V_{red} while maintaining or improving benchmark performance:

$$N^V_{red} \ll N^V_{full}, \quad \text{with comparable or better task performance.} \tag{5.2}$$

This objective differs from simple feature compression because the importance of a visual token depends on the instruction. For example, a token corresponding to small printed text may be crucial for OCR-style questions but less important for coarse image captioning.

5.2.2 Standard MLLM Pipeline

Most MLLMs [134, 135, 180] follow a vision-encoder–projector–LLM design. First, a vision encoder $f_{vis}(\cdot)$ converts the input image $\mathbf{x}$ into a sequence of visual tokens:

$$I = f_{vis}(\mathbf{x}) \in \mathcal{R}^{N^V \times d^V}, \tag{5.3}$$

where N^V is the number of visual tokens and d^V is the visual feature dimension. Second, a projector $g(\cdot)$ maps these visual tokens into the LLM embedding space:

$$\tilde{I} = g(I) \in \mathcal{R}^{N^V \times d_{\text{LLM}}}, \tag{5.4}$$

where d_{LLM} is the LLM hidden dimension. Finally, the projected visual tokens $\tilde{I}$ are concatenated or otherwise inserted into the LLM context together with the text instruction tokens T. The LLM then generates the response autoregressively.

This pipeline explains why visual token count is a central bottleneck. Although the vision encoder extracts the visual features, the large language model must repeatedly attend to the visual tokens during generation. As reviewed in Section 2.4, high-resolution images [123], multi-image inputs, and video inputs can dramatically increase N^V. Reducing visual tokens before they enter the LLM can therefore improve both training and inference efficiency. However, a naive reduction strategy [19] may discard task-relevant information, especially when the instruction requires fine-grained perception. This motivates the conditional token reduction and multi-modal expert design introduced in the next section.

5.3 Methodology

In Section 5.3.1, we introduce the overall framework of LEO-MINI. In Section 5.3.2, we present our proposed token reduction module, COTR, which consolidates a large number of visual tokens into a smaller, more informative set. Finally, in Section 5.3.3, we introduce our mixture of

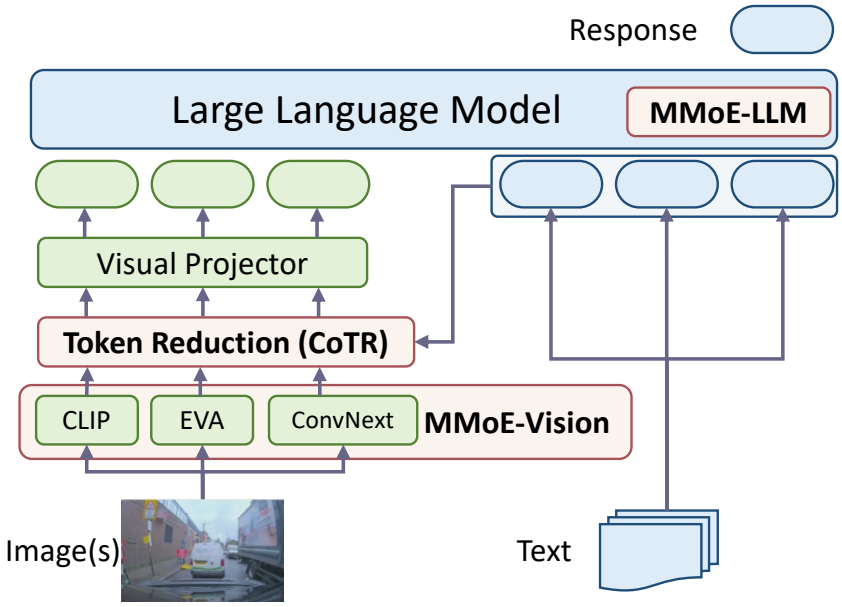


Figure 5.2: The overview of the proposed LEO-MINI. LEO-MINI is designed with MMOE (Section 5.3.3) to enhance visual comprehension and COTR (Section 5.3.2) to reduce the number of visual tokens for efficiency.

multi-modal experts module, MMOE, which is designed to enhance the efficiency of fine-tuning MLLMs while preserving their strong performance. The design is motivated by two observations from the background discussion of efficient generative VLMs in Section 2.4. First, most visual tokens are passed into the LLM even though only a subset may be relevant to a particular instruction. Second, different vision encoders capture different visual cues, so reducing tokens from a single feature stream can discard information that another expert would preserve. LEO-MINI addresses these two issues jointly by making token reduction conditional on both visual and textual context, and by using multi-modal experts to increase capacity without full-model fine-tuning.

5.3.1 Architecture

The overall architecture of LEO-MINI is presented in Figure 5.2. It follows the general design (vision expert-projector-LLM) of existing MLLMs [6, 180, 247], while incorporating a mixture of multi-modal experts and a visual feature compression module.

Specifically, MMOE introduces *multiple vision experts*, each trained on a domain-specific vision task to extract diverse and informative visual features from the input image. These experts

embed the input image into visual token groups $\{I_i\}_{i\in[m]}$, where $I_i \in \mathcal{R}^{N_i^V \times d_i^V}$, m is the number of vision experts, N_i^V is the number of visual tokens generated by the i-th vision expert, and d_i^V is the feature dimension.

Visual feature compression is then applied to the group of visual tokens $\{I_i\}_{i\in[m]}$. First, as the visual tokens generated by different vision experts have different lengths, *i.e.*, $\{N_i^V\}_{i\in[m]}$, the CoTR module (Section 5.3.2) projects them into consolidated visual token groups $\{\bar{I}_i\}_{i\in[m]}$. Each group has $\bar{I}_i \in \mathcal{R}^{N^V \times d_i^V}$, with the same token length N^V. N^V is much smaller than $\sum_{i\in[m]} N_i^V$, significantly reducing the number of visual tokens for efficiency. Then, the consolidated visual tokens are concatenated channel-wise to form the concatenated visual tokens $\bar{I} \in \mathcal{R}^{N^V \times d^V}$, where $d^V = \sum_{i\in[m]} d_i^V$.

After that, a *visual projector* is applied to project the concatenated visual token $\bar{I}$ to have the same dimension with the language model input, resulting in $\tilde{I} \in \mathcal{R}^{N^V \times d_{\text{LLM}}}$, where d_{LLM} is the feature dimension of the LLM's input.

An LLM $f_{\text{LLM}}(\cdot)$ then takes the visual tokens $\tilde{I}$ and the textual instruction tokens T as input to generate the instruction-following response $Y = \{y_i\}_{i\in[L]}$ as,

$$p(Y|\tilde{I}, T) = \prod_{i=1}^{L} p(y_i|\tilde{I}, T, y_{<i}), \tag{5.5}$$

where L is the length of the response, and $y_{<i}$ is the previous tokens of y_i.

5.3.2 Conditional Token Reduction (CoTR)

The CoTR module, illustrated in Figure 5.3, is a query-based module that takes a group of visual tokens $\{I_i\}_{i\in[m]}$, text tokens T, and a group of query tokens $\{Q_i \in \mathcal{R}^{N^V \times d_i^V}\}_{i\in[m]}$ as input, and outputs a group of consolidated visual tokens $\{\bar{I}_i\}_{i\in[m]}$. Specifically, for the visual tokens I_i generated by the i-th vision expert, the CoTR module computes an attention score $\alpha_i \in \mathcal{R}^{N^V \times N_i^V}$ using the query token $Q_i \in \mathcal{R}^{N^V \times d_i^V}$, text tokens T, the visual tokens from other vision experts $\{I_j\}_{j\in[m]\setminus\{i\}}$, and the visual token I_i, as

$$s_i^{\text{QUERY}} = \hat{Q}_i \hat{I}_i^\top \in \mathcal{R}^{N^V \times N_i^V}, \tag{5.6}$$

$$s_i^{\text{SELF}} = \mathbf{1}\hat{I}_i \hat{I}_i^\top \in \mathcal{R}^{1 \times N_i^V}, \tag{5.7}$$

$$s_i^{\text{CROSS}} = \sum_{j\in[m]\setminus\{i\}} \hat{I}_j \hat{I}_i^\top \in \mathcal{R}^{1 \times N_i^V}, \tag{5.8}$$

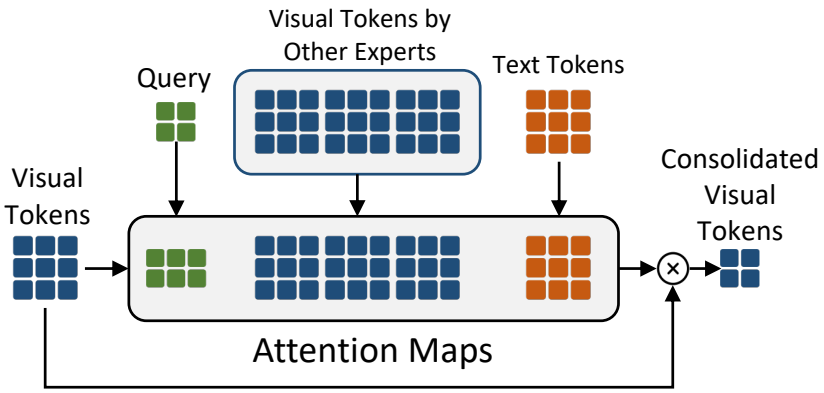


Figure 5.3: The overview of the proposed CoTR. The CoTR module takes a group of visual tokens, a learnable query, and text tokens as input, and outputs a consolidated group of visual tokens to reduce the number of visual tokens.

$$s_i^{\text{TEXT}} = \mathbf{1}\hat{T}\hat{I}_i^{\top} \in \mathcal{R}^{1\times N_i^V}, \tag{5.9}$$

$$\alpha_i = \frac{\text{softmax}(s_i^{\text{QUERY}} + s_i^{\text{SELF}} + s_i^{\text{CROSS}} + s_i^{\text{TEXT}})}{\sqrt{d_i^V}}, \tag{5.10}$$

where $\hat{Q}_i$, $\hat{I}_i, \forall i \in [m]$, and $\hat{T}$ denote the query token, visual tokens, and text tokens projected by learnable linear projections, respectively. The term $\mathbf{1}$ is a vector of ones used to compute the self-attention score. The attention score α_i is then used to compute the consolidated visual tokens $\bar{I}_i$ as,

$$\bar{I}_i = \alpha_i I_i \in \mathcal{R}^{N^V \times d_i^V}. \tag{5.11}$$

In this way, redundant visual tokens are aggregated into a compact set of consolidated visual tokens which significantly improves the efficiency. The length of the query tokens can be adjusted to control the number of visual tokens according to the specific task requirements. Moreover, as we use four different similarities, CoTR can capture the complex relationships between multiple sources of features, leading to more informative consolidated visual tokens.

Finally, the consolidated visual tokens $\{\bar{I}_i\}_{i\in[m]}$ are concatenated channel-wise to form the concatenated visual tokens $\bar{I} \in \mathcal{R}^{N^V \times d^V}$, where $d^V = \sum_{i\in[m]} d_i^V$. These concatenated tokens are then projected using a vision projector to have the same dimension as the LLM input size, resulting in $\tilde{I} \in \mathcal{R}^{N^V \times d_{\text{LLM}}}$.

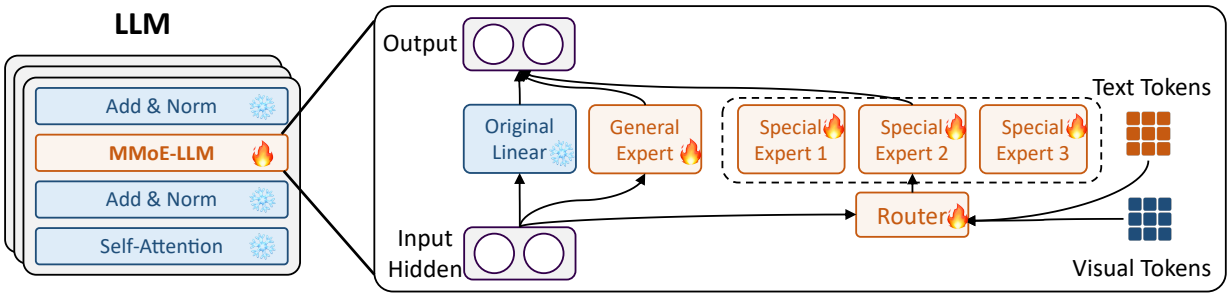


Figure 5.4: The overview of the proposed MMoE-LLM for LLM fine-tuning. MMoE (language) consists of a set of LoRA experts and a routing network that selects the appropriate expert based on the input visual tokens and textual instructions. Moreover, a general expert is employed to learn the general knowledge and improve the robustness of the model.

5.3.3 Mixture of Multi-modal Experts (MMoE)

Our mixture of multi-modal experts module, MMoE, incorporates multiple vision experts to boost visual understanding and multiple LoRA language experts to enhance understanding. MMoE comprises MMoE-Vision and MMoE-LLM.

Effective visual comprehension (MMoE-Vision). As described in Section 5.3.1, drawing inspiration from previous work [6, 180], visual tokens are generated by multiple vision experts, each extracting informative features from different perspectives to enrich visual understanding.

Effective understanding ability (MMoE-LLM). To ensure efficient training with minimal computational overhead, we introduce MMoE for LLM tuning, following the mixture of LoRA experts [48], as shown in Figure 5.4. The vanilla mixture of LoRA experts consists of a set of experts [45, 192], *i.e.*, $\{f_i^E(\cdot)\}_{i\in[E]}$, and a routing network $f^{\text{ROUTING}}(\cdot)$ that outputs the routing probability $R \in \mathcal{R}^E$ taking the hidden state of the current layer as input, where E is the number of experts. Then, based on the routing probability, only the top-k experts will be activated.

Different from the vanilla version, the routing network in MMoE-LLM takes the visual tokens $\tilde{I}$, the textual instruction tokens T, and the hidden state x as the input to compute the routing probability $R = \text{softmax}(f^{\text{ROUTING}}(\tilde{I}, T, x)) \in \mathcal{R}^E$, which facilitates better switching between the experts. Moreover, MMoE-LLM also employs a general expert $f_{\text{GEN}}^E(\cdot)$ to capture the general knowledge and improve the overall robustness of the model.

We select k experts with the highest routing probabilities, *i.e.*, $E' = \text{Top}_k(R)$, and compute the output of the MMoE-LLM as,

$$\text{MMoE-LLM} = f_{\text{GEN}}^E(x) + \sum_{i\in E'} f_i^E(x)/k\,. \tag{5.12}$$

	Stage 1	Stage 2	Stage 3
LLM	❄	🔥	❄
Visual Experts	❄	🔥	❄
Visual Projector	🔥	🔥	🔥
CoTR	-	-	🔥
MMoE-LLM	-	-	🔥

Table 5.1: The training stages of LEO-Mini. CoTR and MMoE-LLM are added and fine-tuned in the third stage, while the LLM and vision experts (MMoE-Vision) remain frozen.

With the original linear layer $f^{\text{ORI}}(\cdot)$, the final output is computed as,

$$\underbrace{f^{\text{ORI}}(x)}_{\text{Original Linear}} + \underbrace{f^{E}_{\text{GEN}}(x)}_{\text{General Expert}} + \underbrace{\sum_{i \in E'} f^{E}_{i}(x)/k}_{\text{Selected Experts}} . \tag{5.13}$$

5.3.4 Training stages

The training of LEO-Mini consists of three stages, as shown in Table 5.1. **Stage 1: Warming up for the visual projector**. This stage pre-trains the visual projector while keeping the LLM and vision experts frozen. The LLM and vision experts are initialized from the base models, while the vision projector is randomly initialized. **Stage 2: Supervised fine-tuning**. In this stage, we fine-tune the entire model, including the LLM, vision experts, and visual projector. **Stage 3: Supervised fine-tuning for reducing the number of visual tokens**. In this stage, we introduce CoTR to the model and perform LoRA fine-tuning (MMoE-LLM) for efficiency. Specifically, CoTR and MMoE-LLM are randomly initialized and fine-tuned, while the LLM and vision experts remain frozen. To avoid the risk of routing collapse [179], we employ a balanced loss for MMoE with a hyperparameter $\lambda = 0.05$, following previous works [179, 192].

We expect the model to learn a comprehensive understanding of the input images in the first two stages, and then in the third stage, the model is guided to focus on the most important information for improved efficiency.

5.4 Experiments

5.4.1 Implementations

Baselines. We compare LEO-MINI with general MLLMs [24, 38, 123, 125, 130, 133, 134, 180] and token-reduction MLLMs [78, 118, 122, 178, 232, 247].

Models. We use Vicuna-v1.5-7B [28] and Llama3-8B [145] as the LLM. For vision experts, we follow the general design of EAGLE [180] and use CLIP [174], ConvNeXt [144], Pix2Struct [103] and EVA-02 [52] for LEO-MINI-Llama-8B. Similarly, we add another vision expert, SAM [100], for LEO-MINI-Vicuna-7B. The visual projector is a 2-layer MLP with the GELU activation function [73]. MMOE-LLM is only applied to the MLP in each block of the LLM. We use 3 experts for MMOE-LLM with $k = 1$ and 1 general expert being consistently activated. Each expert is a LoRA block [48] with rank of 16. The routing network is a 2-layer MLP with the GELU activation function.

Training. LEO-MINI uses the same training data as LLaVA-v1.5 [133] in stage 3 with 665K instruction data. For stage 1 and stage 2, we follow the same data as EAGLE [180]. The training was conducted on 8 A6000 GPUs (48 GB) using DeepSpeed's Zero2 strategy [175].

5.4.2 Benchmark Datasets

We evaluate our method on the following benchmark datasets: MME [55], MMBench [139], Seed-Bench [107], GQA [84], SQA [148], MMMU [234], POPE [124], AI2D [95], VizWiz [65], TextVQA [184], DocVQA [155], ChartQA [154], and OCRBench [140].

MME [55]. The MME benchmark is designed to rigorously evaluate a model's perceptual and cognitive abilities through 14 subtasks. It employs carefully constructed instruction-answer pairs and concise instructions to minimize data leakage and ensure fair evaluation. This setup provides a robust measure of a model's performance across various tasks.

MMBench [139]. MMBench offers a hierarchical evaluation framework, categorizing model capabilities into three levels. The first level (L-1) focuses on perception and understanding. The second level (L-2) expands this to six sub-abilities, while the third level (L-3) further refines these into 20 specific dimensions. This structured approach allows for a nuanced and comprehensive assessment of a model's multifaceted abilities.

Seed-Bench [107]. SEED-Bench consists of 19K multiple-choice questions with accurate human annotations, covering 12 evaluation dimensions, including both the spatial and temporal understanding.

GQA [84]. GQA is structured around three core components: scene graphs, questions, and images. It includes not only the images themselves but also detailed spatial features and object-level attributes. The questions are crafted to assess a model's ability to comprehend visual scenes and perform understanding tasks based on the image content.

ScienceQA [148]. ScienceQA spans a wide array of domains, including natural, language, and social sciences. Questions are hierarchically categorized into 26 topics, 127 categories, and 379 skills, providing a diverse and comprehensive testbed for evaluating multimodal understanding, multi-step understanding, and interpretability.

MMMU [234]. MMMU includes 11.5K meticulously collected multimodal questions from college exams, quizzes, and textbooks, covering six core disciplines: Art & Design, Business, Science, Health & Medicine, Humanities & Social Science, and Tech & Engineering. These questions span 30 subjects and 183 subfields, comprising 30 highly heterogeneous image types, such as charts, diagrams, maps, tables, music sheets, and chemical structures.

POPE [124]. POPE is tailored to assess object hallucination in models. It presents a series of binary questions about the presence of objects in images, using accuracy, recall, precision, and F1 score as metrics. This approach offers a precise evaluation of hallucination levels under different sampling strategies.

AI2D [95]. AI2D is a dataset of over 5000 grade school science diagrams with over 150000 rich annotations, their ground truth syntactic parses, and more than 15000 corresponding multiple choice questions.

VizWiz [65]. VizWiz consists of over 31,000 visual questions originating from blind people who each took a picture using a mobile phone and recorded a spoken question about it, together with 10 crowdsourced answers per visual question.

TextVQA [184]. TextVQA emphasizes the integration of textual information within images. It evaluates a model's proficiency in reading and understanding about text embedded in visual content, requiring both visual and textual comprehension to answer questions accurately.

DocVQA [155]. DocVQA consists of 50,000 questions defined on 12,000+ document images.

ChartQA [154]. CharQA is a large benchmark covering 9.6K human-written questions as well as 23.1K questions generated from human-written chart summaries.

OCRBench [140]. OCRBench is a comprehensive benchmark for evaluating the OCR capabilities of multi-modal language models across five key tasks: text recognition, scene text-centric and document-oriented VQA, key information extraction, and handwritten mathematical expression recognition.

Model	# of Visual Tokens	General					OCR	Knowledge		
		MME^P	MMBench	$SEED^I$	GQA	VizWiz	TextVQA	SQA	POPE	MMMU
VoCo-LLaMA [232]	1	1323.3	58.8	53.7	57.0	-	-	65.4	81.4	-
LLaMA-VID [122]	2	-	-	-	55.5	-	49.0	67.7	83.1	-
PruMerge [178]	32	1350.3	60.9	-	-	-	56.0	72.0	76.3	-
MQT-LLaVA [78]	64	1464.3	63.5	-	60.0	51.5	-	67.0	83.6	34.4
Token-Packer [118]	64	-	64.1	-	61.1	50.7	-	-	86.3	-
LLaVA-Mini [247]	64	1476.8	67.5	60.2	61.8	58.5	59.1	69.7	85.3	-
LEO-MINI-Vicuna-7B	64	1542.6	67.8	73.2	64.0	51.4	70.1	73.3	90.0	34.1
Δ	-	↑ 65.8	↑ 0.3	↑ 13.0	↑ 2.2	↓ 7.1	↑ 11.0	↑ 1.3	↑ 3.7	↓ 0.3
LEO-MINI-Llama-8B	64	**1583.0**	**77.0**	**75.8**	**64.5**	**69.3**	**75.1**	**84.5**	**90.3**	**38.8**
Δ	-	↑ 106.2	↑ 9.5	↑ 15.6	↑ 2.7	↑ 10.8	↑ 16.0	↑ 12.5	↑ 4.0	↑ 4.4

Table 5.2: Comparison to token reduction methods on general, OCR, and knowledge-based tasks. Best in **Bold**. Second best in Underline.

5.4.3 Evaluation Metrics

MLLM benchmarks use task-specific evaluation protocols, so the reported numbers in Tables 5.2 to 5.5 should be interpreted according to each benchmark's official metric. Unless otherwise stated, all task-performance metrics are higher-is-better.

For multiple-choice benchmarks, including MMBench [139], SEED-Bench [107], ScienceQA (SQA) [148], MMMU [234], and AI2D [95], we report accuracy under the official evaluation protocol. For GQA [84], TextVQA [184], DocVQA [155], and ChartQA [154], we report the official VQA-style score used by each benchmark, which measures whether the generated answer matches the ground-truth answer under the benchmark's answer normalization or scoring rule. For POPE [124], which evaluates object hallucination through binary object-presence questions, we report accuracy, following common MLLM evaluation practice. For MME [55], we report the official perception score MME^P, which aggregates the perception-related subtasks; a larger score indicates stronger perceptual understanding. For OCRBench [140], we report the official OCRBench score, where higher values indicate stronger text recognition and OCR-oriented visual question answering ability.

In addition to task performance, this chapter evaluates efficiency. The column "# V tokens" or "# of VT" denotes the number of visual tokens passed into the LLM, where fewer tokens generally indicate lower LLM-side computation. In Table 5.6, FLOPs are measured in tera floating-point operations (T), CUDA Time is measured in seconds for one forward inference example, and GPU Memory is measured in gigabytes (GB). For these efficiency metrics, lower values are better. Thus, the goal of LEO-MINI is to maintain or improve the benchmark scores while reducing the number of visual tokens, FLOPs, and inference time.

LLM	Model	# V tokens	MME^P	MMBench	SEED^I	GQA	SQA	MMMU	POPE	AI2D	VizWiz	TextVQA	DocVQA	ChartQA	OCRBench
Vicuna-7B [28] & Qwen-7B [8]	InstructBLIP [38]	32	-	-	-	49.2	60.5	-	-	-	34.5	50.1	-	-	-
	LLaVA-1.5 [133]	576	1511	64.3	66.1	62.0	66.8	-	85.9	-	50.0	58.2	-	-	
	LLaVA-NeXt [134]	2880	1519	-	-	64.2	70.1	35.1	-	66.6	57.6	64.9	74.4	54.8	-
	InternVL [24]	1792	1525	64.3	-	62.9	-	-	86.4	-	52.5	57.0	-	-	-
	VILA [130]	576	1533	68.9	61.1	62.3	68.2	-	85.5	-	57.8	64.4	-	-	-
	Monkey [125]	256	-	-	-	60.7	69.4	-	-	62.6	**61.2**	67.6	66.5	65.1	-
	MoVE-based MLLMs														
	Brave-X5 [94]	160	-	-	-	52.7		-	87.6	-	54.2	-	-	-	-
	Mini-Gemini [123]	576	1523	65.8	-	64.5	71.1	36.1	-	-	-	65.2	-	-	-
	Mousi-X3 [51]	576	-	66.8	66.0	63.3	70.2	-	87.3	-	-	58.0	-	-	-
	LEO [6]	512	-	72.9	72.2	64.8	**78.5**	36.4	88.0	69.6	57.9	68.8	**80.1**	**71.0**	-
	DeepSeek-VL [146]	576	-	**73.2**	70.4	-	-	**36.6**	88.1	-	-	-	-	-	45.6
	Eagle-X5 [180]	1024	1528	68.4	**73.9**	**64.9**	69.8	36.3	88.8	-	54.4	**71.2**	78.6	67.8	52.9
	LEO-MINI-Vicuna-7B	**64** (↓ 97.77%)	**1543**	67.8	73.2	64.0	73.3	34.1	**90.0**	**72.2**	51.4	70.1	75.3	66.8	**55.6**
Llama-8B [145]	Cambrian-1 [193]	576	1547	75.9	74.7	64.6	80.4	42.7	-	73.0	-	71.7	77.8	73.3	62.4
	LLaVA-NeXT [134]	2880	1604	72.5	72.7	65.2	72.8	41.7	-	71.6	-	64.6	72.8	69.5	49.0
	MoVE-based MLLMs														
	Mini-Gemini-HD [123]	2880	**1606**	72.7	73.2	64.5	75.1	37.3	-	73.5	-	70.2	74.6	59.1	47.7
	Eagle-X4-Plus [180]	1024	1559	75.9	**76.3**	**64.9**	84.3	**43.4**	-	**76.1**	-	**77.1**	**86.6**	80.1	**62.6**
	LEO-MINI-Llama-8B	**64** (↓ 97.77%)	1583	**77.0**	75.8	64.5	**84.5**	38.8	**90.3**	75.7	**69.3**	75.1	86.3	**80.5**	62.4

Table 5.3: Comparison to general and MoVE (mixture of vision experts)-based MLLMs. Best in **Bold**. Second best in Underline.

5.4.4 Main Results

We compare our LEO-MINI with several state-of-the-art token reduction MLLMs (Table 5.2), MoE-based MLLMs (Table 5.3), and general MLLMs (Table 5.3) on various tasks.

Comparison with token reduction MLLMs [78, 118, 122, 178, 232, 247]. The results are shown in Table 5.2. LEO-MINI outperforms all the token reduction MLLMs on all tasks. Specifically, LEO-MINI with Llama3-8B achieves 1583.0 on MME (perception), 77.0 on MMBench, 75.8 on SeedBench, 64.5 on GQA, 69.3 on VizWiz, 75.1 on TextVQA, 84.5 on SQA, 90.3 on POPE, and 38.8 on MMMU, outperforming the best baseline by a large margin. This demonstrates the effectiveness of LEO-MINI in improving the performance of MLLMs on various tasks. Moreover, LEO-MINI with Vicuna-7B also outperforms the best baseline on all tasks except for GQA and MMMU, showing the generalization ability of LEO-MINI with different LLMs.

Comparison with general and MoVE-based MLLMs [6, 24, 38, 51, 123, 125, 130, 133, 134, 146, 149, 180, 193]. The results are shown in Table 5.3. With Vicuna-7B, LEO-MINI outperforms all the general and MoVE-based MLLMs on MME, POPE, AI2D, and OCRBench by 15, 1.2, 2.6, and 2.7 points, respectively, with only 64 visual tokens. LEO-MINI with Vicuna-7B

also achieves the second best performance on SeedBench^I, SQA, and TextVQA. On the other side, with Llama3-8B, LEO-MINI achieves stronger performance as it outperforms all the general and MoVE-based MLLMs on MMBench, SQA, and CharQA by 1.1, 0.2, and 0.4 points, respectively, while reducing the number of visual tokens by 97.77% compared to LLaVA-NeXt and Mini-Gemeni-HD. Moreover, compared to the general MLLMs, LEO-MINI with Llama3-8B achieves comparable performance on other benchmark datasets as the discrepancy is marginal. Specifically, LEO-MINI with Llama3-8B achieves the second-best performance on SeedBench^I, AI2D, TextVQA, DocVQA, and OCRBench.

5.4.5 Ablation Studies

In this section, we conduct ablation studies to analyze the effectiveness of different components in LEO-MINI, the number of visual tokens, using other token reduction modules and MoE, and the amount of training data for stage 3 using LEO-MINI-Llama-8B.

Is 1 visual token enough for MLLMs? To understand whether 1 visual token is enough for representing the visual information, we conduct the experiments and present the results in Table 5.4. LEO-MINI with 1 visual token ($N^V = 1$) shows a slight decrease across some metrics compared to LEO-MINI with 64 tokens, which is reasonable. For example, in the MME^P, the score dropped from 1583 to 1565, and in SEED^I, the performance slightly decreased from 75.8 to 75.6. Similarly, in the domains of GQA and SQA, the scores decrease from 64.5 to 64.2 and 84.5 to 83.6, respectively. This trend continues across other evaluated areas such as MMMU, POPE, AI2D, TextVQA, ChartQA, and OCRBench. While the use of only 1 visual token consistently leads to lower performance, it greatly improves model's efficiency as shown in Table 5.6. To further understand how increasing the number of visual tokens impacts the performance, we conduct experiments with 16 and 256 visual tokens. The results show that the performance on most of the benchmarks is improved as the number of visual tokens increases. This is reasonable as more visual tokens can provide more detailed visual information to the model, which can help the model to better understand the visual information. However, we also observe that the performance on some benchmarks is slightly decreased, *e.g.*, AI2D, which might be due to the overfitting issue.

Will more training data help in summarizing the visual information? To understand the impact of the amount of training data on the performance, we conduct the experiment with EAGLE-1.8M SFT data [180] for stage 3 in Table 5.4 with 64 tokens. The results show that the performance is improved on most of the benchmarks, *i.e.*, SeedBench^I, SQA, MMU, POPE, AI2D, TextVQA, ChartQA, and OCRBench. For example, the performance on SeedBench^{Image} is improved from 75.8 to 76.6, and the performance on SQA is improved from 84.5 to 85.9. This

Model	MME^P	SEED^I	GQA	SQA	MMMU	POPE	AI2D	TextVQA	ChartQA	OCRBench
LEO-MINI (64 tokens)	1583	75.8	64.5	84.5	38.8	90.3	75.7	75.1	80.5	62.4
w/ 1 visual token	1565	74.6	64.2	83.6	37.7	89.0	74.5	73.5	80.0	61.7
w/ 16 visual token	1577	75.4	64.4	84.3	38.2	90.1	75.4	74.2	80.2	62.6
w/ 256 visual token	1584	76.1	64.2	85.5	39.0	90.7	75.5	75.5	80.5	63.0
w/ 1.8m SFT data (stage 3)	1548	76.6	64.1	85.9	39.4	90.3	77.9	75.7	80.9	63.1

Table 5.4: Ablation studies of LEO-MINI. We explore the impact of the number of visual tokens, the amount of training data, and the MMOE. Numbers in green indicate the performance is improved compared to LEO-MINI (64 tokens).

Token Reduction	MoE	MME^P	SEED^I	GQA	SQA	MMMU	POPE	AI2D	TextVQA	ChartQA	OCRBench
COTR	MMOE-Vision + MMOE-LLM	**1583**	**75.8**	**64.5**	**84.5**	**38.8**	90.3	**75.7**	**75.1**	**80.5**	**62.4**
MQT	MMOE-Vision + MMOE-LLM	1537	75.3	64.4	84.2	37.0	**90.4**	75.3	74.2	80.4	61.6
MQT	-	1435	-	61.6	67.6	34.8	84.4	-	-	-	-
COTR	MMOE-Vision + LORA-MOE	1521	75.3	**64.5**	83.4	38.2	90.3	75.4	74.0	80.0	61.7

Table 5.5: Ablation study on the effectiveness of COTR and MMOE-LLM. We use Llama3-8B as the backbone.

indicates that more training data can help the model to better summarize the visual information and improve the performance on various tasks.

How do other token reduction methods work? To understand how our proposed token reduction module, *i.e.*, COTR, works, we compare it with a representative token reduction method, *i.e.*, MQT-LLaVA [78], as shown in Table 5.5, while keeping the MMOE unchanged. The results show that COTR outperforms MQT on most benchmarks, demonstrating the effectiveness of our proposed method. Moreover, compared to MQT-LLaVA (the third row), MMOE improves the performance of MQT-LLaVA on most benchmarks, showing the importance of introducing multiple experts for extracting informative tokens.

How does MMOE-LLM improve the performance? We conduct an ablation study to investigate the effectiveness of MMOE-LLM in Table 5.5. Results show that MMOE-LLM significantly improves the performance on most benchmarks, compared to the baseline with MMOE-Vision and LoRA-MoE [216]. MMOE-LLM improves performance by 62, 0.5, 1.1, 0.6, 0.3, 1.1, 0.5, and 0.7 on MME^P, SEEDBench^I, SQA, MMMU, AI2D, TextVQA, ChartQA, and OCRBench, respectively. This demonstrates the effectiveness of MMOE-LLM in better switching between different experts to better understand visual information.

How does MMOE-LLM switch between different LORA experts? To understand how our MMOE-LLM switches between different LoRA experts, we visualize the expert choice in Fig-

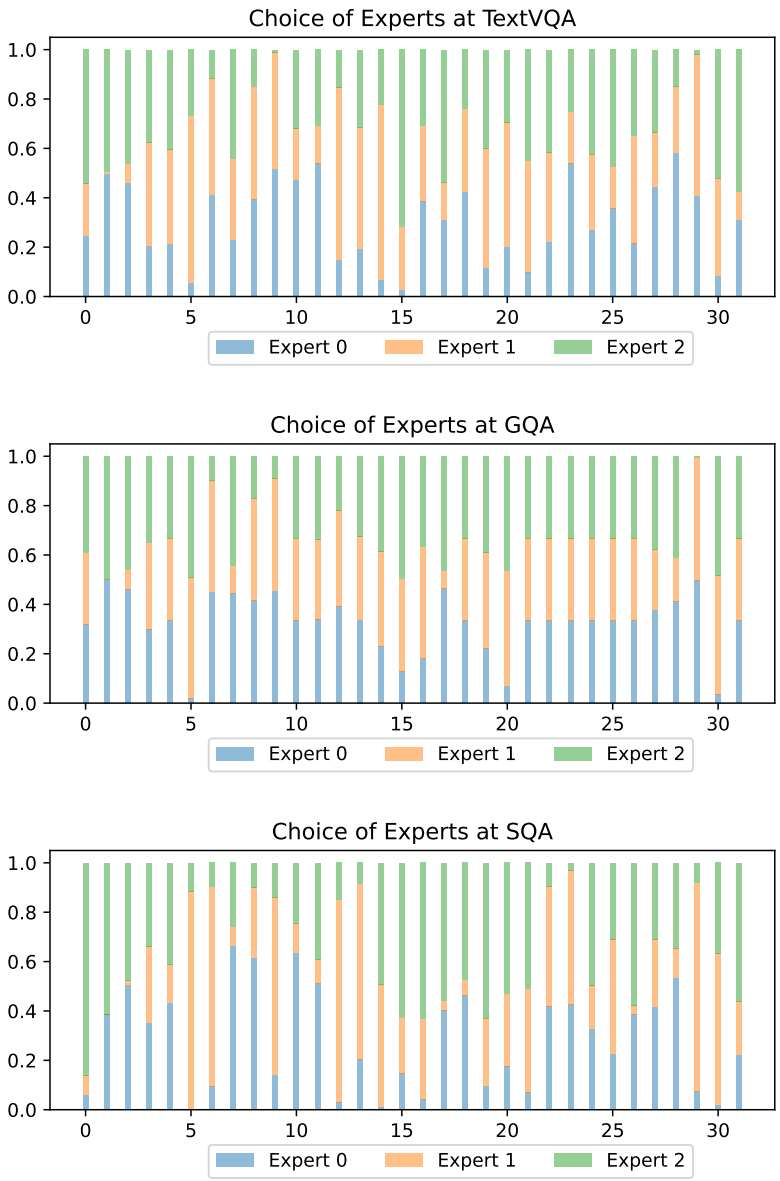


Figure 5.5: Visualization of the expert choice using LEO-MINI-Llama-8B on TextVQA, ScienceQA, and GQA. Best viewed in color.



Figure 5.6: Visualization of the expert choice using LEO-MINI-Llama-8B (right) on TextVQA, ScienceQA, and GQA compared to the vanilla MoE (left) on selected layers. Best viewed in color.

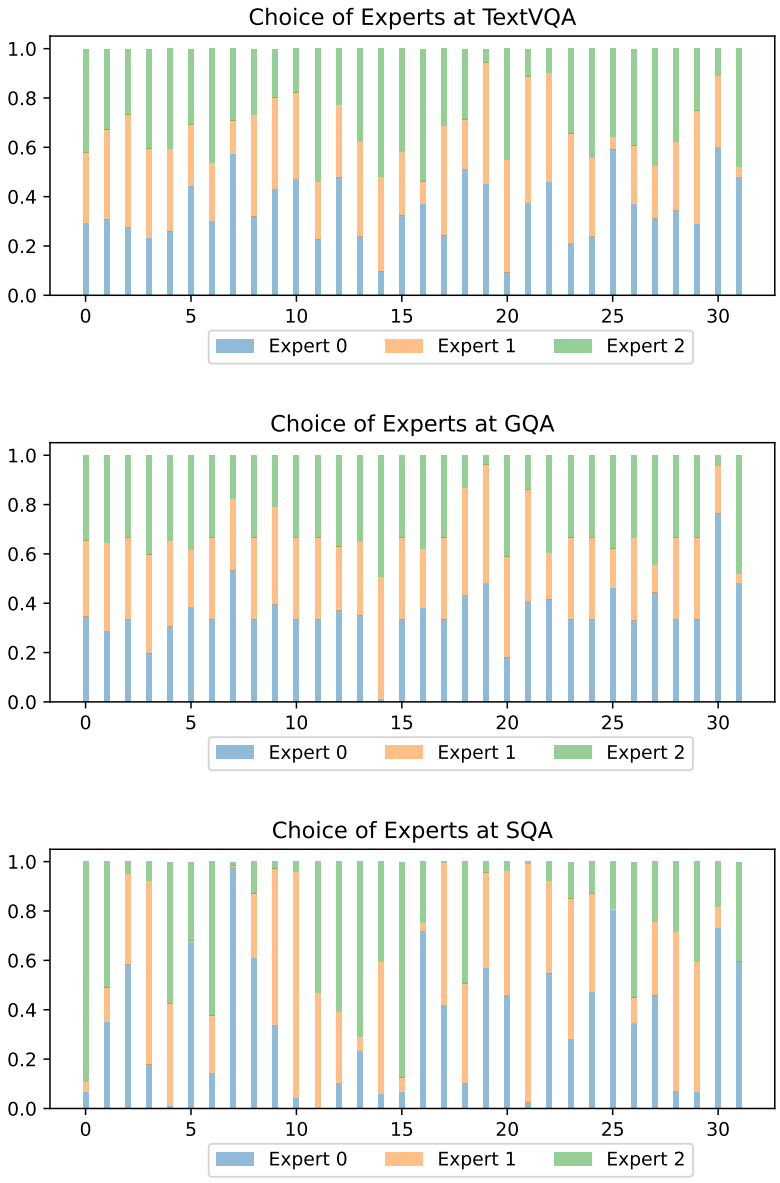


Figure 5.7: Visualization of the expert choice using the vanilla MoE on TextVQA, ScienceQA, and GQA. Best viewed in color.

ure 5.5 with GQA (general), TextVQA (OCR), and SQA (Knowledge). The visualization shows that MMoE-LLM can effectively switch between different experts to better understand visual

Models	# of VT	FLOPs (T)	CUDA Time (s)	GPU Memory (GB)
LLaVA-Next [134]	2880	45.5	7.037	14.74
Eagle-X4-Plus [180]	1024	29.4	2.073	20.10
LEO-MINI-Llama-8B	1	13.7	0.593	19.28
LEO-MINI-Llama-8B	64	16.4	0.655	19.34
Δ	↓ 97.77%	↓ 63.83%	↓ 90.69%	

Table 5.6: Efficiency analysis of LEO-MINI and other MLLMs using Llama3-8B. "VT" represents visual tokens. Δ indicates the difference between LEO-MINI and LLaVA-Next.

information. For example, in the TextVQA and SQA tasks, the model mainly activates expert 2 at layer 0, while for GQA, the model evenly activates three experts. When it comes to the middle layer, such as layer 29, the model evenly activates experts 0 and 1 for TextVQA and GQA, while for SQA, the model mainly activates expert 1, respectively. Also, we notice that for some layers, such as layers 0, 3, 5, 6, 23, 25, 29, and 31, MMOE-LLM selects different experts for different benchmarks, while for some layers, such as layers 1, 2, 17, and 27, the model selects similar experts for different benchmarks. This demonstrates the effectiveness of MMOE-LLM in better understanding visual information by switching between different experts.

How does MMOE-LLM switch between different LORA experts compared to the vanilla MoE? To understand how MMOE-LLM switches using the additional visual and input signals, we compare it with the vanilla MoE in Figure 5.6. The full routing results of the vanilla MoE are shown in Figure 5.7, while the results of MMOE-LLM are shown in Figure 5.5. The results show that at different layers, MMOE-LLM shows different expert preferences for different benchmarks, while the vanilla MoE shows similar expert preferences for different benchmarks.

5.4.6 Efficiency Analysis

To understand how efficient LEO-MINI is, we compare the number of visual tokens, FLOPs, and CUDA processing time of LEO-MINI with other MLLMs [134, 180, 193] using Llama3-8B. Specifically, we use Pytorch Profiler to measure the FLOPs and CUDA time using the figure and "What is shown in this image?" as input. We use one A6000 GPU with 48 GB memory to inference the models.

The comparison is shown in Table 5.6. LEO-MINI with Llama3-8B is 97.77% more efficient in terms of visual tokens, 63.83% in terms of FLOPs, and 90.69% in terms of CUDA time compared to LLaVA-Next. Moreover, as shown in Figure 5.1, LEO-MINI outperforms LLaVA-Next

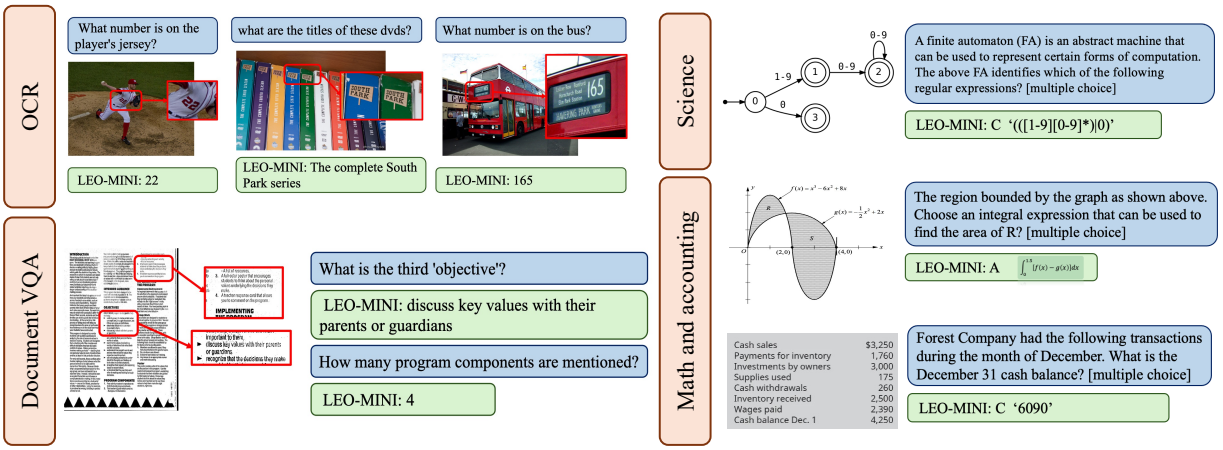


Figure 5.8: Qualitative results across four vision-language tasks demonstrating LEO-MINI's detailed visual understanding. The images are taken from TextVQA [184], DocVQA [155], and MMMU [234].

by 6% on average performance with only 64 visual tokens. Even compared to the most powerful MLLM with similar performance, *i.e.*, Eagle-X4-Plus, LEO-MINI is 93.75% more efficient in terms of visual tokens, 44.10% in terms of FLOPs, and 68.40% in terms of CUDA time. More excitingly and importantly, LEO-MINI with only 1 token achieves comparable performance to the state-of-the-art MLLMs with thousands of visual tokens with faster inference time and lower computational cost in terms of FLOPs.

5.4.7 Qualitative Analysis

To understand the detailed visual understanding of LEO-MINI, we conduct a case study for qualitative analysis on four vision-language tasks [155, 184, 234] as shown in Figure 5.8. We use LEO-MINI-Llama3-8B with 64 tokens.

Though the model only takes 64 visual tokens, LEO-MINI performs effectively in capturing visual details, such as accurately identifying the numbers and small book title in OCR. Moreover, LEO-MINI is able to understand the order and count numbers, as for document VQA, LEO-MINI precisely finds the correct results. For science and math questions, LEO-MINI also shows incredible understanding ability. For science, LEO-MINI can effectively translate the computation diagram into mathematical equations. For math and accounting, LEO-MINI successfully finds the true expression and does calculations correctly.

Overall, LEO-MINI shows strong visual understanding with improved efficiency, making it a practical solution for multi-modal understanding.

5.5 Conclusion and Limitations

In this chapter, to address the redundancy of visual tokens in MLLMs, we propose a novel MLLM, LEO-MINI, that significantly reduces the number of visual tokens while boosting visual understanding capabilities. LEO-MINI incorporates a novel token reduction module, COTR, to consolidate a large number of visual tokens into a smaller set of tokens, using the similarity between visual tokens, text tokens, and a compact learnable query. However, simply reducing the number of visual tokens leads to information loss. To avoid the loss and boost the visual comprehension ability with minimal computational overhead, LEO-MINI employs a novel mixture of multi-modal experts module, MMOE, that includes a set of language (LoRA) experts and a set of vision experts trained on diverse domain-specific data. For better switching between different LoRA experts, MMOE employs a new router that takes the text and visual tokens as additional inputs. MMOE also includes a general LoRA expert that is always activated to learn general knowledge. We evaluate LEO-MINI on various vision-and-language tasks, showcasing its potential for practical applications with improved efficiency and performance compared to existing efficient MLLMs.

Limitations. First, the proposed COTR needs training. Introducing a training-free token reduction module might be a promising direction. Second, the proposed MMOE is designed to be efficient and scalable, but it may not be optimal for all tasks. For vision experts, due to the computational limitations, we use a fixed set of experts. It would be interesting to explore more vision experts from a wide range of domains and introduce more advanced expert selection mechanisms, such as dynamic vision expert routing. Last, due to the computational limitations, we did not test the efficiency of LEO-MINI using bigger LLMs with 13B or 67B parameters.

Potential Risks. LEO-MINI is an MLLM that can be used for various tasks, including visual question answering, image captioning, and document understanding. However, as an inherent risk of (multi-modal) LLMs, LEO-MINI may generate biased or harmful content, especially when the query data contains sensitive information. We recommend that users carefully review the generated content and apply appropriate filters to mitigate potential risks.

Chapter 6

Conclusion and Future Work

This thesis has presented three interconnected methods that leverage multi-modal feature alignment to address OOD challenges for robust AI deployment in open-world environments. We systematically addressed three levels of OOD handling: OOD detection (binary identification of whether data is OOD), OOD recognition (assigning semantic labels to novel categories via 3D open-vocabulary semantic segmentation), and OOD understanding (understanding and responding to unfamiliar scenarios). To summarize, we have made the following contributions:

- In Chapter 3, we proposed SUPREME, a novel few-shot OOD detection method that mitigates excessive or poorly calibrated modality gaps between image and text embeddings in VLMs. By incorporating ID image prototypes alongside ID text prototypes, and introducing the Bias Prompt Generation (BPG) and Image-Text Contrastive (ITC) modules, SUPREME effectively reduces false positive rates and achieves state-of-the-art performance across multiple OOD detection benchmarks.
- In Chapter 4, we proposed AIDE, an aligned 3D open-vocabulary semantic segmentation framework that addresses the misalignment between 3D models and text encoders. Through our CLIP-Rewarded Alignment Module for generating high-quality 3D-scene-image-to-text pairs and the Adaptive Segmentation Module for adapting text encoders to 3D contexts, AIDE significantly outperforms previous methods on representative benchmarks.
- In Chapter 5, we proposed LEO-MINI, an efficient MLLM that incorporates the novel Conditional Token Selection (COTR) module for visual token reduction and the Conditional Mixture of Multi-Modal Experts (MMOE) module for enhanced visual understanding. By leveraging the strong generalization capabilities of MLLMs while addressing their

computational constraints, LEO-MINI enables practical OOD understanding in resource-constrained environments, achieving competitive performance while using significantly fewer visual tokens.

While each method targets a different aspect of OOD problems, they share a common technical foundation: improved multi-modal feature alignment to calibrate cross-modal compatibility—whether between image and text embeddings for OOD detection, between 3D point clouds and language for OOD recognition, or between visual tokens and language for OOD understanding. This unified perspective on multi-modal feature alignment provides a principled approach for addressing complex OOD challenges in open-world AI deployment.

These methods have potentially wide applicability across various AI systems. SUPREME can be integrated into any VLM-based recognition system to improve robustness against novel inputs, addressing both semantic shift (novel categories) and covariate shift (distribution changes). AIDE enables 3D perception systems to recognize unlimited categories beyond their training label space, which is essential for autonomous systems operating in dynamic environments where novel objects frequently appear. LEO-MINI can be adopted in resource-constrained applications where OOD understanding is critical, such as edge devices in autonomous vehicles or portable medical diagnostic systems, enabling AI systems to understand unfamiliar scenarios in real-time.

However, there are several practical considerations when applying our methods:

1. **The reliance on pre-trained VLMs:** All three approaches build upon pre-trained VLMs such as CLIP. While this enables strong zero-shot and few-shot capabilities, the performance is inherently bounded by the quality and biases of these foundation models. Future work should investigate methods to further adapt or fine-tune these models for specific domains while preserving their generalization capabilities.

2. **The trade-off between efficiency and performance:** While LEO-MINI demonstrates that aggressive token reduction can maintain competitive performance, there may be scenarios requiring fine-grained visual understanding where more visual tokens are necessary. Understanding when and how to dynamically adjust the number of visual tokens based on task requirements remains an open question.

3. **Evaluation in real-world deployment scenarios:** While we report improvements on standard benchmarks, translating these improvements to real-world applications requires additional validation. For instance, in autonomous driving, improved OOD detection metrics may not directly correlate with safer driving outcomes. A more comprehensive evaluation framework that considers downstream task performance is essential.

Building on the progression from OOD detection through recognition to understanding established in this thesis, several directions emerge as promising avenues for future research:

1. **Unified detection–recognition–understanding pipelines:** While our methods address OOD detection, recognition, and understanding as separate tasks, a natural extension is to develop unified pipelines that seamlessly integrate all three capabilities. For instance, an autonomous system could first detect novel objects using SUPREME, recognize them in 3D space using AIDE, and then understand the scenario using LEO-MINI. Developing end-to-end architectures that jointly optimize these capabilities could lead to more robust open-world AI systems.

2. **Advancing OOD detection with richer semantics:** Our work on multi-modal prototypes demonstrates the value of incorporating both image and text modalities for OOD detection. Future work could extend this approach by leveraging the understanding capabilities of MLLMs to provide richer semantic understanding of why certain inputs are OOD, enabling more interpretable and actionable detection results.

3. **Extending open-vocabulary recognition to dynamic environments:** While AIDE enables recognition of novel 3D categories, real-world environments are dynamic and continuously evolving. Future research could investigate continual learning approaches that allow models to incrementally learn new categories while maintaining performance on previously seen ones, without catastrophic forgetting.

4. **Reliable and factual understanding over OOD scenarios:** While LEO-MINI improves understanding efficiency, the reliability and factual correctness of understanding outputs remain challenging, particularly for OOD scenarios. Current VLMs are prone to hallucinations when encountering unfamiliar inputs. Future work should investigate methods to improve the reliability of understanding outputs, potentially by integrating uncertainty estimation from OOD detection into the understanding process.

5. **Multi-modal knowledge distillation:** Building on our recent work on efficient MLLMs, distilling knowledge from multiple specialized visual experts into a single efficient encoder could further reduce computational overhead while maintaining diverse visual understanding capabilities [204]. This is particularly relevant for deployment on edge devices where OOD scenarios are most likely to occur.

6. **Temporal understanding for sequential OOD scenarios:** Extending our methods to handle temporal sequences—such as video understanding where OOD objects appear and

evolve over time—presents both opportunities and challenges. The token reduction strategies in LEO-MINI could be valuable for processing long video sequences, while the detection and recognition capabilities could be extended to track and reason about novel objects across frames.

In conclusion, this thesis addresses the fundamental challenge of handling OOD data, encompassing both semantic shift and covariate shift, through systematic improvements in multimodal feature alignment. We demonstrate that improved alignment between visual and semantic representations can address three levels of OOD handling: binary detection of OOD inputs (SUPREME), semantic recognition of novel categories (AIDE), and understanding unfamiliar scenarios (LEO-MINI). These contributions establish a foundation for deploying AI systems in open-world environments, where the ability to reliably detect, recognize, and understand novel scenarios is essential for safe and robust operation in safety-critical applications such as autonomous driving and medical diagnosis. We hope this thesis will inspire further investigation into multimodal feature alignment techniques for OOD challenges, ultimately advancing the reliability of AI systems in real-world deployment scenarios.

References

[1] Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. SATR: Zero-Shot Semantic Segmentation of 3D Shapes. In *IEEE/CVF International Conference on Computer Vision*, pages 15120–15133, Paris, France, 2023. IEEE.

[2] Nikolas Adaloglou, Felix Michels, Tim Kaiser, and Markus Kollmann. Adapting Contrastive Language-Image Pretrained (CLIP) Models for Out-of-Distribution Detection. *Transactions on Machine Learning Research*, November 2023.

[3] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1543, Las Vegas, NV, USA, 2016. IEEE.

[4] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection, October 2023. arXiv:2310.11511 [cs].

[5] Reza Averly and Wei-Lun Chao. Unified out-of-distribution detection: A model-specific perspective. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1453–1463, 2023.

[6] Mozhgan Nasr Azadani, James Riddell, Sean Sedwards, and Krzysztof Czarnecki. LEO: Boosting Mixture of Vision Encoders for Multimodal Large Language Models, January 2025. arXiv:2501.06986 [cs].

[7] Donghyeon Baek, Youngmin Oh, and Bumsub Ham. Exploiting a Joint Embedding Space for Generalized Zero-Shot Semantic Segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 9536–9545, 2021.

[8] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[9] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2024.

[10] Yichen Bai, Zongbo Han, Bing Cao, Xiaoheng Jiang, Qinghua Hu, and Changqing Zhang. ID-like Prompt Learning for Few-Shot Out-of-Distribution Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17480–17489, 2024.

[11] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – Mining Discriminative Components with Random Forests. In *European Conference on Computer Vision*, pages 446–461, 2014.

[12] Maxime Bucher, Tuan-Hung VU, Matthieu Cord, and Patrick Pérez. Zero-Shot Semantic Segmentation. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[13] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11618–11628, Seattle, WA, USA, June 2020. IEEE.

[14] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and Stuff Classes in Context. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1209–1218. IEEE Computer Society, 2018.

[15] Zhaowei Cai, Gukyeong Kwon, Avinash Ravichandran, Erhan Bas, Zhuowen Tu, Rahul Bhotika, and Stefano Soatto. X-DETR: A Versatile Architecture for Instance-wise Vision-Language Tasks, April 2022. arXiv:2204.05626 [cs].

[16] Chentao Cao, Zhun Zhong, Zhanke Zhou, Yang Liu, Tongliang Liu, and Bo Han. Envisioning Outlier Exposure by Large Language Models for Out-of-Distribution Detection. In *ICML*, June 2024.

[17] David M. Chan, Austin Myers, Sudheendra Vijayanarasimhan, David A. Ross, and John Canny. IC^3: Image Captioning by Committee Consensus, February 2023. arXiv:2302.01328 [cs].

[18] Jun Chen, Deyao Zhu, Guocheng Qian, Bernard Ghanem, Zhicheng Yan, Chenchen Zhu, Fanyi Xiao, Sean Chang Culatana, and Mohamed Elhoseiny. Exploring Open-Vocabulary Semantic Segmentation from CLIP Vision Encoder Distillation Only. In *IEEE/CVF International Conference on Computer Vision*, pages 699–710, 2023.

[19] Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An Image is Worth 1/2 Tokens After Layer 2: Plug-and-Play Inference Acceleration for Large Vision-Language Models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, volume 15139, pages 19–35, Cham, 2024. Springer Nature Switzerland. Series Title: Lecture Notes in Computer Science.

[20] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. CLIP2Scene: Towards label-efficient 3D scene understanding by CLIP. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 7020–7030, June 2023.

[21] Shaoxiang Chen, Zequn Jie, and Lin Ma. LLaVA-MoLE: Sparse Mixture of LoRA Experts for Mitigating Data Conflicts in Instruction Finetuning MLLMs, January 2024. arXiv:2401.16160 [cs].

[22] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1597–1607. PMLR, November 2020. ISSN: 2640-3498.

[23] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. VoxelNeXt: Fully Sparse VoxelNet for 3D Object Detection and Tracking, March 2023. arXiv:2303.11301 [cs].

[24] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and

Jifeng Dai. Intern VL: Scaling up Vision Foundation Models and Aligning for Generic Visual-Linguistic Tasks. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24185–24198, Seattle, WA, USA, June 2024. IEEE. TLDR: A large-scale vision-language foun-dation model (Intern VL), which scales up the vision foun-dation model to 6 billion parameters and progressively aligns it with the LLM, using web-scale image-text data from various sources is designed.

[25] Zhimin Chen, Longlong Jing, Yingwei Li, and Bing Li. Bridging the domain gap: Self-supervised 3D scene understanding with foundation models. In *Thirty-seventh conference on neural information processing systems*, 2023.

[26] Ran Cheng, Ryan Razani, Ehsan Taghavi, Enxu Li, and Bingbing Liu. (af)2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12542–12551, 2021.

[27] Ali Cheraghian, Shafin Rahman, Dylan Campbell, and Lars Petersson. Transductive Zero-Shot Learning for 3D Point Cloud Classification. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 912–922, Snowmass Village, CO, USA, March 2020. IEEE.

[28] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023.

[29] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3070–3079, Long Beach, CA, USA, June 2019. IEEE.

[30] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. NUS-WIDE: a real-world web image database from National University of Singapore. In *CIVR*, pages 1–9, 2009.

[31] Sanghyuk Chun, Wonjae Kim, Song Park, Minsuk Chang, and Seong Joon Oh. ECCV Caption: Correcting False Negatives by Collecting Machine-and-Human-verified Image-Caption Associations for MS-COCO. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, volume 13668, pages 1–19, Cham, 2022. Springer Nature Switzerland. Series Title: Lecture Notes in Computer Science.

[32] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and and A. Vedaldi. Describing Textures in the Wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[33] Eric Nuertey Coleman, Luigi Quarantiello, Ziyue Liu, Qinwen Yang, Samrat Mukherjee, Julio Hurtado, and Vincenzo Lomonaco. Parameter-efficient continual fine-tuning: A survey, 2025.

[34] NLP Connect. vit-gpt2-image-captioning, 2022.

[35] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds for autonomous driving, 2020.

[36] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2443, Honolulu, HI, July 2017. IEEE.

[37] Damai Dai, Chengqi Deng, Chenggang Zhao, R.x. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y.k. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1280–1297, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[38] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. November 2023.

[39] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[40] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, May 2019. arXiv: 1810.04805.

[41] Terrance DeVries and Graham W. Taylor. Learning Confidence for Out-of-Distribution Detection in Neural Networks, February 2018. arXiv:1802.04865 [stat].

[42] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Lowis3D: Language-driven open-world instance-level 3D scene understanding, 2023. arXiv: 2308.00353 [cs.CV].

[43] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. PLA: Language-Driven Open-Vocabulary 3D Scene Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7010–7019, 2023.

[44] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. MaskCLIP: Masked Self-Distillation Advances Contrastive Language-Image Pretraining. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10995–11005, Vancouver, BC, Canada, June 2023. IEEE.

[45] Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Wei Shen, Limao Xiong, Yuhao Zhou, Xiao Wang, Zhiheng Xi, Xiaoran Fan, Shiliang Pu, Jiang Zhu, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. LoRAMoE: Alleviating World Knowledge Forgetting in Large Language Models via MoE-Style Plugin. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1932–1945, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[46] Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR, 2022.

[47] Xuefeng Du, Yiyou Sun, Jerry Zhu, and Yixuan Li. Dream the Impossible: Outlier Imagination with Diffusion Models. In *Advances in Neural Information Processing Systems*, volume 36, pages 60878–60901, December 2023.

[48] 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, October 2021. arXiv:2106.09685 [cs].

[49] Sepideh Esmaeilpour, Bing Liu, Eric Robertson, and Lei Shu. Zero-Shot Out-of-Distribution Detection Based on the Pre-trained Model CLIP. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6568–6576, June 2022. arXiv:2109.02748 [cs].

[50] Andre Esteva and Eric Topol. Can skin cancer diagnosis be transformed by ai? *The Lancet*, 394(10211):1795, Nov 2019.

[51] Xiaoran Fan, Tao Ji, Changhao Jiang, Shuo Li, Senjie Jin, Sirui Song, Junke Wang, Boyang Hong, Lu Chen, Guodong Zheng, Ming Zhang, Huangcaishuang, Rui Zheng, Zhiheng Xi, Yuhao Zhou, Shihan Dou, Junjie Ye, Hang Yan, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Poly-Visual-Expert Vision-Language Models. August 2024.

[52] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, page 105171, 2024.

[53] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[54] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the Limits of Out-of-Distribution Detection. In *Advances in Neural Information Processing Systems*, volume 34, pages 7068–7081. Curran Associates, Inc., 2021.

[55] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models, March 2024. arXiv:2306.13394 [cs].

[56] Hao Fu, Naman Patel, Prashanth Krishnamurthy, and Farshad Khorrami. CLIPScope: Enhancing Zero-Shot OOD Detection with Bayesian Scoring. pages 5346–5355, 2025.

[57] Zijian Gao, Jingyu Liu, Weiqi Sun, Sheng Chen, Dedan Chang, and Lili Zhao. CLIP2TV: Align, Match and Distill for Video-Text Retrieval, July 2022. arXiv:2111.05610 [cs].

[58] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling Open-Vocabulary Image Segmentation with Image-Level Labels, July 2022. arXiv:2112.12143 [cs].

[59] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One Embedding Space To Bind Them All, May 2023. arXiv:2305.05665 [cs].

[60] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering.

[61] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9224–9232, Salt Lake City, UT, USA, June 2018. IEEE.

[62] Joris Guérin, Kevin Delmas, Raul Ferreira, and Jérémie Guiochet. Out-of-distribution detection is not all you need. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023.

[63] Tao Guo, Song Guo, and Junxiao Wang. pFedPrompt: Learning Personalized Prompt for Vision-Language Models in Federated Learning. In *Proceedings of the ACM Web Conference 2023*, WWW '23, pages 1364–1374, New York, NY, USA, April 2023. Association for Computing Machinery.

[64] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, and Pheng-Ann Heng. Point-Bind & Point-LLM: Aligning Point Cloud with Multi-modality for 3D Understanding, Generation, and Instruction Following, September 2023. arXiv:2309.00615 [cs].

[65] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz Grand Challenge: Answering Visual Questions from Blind People. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3608–3617, Salt Lake City, UT, USA, June 2018. IEEE.

[66] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, Las Vegas, NV, 2016. IEEE Computer Society.

[67] Qingdong He, Jinlong Peng, Zhengkai Jiang, Kai Wu, Xiaozhong Ji, Jiangning Zhang, Yabiao Wang, Chengjie Wang, Mingang Chen, and Yunsheng Wu. UniM-OV3D: Uni-Modality Open-Vocabulary 3D Scene Understanding with Fine-Grained Feature Representation, January 2024. arXiv:2401.11395 [cs].

[68] Shuting He, Xudong Jiang, Wei Jiang, and Henghui Ding. Prototype Adaption and Projection for Few- and Zero-Shot 3D Point Cloud Semantic Segmentation. *IEEE Transactions on Image Processing*, 32:3199–3211, 2023. Conference Name: IEEE Transactions on Image Processing.

[69] Yong He, Hongshan Yu, Xiaoyan Liu, Zhengeng Yang, Wei Sun, Saeed Anwar, and Ajmal Mian. Deep learning based 3d segmentation: A survey, 2024.

[70] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *IEEE conference on computer vision and pattern recognition, CVPR 2015, boston, MA, USA, june 7-12, 2015*, pages 961–970. IEEE Computer Society, 2015. tex.bibsource: dblp computer science bibliography, https://dblp.org tex.timestamp: Fri, 24 Mar 2023 00:02:59 +0100 TLDR: This paper introduces ActivityNet, a new large-scale video benchmark for human activity understanding that aims at covering a wide range of complex human activities that are of interest to people in their daily living.

[71] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why ReLU Networks Yield High-Confidence Predictions Far Away From the Training Data and How to Mitigate the Problem. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 41–50, Long Beach, CA, USA, June 2019. IEEE.

[72] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. February 2017.

[73] Dan Hendrycks and Kevin Gimpel. Gaussian Error Linear Units (GELUs), June 2023. arXiv:1606.08415 [cs].

[74] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep Anomaly Detection with Outlier Exposure. September 2018.

[75] Georg Hess, Adam Tonderski, Christoffer Petersson, Kalle Åström, and Lennart Svensson. LidarCLIP or: How I learned to talk to point clouds. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision (WACV)*, pages 7438–7447, January 2024.

[76] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A Comprehensive Survey of Deep Learning for Image Captioning. *ACM Computing Surveys*, 51(6), February 2019. Place: New York, NY, USA Publisher: Association for Computing Machinery.

[77] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11105–11114, 2020.

[78] Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Matryoshka Query Transformer for Large Vision-Language Models. November 2024.

[79] Rui Huang, Andrew Geng, and Yixuan Li. On the Importance of Gradients for Detecting Distributional Shifts in the Wild. In *Advances in Neural Information Processing Systems*, volume 34, pages 677–689. Curran Associates, Inc., 2021.

[80] Rui Huang and Yixuan Li. MOS: Towards Scaling Out-of-distribution Detection for Large Semantic Space. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8706–8715, Nashville, TN, USA, June 2021. IEEE.

[81] Siteng Huang, Biao Gong, Yulin Pan, Jianwen Jiang, Yiliang Lv, Yuyuan Li, and Donglin Wang. VoP: Text-Video Co-Operative Prompt Tuning for Cross-Modal Retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6565–6574, 2023.

[82] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson W.H. 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 (ICCV)*, pages 22157–22167, October 2023.

[83] Wenxuan Huang, Zijie Zhai, Yunhang Shen, Shaosheng Cao, Fei Zhao, Xiangfeng Xu, Zheyu Ye, and Shaohui Lin. Dynamic-LLaVA: Efficient Multimodal Large Language Models via Dynamic Vision-language Context Sparsification. October 2024.

[84] Drew A. Hudson and Christopher D. Manning. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6693–6702, Long Beach, CA, USA, June 2019. IEEE.

[85] Mark J. Huiskes and Michael S. Lew. The MIR Flickr Retrieval Evaluation. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, pages 39–43, New York, NY, USA, 2008. Association for Computing Machinery.

[86] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[87] Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *J Pathol Inform*, 7:29, July 2016.

[88] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. ISSN: 2640-3498.

[89] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual Prompt Tuning, July 2022. arXiv:2203.12119 [cs].

[90] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[91] Wenyu Jiang, Hao Cheng, MingCai Chen, Chongjun Wang, and Hongxin Wei. DOS: Diverse Outlier Sampling for Out-of-Distribution Detection. October 2023.

[92] Xue Jiang, Feng Liu, Zhen Fang, Hong Chen, Tongliang Liu, Feng Zheng, and Bo Han. Negative label guided OOD detection with pretrained vision-language models. In *ICLR*, 2024.

[93] K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards Open World Object Detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5826–5836, Nashville, TN, USA, June 2021. IEEE.

[94] Oğuzhan Fatih Kar, Alessio Tonioni, Petra Poklukar, Achin Kulshrestha, Amir Zamir, and Federico Tombari. BRAVE: Broadening the Visual Encoding of Vision-Language Models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 113–132, Cham, 2024. Springer Nature Switzerland.

[95] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A Diagram is Worth a Dozen Images. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 235–251, Cham, 2016. Springer International Publishing.

[96] Omar Khattab and Matei Zaharia. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT, June 2020. arXiv:2004.12832 [cs].

[97] Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating Prompts: Foundational Model Adaptation without Forgetting. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15144–15154, Paris, France, October 2023. IEEE. TLDR: This work

introduces a self-regularization framework for prompting called PromptSRC (Prompting with Self-regulating Constraints), the first regularization framework for prompt learning that avoids overfitting by jointly attending to pre-trained model features, the training trajectory during prompting, and the textual diversity.

[98] Tung Kieu, Bin Yang, and Christian S. Jensen. Outlier Detection for Multidimensional Time Series Using Deep Neural Networks. In *IEEE International Conference on Mobile Data Management*, pages 125–134, Aalborg, Denmark, 2018. IEEE.

[99] Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5583–5594. PMLR, July 2021. ISSN: 2640-3498.

[100] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment Anything, April 2023. arXiv:2304.02643 [cs].

[101] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, pages 1097–1105, 2012.

[102] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical Transformer for LiDAR-Based 3D Recognition. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17545–17555, Vancouver, BC, Canada, June 2023. IEEE.

[103] Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot Parsing as Pretraining for Visual Language Understanding. In *Proceedings of the 40th International Conference on Machine Learning*, pages 18893–18912. PMLR, July 2023. ISSN: 2640-3498.

[104] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[105] Brian Lester, Rami Al-Rfou, and Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning, September 2021. arXiv:2104.08691 [cs].

[106] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In

Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, 2020. Association for Computational Linguistics.

[107] Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. SEED-Bench: Benchmarking Multimodal Large Language Models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13299–13308, Seattle, WA, USA, June 2024. IEEE.

[108] Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven Semantic Segmentation. In *International Conference on Learning Representations*, February 2022.

[109] Fei-Fei Li, Marco Andreeto, Marc'Aurelio Ranzato, and Pietro Perona. Caltech 101, April 2022.

[110] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. LLaVA-NeXT-Interleave: Tackling Multi-image, Video, and 3D in Large Multimodal Models, July 2024. arXiv:2407.07895 TLDR: LLaVA-NeXT-Interleave is introduced, which simultaneously tackles Multi-image, Multi-frame (video), Multi-view (3D), and Multi-patch (single-image) scenarios in LMMs and exhibits several emerging capabilities, e.g., transferring tasks across different settings and modalities.

[111] Jiale Li, Hang Dai, Hao Han, and Yong Ding. Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21694–21704, 2023.

[112] Jingyao Li, Pengguang Chen, Shaozuo Yu, Shu Liu, and Jiaya Jia. MOODv2: Masked Image Modeling for Out-of-Distribution Detection, January 2024. arXiv:2401.02611 [cs].

[113] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19730–19742. PMLR, 2023.

[114] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, February 2022. arXiv:2201.12086 [cs].

[115] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. VideoChat: Chat-Centric Video Understanding, January 2024.

arXiv:2305.06355 TLDR: An end-to-end chat-centric video understanding system that integrates video foundation models and large language models via a learnable neural interface, excelling in spatiotemporal reasoning, event localization, and causal relationship inference is initiated.

[116] Tianqi Li, Guansong Pang, Xiao Bai, Wenjun Miao, and Jin Zheng. Learning Transferable Negative Prompts for Out-of-Distribution Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17584–17594, 2024.

[117] Wei Li, Linchao Zhu, Longyin Wen, and Yi Yang. DeCap: Decoding CLIP Latents for Zero-Shot Captioning via Text-Only Training. September 2022.

[118] Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jie Qin, Jianke Zhu, and Lei Zhang. TokenPacker: Efficient Visual Projector for Multimodal LLM, August 2024. arXiv:2407.02392 [cs].

[119] Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online, August 2021. Association for Computational Linguistics. TLDR: Prefix-tuning is proposed, a lightweight alternative to fine- Tuning for natural language generation tasks, which keeps language model parameters frozen and instead optimizes a sequence of continuous task-specific vectors, which is called the prefix.

[120] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. *arXiv:2004.06165 [cs]*, July 2020. arXiv: 2004.06165.

[121] Yan Li, Ethan X.Fang, Huan Xu, and Tuo Zhao. Implicit Bias of Gradient Descent based Adversarial Training on Separable Data. December 2019.

[122] Yanwei Li, Chengyao Wang, and Jiaya Jia. LLaMA-VID: An Image is Worth 2 Tokens in Large Language Models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, volume 15104, pages 323–340, Cham, 2024. Springer Nature Switzerland. Series Title: Lecture Notes in Computer Science.

[123] Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-Gemini: Mining the Potential of Multi-modality Vision

Language Models, March 2024. arXiv:2403.18814 TLDR: This work introduces Mini-Gemini, a simple and effective framework enhancing multi-modality Vision Language Models (VLMs) and empowers current frameworks with image understanding, reasoning, and generation simultaneously, and constructs a high-quality dataset that promotes precise image comprehension and reasoning-based generation.

[124] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating Object Hallucination in Large Vision-Language Models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, Singapore, December 2023. Association for Computational Linguistics.

[125] Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image Resolution and Text Label are Important Things for Large Multi-Modal Models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26753–26763, Seattle, WA, USA, June 2024. IEEE. TLDR: Monkey is introduced to enhance LMM capabilities and surpasses existing LMMs in many tasks like Image Captioning and various Visual Question Answering formats, and in qualitative tests focused on dense text question answering, Monkey has exhibited encouraging results.

[126] Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, pages 4208–4215, Stockholm, Sweden, July 2018. AAAI Press.

[127] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks, August 2020. arXiv:1706.02690 [cs].

[128] Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in neural information processing systems*, 2022.

[129] Bin Lin, Zhenyu Tang, Yang Ye, Jiaxi Cui, Bin Zhu, Peng Jin, Jinfa Huang, Junwu Zhang, Yatian Pang, Munan Ning, and Li Yuan. MoE-LLaVA: Mixture of Experts for Large Vision-Language Models, July 2024. arXiv:2401.15947 TLDR: The MoE-LLaVA is presented, a MoE-based sparse LVLM architecture, which uniquely activates only the top-k experts through routers during deployment, keeping the remaining experts inactive, and establishes a baseline for sparse LVLMs.

[130] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. VILA: On Pre-training for Visual Language Models, May 2024. arXiv:2312.07533 [cs].

[131] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

[132] Dongxu Liu, Bing Xu, Yinzhuo Chen, Bufan Xu, Wenpeng Lu, Muyun Yang, and Tiejun Zhao. Pmol: Parameter efficient moe for preference mixing of llm alignment. *arXiv preprint arXiv:2411.01245*, 2024.

[133] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved Baselines with Visual Instruction Tuning. arXiv, October 2023. arXiv:2310.03744 [cs].

[134] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024.

[135] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual Instruction Tuning. November 2023.

[136] Kai Liu, Zhihang Fu, Chao Chen, Sheng Jin, Ze Chen, Mingyuan Tao, Rongxin Jiang, and Jieping Ye. Category-Extensible Out-of-Distribution Detection via Hierarchical Context Descriptions. In *Advances in Neural Information Processing Systems*, volume 36, pages 33241–33261, December 2023.

[137] 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, June 2023. arXiv:2305.10764 [cs].

[138] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based Out-of-distribution Detection. In *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc., 2020.

[139] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is Your Multi-modal Model an All-Around Player? In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, volume 15064, pages 216–233. Springer Nature Switzerland, Cham, 2025. Series Title: Lecture Notes in Computer Science.

[140] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: on the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67(12), December 2024. Publisher: Springer Science and Business Media LLC.

[141] Yuyuan Liu, Choubo Ding, Yu Tian, Guansong Pang, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Residual Pattern Learning for Pixel-Wise Out-of-Distribution Detection in Semantic Segmentation. pages 1151–1161, 2023.

[142] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[143] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. In *Advances in Neural Information Processing Systems*, 2019.

[144] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 11966–11976. IEEE, 2022.

[145] AI @ Meta1 Llama Team. The llama 3 herd of models, 2024.

[146] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. DeepSeek-VL: Towards Real-World Vision-Language Understanding, March 2024. arXiv:2403.05525 [cs].

[147] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[148] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th conference on neural information processing systems (NeurIPS)*, 2022.

[149] Gen Luo, Yiyi Zhou, Yuxin Zhang, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Feast Your Eyes: Mixture-of-Resolution Adaptation for Multimodal Large Language Models, March 2024. arXiv:2403.03003 TLDR: This work proposes a novel and efficient method

for MLLMs, termed Mixture-of-Resolution Adaptation (MRA), which adopts two visual pathways for images with different resolutions, where high-resolution visual information is embedded into the low-resolution pathway via the novel mixture-of-resolution adapters (MR-Adapters).

[150] Kaifeng Lyu and Jian Li. Gradient Descent Maximizes the Margin of Homogeneous Neural Networks. In *International Conference on Learning Representations*, September 2019.

[151] Akshayaa Magesh, Venugopal V. Veeravalli, Anirban Roy, and Susmit Jha. Principled out-of-distribution detection via multiple testing. *Journal of Machine Learning Research*, 24(378):1–35, 2023.

[152] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Robotics: Science and Systems (RSS)*, 2017.

[153] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-Grained Visual Classification of Aircraft. Technical report, 2013. _eprint: 1306.5151.

[154] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[155] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A Dataset for VQA on Document Images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2199–2208, Waikoloa, HI, USA, January 2021. IEEE.

[156] Björn Michele, Alexandre Boulch, Gilles Puy, Maxime Bucher, and Renaud Marlet. Generative Zero-Shot Learning for Semantic Segmentation of 3D Point Clouds. In *2021 International Conference on 3D Vision (3DV)*, pages 992–1002, December 2021. ISSN: 2475-7888.

[157] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. arXiv: 1301.3781 [cs.CL].

[158] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet ++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4213–4220, 2019.

[159] George A. Miller. WordNet: A lexical database for english. *Communications of The Acm*, 38(11):39–41, 1995. tex.bibsource: dblp computer science bibliography, https://dblp.org tex.timestamp: Fri, 24 Mar 2023 16:31:07 +0100.

[160] Yifei Ming, Ziyang Cai, Jiuxiang Gu, Yiyou Sun, Wei Li, and Yixuan Li. Delving into out-of-distribution detection with vision-language representations. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in neural information processing systems*, 2022.

[161] Atsuyuki Miyai, Jingkang Yang, Jingyang Zhang, Yifei Ming, Yueqian Lin, Qing Yu, Go Irie, Shafiq Joty, Yixuan Li, Hai Li, Ziwei Liu, Toshihiko Yamasaki, and Kiyoharu Aizawa. Generalized Out-of-Distribution Detection and Beyond in Vision Language Model Era: A Survey, July 2024. arXiv:2407.21794 [cs].

[162] Atsuyuki Miyai, Qing Yu, Go Irie, and Kiyoharu Aizawa. LoCoOp: Few-Shot Out-of-Distribution Detection via Prompt Learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 76298–76310, December 2023.

[163] Harikrishna Narasimhan, Aditya Krishna Menon, Wittawat Jitkrittum, and Sanjiv Kumar. Plugin estimators for selective classification with out-of-distribution detection. October 2023.

[164] Jun Nie, Yonggang Zhang, Zhen Fang, Tongliang Liu, Bo Han, and Xinmei Tian. Out-of-Distribution Detection with Negative Prompts. In *ICLR*, 2024.

[165] Jie Oin, Jie Wu, Pengxiang Yan, Ming Li, Ren Yuxi, Xuefeng Xiao, Yitong Wang, Rui Wang, Shilei Wen, Xin Pan, and Xingang Wang. FreeSeg: Unified, Universal and Open-Vocabulary Image Segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19446–19455, Vancouver, BC, Canada, June 2023. IEEE.

[166] OpenAI. GPT-4 Technical Report, March 2023. arXiv:2303.08774 [cs].

[167] Jaewoo Park, Yoon Gyo Jung, and Andrew Beng Jin Teoh. Nearest Neighbor Guidance for Out-of-Distribution Detection. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1686–1695, Paris, France, October 2023. IEEE. TLDR: This work proposes a method called Nearest Neighbor Guidance (NNGuide) that guides the classifier-based score to respect the boundary geometry of the data manifold, reducing the overconfidence of OOD samples while preserving the fine-grained capability of the classifier-based score.

[168] Sangha Park, Jisoo Mok, Dahuin Jung, Saehyung Lee, and Sungroh Yoon. On the Powerfulness of Textual Outlier Exposure for Visual OoD Detection. In *Advances in Neural Information Processing Systems*, volume 36, pages 51675–51687, December 2023.

[169] Youngmin Park, Vincent Lepetit, and Woontack Woo. Multiple 3d object tracking for augmented reality. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 117–120, 2008.

[170] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and Dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[171] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D Scene Understanding With Open Vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–824, 2023.

[172] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding Multimodal Large Language Models to the World, July 2023. arXiv:2306.14824 [cs].

[173] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv, April 2017. arXiv:1612.00593 [cs].

[174] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. ISSN: 2640-3498.

[175] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE Press, 2020.

[176] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. DenseCLIP: Language-Guided Dense Prediction with Context-Aware Prompting. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18061–18070, New Orleans, LA, USA, June 2022. IEEE.

[177] Zhu Rui and Zhao Yongjia. Real-time plane segmentation for scene understanding in robot navigation. In *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2017.

[178] Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive Token Reduction for Efficient Large Multimodal Models, May 2024. arXiv:2403.15388 [cs].

[179] Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. February 2017.

[180] Min Shi, Fuxiao Liu, Shihao Wang, Shijia Liao, Subhashree Radhakrishnan, De-An Huang, Hongxu Yin, Karan Sapra, Yaser Yacoob, Humphrey Shi, Bryan Catanzaro, Andrew Tao, Jan Kautz, Zhiding Yu, and Guilin Liu. Eagle: Exploring The Design Space for Multimodal LLMs with Mixture of Encoders, August 2024. arXiv:2408.15998 TLDR: This study provides an extensive exploration of the design space for MLLMs using a mixture of vision encoders and resolutions, finding that simply concatenating visual tokens from a set of complementary vision encoders is as effective as more complex mixing architectures or strategies.

[181] Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts, November 2020. arXiv:2010.15980 [cs].

[182] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.

[183] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. FLAVA: A Foundational Language And Vision Alignment Model. *arXiv:2112.04482 [cs]*, December 2021. arXiv: 2112.04482.

[184] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA Models That Can Read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8309–8318, Long Beach, CA, USA, June 2019. IEEE.

[185] Santokh Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. 2015.

[186] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, June 2020.

[187] Yiyou Sun, Chuan Guo, and Yixuan Li. ReAct: Out-of-distribution Detection With Rectified Activations. In *Advances in Neural Information Processing Systems*, volume 34, pages 144–157. Curran Associates, Inc., 2021.

[188] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-Distribution Detection with Deep Nearest Neighbors. In *Proceedings of the 39th International Conference on Machine Learning*, pages 20827–20840. PMLR, June 2022. ISSN: 2640-3498.

[189] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *European Conference on Computer Vision*, 2020.

[190] Leitian Tao, Xuefeng Du, Jerry Zhu, and Yixuan Li. Non-parametric Outlier Synthesis. In *ICLR*, 2023.

[191] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas Guibas. KPConv: Flexible and Deformable Convolution for Point Clouds. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6410–6419, Seoul, Korea (South), October 2019. IEEE.

[192] Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Chengzhong Xu. HydraLoRA: An Asymmetric LoRA Architecture for Efficient Fine-Tuning, May 2024. arXiv:2404.19245 [cs] TLDR: HydraLoRA, a LoRA framework with an asymmetric structure that eliminates the need for domain expertise is developed, and experiments demonstrate that HydraLoRA outperforms other PEFT approaches, even those that rely on domain knowledge during the training and inference phases.

[193] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024.

[194] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel,

Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open Foundation and Fine-Tuned Chat Models, July 2023. arXiv:2307.09288 [cs].

[195] Michael Tschannen, Basil Mustafa, and Neil Houlsby. CLIPPO: Image-and-Language Understanding from Pixels Only, April 2023. arXiv:2212.08045 [cs].

[196] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist Species Classification and Detection Dataset. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, Salt Lake City, UT, June 2018. IEEE.

[197] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems*, pages 5998–6008, Long Beach, CA, 2017.

[198] Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Will we run out of data? limits of llm scaling based on human-generated data, 2024.

[199] Catherine Wah, Steve Branson, Peter Welinder, and Serge Belongie Pietro Perona. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[200] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. ViM: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4921–4930, June 2022.

[201] Hualiang Wang, Yi Li, Huifeng Yao, and Xiaomeng Li. CLIPN for Zero-Shot OOD Detection: Teaching CLIP to Say No. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1802–1812, Paris, France, October 2023. IEEE.

[202] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. OFA: Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. ISSN: 2640-3498.

[203] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9929–9939. PMLR, 13–18 Jul 2020.

[204] Yimu Wang, Mozhgan Nasr Azadani, Sean Sedwards, and Krzysztof Czarnecki. HAWAII: Hierarchical visual knowledge transfer for efficient vision-language models. In *NeurIPS*, 2025. arXiv: 2506.19072 [cs.CV].

[205] Yimu Wang, Mozhgan Nasr Azadani, Sean Sedwards, and Krzysztof Czarnecki. LEO-MINI: An efficient multimodal large language model using conditional token reduction and mixture of multi-modal experts. In *EMNLP*, 2025. arXiv: 2504.04653 [cs.CV].

[206] Yimu Wang and Krzysztof Czarnecki. AiDe: Improving 3D open-vocabulary semantic segmentation by aligned vision-language learning. In *2025 IEEE/CVF winter conference on applications of computer vision (WACV)*, pages 2674–2685, 2025.

[207] Yimu Wang, Evelien Riddell, Adrian Chow, Sean Sedwards, and Krzysztof Czarnecki. Mitigating the modality gap: Few-shot out-of-distribution detection with multi-modal prototypes and image bias estimation, 2025. arXiv: 2502.00662 [cs.CV].

[208] Yimu Wang and Peng Shi. Video-Text Retrieval by Supervised Multi-Space Multi-Grained Alignment, February 2023. arXiv:2302.09473 [cs].

[209] Yuanbin Wang, Shaofei Huang, Yulu Gao, Zhen Wang, Rui Wang, Kehua Sheng, Bo Zhang, and Si Liu. Transferring CLIP's knowledge into zero-shot point cloud semantic segmentation. In *Proceedings of the 31st ACM international conference on multimedia*, MM '23, pages 3745–3754, New York, NY, USA, 2023. Association for Computing Machinery.

[210] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating Neural Network Overconfidence with Logit Normalization, 2022. Version Number: 2.

[211] Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. Stop Looking for Important Tokens in Multimodal Language Models: Duplication Matters More, February 2025. arXiv:2502.11494 [cs].

[212] Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R. Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, Taylan Cemgil, S. M. Ali Eslami, and Olaf Ronneberger. Contrastive training for improved out-of-distribution detection, 2020.

[213] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4376–4382, 2019.

[214] Jialin Wu, Xia Hu, Yaqing Wang, Bo Pang, and Radu Soricut. Omni-SMoLA: Boosting generalist multimodal models with soft mixture of low-rank experts. In *IEEE/CVF conference on computer vision and pattern recognition, CVPR 2024, seattle, WA, USA, june 16-22, 2024*, pages 14205–14215. IEEE, 2024. tex.bibsource: dblp computer science bibliography, https://dblp.org tex.timestamp: Tue, 17 Dec 2024 09:07:28 +0100.

[215] Jianzong Wu, Xiangtai Li, Henghui Ding, Xia Li, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Betrayed by Captions: Joint Caption Grounding and Generation for Open Vocabulary Instance Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21938–21948, 2023.

[216] Xun Wu, Shaohan Huang, and Furu Wei. Mixture of LoRA Experts. October 2023.

[217] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic Projection Network for Zero- and Few-Label Semantic Segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8248–8257, Long Beach, CA, USA, June 2019. IEEE.

[218] Aoran Xiao, Xiaofei Yang, Shijian Lu, Dayan Guan, and Jiaxing Huang. Fps-net: A convolutional fusion network for large-scale lidar point cloud segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 176:237–249, 2021.

[219] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, San Francisco, CA, USA, June 2010. IEEE.

[220] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation, 2021.

[221] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5288–5296, Las Vegas, NV, USA, June 2016. IEEE. TLDR: A detailed analysis of MSR-VTT in comparison to a complete set of existing datasets, together with a summarization of different state-of-the-art video-to-text approaches, shows that the hybrid Recurrent Neural Networkbased approach, which combines single-frame and motion representations with soft-attention pooling strategy, yields the best generalization capability on this dataset.

[222] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A Simple Baseline for Open-Vocabulary Semantic Segmentation with Pre-trained Vision-Language Model. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, volume 13689, pages 736–753, Cham, 2022. Springer Nature Switzerland. Series Title: Lecture Notes in Computer Science.

[223] Mingyu Xu, Zheng Lian, Bin Liu, and Jianhua Tao. VRA: Variational Rectified Activation for Out-of-distribution Detection. November 2023.

[224] Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. PAConv: Position Adaptive Convolution with Dynamic Kernel Assembling on Point Clouds. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3172–3181, Nashville, TN, USA, June 2021. IEEE.

[225] 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 (CVPR)*, pages 1179–1189, June 2023.

[226] Jihan Yang, Runyu Ding, Zhe Wang, and Xiaojuan Qi. RegionPLC: Regional Point-Language Contrastive Learning for Open-World 3D Scene Understanding, July 2023. arXiv:2304.00962 [cs].

[227] Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, Wenxuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Dan Hendrycks, Yixuan Li, and Ziwei Liu. OpenOOD: Benchmarking Generalized Out-of-Distribution Detection, 2022. Version Number: 1.

[228] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *Int. J. Comput. Vision*, 132(12):5635–5662, June 2024.

[229] Jingkang Yang, Kaiyang Zhou, and Ziwei Liu. Full-spectrum out-of-distribution detection. *Int. J. Comput. Vision*, 131(10):2607–2622, June 2023.

[230] Yuwei Yang, Munawar Hayat, Zhao Jin, Hongyuan Zhu, and Yinjie Lei. Zero-Shot Point Cloud Segmentation by Semantic-Visual Aware Synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11552–11562, Paris, France, October 2023. IEEE.

[231] Hantao Yao, Rui Zhang, and Changsheng Xu. Visual-Language Prompt Tuning With Knowledge-Guided Context Optimization. pages 6757–6767, 2023.

[232] Xubing Ye, Yukang Gan, Xiaoke Huang, Yixiao Ge, Ying Shan, and Yansong Tang. VoCo-LLaMA: Towards Vision Compression with Large Language Models, June 2024. arXiv:2406.12275 [cs].

[233] Geng Yu, Jianing Zhu, Jiangchao Yao, and Bo Han. Self-Calibrated Tuning of Vision-Language Models for Out-of-Distribution Detection. November 2024.

[234] Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A Massive Multi-Discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9556–9567, Seattle, WA, USA, June 2024. IEEE.

[235] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access*, 8:58443–58469, 2020. arXiv: 1906.05113.

[236] Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, and Sara Hooker. Pushing mixture of experts to the limit: Extremely parameter efficient moe for instruction tuning. *arXiv preprint arXiv:2309.05444*, 2023.

[237] John R. Zech, Marcus A. Badgeley, Mingping Liu, Anthony B. Costa, Joseph J. Titano, and Eric K. Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine*, 15(11):e1002683, 2018.

[238] Fanhu Zeng, Zhen Cheng, Fei Zhu, Hongxin Wei, and Xu-Yao Zhang. Local-Prompt: Extensible Local Prompts for Few-Shot Out-of-Distribution Detection. 2025.

[239] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. CLIP2: Contrastive language-image-point pretraining from real-world point cloud data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 15244–15253, June 2023.

[240] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid Loss for Language Image Pre-Training, September 2023. arXiv:2303.15343 [cs].

[241] Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language models, 2023.

[242] Cheng Zhang, Haocheng Wan, Xinyi Shen, and Zizhao Wu. PVT: Point-Voxel Transformer for Point Cloud Learning, May 2022. arXiv:2108.06076 [cs].

[243] Jingyang Zhang, Jingkang Yang, Pengyun Wang, Haoqi Wang, Yueqian Lin, Haoran Zhang, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Yixuan Li, Ziwei Liu, Yiran Chen, and Hai Li. OpenOOD v1.5: Enhanced Benchmark for Out-of-Distribution Detection, June 2023. arXiv:2306.09301 [cs].

[244] Junbo Zhang, Runpei Dong, and Kaisheng Ma. CLIP-FO3D: Learning free open-world 3D scene representations from 2D dense CLIP. In *2023 IEEE/CVF international conference on computer vision workshops (ICCVW)*, pages 2040–2051, 2023.

[245] Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. [CLS] Attention is All You Need for Training-Free Visual Token Pruning: Make VLM Inference Faster, December 2024. arXiv:2412.01818 [cs].

[246] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8542–8552, New Orleans, LA, USA, June 2022. IEEE.

[247] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-Mini: Efficient Image and Video Large Multimodal Models with One Vision Token, January 2025. arXiv:2501.03895 [cs].

[248] Yabin Zhang, Wenjie Zhu, Chenhang He, and Lei Zhang. LAPT: Label-driven Automated Prompt Tuning for OOD Detection with Vision-Language Models. In *ECCV*. arXiv, July 2024. arXiv:2407.08966 [cs].

[249] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[250] Shizhen Zhao, Jiahui Liu, Xin Wen, Haoru Tan, and Xiaojuan Qi. Rethinking Out-of-Distribution Detection in Vision Foundation Models. October 2024.

[251] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, June 2018.

[252] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning Placeholders for Open-Set Recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4399–4408, Nashville, TN, USA, June 2021. IEEE.

[253] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional Prompt Learning for Vision-Language Models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16795–16804, New Orleans, LA, USA, June 2022. IEEE. TLDR: Conditional Context Optimization (CoCoOp), which extends CoOp by further learning a lightweight neural network to generate for each image an input-conditional token (vector), and yields stronger domain generalization performance as well.

[254] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to Prompt for Vision-Language Models. *International Journal of Computer Vision*, 130(9):2337–2348, September 2022. TLDR: Context Optimization (CoOp) is proposed, a simple approach

specifically for adapting CLIP-like vision-language models for downstream image recognition that achieves superb domain generalization performance compared with the zero-shot model using hand-crafted prompts.

[255] Mingyang Zhou, Luowei Zhou, Shuohang Wang, Yu Cheng, Linjie Li, Zhou Yu, and Jingjing Liu. UC2: Universal Cross-lingual Cross-modal Vision-and-Language Pre-training. *CoRR*, abs/2104.00332, 2021. arXiv: 2104.00332.

[256] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. InternVL3: Exploring Advanced Training and Test-Time Recipes for Open-Source Multimodal Models, April 2025. arXiv:2504.10479 [cs].

[257] Muzhi Zhu, Hengtao Li, Hao Chen, Chengxiang Fan, Weian Mao, Chenchen Jing, Yifan Liu, and Chunhua Shen. SegPrompt: Boosting Open-World Segmentation via Category-Level Prompt Learning. In *IEEE/CVF International Conference on Computer Vision*, pages 999–1008, 2023.

[258] 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 (ICCV)*, pages 2639–2650, October 2023.

[259] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Wei Li, Yuexin Ma, Hongsheng Li, Ruigang Yang, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar-based perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 6807–6822, 2022.

[260] Zhuofan Zong, Bingqi Ma, Dazhong Shen, Guanglu Song, Hao Shao, Dongzhi Jiang, Hongsheng Li, and Yu Liu. MoVA: Adapting Mixture of Vision Experts to Multimodal Context, October 2024. arXiv:2404.13046 [cs].