From Drop-off to Recovery: A Mechanistic Analysis of Segmentation in MLLMs

Boyong Wu[1,2], Sanghwan Kim[1,2,3], and Zeynep Akata[1,2,3]

[1] Technical University of Munich, Munich, Germany
[2] Helmholtz Munich, Munich, Germany
[3] Munich Center for Machine Learning (MCML), Munich, Germany
b.wu@tum.de

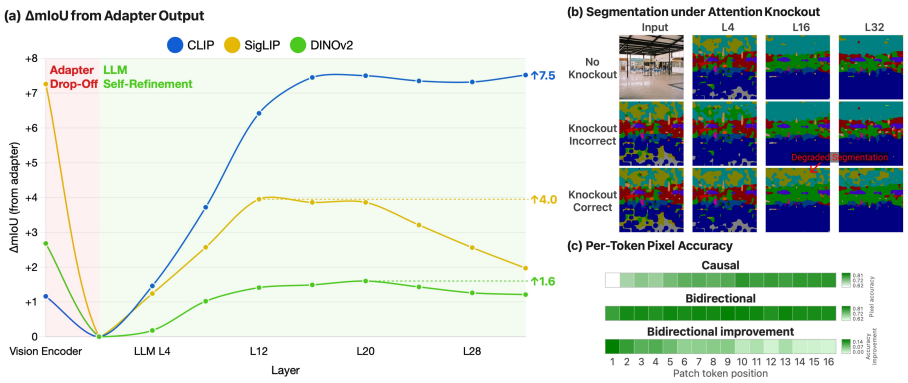

Fig. 1: Overview of main findings. (a) Layerwise linear probing on ADE20K: the adapter introduces a representation drop-off, but LLM layers progressively recover segmentation quality. **(b)** Attention knockout on conflicting class pairs: knocking out attention from correctly classified tokens degrades segmentation, confirming that cross-token self-refinement is driven by semantic anchors. **(c)** Per-token pixel accuracy at an intermediate LLM layer: causal attention starves early position tokens of semantic anchors, while bidirectional attention among image tokens alleviates this bottleneck.

Abstract. Multimodal Large Language Models (MLLMs) are increasingly applied to pixel-level vision tasks, yet their intrinsic capacity for spatial understanding remains poorly understood. We investigate segmentation capacity through a layerwise linear probing evaluation across the entire MLLM pipeline: vision encoder, adapter, and LLM. We further conduct an intervention based attention knockout analysis to test whether cross-token attention progressively refines visual representations, and an evaluation of bidirectional attention among image tokens on spatial consistency. Our analysis reveals that the adapter introduces a segmentation representation drop-off, but LLM layers progressively recover through attention-mediated refinement, where correctly classified tokens steer misclassified neighbors toward the correct label. At early image token positions, this recovery is bounded by causal attention, which bidirectional attention among image tokens alleviates. These findings provide a mechanistic account of how MLLMs process visual information for segmentation, informing the design of future segmentation-capable models.

Keywords: Multimodal Large Language Models · Semantic Segmentation · Mechanistic Interpretability

1 Introduction

Detecting and segmenting content in visual scenes is foundational for applications in robotics, AR/VR, autonomous driving, and medical imaging [9,16,42]. Vision Transformers pretrained with contrastive image-text objectives or self-supervised distillation have emerged as powerful feature extractors: their patch-level embeddings capture rich dense semantics and transfer strongly to segmentation when paired with lightweight decoders or linear probes [3,30,31]. More recently, large-scale pretraining has yielded specialized vision encoders such as SAM [5,18,33] that further push segmentation performance. In parallel, Multimodal Large Language Models (MLLMs) have demonstrated impressive capabilities in multimodal reasoning, instruction following, and language conditioned control [2, 8, 24]. Among MLLMs, adapter style architectures are notable for their simplicity and have demonstrated state-of-the-art performance by combining a pretrained Vision Encoder with a pretrained Large Language Model (LLM) through a learned adapter network, that maps vision embeddings into the LLM's token space [24,25,28]. These capabilities have motivated a growing body of work that adapts MLLMs for pixel-level semantic segmentation tasks, arguing that language supervision can improve interpretability and enable compositional segmentation driven by Referring Expression Segmentation, which allows segmentation through instruction following beyond what traditional segmentation models support [17,19].

While recent works have proposed MLLM-based methods specialized for semantic segmentation [7, 34, 39, 43], it remains unclear whether the MLLM as a whole is actually better at image segmentation than its underlying vision encoder under a controlled probing setup. Recent analyses suggest that, without task-specific tuning, MLLMs may underperform on classical vision tasks and may overlook visual evidence that is already present in their vision backbones [13,14,36,45].

To address this gap in the literature, we propose a systematic, intervention-driven framework to dissect where segmentation competence arises or degrades across the MLLM stack. First, we perform linear probing of frozen embeddings at the vision encoder, the adapter, and every intermediate LLM layer (Sec. 3). This reveals a *representation drop-off* at the adapter: projecting into the LLM embedding space trades fine-grained spatial fidelity for cross-modal alignment, degrading token-level separability. However, downstream LLM layers progressively *self-refine* these visual representations and gradually recover segmentation quality.

This recovery raises the question of whether it is actively driven by cross-token attention or is merely a byproduct of residual connections and normalization. To further study these mechanisms, we design attention knockout interventions on conflicting class pairs (e.g., ceiling vs. sky), which suppress specific

attention pathways while leaving all other operations unchanged (Sec. 4). We find that correctly classified tokens act as semantic anchors whose attention signals pull misclassified neighbors toward the correct label, indicating that cross-token attention drives the observed self-refinement.

Finally, under standard causal attention, early image tokens can only attend to preceding tokens in sequence order and thus lack global spatial context, limiting self-refinement at those positions. We ablate this context starvation by introducing bidirectional attention exclusively among image tokens and by varying the vision encoder (Sec. 5). Bidirectional attention eliminates the positional bias at early patches, and the degree of recovery depends on the compatibility between the vision encoder's representation space and the LLM's embedding space, with text-aligned encoders (e.g., CLIP, SigLIP) benefiting more than vision-only ones (e.g., DINOv2).

Our contribution can be summarized as follows (Fig. 1):

- We reveal a representation drop-off at the adapter and progressive self-refinement across LLM layers by performing layerwise linear probing that analyzes segmentation competence across the MLLM stack.
- Our attention knockout experiments show that self-refinement is driven by cross-token attention, with correctly classified tokens acting as semantic anchors that guide their misclassified neighbors.
- We further find that applying bidirectional attention alleviates context starvation at early image tokens and that text-aligned encoders benefit more from LLM-layer refinement than vision-only encoders.

Across experiments, we observe that LLM layers provide structure- and constraint-aware refinement that improves local consistency and resolves certain class conflicts, but they do not recover fine-grained spatial details absent from the encoder. These findings clarify when and where MLLMs aid semantic segmentation and where they fall short, offering practical guidance for the design of segmentation-capable MLLMs.

2 Related Work

Vision Encoders for Segmentation. Vision Transformers pretrained at scale with supervised, contrastive, or self-supervised objectives have become powerful feature extractors for dense prediction tasks [6,11,31,33,41]. Notably, patch-level features from these models already transfer strongly to segmentation and other dense tasks when decoded with lightweight heads or linear probes [3,30], establishing that rich spatial information is present in the encoder representations themselves. Our work builds on this probing paradigm but extends it beyond the vision encoder in isolation: we probe representations at every stage of the full MLLM pipeline, from the encoder through the adapter and into each LLM layer, to understand whether the downstream components preserve, degrade, or enhance the spatial information already present in the encoder features.

MLLMs for Segmentation. Adapter-style MLLMs connect a pretrained vision encoder to a pretrained LLM through a learned projection [25,28,40], enabling multimodal reasoning without training either component from scratch. A growing line of work adapts this architecture for segmentation, leveraging instruction following and language-conditioned control to produce pixel-level outputs. A common strategy introduces special tokens that bridge language reasoning with mask prediction. LISA [19] pioneered this by connecting an MLLM to SAM via a learned `[SEG]` token, enabling reasoning segmentation from complex language instructions. GSVA [39] extended this to multi-target and empty-target cases with multiple `[SEG]` and `[REJ]` tokens. PixelLM [34] takes a SAM-free approach, generating multi-scale segment tokens decoded by a lightweight head. Other recent notable efforts include GLaMM [32], SAM4MLLM [7], OMG-LLaVA [43], and others [35,38,44,46,48], each proposing architectural variants for grounding or segmentation. More recently, STAMP [26] predicts segmentation masks non-autoregressively through simultaneous textual mask prediction, granting bidirectional attention among mask tokens while preserving causal attention for text. Our works are complementary, STAMP introduces this design empirically, while our Sec.5 analysis reaches the same architectural conclusion from probing MLLM internals. While these works report competitive segmentation results, they focus on system-level performance and do not investigate where within the MLLM stack segmentation competence actually resides. Our work complements this line of research by providing a diagnostic analysis of the representations these architectures produce.

Diagnostic Analyses of MLLM Representations. Recent analyses caution that MLLMs may underperform on standard vision tasks without task-specific finetuning. Zhang *et al.* [45] examine the performance of MLLMs on classification tasks and find that they can overlook visual evidence encoded by their own vision backbone. Other works in this area have shown that MLLMs substantially underperform their vision encoders on vision-centric tasks such as depth estimation and correspondence, identifying the LLM as the primary bottleneck. Their analysis probes representations across the VLM but focuses on general perceptual tasks and does not perform causal interventions [13]. Conversely, Li *et al.* [21] show that generative MLLMs can extract visual information more effectively than CLIP from the same frozen encoder, suggesting a more nuanced picture. Liang *et al.* [23] demonstrate that intermediate MLLM layers can hold richer region-level descriptions than the final layer. We build on these findings but focus specifically on semantic segmentation, introduce attention knockout experiments to test whether LLM layers actively resolve ambiguity, and ablate architectural choices such as attention directionality and vision encoder selection. Our work contributes a segmentation-focused, causal-intervention view, complementing classification- and correspondence-oriented findings.

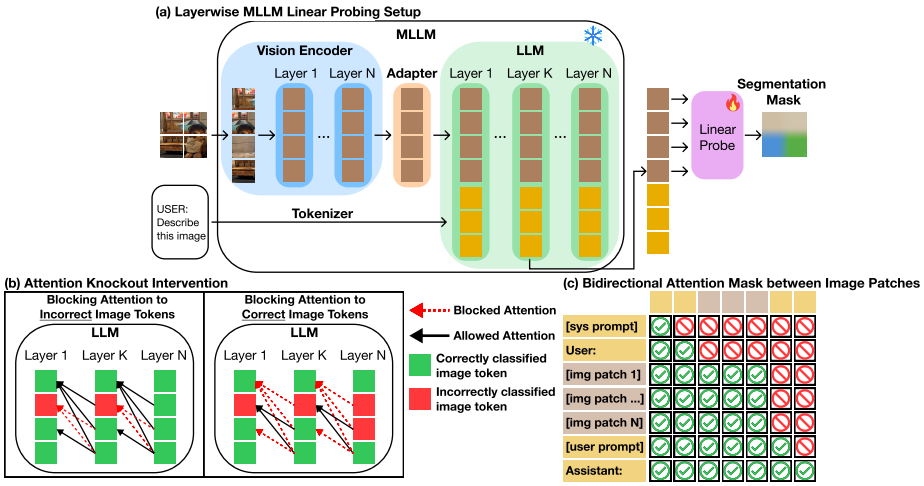

Fig. 2: Overview of the three analysis methods. (a) Layerwise linear probing: given an input image, the vision encoder produces patch token embeddings (brown), which are projected by the adapter into the LLM's embedding space. Inside the LLM, image tokens are processed jointly with text prompt tokens (yellow). At each layer ℓ, we extract only the image token representations and train an independent linear probe to predict per-patch semantic classes, reassembled into a 2D segmentation map. **(b)** Attention knockout: we selectively block attention to incorrectly classified tokens (left) or correctly classified tokens (right) across all LLM layers, testing whether cross-token attention drives self-refinement. **(c)** Bidirectional attention mask: image tokens attend to each other bidirectionally while all other token pairs retain causal masking, alleviating context starvation at early image positions.

3 Layerwise Linear Probing

Models and notation. Given an input image, the vision encoder divides it into a grid of T non-overlapping patches (e.g., $T = 576$ for a 336×336 image with patch size 14) and maps them to a sequence of token embeddings. The adapter, a learned two-layer MLP, projects each token embedding into the LLM's d-dimensional input space. Within each LLM layer, the full input sequence contains system tokens, image tokens, and text prompt tokens. We denote by $\mathbf{X}^{(\ell)} \in \mathbb{R}^{T \times d}$ the representations corresponding to the T image patch tokens only at layer ℓ. Under standard causal attention, image tokens can only attend to preceding tokens in the sequence (i.e., system tokens and earlier image patches) and are therefore not influenced by the text prompt that follows them. Since each token corresponds to a fixed spatial position in the image, per-token predictions can be reshaped into a 2D grid and upsampled to the original resolution for pixel-level evaluation.

Datasets. We evaluate on the following three standard segmentation benchmarks: ADE20K (150 classes, diverse indoor and outdoor scenes) [47]; PASCAL VOC 2012 (20 foreground classes, augmented training set) [12]; and Cityscapes

(urban street scenes) [10]. We use standard train/validation splits and report mIoU as the primary metric and pixel accuracy (pAcc) as a secondary measure.

Probing protocol. We introduce a probing protocol to systematically evaluate where segmentation competence arises, degrades, or is refined across the MLLM stack. Our framework targets adapter-style MLLMs, which combine a frozen vision encoder, a trained adapter, and an LLM. This modular structure allows us to isolate and compare representations at three stages: the vision encoder output, the adapter output, and each intermediate LLM layer. To evaluate the segmentation quality of representations at each stage of the MLLM, we train an independent linear probe per layer [1]. The procedure is identical across all stages for the vision encoder, adapter, and LLM layers, and differs only in which hidden state is extracted and the dimension of the hidden state.

For a target layer ℓ, we freeze the entire MLLM and extract $\mathbf{X}^{(\ell)}$ for every image in the training set. Each token is independently classified by a linear probe trained with cross-entropy on the frozen features. Additional training and implementation details are provided in the supplementary material.

We train three MLLM variants by pairing Vicuna-7B [37] with CLIP ViT-L/14@336, DINOv2 Large@336, and SigLIP SO400M/14, each following the standard LLaVA-1.5 two-stage procedure [24]: pretraining the adapter on 558K image-caption pairs, then finetuning the full model on 665K visual instruction data. By comparing mIoU across layers under identical probe training conditions, we obtain a complete profile of how segmentation-relevant information evolves from the vision encoder through the adapter and into the LLM.

3.1 The Adapter Introduces a Representation Drop-off

We first compare the segmentation quality of features immediately before and after the adapter. Fig. 3 reports the mIoU for three vision encoders at the encoder output and the adapter output. Across all encoders, we observe a consistent drop in mIoU after the adapter projects the visual features into the LLM's embedding space. The magnitude of this drop varies: CLIP experiences a modest decline, DINOv2 and SigLIP suffer larger degradations. This representation drop-off indicates that the adapter introduces a structural bottleneck, trading fine-grained spatial fidelity for cross-modal alignment with the language embedding space. Analogous representation gaps have been documented in contrastive vision-language spaces, where image and text embeddings occupy geometrically separated regions [22], while Fu *et al.* [13] show that VLMs can underperform their own vision encoders on tasks such as correspondence.

3.2 LLM Layers Progressively Recover Segmentation Quality

Segmentation quality does not continue to degrade as features propagate through the LLM. On the contrary, as shown in Fig. 3, mIoU steadily recovers across the LLM layers. The recovery is characterized by a sharp increase in mIoU in the early LLM layers, followed by a plateau in the mid-to-late layers where peak performance is reached. This is a notable finding: The LLM layers, which were

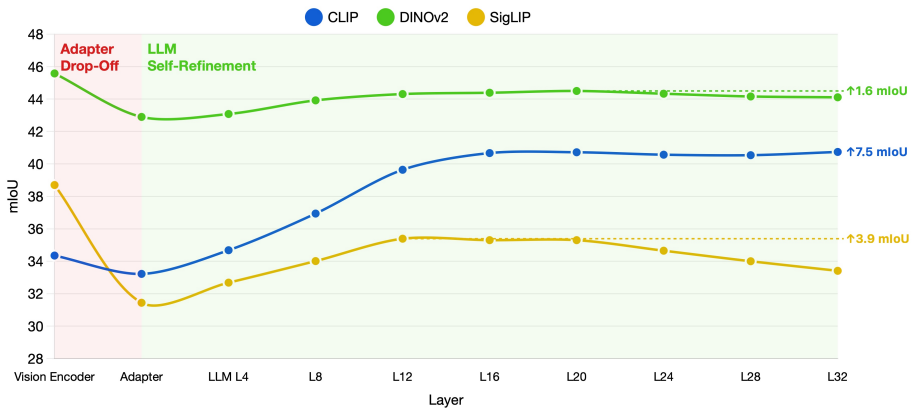

Fig. 3: Layerwise linear probing results across the MLLM stack. mIoU on ADE20K for CLIP, DINOv2, and SigLIP encoders paired with Vicuna-7B, measured at the vision encoder output, adapter output, and each LLM layer. All three encoders exhibit a drop at the adapter followed by progressive recovery across LLM layers. Dashed lines mark the best-performing layer; values on the right indicate the total mIoU improvement across LLM layers.

pretrained for language modeling and not for spatial reasoning, are able to refine the visual representations and restore segmentation-relevant structure that was lost at the adapter.

The drop-off and recovery pattern is consistent across all three encoders, but the strength of recovery varies. CLIP, which is pretrained with a contrastive objective that aligns visual features to text, exhibits the strongest recovery: its LLM-layer representations ultimately exceed the vision encoder baseline. SigLIP, which shares a similar contrastive pretraining objective, also shows meaningful recovery. DINOv2, a vision-only encoder with no text alignment, recovers less strongly. This suggests that the degree of compatibility between the vision encoder's representation space and the LLM's embedding space influences how effectively the LLM layers can refine the visual features.

3.3 Qualitative Evidence and Semantic Clustering

Fig. 4 shows segmentation predictions from the linear probe at different stages of the MLLM for representative ADE20K validation images. At the adapter output, predictions exhibit noisy boundaries and more frequent confusion between classes. At deeper LLM layers, these errors are progressively resolved: boundaries become more spatially coherent and class assignments stabilize. This visual evidence complements the quantitative mIoU improvements and suggests that the LLM layers perform a form of contextual refinement, leveraging global token interactions and semantics to re-impose structure to disambiguate local patch-level predictions, yielding net gains for specific conflicts.

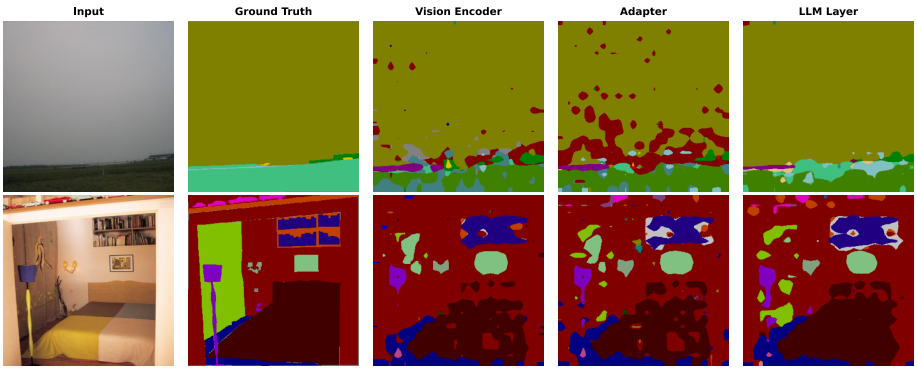

Fig. 4: Qualitative segmentation predictions across the MLLM stack. From left to right: input image, ground truth, linear probe prediction at the vision encoder output, adapter output, and at an intermediate LLM layer. Representation drop-off at the adapter but deeper layers appear to resolve class confusions (e.g., wall vs. bed) and produce more spatially coherent predictions.

To further investigate how the LLM layers refine visual representations, we visualize the hidden states of individual image patches using UMAP projections at different depths of the model. Fig. 5 shows the 2D embeddings of all 576 patch tokens from a single image of the CLIP MLLM at the adapter output, an intermediate LLM layer, and the layer at which linear probing performance peaks. Each patch is colored by the semantic class and classes not among the four most prevalent are shown in gray. At the adapter output, patches from different semantic classes are heavily interleaved, confirming that the projected features lack clear category-level organization. As representations pass through the LLM, same-class patches progressively cluster together, and by layer 20, distinct semantic groups such as floor, ceiling, wall, and building occupy clearly separated regions of the embedding space, wall and building cluster close together because of semantic similarity. This provides a complementary, geometric perspective on the recovery and corroborates the mIoU improvements observed in the linear probing results and qualitative results: The LLM does not merely make features more linearly separable, but actively re-organizes them into semantically coherent clusters.

We repeat the layerwise linear probing experiments across the MLLM stack on more multimodal architectures. In addition to LLaVA (CLIP ViT-L and Vicuna-7B) we evaluate OneVision [20], which uses SigLIP SO400M with Qwen2-7B as LLM (providing evidence relevant to LISA style segmentation MLLMs that share these adapter-style backbones) and DeepSeek-VL [27], which uses a hybrid Vision Encoder, combining SigLIP-L and SAM-B embeddings with DeepSeek-LLM 7B as LLM. Our supplementary shows that the drop-off and recovery mechanism we identify is a property that generalizes across architectures with different vision encoders and LLM backbones.

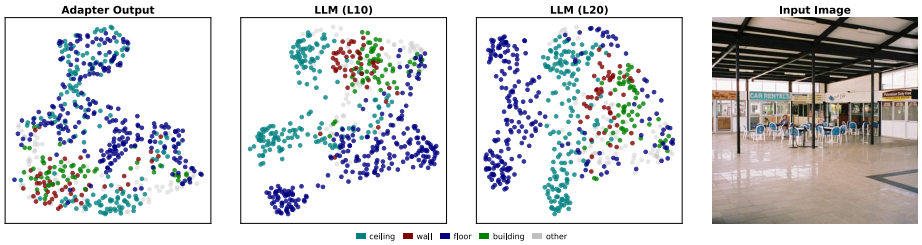

Fig. 5: UMAP projections of patch-level hidden states across the MLLM stack. Each point represents one of 576 image patch tokens, colored by semantic class. At the adapter output, classes are interleaved, by layer 20, same-class patches form distinct clusters, illustrating the progressive emergence of semantic structure through the LLM layers.

4 Attention Knockout

The layerwise probing results from Sec. 3.1 show that segmentation quality improves across LLM layers, suggesting that the model progressively resolves classification errors. To test whether this self-refinement is actively mediated by cross-token attention rather than being a passive byproduct of residual connections or layer normalization, we design a global attention knockout experiment. We adapt the attention knockout technique introduced by Geva *et al.* [15] for tracing information flow in auto-regressive language models, and applied to MLLM by Neo *et al.* [29] to study how visual information is extracted at the output position. While both prior works use knockout to identify which tokens inform the model's generated text output, we repurpose the technique to probe a different question: whether cross-token attention among image patch tokens themselves drives the self-refinement of spatial representations across LLM layers.

Procedure. Given a test image, we first obtain predicted labels $\hat{y}_t$ for each image patch token t. We select a target class c to block and identify the corresponding token set $\mathcal{B}_c = \{s \in \mathcal{I} : \hat{y}_s = c\}$, where $\mathcal{I}$ denotes the full set of image tokens. At every LLM layer ℓ, we mask out all attention from any image token to any token in $\mathcal{B}_c$ by setting the pre-softmax attention logits to $-\infty$:

$$\mathbf{A}^{(\ell)}_{t \leftarrow s} \leftarrow -\infty \quad \forall\, t \in \mathcal{I},\ s \in \mathcal{B}_c. \tag{1}$$

This renders class c completely invisible: No image token, including tokens of class c themselves, can attend to the blocked tokens. Blocked tokens can still attend to all non-blocked tokens, so their representations continue to evolve but without any self-reinforcement from same-class neighbors. We then extract hidden states at every layer and evaluate segmentation via the same per-layer linear probes used in the earlier experiment.

Experimental conditions. We focus on images where the unmodified model exhibits characteristic class confusions, e.g., patches of ceiling misclassified as sky. For each such image, we run two complementary conditions and compare the resulting layerwise segmentation against the unmodified model (Fig. 2b):

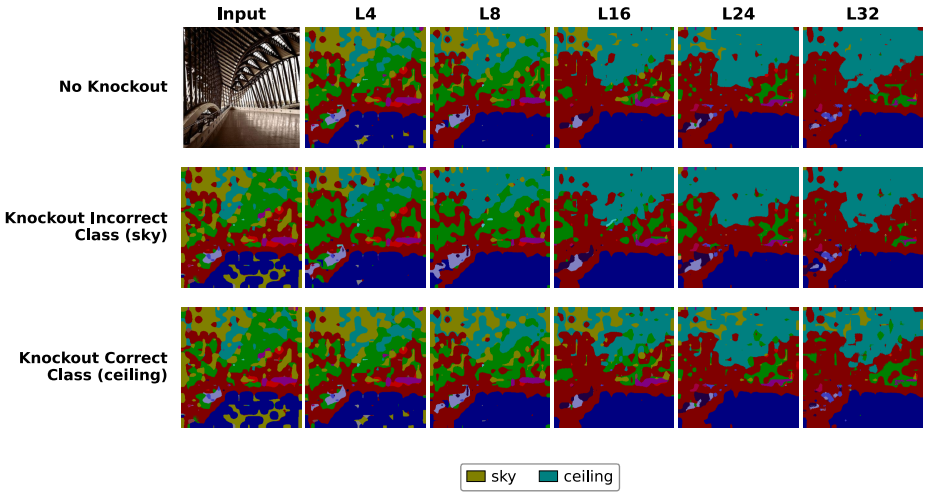

Fig. 6: Global attention knockout: layerwise segmentation comparison. Top row: unmodified baseline. Middle row: incorrect class blocked (e.g. sky). Bottom row: correct class blocked (e.g. ceiling). Blocking the incorrect class accelerates the resolution of misclassified patches across layers, while blocking the correct class impairs self-correction and leaves more errors in mid-to-late layers.

1. **Block incorrect class.** We block the class that the model *incorrectly* assigns to some tokens (e.g. block *sky* when the ground truth is *ceiling*). If attention to incorrectly classified tokens reinforces errors, removing their influence should accelerate self-correction through global context.
2. **Block correct class.** We block the class that is *correctly* assigned (e.g. block *ceiling*). If correctly classified tokens serve as semantic anchors in a global context that pull misclassified neighbors toward the right label, removing them should impair self-correction.

4.1 Blocking the Incorrect Class Accelerates Self-Correction

We select ADE20K validation images where the unmodified model exhibits characteristic class confusions, such as ceiling patches misclassified as sky, and compare layerwise segmentation maps under the two blocking conditions against the unmodified model.

When we block the incorrectly assigned class (e.g. making *sky* invisible in an image where the ground truth is *ceiling*), misclassified tokens are corrected faster across layers compared to the unmodified model. Already in the early LLM layers, the segmentation maps show reduced confusion in the affected region, and by the final layer the model, with incorrect classes knocked out, produces fewer residual misclassifications than the unmodified model. This indicates that tokens carrying the incorrect label were actively reinforcing erroneous predictions through attention: by silencing them, the remaining contextual cues from cor-

rectly classified neighbors (e.g. walls, floors, and other indoor elements) dominate the attention field, enabling the model to resolve the ambiguity more efficiently.

4.2 Blocking the Correct Class Impairs Self-Correction

The complementary condition reveals the opposite effect. When we block the correctly assigned class (e.g. making ceiling invisible), the model's ability to self-correct deteriorates markedly. Misclassified tokens persist longer across the layer progression, and in the mid-to-late layers the segmentation quality falls below that of the unmodified model, with more misclassified patches remaining than when no intervention is applied. This demonstrates that correctly classified tokens serve as semantic anchors: Their attention signals help pull misclassified neighbors toward the correct label. Without these anchors, the model loses its primary self-correction mechanism and errors are left unresolved or even amplified.

4.3 Cross-Token Attention Drives Self-Refinement

Together, these two conditions provide direct evidence that the representation refinement documented in Sec. 3.1 is not a passive byproduct of residual connections or layer normalization, but is actively mediated by cross-token attention. The LLM layers leverage semantic context, attending to tokens of the correct class, to progressively resolve local classification errors. However, because self-correction relies on the presence of correctly classified anchors, this mechanism cannot recover fine-grained spatial details absent from the encoder features or overcome systematic encoder biases where no correct anchors exist. These findings confirm the self-refinement hypothesis and identify cross-token attention as its operative mechanism.

We have also extended our attention knockout analysis to OneVision and DeepSeek-VL. Our supplementary shows both architectures reproduce the semantic anchor effect we identify for LLaVA: blocking the incorrect class accelerates self-correction, while blocking the correct class impairs it, demonstrating that cross-token attention drives self-refinement across diverse MLLM architectures, beyond the LLaVA framework.

5 Causal vs. Bidirectional Attention

The preceding experiments established that cross-token attention drives self-refinement across LLM layers (Secs. 3.1 and 4.1). However, under standard causal attention, image tokens are processed in raster order (left-to-right, top-to-bottom) and each token can only attend to preceding tokens in the sequence. The first image token cannot see any other image tokens, while the last token attends to all T image patches. This creates a structural asymmetry in which early tokens lack global spatial context, a limitation particularly relevant for segmentation, where every patch should ideally access scene-level information.

We observed this effect in our layerwise probing experiments: tokens in the first row, and especially the top-left corner, are often misclassified in the qualitative visualizations and exhibited consistently lower classification accuracy that did not improve across LLM layers.

Image-only bidirectional attention. To test whether this positional bias limits segmentation quality, we modify the attention mask to grant bidirectional attention exclusively among image tokens while preserving causal attention for text generation (Fig. 2c). Let $\mathcal{I}$ denote the set of image token positions within the input sequence. The modified mask function is:

$$M(q,k) = \underbrace{(q \in \mathcal{I} \ \wedge \ k \in \mathcal{I})}_{\text{image–image: bidirectional}} \quad \vee \quad \underbrace{(q \geq k)}_{\text{causal}}, \tag{2}$$

where q and k index the query and key positions in the full input sequence, respectively. When both tokens lie within the image region, attention is permitted regardless of their relative position, but for all other pairs, the standard causal constraint applies. This design differs from the prefix-LM strategy employed by PaLiGemma [4], which grants bidirectional attention across *all* input tokens, image, system prompt, and task prefix alike. Our variant targets spatial self-refinement among image tokens specifically, isolating its effect on segmentation while preserving the sequential structure required for autoregressive text generation.

Training. We follow the standard LLaVA-1.5 two-stage procedure: (1) pretraining the adapter on 558K image-caption pairs with the LLM frozen, and (2) finetuning the full model on 665K visual instruction data. Both stages use the same attention type, either fully causal or fully bidirectional, so that the comparison reflects the cumulative effect of attention directionality across the entire training process. All MLLM hyperparameters, training data, and vision encoders (CLIP, DINOv2, SigLIP) are identical between the two conditions, only the attention mask differs.

5.1 Early Tokens Suffer Context Starvation

We compare per-patch classification accuracy between the causal and bidirectional CLIP MLLM models at the same layer across the ADE20K validation set. To isolate the positional effect of the attention mask from the content-distribution bias shared by both models, we report the difference in pixel accuracy between the bidirectional and causal attention mechanisms evaluated at patch positions. Fig. 7 reports accuracy for the first 50 patch tokens in patch order. The gap is striking: Patch 0, which under causal attention attends to no visual context, shows a +14.35 percentage-point pixel accuracy increase, a 23.2% relative improvement under bidirectional attention; the gap decays in later image patches.

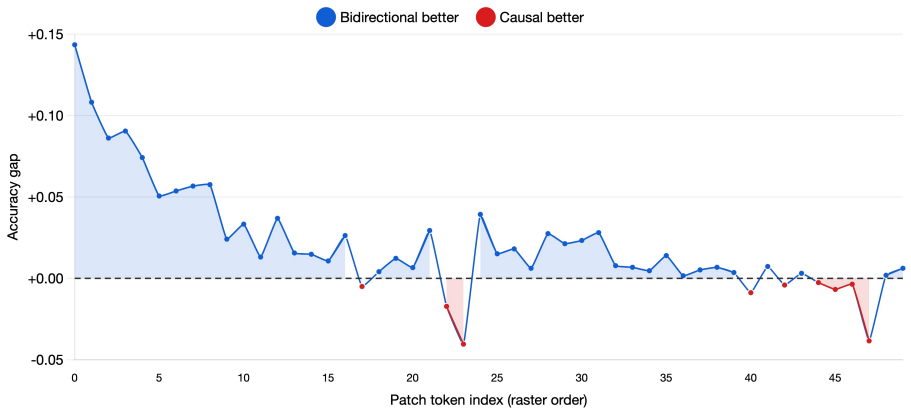

Fig. 7: Context starvation under causal attention. Per-patch pixel accuracy for the first 50 tokens in the LLM layer of the MLLM. Accuracy Gap: Bidirectional minus causal accuracy. Early patches suffer severe context starvation under causal masking.

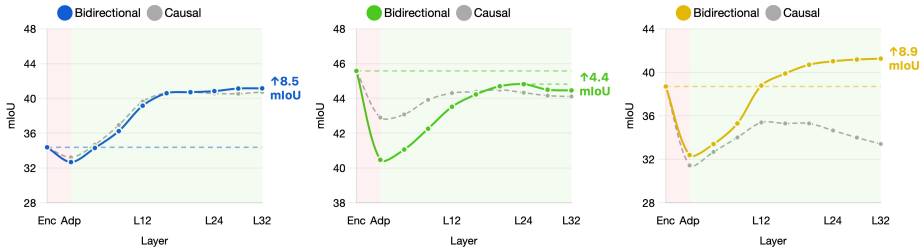

Fig. 8: Causal vs. bidirectional attention: Layerwise probing on ADE20K across the MLLM Stack for different vision encoders. Bidirectional attention yields modest peak mIoU gains for CLIP and DINOv2, while SigLIP shows a larger effect.

5.2 Bidirectional Attention Sustains Recovery Across Layers

Fig. 8 compares layerwise linear probing mIoU for causal and bidirectional MLLMs across all three encoder configurations. Bidirectional attention yields modest peak mIoU gains for CLIP (+0.42) and DINOv2 (+0.32), consistent with the per-patch analysis: The improvement is concentrated at the few context-starved positions rather than reflecting a broad representational change. SigLIP shows a larger effect: Under causal attention, recovery peaks at layer 12 (mIoU 35.39) and then declines through deeper layers, while bidirectional attention sustains monotonic improvement to 41.26 at layer 32.

Table 1: VQA benchmark: causal vs. bidirectional LLaVA 1.5 (7B) across three vision encoders. All models use Vicuna-7B as the LLM backbone. Bidirectional attention preserves language understanding with segmentation gains that vary across encoders. DINOv2 shows broader regressions, consistent with its lack of text alignment.

Vision Encoder	Attention	GQA	MMB	MME^P	MME^C	MMMU	POPE	SQA^I	TextVQA	VizWiz
CLIP ViT-L/14	Causal	62.6	66.4	1483	284	35.3	86.8	68.7	46.9	56.0
	Bidirectional	62.7	65.5	1538	288	36.2	86.9	69.7	47.0	57.5
DINOv2 ViT-L/14	Causal	62.1	57.7	1304	324	34.6	87.2	66.1	14.0	51.4
	Bidirectional	60.5	55.3	1247	326	32.4	85.1	66.3	13.7	45.8
SigLIP SO400M/14	Causal	61.1	63.7	1414	275	34.7	84.4	70.4	50.2	58.2
	Bidirectional	62.1	66.9	1402	298	34.9	84.8	70.7	53.6	53.3

5.3 Self-Refinement Requires Access to Neighbors

The bidirectional attention experiment reveals context starvation as a real but sharply localized cost of causal masking. For the first few image tokens, the causal mask creates a persistent representation penalty that is not resolved across LLM layers; bidirectional attention eliminates it by granting immediate access to all visual neighbors. Beyond these early positions, causal and bidirectional attention produce comparable representations, indicating that self-refinement through partial context is sufficient once a token has access to even a modest number of visual neighbors. This finding reinforces the self-refinement narrative from Sec. 4.1: Self-refinement depends on access to semantically consistent neighbors, and causal masking delays the formation of these contextual anchors for early tokens, whereas bidirectional attention restores immediate global context and enables refinement to operate uniformly across spatial positions.

5.4 Effect on Language Understanding

The preceding sections show that bidirectional attention among image tokens improves segmentation, particularly for context-starved early patches. A natural concern is whether this modification degrades the model's language capabilities. Tab. 1 compares VQA performance across nine benchmarks for causal and bidirectional variants of all three encoder configurations. CLIP and SigLIP models maintain comparable performance to their causal baselines across most benchmarks, indicating that bidirectional image attention preserves language understanding. DINOv2 shows minor regressions, consistent with its weaker text alignment observed throughout our experiments. These results suggest that bidirectional attention among image tokens offers a favorable trade-off: it alleviates context starvation for segmentation without sacrificing language capabilities for text-aligned encoders.

6 Conclusion

Our probing and intervention framework reveals that adapter-style MLLMs introduce a representation drop-off at the adapter that degrades token-level separability, but LLM layers progressively recover segmentation quality through attention-mediated self-refinement. Attention knockout experiments confirm that correctly classified tokens act as semantic anchors whose attention signals pull misclassified neighbors toward the correct label, identifying cross-token attention as the operative refinement mechanism. Causal attention creates context starvation at early image token positions that bidirectional attention among image tokens alleviates by restoring immediate global context and enabling refinement to operate uniformly across spatial positions. These findings provide a mechanistic account of how MLLMs process visual information for segmentation, informing the design of future segmentation-capable MLLMs and hopefully providing a step for more interpretable multimodal systems.

Limitations. We used LLaVA-type models as a starting point and varied the vision encoder across the configurations for interpreting MLLMs, but our conclusions might not generalize for significantly different architectures. Our linear probes may underestimate actual segmentation capacity achievable with richer task heads and the knockout interventions block attention globally across all layers, which does not isolate the contribution of individual layers to the refinement process. Future work could explore whether these findings extend to broader task types and model architectures.

Acknowledgements

This work was partially funded by the ERC (853489 DEXIM) and the Alfried Krupp von Bohlen und Halbach Foundation, for which we thank them for their generous support. The authors gratefully acknowledge the scientific support and resources of the AI service infrastructure LRZ AI Systems provided by the Leibniz Supercomputing Centre (LRZ) of the Bavarian Academy of Sciences and Humanities (BAdW), funded by Bayerisches Staatsministerium für Wissenschaft und Kunst (StMWK). We also acknowledge the use of the HPC cluster at Helmholtz Munich for the computational resources used in this study.

References

1. Alain, G., Bengio, Y.: Understanding intermediate layers using linear classifier probes. In: ICLR Workshop (2017). https://doi.org/10.48550/arXiv.1610.01644
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-VL Technical Report (Feb 2025). https://doi.org/10.48550/arXiv.2502.13923
3. Banani, M.E., Raj, A., Maninis, K.K., Kar, A., Li, Y., Rubinstein, M., Sun, D., Guibas, L., Johnson, J., Jampani, V.: Probing the 3D Awareness of Visual Foundation Models. In: CVPR (2024). https://doi.org/10.48550/arXiv.2404.08636
4. Beyer, L., Steiner, A., Pinto, A.S., Kolesnikov, A., Wang, X., Salz, D., Neumann, M., Alabdulmohsin, I., Tschannen, M., Bugliarello, E., Unterthiner, T., Keysers, D., Koppula, S., Liu, F., Grycner, A., Gritsenko, A., Houlsby, N., Kumar, M., Rong, K., Eisenschlos, J., Kabra, R., Bauer, M., Bošnjak, M., Chen, X., Minderer, M., Voigtlaender, P., Bica, I., Balazevic, I., Puigcerver, J., Papalampidi, P., Henaff, O., Xiong, X., Soricut, R., Harmsen, J., Zhai, X.: PaliGemma: A versatile 3B VLM for transfer (Oct 2024). https://doi.org/10.48550/arXiv.2407.07726
5. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: SAM 3: Segment Anything with Concepts. In: ICLR (2026). https://doi.org/10.48550/arXiv.2511.16719
6. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging Properties in Self-Supervised Vision Transformers. In: ICCV (2021). https://doi.org/10.48550/arXiv.2104.14294
7. Chen, Y.C., Li, W.H., Sun, C., Wang, Y.C.F., Chen, C.S.: SAM4MLLM: Enhance Multi-Modal Large Language Model for Referring Expression Segmentation. In: ECCV (2024). https://doi.org/10.48550/arXiv.2409.10542
8. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., Gu, L., Wang, X., Li, Q., Ren, Y., Chen, Z., Luo, J., Wang, J., Jiang, T., Wang, B., He, C., Shi, B., Zhang, X., Lv, H., Wang, Y., Shao, W., Chu, P., Tu, Z., He, T., Wu, Z., Deng, H., Ge, J., Chen, K., Zhang, K., Wang, L., Dou, M., Lu, L., Zhu, X., Lu, T., Lin, D., Qiao, Y., Dai, J., Wang, W.: Expanding Performance Boundaries of Open-Source Multimodal Models with Model, Data, and Test-Time Scaling (Dec 2024). https://doi.org/10.48550/arXiv.2412.05271
9. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention Mask Transformer for Universal Image Segmentation. In: CVPR (2022). https://doi.org/10.48550/arXiv.2112.01527
10. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In: CVPR (2016). https://doi.org/10.48550/arXiv.1604.01685
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In: ICLR (2021). https://doi.org/10.48550/arXiv.2010.11929

12. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision **88**(2), 303–338 (Jun 2010). https://doi.org/10.1007/s11263-009-0275-4
13. Fu, S., Bonnen, T., Guillory, D., Darrell, T.: Hidden in plain sight: VLMs overlook their visual representations. In: Conference on Language Modeling (COLM) (2025). https://doi.org/10.48550/arXiv.2506.08008
14. Fu, X., Hu, Y., Li, B., Feng, Y., Wang, H., Lin, X., Roth, D., Smith, N.A., Ma, W.C., Krishna, R.: BLINK: Multimodal Large Language Models Can See but Not Perceive. In: ECCV (2024). https://doi.org/10.48550/arXiv.2404.12390
15. Geva, M., Bastings, J., Filippova, K., Globerson, A.: Dissecting Recall of Factual Associations in Auto-Regressive Language Models. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 12216–12235. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.emnlp-main.751
16. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV (2017). https://doi.org/10.48550/arXiv.1703.06870
17. Hu, R., Rohrbach, M., Darrell, T.: Segmentation from Natural Language Expressions. In: ECCV (2016). https://doi.org/10.48550/arXiv.1603.06180
18. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment Anything. In: ICCV (2023). https://doi.org/10.48550/arXiv.2304.02643
19. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: LISA: Reasoning Segmentation via Large Language Model. In: CVPR (2024). https://doi.org/10.48550/arXiv.2308.00692
20. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., Li, C.: LLaVA-OneVision: Easy Visual Task Transfer (2024). https://doi.org/10.48550/arXiv.2408.03326
21. Li, S., Koh, P.W., Du, S.S.: Exploring How Generative MLLMs Perceive More Than CLIP with the Same Vision Encoder. In: Annual Meeting of the Association for Computational Linguistics (ACL) (2025). https://doi.org/10.48550/arXiv.2411.05195
22. Liang, W., Zhang, Y., Kwon, Y., Yeung, S., Zou, J.: Mind the Gap: Understanding the Modality Gap in Multi-modal Contrastive Representation Learning. In: NeurIPS (2022). https://doi.org/10.48550/arXiv.2203.02053
23. Liang, Y., Cai, Z., Xu, J., Huang, G., Wang, Y., Liang, X., Liu, J., Li, Z., Wang, J., Huang, S.L.: Unleashing Region Understanding in Intermediate Layers for MLLM-based Referring Expression Generation. In: NeurIPS (Nov 2024), https://openreview.net/forum?id=168NLzTpw8
24. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved Baselines with Visual Instruction Tuning. In: CVPR (2024). https://doi.org/10.48550/arXiv.2310.03744
25. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual Instruction Tuning. In: NeurIPS (2023). https://doi.org/10.48550/arXiv.2304.08485
26. Liu, J., Feng, M., Chen, L.: Better, Stronger, Faster: Tackling the Trilemma in MLLM-based Segmentation with Simultaneous Textual Mask Prediction. In: CVPR (2026). https://doi.org/10.48550/arXiv.2512.00395
27. Lu, H., Liu, W., Zhang, B., Wang, B., Dong, K., Liu, B., Sun, J., Ren, T., Li, Z., Yang, H., Sun, Y., Deng, C., Xu, H., Xie, Z., Ruan, C.: DeepSeek-VL: Towards Real-World Vision-Language Understanding (2024). https://doi.org/10.48550/arXiv.2403.05525

28. Merullo, J., Castricato, L., Eickhoff, C., Pavlick, E.: Linearly Mapping from Image to Text Space. In: ICLR (2023). https://doi.org/10.48550/arXiv.2209.15162
29. Neo, C., Ong, L., Torr, P., Geva, M., Krueger, D., Barez, F.: Towards Interpreting Visual Information Processing in Vision-Language Models. In: ICLR (2025). https://doi.org/10.48550/arXiv.2410.07149
30. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning Robust Visual Features without Supervision. TMLR (2024). https://doi.org/10.48550/arXiv.2304.07193
31. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021), http://proceedings.mlr.press/v139/radford21a
32. Rasheed, H., Maaz, M., Mullappilly, S.S., Shaker, A., Khan, S., Cholakkal, H., Anwer, R.M., Xing, E., Yang, M.H., Khan, F.S.: GLaMM: Pixel Grounding Large Multimodal Model. In: CVPR (2024). https://doi.org/10.48550/arXiv.2311.03356
33. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: SAM 2: Segment Anything in Images and Videos. In: ICLR (2025). https://doi.org/10.48550/arXiv.2408.00714
34. Ren, Z., Huang, Z., Wei, Y., Zhao, Y., Fu, D., Feng, J., Jin, X.: PixelLM: Pixel Reasoning with Large Multimodal Model. In: CVPR (2024). https://doi.org/10.48550/arXiv.2312.02228
35. Tong, S., Brown, E., Wu, P., Woo, S., Middepogu, M., Akula, S.C., Yang, J., Yang, S., Iyer, A., Pan, X., Wang, Z., Fergus, R., LeCun, Y., Xie, S.: Cambrian-1: A Fully Open, Vision-Centric Exploration of Multimodal LLMs. In: NeurIPS (2024). https://doi.org/10.48550/arXiv.2406.16860
36. Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., Xie, S.: Eyes Wide Shut? Exploring the Visual Shortcomings of Multimodal LLMs. In: CVPR (2024). https://doi.org/10.48550/arXiv.2401.06209
37. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: LLaMA: Open and Efficient Foundation Language Models (Feb 2023). https://doi.org/10.48550/arXiv.2302.13971
38. Wu, S., Jin, S., Zhang, W., Xu, L., Liu, W., Li, W., Loy, C.C.: F-LMM: Grounding Frozen Large Multimodal Models. In: CVPR (2025). https://doi.org/10.48550/arXiv.2406.05821
39. Xia, Z., Han, D., Han, Y., Pan, X., Song, S., Huang, G.: GSVA: Generalized Segmentation via Multimodal Large Language Models. In: CVPR (2024). https://doi.org/10.48550/arXiv.2312.10103
40. Yao, H., Wu, W., Yang, T., Song, Y., Zhang, M., Feng, H., Sun, Y., Li, Z., Ouyang, W., Wang, J.: Dense Connector for MLLMs. In: NeurIPS (2024). https://doi.org/10.48550/arXiv.2405.13800
41. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid Loss for Language Image Pre-Training. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 11941–11952. IEEE, Paris, France (Oct 2023). https://doi.org/10.1109/ICCV51070.2023.01100

42. Zhang, C., Cho, J., Puspitasari, F.D., Zheng, S., Li, C., Qiao, Y., Kang, T., Shan, X., Zhang, C., Qin, C., Rameau, F., Lee, L.H., Bae, S.H., Hong, C.S.: A Survey on Segment Anything Model (SAM): Vision Foundation Model Meets Prompt Engineering (Oct 2024). https://doi.org/10.48550/arXiv.2306.06211
43. Zhang, T., Li, X., Fei, H., Yuan, H., Wu, S., Ji, S., Loy, C.C., Yan, S.: OMG-LLaVA: Bridging Image-level, Object-level, Pixel-level Reasoning and Understanding. In: NeurIPS (2024). https://doi.org/10.48550/arXiv.2406.19389
44. Zhang, Y., Ma, Z., Gao, X., Shakiah, S., Gao, Q., Chai, J.: GROUNDHOG: Grounding Large Language Models to Holistic Segmentation. In: CVPR (2024). https://doi.org/10.48550/arXiv.2402.16846
45. Zhang, Y., Unell, A., Wang, X., Ghosh, D., Su, Y., Schmidt, L., Yeung-Levy, S.: Why are Visually-Grounded Language Models Bad at Image Classification? In: NeurIPS (2024). https://doi.org/10.48550/arXiv.2405.18415
46. Zhang, Z., Ma, Y., Zhang, E., Bai, X.: PSALM: Pixelwise SegmentAtion with Large Multi-Modal Model. In: ECCV (2024). https://doi.org/10.48550/arXiv.2403.14598
47. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic Understanding of Scenes through the ADE20K Dataset. IJCV **127**, 302–321 (2019). https://doi.org/10.1007/s11263-018-1140-0
48. Zou, X., Yang, J., Zhang, H., Li, F., Li, L., Wang, J., Wang, L., Gao, J., Lee, Y.J.: Segment Everything Everywhere All at Once. In: NeurIPS (2023). https://doi.org/10.48550/arXiv.2304.06718