Efficient SAM3 Adaptation for Multi-Class Semantic Segmentation via Dense Competitive Representations

Wenbin Liao[1,2] **Hao Zhu**[1,2] **Yike Ma**[1] **Hao Jiang**[1] **Feng Dai**[1,*]
[1]Institute of Computing Technology, Chinese Academy of Science
[2]School of Computer Science and Technology, University of Chinese Academy of Sciences
{liaowenbin23z, zhuhao22z, ykma, jianghao, fdai}@ict.ac.cn

Abstract

The Segment Anything Model 3 (SAM3) has achieved significant progress in *Promptable Concept Segmentation (PCS)* by processing short noun phrases to generate segmentation masks with unique instance identifiers. While SAM3 excels at single-concept segmentation, it falls short in complex, real-world environments where multiple concepts must be segmented simultaneously. In particular, SAM3 typically fails to simultaneously process concepts with semantic overlap, leading to semantic misclassifications and redundant mask predictions. Moreover, as the number of concepts increases, the independent inference paradigm introduces prohibitive inference latency. To address these limitations, we propose the **D**ense **C**ompetitive **R**epresentations **SAM** (**DCR-SAM**) framework. By decoupling cross-modal interactions, the proposed method alleviates the inference latency. To resolve semantic misclassifications, DCR-SAM dynamically aligns textual concepts with visual representations and incorporates a dense competition head to enforce explicit inter-class competition. Furthermore, a learnable background token is applied to absorb non-target objects dynamically. These mechanisms effectively suppress overlapping predictions and generate high-fidelity masks. Extensive experimental evaluations across multiple benchmarks demonstrate that the proposed framework: ❶ achieves a 2.3% $\sim$ 10.7% performance improvement over state-of-the-art methods, ❷ delivers a 33$\times$ inference speedup compared to the native SAM3, and ❸ exhibits strong robustness across diverse scenarios. Our code is publicly available at GitHub.

1 Introduction

The evolution of Visual Foundation Models (VFMs) [10, 42, 49, 27, 46, 9] achieved a breakthrough with the Segment Anything Model 3 [9] (SAM3). As a pioneering architecture in *Promptable Concept Segmentation (PCS)*, SAM3 [9] demonstrates exceptional capability in single-concept segmentation guided by short noun phrases (*e.g.*, a red billboard). By incorporating deep cross-modal attention mechanisms, the model leverages textual prompts to dynamically condition visual representations, enabling precise mask generation.

While SAM3 [9] excels at single-concept segmentation, complex real-world scenarios [31, 33, 6] (*e.g.*, autonomous driving and remote sensing) require simultaneous identification and segmentation of multiple concepts within a single scene. Conventional semantic segmentation methods [62, 11, 19, 25] achieve multi-class segmentation by assigning individual pixels to a set of predefined classes. However, these methods face limitations in complex environments. They typically rely on narrow, task-specific feature representations and lack broad semantic priors of large-scale foundation models. On the other hand, SAM3 [9] leverage robust visual-linguistic representations to achieve remarkable

*Corresponding Author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

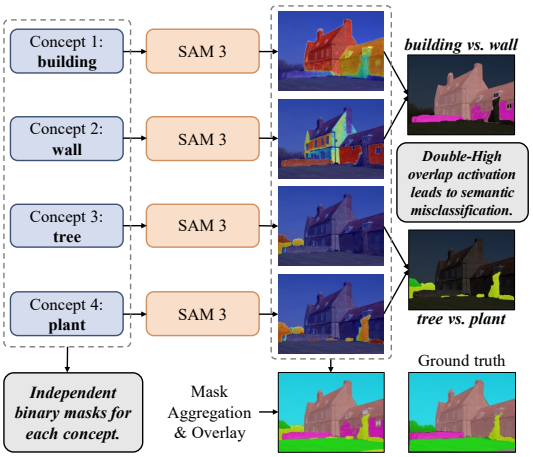


Figure 1: **Limitations of direct mask aggregation in SAM3.** Treating multi-class segmentation as isolated binary segmentation tasks lacks inter-class competition, causing "double-high" activation for similar concepts and leading to severe semantic misclassifications in the final mask.

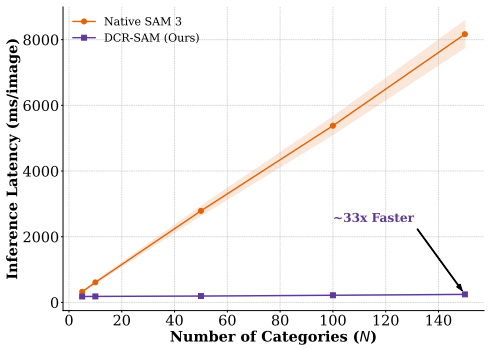


Figure 2: **Comparison of inference latency.** The inference latency of the original SAM3 increases linearly $\mathcal{O}(n)$ with the number of categories, whereas our method achieves a 33x speedup.

performance in single-concept segmentation. However, it faces two critical challenges when extended to multi-class scenarios: ❶ SAM3 typically fails to simultaneously process similar concepts, leading to semantic misclassifications and redundant mask predictions. ❷ As the number of target concepts increases, the original SAM3 inference paradigm incurs exceptionally high inference latency.

Recent works [31, 14] achieve multi-class segmentation by aggregating the individual masks independently predicted by SAM3 for multiple categories. However, this paradigm inherently treats the multi-class segmentation as a series of independent binary segmentation tasks. Consequently, there is a complete absence of competition among categories, which leaves the model without explicit constraints to differentiate semantically similar concepts. As shown in Figure 1, the model frequently produces "double-high" activations for identical pixel regions when processing similar concepts (*e.g.*, tree *vs.* plant). This unconstrained behavior inevitably leads to semantic misclassifications and overlapping masks, severely degrading the reliability of the dense prediction results.

On the other hand, the standard architecture of SAM3 requires executing independent and computationally expensive cross-attention operations for each concept to acquire concept-specific dense visual features, followed by an independent decoding process. As shown in Figure 2, our inference latency scaling analysis reveals that on the ADE20K [70] dataset with 150 categories, the inference time scales linearly ($\mathcal{O}(n)$) as the number of categories increases, ultimately resulting in severe inference latency.

To address these limitations, we propose the **D**ense **C**ompetitive **R**epresentations **SAM** (**DCR-SAM**) framework, which adapts SAM3 into an efficient architecture for multi-class segmentation. Specifically, the proposed framework resolves the aforementioned challenges through three key designs: ❶ To overcome the $\mathcal{O}(n)$ inference latency, we decouple the deep cross-modal interactions by replacing class-specific text queries with a unified learnable prompt sequence. This approach enables the extraction of a globally contextualized visual representation shared across all concepts through a single forward pass, achieving a nearly 33$\times$ inference speedup compared to the native SAM3, as shown in Figure 2. ❷ To resolve semantic misclassifications among similar concepts, we transition from isolated decoding to a dense competitive paradigm through the introduction of the *Vision-Conditioned Text Refinement (VCTR)* and *Vision-Guided Text Pooling (VGTP)* modules, coupled with a dense competition head. *VCTR* explicitly aligns textual sequence with the modified visual representation by injecting image-specific visual cues, whereas *VGTP* dynamically compresses this aligned sequence into highly discriminative global class anchors. By enforcing explicit inter-class competition, the proposed method effectively alleviates "double-high" activations and generates

high-fidelity masks. ❸ Furthermore, to enhance adaptability across diverse unconstrained scenarios, we incorporate a dynamically learnable background token to absorb non-target objects adaptively.

The main contributions of our work are summarized as follows:

- We reveal how the independent inference paradigm of SAM3 induces severe semantic misclassification among conceptually similar categories and results in an $\mathcal{O}(n)$ inference latency.
- We propose **DCR-SAM**, by decoupling cross-modal interactions to alleviate inference latency and dynamically aligning textual concepts with visual representation, the proposed method effectively suppresses overlapping predictions to generate high-fidelity masks.
- Extensive experimental evaluations demonstrate that our method achieves state-of-the-art (SOTA) performance across multiple domain benchmarks while exhibiting excellent inference efficiency.

2 Related Work

Promptable Segmentation Models The Segment Anything Model (SAM) [27, 46, 9] represents a significant milestone for promptable segmentation models, demonstrating remarkable cross-domain generalization across diverse downstream applications [31, 3, 54, 41, 6, 33]. Central to this success is the *Promptable Visual Segmentation (PVS)* paradigm introduced by SAM [27] and SAM 2 [46], where visual prompts (*e.g.*, points, boxes, or masks) guide the segmentation of specific targets. Driven by large-scale, high-quality datasets, this paradigm establishes a robust baseline for generalized segmentation. Building upon this foundation, SAM3 [9] extends the architecture from *PVS* to *Promptable Concept Segmentation (PCS)*, enabling the model to segment all instances of a given concept using short noun phrases (*e.g.*, a red billboard). Recent studies [30, 44, 63, 64, 13, 72] explore improvements to this architecture. For example, EfficientSAM3 [63, 64] enhances the overall inference efficiency of SAM3 through distillation, SAM3-Adapter [13] injects prior heuristic information for downstream tasks via a lightweight staged adapter. However, these works fundamentally retain the inherent serial inference mechanism of the native architecture. Consequently, they incur non-negligible inference latency and semantic misclassification when applied to real-world multi-class segmentation tasks. Meanwhile CoCo-SAM3 [14] proposes a training-free framework that separates the inference process into intra-class enhancement and inter-class competition modules to alleviate semantic misclassification. Nevertheless, this method relies heavily on complex score calibration and multi-prompt encoding. In this work, we alleviate these issues at the architectural level by decoupling cross-modal interactions. Furthermore, by dynamically aligning textual anchors with visual representations and introducing explicit inter-class competition, the proposed framework effectively mitigates semantic misclassification and generates high-fidelity masks.

Multi-Class Semantic Segmentation and Foundation Models Multi-class semantic segmentation is a fundamental task in computer vision, with the objective of assigning a specific semantic label to every pixel within an image. The field has witnessed remarkable progress, evolving from early CNN-based models, such as FCN [35] and U-Net [47], to modern Transformer-based architectures, such as SegFormer [61], Mask2Former [16] and EoMT [25]. While these methods excel at efficient dense prediction, they typically rely on narrow, task-specific feature representations, thereby lacking the broad semantic priors inherent in large-scale foundation models. Several vision-language models, such as CLIP [45], SigLIP [65], and ALIGN [24], emphasize global image-text alignment and inherently lack pixel-level granularity, rendering them unsuitable for direct application to downstream dense prediction tasks. Conversely, self-supervised vision foundation models, such as DINO [10, 42, 49, 71], capture excellent fine-grained pixel-level representations on massive image corpora, yet they lack explicit cross-modal semantic guidance and require the training of task-specific decoder heads for multi-class segmentation. In this work, we fully leverage the powerful cross-modal priors of SAM3 [9] to enhance the potential of the model in dense prediction tasks.

3 Methodology

In this section, we first briefly review the *Promptable Concept Segmentation (PCS)* pipeline of SAM3 (§3.1). Subsequently, we elaborate on the overall architecture of our proposed DCR-SAM

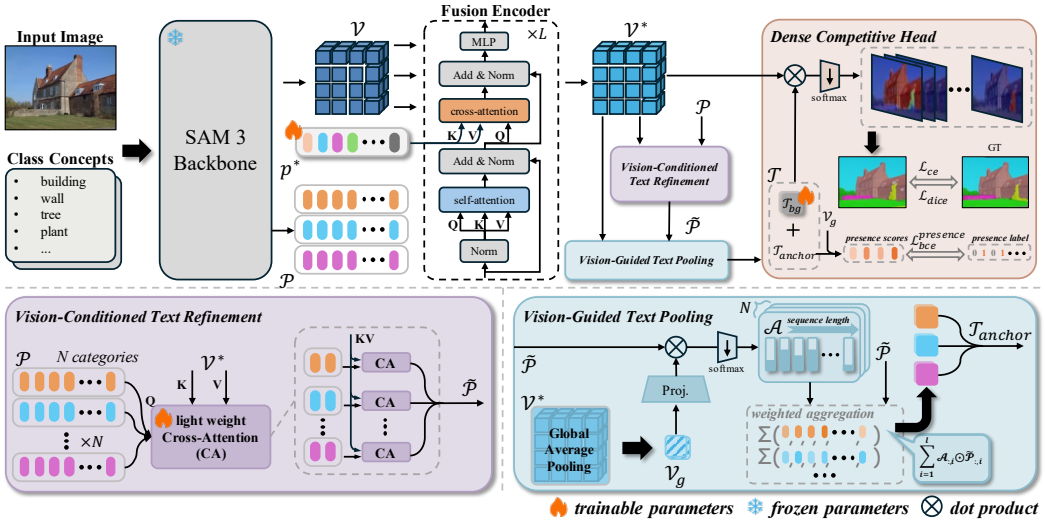


Figure 3: **Overview of our proposed DCR-SAM framework.** It transforms SAM3 from an independent decoding paradigm to a dense competitive paradigm. First, we alleviate the inference latency via Decoupled Visual Conditioning to obtain a globally contextualized visual representation. Then, we align text and visual features using the *VCTR* and *VGTP* modules. Finally, we design a competition head to enforce explicit inter-class competition and generate high-fidelity masks.

framework (§3.2). The overview of our method is illustrated in Figure 3. Finally, we introduce the joint optimization and the corresponding objective functions (§3.3).

3.1 Preliminary

Unlike dual-encoder architectures such as CLIP [45], SAM3 [9] employs an encoder-decoder architecture where the core modules comprise the vision encoder, the text encoder, the fusion encoder, and the semantic decoding head (for simplicity, we exclusively discuss the modules relevant to this work). Given an input image $I \in \mathbb{R}^{H\times W\times 3}$ and a target concept set comprising N categories $C = \{c_1, c_2, \ldots, c_N\}$, the network initially processes the image through a vision encoder to extract the unconditioned visual embeddings $\mathcal{V} \in \mathbb{R}^{hw\times d}$, where d denotes the embedding dimension, h and w depend on the patch resolution of the vision encoder. Simultaneously, each specific concept c_i is encoded into prompt tokens $p_i \in \mathbb{R}^{l\times d}$ via a text encoder, where l denotes the sequence length. The complete set of prompt tokens is denoted as $\mathcal{P} = \{p_1, p_2, \ldots, p_N\}$. Subsequently, to achieve concept-specific localization, the unconditioned visual embeddings must be dynamically conditioned on the prompt tokens. Following the native architecture of SAM3 [9], a fusion encoder E_{fusion} accepts the unconditioned visual embeddings $\mathcal{V}$ and conditions these embeddings on the prompt tokens p_i using a stack of Transformer [55] blocks equipped with self-attention and cross-attention layers. The output of fusion encoder constitutes the conditioned visual embeddings:

$$\mathcal{V}^i_{cond} = E_{fusion}(\mathcal{V}, p_i), \quad \mathcal{V}^i_{cond} \in \mathbb{R}^{hw\times d}. \tag{1}$$

Finally, these conditioned embeddings are fed into a semantic decoding head, which is adapted from the MaskFormer [15] architecture, to decode the binary semantic logits map $\mathcal{M}_i$ for the specific concept c_i:

$$\mathcal{M}_i = \text{SegHead}(V^i_{cond}), \quad \mathcal{M}_i \in \mathbb{R}^{h\times w}. \tag{2}$$

In a dense multi-class scenario, to obtain the comprehensive multi-class prediction $\mathcal{M}$, the independently generated binary semantic logit maps must first be aggregated. Subsequently, a standard softmax activation is applied to normalize the probabilities across categories, followed by an `argmax` operation to derive the final segmentation mask:

$$\mathcal{M} = \text{Concat}^N_{i=1}(\mathcal{M}_i), \quad \mathcal{M} \in \mathbb{R}^{N\times h\times w}. \tag{3}$$

Limitations This pipeline must execute the feature fusion and decoding processes independently for each category, which introduces a inference latency with a complexity of $\mathcal{O}(n)$. Furthermore, the independent prediction paradigm for each category inherently lacks inter-class competition, thereby provoking severe semantic misclassifications and overlapping masks. In contrast, by decoupling the cross-modal fusion process and constructing a dense competition head, our method transforms the native serial binary segmenter into a highly efficient competitive multi-class architecture.

3.2 Architecture of DCR-SAM

Decoupled Visual Conditioning To alleviate the prohibitive $\mathcal{O}(n)$ inference latency caused by the concept-specific fusion and decoding process, we design a decoupled visual conditioning strategy. Instead of repeatedly feeding the specific text prompt p_i into the computationally intensive fusion encoder (*cf.* Eq. 1), we introduce a unified learnable prompt sequence $p^* \in \mathbb{R}^{l\times d}$. By substituting the concept-specific tokens with p^*, We completely decouple the deep cross-modal feature fusion from specific prompt tokens. Consequently, the fusion encoder integrates the visual features with the unified prompt sequence through a single forward pass, yielding the joint representation $\mathcal{V}^*$:

$$\mathcal{V}^* = E_{fusion}(\mathcal{V}, p^*), \quad \mathcal{V}^* \in \mathbb{R}^{hw\times d}. \tag{4}$$

This modification adapts the native fusion encoder to extract high-quality, globally contextualized visual representations shared across all concepts. These representations serve as a universal visual foundation for all subsequent multi-class prediction tasks. Benefiting from this strictly category-agnostic feature fusion process, our approach fundamentally alleviates the inherent inference latency bottleneck of the native architecture.

Vision-Conditioned Text Refinement The native SAM3 [9] independent inference paradigm lacks inter-class competition, leading to severe semantic misclassification and overlapping masks among conceptually similar categories. To fundamentally resolve this issue, our DCR-SAM framework shifts from isolated decoding to a dense competitive paradigm, where textual features are explicitly aligned with globally contextualized visual representations to ensure robust cross-modal grounding. However, original textual features extracted by the text encoder are static and context-free, rendering direct alignment suboptimal. Therefore, we propose the *Vision-Conditioned Text Refinement (VCTR)* module, which achieves dynamic alignment by integrating image-specific visual cues into textual features. Specifically, we design a lightweight Cross-Attention (CA) module to refine the text representations. The text tokens p_i for each category serve as the *Query* (Q), while the globally contextualized visual features $\mathcal{V}^*$ act as the *Key* (K) and *Value* (V). Finally, we aggregate all category features to obtain the aligned textual feature set $\widetilde{\mathcal{P}} = \{\tilde{p}_1, \tilde{p}_2, \ldots, \tilde{p}_N\}$:

$$\tilde{p}_i = \mathrm{MHA}(\mathrm{LN}(p_i), \mathrm{LN}(\mathcal{V}^*), \mathrm{LN}(\mathcal{V}^*)) + p_i, \quad \text{where } \tilde{p}_i \in \mathbb{R}^{l\times d},\ \widetilde{\mathcal{P}} \in \mathbb{R}^{N\times l\times d}. \tag{5}$$

where MHA and LN denote the Multi-Head Attention and Layer Normalization, respectively. Notably, this lightweight module involves only simple matrix operations and a minimal number of additional parameters, incurring a negligible inference latency.

Vision-Guided Text Pooling For highly efficient dense matching, the aligned text sequence $\widetilde{\mathcal{P}}$ must be compressed into a global feature embedding (*anchor*) for each category. An intuitive approach applies average pooling across the sequence. However, this introduces redundant noise from less relevant content or padding tokens, compromising the discriminative capability of semantic anchors. Instead, we propose the *Vision-Guided Text Pooling (VGTP)* module. First, we apply Global Average Pooling (GAP) and a linear projection layer $\mathcal{F}_{proj}$ to the globally contextualized visual representation $\mathcal{V}^*$ to generate a global visual embedding $\mathcal{V}_g$:

$$\mathcal{V}_g = \mathcal{F}_{proj}(\mathrm{GAP}(\mathcal{V}^*)), \quad \mathcal{V}_g \in \mathbb{R}^{1\times d}. \tag{6}$$

Next, we compute the attention scores matrix $\mathcal{A}$ between visual embedding and the aligned text sequence. These scores facilitate a weighted aggregation along the sequence length l to generate the final global text anchors $\mathcal{T}_{anchor}$ for each category:

$$\mathcal{A} = \texttt{softmax}\left(\frac{\widetilde{\mathcal{P}}\,\mathcal{V}_g^T}{\sqrt{d}}\right), \quad \mathcal{A} \in \mathbb{R}^{N\times l}, \tag{7}$$

$$\mathcal{T}_{anchor} = \sum_{i=1}^{l} \mathcal{A}_{:,i} \odot \widetilde{\mathcal{P}}_{:,i}, \quad \mathcal{T}_{anchor} \in \mathbb{R}^{N\times d}. \tag{8}$$

Dynamic Background In multi-class segmentation tasks, pixels not belonging to predefined target categories are typically assigned to a background class. To adapt the SAM3 [9] to this formulation, a straightforward solution uses a generic text prompt, *e.g.*, background. However, this concept is overly broad and lacks explicit visual grounding. Alternatively, hard-coding background categories (*e.g.*, sky or cloud) severely degrades architectural universality, requires extensive dataset-specific engineering, and fails to exhaust all potential unknown distractors. To address these limitations, we introduce an independent, learnable background token $\mathcal{T}_{bg} \in \mathbb{R}^{1\times d}$ and concatenate it with the generated text anchors to form a complete semantic anchor set $\mathcal{T}$:

$$\mathcal{T} = \mathrm{Concat}(\mathcal{T}_{anchor}, \mathcal{T}_{bg}), \quad \mathcal{T} \in \mathbb{R}^{(N+1)\times d}. \tag{9}$$

Dense Competitive Head Equipped with the complete set of text anchors $\mathcal{T}$ and the globally contextualized visual representation $\mathcal{V}^*$, we discard the isolated decoding paradigm in favor of a dense competitive head. To ensure stable optimization and prevent feature magnitude discrepancies from dominating the logit computation, we apply L_2 normalization along the channel dimension and introduce a temperature scaling parameter τ prior to the dot product. Specifically, we compute the scaled similarity between the normalized dense visual features and the text anchors to generate the dense logits map $\mathcal{M}^*$:

$$\mathcal{M}^* = \frac{1}{\tau}\left(\frac{\mathcal{T}}{\|\mathcal{T}\|_2}\right)\left(\frac{\mathcal{V}^*}{\|\mathcal{V}^*\|_2}\right)^T, \quad \mathcal{M}^* \in \mathbb{R}^{(N+1)\times hw}. \tag{10}$$

3.3 Joint Optimization for DCR-SAM

Global Presence Score While the dense competitive head suppresses semantic misclassification, relying solely on local dense features can occasionally lead to false positive predictions due to localized visual similarities. To provide a robust constraint, we follow the training strategy of SAM3 [9] and introduce a global presence score during the training phase. Specifically, we compute the dot product between the global visual embedding $\mathcal{V}_g$ and the set of target semantic anchors $\mathcal{T}_{anchor}$, followed by a `sigmoid` operation, to generate the image-level presence scores $S_{presence}$:

$$S_{presence} = \texttt{sigmoid}(\mathcal{T}_{anchor}\,\mathcal{V}_g^T), \quad S_{presence} \in \mathbb{R}^{N\times 1}. \tag{11}$$

Joint Objective To holistically optimize the proposed model, we formulate a joint end-to-end training objective. We employ the cross-entropy loss and the dice loss to supervise the final mask, and utilize the binary cross-entropy loss to supervise the global presence score, and λ_i denotes the weight to rescale the loss terms:

$$\mathcal{L}_{total} = \lambda_1\mathcal{L}_{ce} + \lambda_2\mathcal{L}_{dice} + \lambda_3\mathcal{L}_{bce}^{presence}. \tag{12}$$

4 Experiments

4.1 Experimental Setup

Datasets and Evaluation Metric We evaluate our DCR-SAM on four highly challenging natural-scene benchmarks: **ADE20K** [70], which encompasses a dense label space of 150 classes, **Cityscapes** [17], which contains 19 classes tailored for urban driving scenarios, COCO-Stuff-164K [7] (**COCO-Stuff**), which features a vast and diverse label space of 171 classes, and PASCAL-Context59 [40] (**Context59**), which provides dense annotations for 59 prevalent categories. To demonstrate cross-domain robustness, we extend our evaluation to four representative remote sensing benchmarks: **LoveDA** [56] with 7 classes, ISPRS **Potsdam** [48] with 6 classes, ISPRS **Vaihingen** [39] with 6 classes, and **iSAID** [58] with 16 classes. Unless stated otherwise, all models are trained on their respective training sets, and the results are reported on the official validation sets. We employ the standard mean Intersection over Union (mIoU) as our primary metric.

Baselines We compare DCR-SAM with state-of-the-art semantic segmentation methods, including InternImage-H [57], EoMT [25], and SegMAN-L [19]. For SAM3 [9], we adopt its complete official architecture, retaining all core components, including the detector and presence prediction modules, and establish two baselines: ① *Native-Binary* (**SAM3**†), which follows the official training

Table 1: **Comparison with state-of-the-art semantic segmentation methods** on natural scene and remote sensing, with the best results in **bold**, the second-best underlined, and all results evaluated on the validation set. † denotes *Native-Binary* paradigm and ‡ denotes *Native-Multi-Class* paradigm.

Methods	Pub. & Year	Backbone	*Train on Natural Scene*			
			ADE20K	Cityscapes	COCO-Stuff	Context59
DeepLabV3 [11]	ECCV'18	ResNet-101	42.6	77.1	37.9	52.0
DeepLabV3+ [12]	ECCV'18	ResNet-101	44.5	73.6	–	53.2
HiPoSeg [32]	ICLR'26	ResNet-101	49.0	84.0	–	–
HRNet [51]	arXiv'19	HRNetV2-W48	41.9	78.5	36.0	51.9
OCRNet [62]	ECCV'20	HRNetV2-W48	41.5	80.0	39.0	54.4
OCRNet [62]	ECCV'20	D-ResNet-101	45.3	81.8	–	54.8
SETR-PUP [68]	CVPR'21	ViT-L	50.1	82.2	–	55.3
EoMT [25]	CVPR'25	ViT-L	58.4	84.2	–	–
DTop [52]	ICCV'23	ViT-L	52.8	–	47.1	62.7
SegViT [66]	NeurIPS'22	ViT-L	53.3	–	47.4	63.0
ECENet [34]	ICCV'23	Swin-L	55.2	84.5	–	65.9
CPT [53]	TIP'25	Swin-L	58.5	84.6	–	65.6
SegMAN-L [19]	CVPR'25	LASS	53.2	84.2	–	–
SegNeXt-T [21]	NeurIPS'22	MSCAN-T	41.1	79.8	38.7	51.2
InternImage-H [57]	CVPR'23	DCNv3	60.3	86.1	–	70.3
SAM3† [9]	ICLR'26	SAM3 (PE-L+/14)	46.5	75.4	48.1	66.3
SAM3‡ [9]	ICLR'26	SAM3 (PE-L+/14)	41.2	72.6	45.4	64.3
DCR-SAM (Ours)	–	SAM3 (PE-L+/14)	**68.7**↑8.4	**89.1**↑3.0	**50.8**↑2.7	**72.6**↑2.3

Methods	Pub. & Year	Backbone	*Train on Remote Sensing*			
			LoveDA	Potsdam	Vaihingen	iSAID
PSPNet [67]	CVPR'17	ResNet-50	48.3	82.0	76.8	60.2
DeepLabV3+ [12]	ECCV'18	ResNet-50	47.6	84.2	77.1	61.9
DANet [18]	CVPR'19	ResNet-50	50.2	81.4	77.3	60.0
SemanticFPN [26]	CVPR'19	ResNet-50	48.2	84.6	77.9	59.3
FarSeg [69]	CVPR'20	ResNet-50	48.2	84.3	79.1	63.7
LOGCAN++ [38]	TGRS'25	ResNet-50	53.4	87.6	83.9	65.0
SCSM [37]	ISPRS'25	ResNet-50	54.6	87.8	84.7	66.9
Segmenter [50]	ICCV'21	ViT-L	47.1	86.5	79.4	–
EfficientViT [8]	ICCV'23	EfficientViT-L2	47.1	84.2	80.5	–
DDP [23]	ICCV'23	Swin-T	52.5	87.4	82.6	–
SAM3† [9]	ICLR'26	SAM3 (PE-L+/14)	55.8	66.4	69.8	61.1
SAM3‡ [9]	ICLR'26	SAM3 (PE-L+/14)	50.4	61.3	65.7	55.4
DCR-SAM (Ours)	–	SAM3 (PE-L+/14)	**66.5**↑10.7	**90.2**↑2.4	**89.2**↑4.5	**75.4**↑8.5

strategy to independently predict binary masks for each category; and ② *Native-Multi-Class* (**SAM3**‡), which concatenates category-wise predictions and is fine-tuned under a standard multi-class training paradigm. Following established SAM3 adaptation practices, semantic masks from both the semantic and detector heads are fused to obtain the final segmentation results. Notably, the latter configuration lacks explicit inter-class competition, potentially leading to semantic overlap among related categories.

Implementation Details The proposed DCR-SAM is implemented in the `PyTorch` framework [43]. All models are initialized with the official pre-trained weights of SAM3 [9]. For the concept set, we directly utilize the original category names provided by the respective datasets. The length of the unified learnable prompt sequence p^* is set to 32. The scaling parameter τ is set to 0.07. The scaling weights (λ_1, λ_2, and λ_3) of the loss functions are set to 1 by default. The dynamic background strategy is optional, it can be omitted when all pixels in the evaluation dataset belong to the target categories. During the training phase, we freeze the original weights of SAM3 backbone and only fine-tune the newly introduced parameters: the learnable prompt sequence, the lightweight cross-modal attention module, and the learnable background token. The network is optimized end-to-end using the AdamW optimizer [36] with an initial learning rate of 1×10^{-4} and a weight decay of 5×10^{-4}. The batch size is set to 8, and all models are trained for $80,000$ iterations. All experiments are conducted on 8 NVIDIA RTX 4090 GPUs. Further details are provided in the Appendix A.1.

4.2 Overall Performance

Quantitative Comparisons As shown in Table 1, our method demonstrates exceptional performance across various benchmarks. In natural scene datasets, the proposed method significantly outperforms previous SOTA approaches, yielding substantial improvements ranging from **2.3%** $\sim$ **8.4%**. Even compared with InternImage-H [57], a vision foundation model built upon deformable

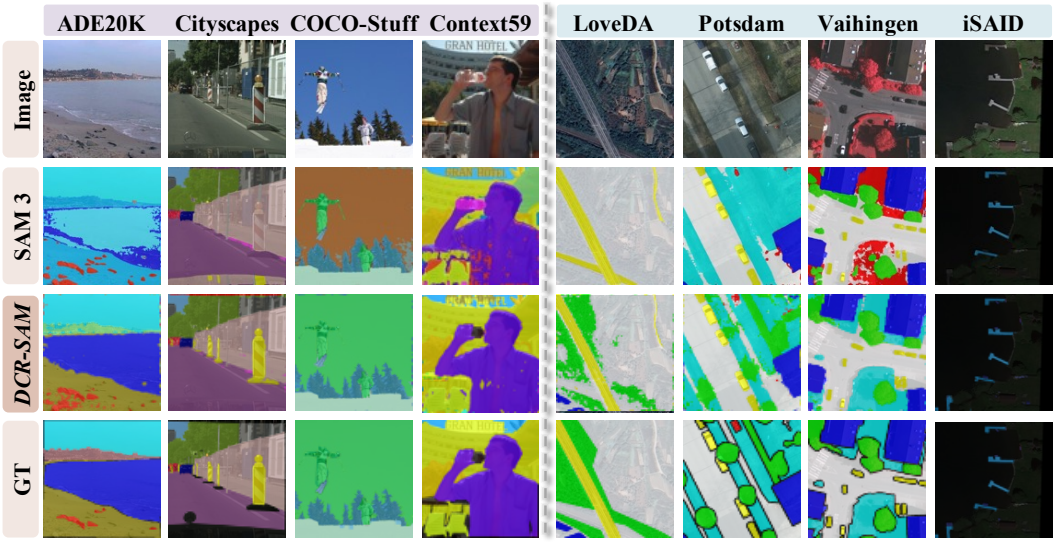


Figure 4: **Qualitative comparison results** on natural scene and remote sensing. We use the SAM3† to present the visualization results of the second row.

convolutions, our method consistently achieves superior results. In cross-domain remote sensing scenarios, it continues to demonstrate substantial superiority, achieving improvements of approximately **2.4%** $\sim$ **10.7%**. Furthermore, it significantly surpasses specialized methods designed for remote sensing, such as LOGCAN++ [38] and SCSM [37], verifying its robust scene adaptability. Notably, the performance gains achieved in remote sensing are generally more pronounced than those in natural scenes. We attribute this phenomenon to the significantly smaller concept space of remote sensing imagery compared to natural scenes, enabling the dense competition head to establish more stable semantic anchors.

Moreover, the comparison with the two SAM3 adaptation baselines demonstrates that both training paradigms fail to achieve optimal results without modifying the core network architecture. Notably, the performance of the *Native-Multi-Class* paradigm (SAM3‡) is inferior to that of the original *Native-Binary* paradigm (SAM3†). This performance degradation indicates that merely adjusting the training strategy fails to compensate for the inherent architectural defects of the model in multi-class segmentation tasks. More experimental analysis in the Appendix B.1.

Qualitative Results We visualize the segmentation results across both natural scene and remote sensing benchmarks in Figure 4. SAM3 exhibits limitations such as semantic fragmentation, noisy artifacts (*e.g.*, on the person in Context59), and regional misclassifications (*e.g.*, omitting vegetation in LoveDA) in complex semantic contexts. Conversely, our DCR-SAM preserves the high-fidelity spatial delineation inherent to SAM3 while demonstrating significant improvements in semantic discriminability.

4.3 Further analysis

Component Ablation To validate the effectiveness of the proposed modules, we conduct an ablation analysis on the ADE20K and LoveDA benchmarks. Table 2 reports the per-image inference time (ms) and the mIoU on both benchmarks. The proposed method achieves a **33**$\times$ inference speedup on the ADE20K benchmark, decreasing the inference time from **8038 ms** to **244 ms** alongside substantial performance gains, which demonstrates the necessity of decoupling the cross-modal feature fusion. Furthermore, the text refinement results indicate that incorporating the *VCTR* module yields significant improvements with only a marginal increase in inference latency, extending the inference time from **224 ms** to **244 ms** on the ADE20K dataset. Regarding the pooling strategy, the proposed *VGTP* module significantly outperforms conventional approaches such as first, last, and mean pooling, confirming that dynamic compression constructs highly discriminative textual anchors. For the background category, it can be observed that the proposed dynamic background strategy

Table 2: **Ablation analysis of core components.** We use the SAM3† as our baseline model.

Components	ADE20K		LoveDA	
	Time.↓	mIoU	Time.↓	mIoU
baseline	8038	46.5	647	55.8
(a) text refinement				
DCR-SAM w/o *VCTR*	224	61.4	179	60.9
DCR-SAM w/ *VCTR*	244	**68.7**	183	**66.5**
(b) pooling strategy				
First	244	63.1	183	61.8
Last	244	62.8	183	60.2
Mean	244	66.5	183	63.2
VGTP.	244	**68.7**	183	**66.5**
(c) background strategy				
generic text	–	–	183	59.5
hard-coding	–	–	193	64.1
dynamic background	–	–	183	**66.5**

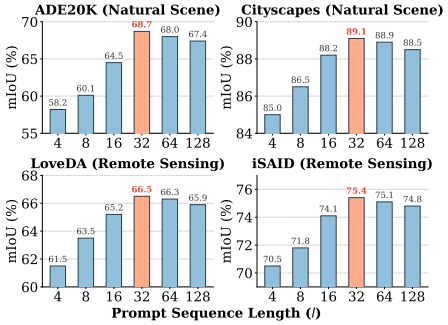


Figure 5: **Analysis of learnable prompt sequence length.** We select six different lengths ($l \in \{4, 8, 16, 32, 64, 128\}$) and report the mIoU on four benchmarks. Best results are highlighted in orange.

($\mathcal{T}_{bg}$) achieves optimal performance. Although expanding the number of hard-coding background categories can further improve the performance, this strategy requires extensive prompt engineering and manual adaptations specific to the dataset, thereby significantly limiting the general applicability of the model. Additional ablation analyses are detailed in Appendix B.1.

Impact of Learnable Prompt Sequence Length We investigate the impact of the learnable prompt sequence length ($l \in \{4, 8, 16, 32, 64, 128\}$) on the final segmentation performance, as illustrated in Figure 5. The performance variations exhibit a highly consistent trend across all four evaluation benchmarks, uniformly achieving optimal results when l is set to 32. When the sequence length is highly restricted ($l \leq 8$), the model lacks the representational capacity to capture complex semantic variations, resulting in suboptimal segmentation performance. Conversely, the performance degrades when the sequence length exceeds 32. This degradation is likely attributable to redundant parameters that introduce noise and increase the risk of overfitting. Therefore, we apply a unified sequence length across all benchmarks in this work.

Analysis of Objective Function We evaluate the individual contributions of the three loss components ($\mathcal{L}_{ce}$, $\mathcal{L}_{dice}$, and $\mathcal{L}_{bce}^{presence}$). As shown in Table 3, the results indicate that $\mathcal{L}_{ce}$ and $\mathcal{L}_{dice}$ are highly complementary, and their joint application yields substantial improvements. Ultimately, the incorporation of the binary cross-entropy loss ($\mathcal{L}_{bce}^{presence}$) for the prediction of category presence achieves peak mIoU scores of 68.7 and 66.5 on the two benchmarks, respectively. This demonstrates the critical role of this component in mitigating false positives. In addition, Figure 6 visualizes the training trajectories, which illustrate a highly stable and synchronous convergence process across both domains, thereby validating the robustness of the joint optimization strategy.

Comparison with SAM 3-Based Methods We further compare DCR-SAM with four recent SAM 3-based methods, including SegEarth-OV3 [31], CoCo-SAM3 [14], EfficientSAM3 [63], and SAM3-Adapter [13]. As shown in Table 4, DCR-SAM consistently outperforms these methods, achieving 68.7 mIoU on ADE20K and 66.5 mIoU on LoveDA, surpassing the strongest baselines by 21.5% and 13.6%, respectively. While SegEarth-OV3 and CoCo-SAM3 adopt training-free strategies, EfficientSAM3 and SAM3-Adapter pri-

Table 3: **Analysis of loss functions.** We report the results on the ADE20K and LoveDA datasets.

$\mathcal{L}_{ce}$	$\mathcal{L}_{dice}$	$\mathcal{L}_{bce}^{presence}$	ADE20K	LoveDA
✓	✗	✗	66.7	64.0
✓	✓	✗	67.3	64.9
✓	✓	✓	**68.7**	**66.5**

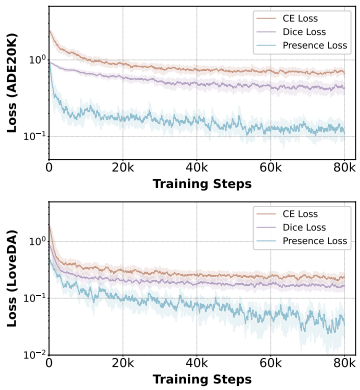


Figure 6: **Loss curves** of three loss functions.

Table 4: **Comparison with SAM 3-based methods** on ADE20K and LoveDA (mIoU, %).

Methods	ADE20K	LoveDA
SegEarth-OV3	36.6	47.4
CoCo-SAM3	38.3	–
EfficientSAM3	43.2	50.3
SAM3-Adapter	47.2	52.9
DCR-SAM	**68.7**	**66.5**

Table 5: **Comparison** of different segmentation architectures using a unified PE-L+/14 backbone on ADE20K (mIoU, %).

Methods	Backbone	mIoU
UPerNet	PE-L+/14	59.6
EoMT-Style	PE-L+/14	52.1
Mask2Former	PE-L+/14	61.3
DCR-SAM	**PE-L+/14**	**68.7**

marily focus on efficient adaptation without explicitly addressing inter-class semantic conflicts. In contrast, DCR-SAM introduces a dense competitive head to enforce explicit inter-class competition, effectively mitigating semantic overlap and improving multi-class segmentation performance.

Comparison under a Unified Perception Encoder Backbone. To further disentangle the contributions of backbone representations and segmentation architecture, we conduct a controlled comparison with representative segmentation heads, including UPerNet [60], Mask2Former [16], and EoMT-Style [25], using the same PE-L+/14 backbone [4]. This setting ensures a consistent visual feature extractor across all methods, allowing us to assess the effectiveness of different segmentation architectures under shared backbone representations. As shown in Table 5, DCR-SAM achieves 68.7 mIoU on ADE20K, outperforming UPerNet, EoMT-Style, and Mask2Former by 9.1%, 16.6%, and 7.4%, respectively. These improvements suggest that the effectiveness of DCR-SAM cannot be attributed solely to the powerful pretrained backbone. Notably, the Perception Encoder is pretrained with an emphasis on open-vocabulary visual representations, which may favor architectures designed to exploit such representations over conventional closed-set segmentation heads. Unlike these methods, DCR-SAM preserves the native promptable architecture of SAM 3 while introducing explicit inter-class competition to resolve semantic ambiguities among candidate categories. This design enables more effective utilization of pretrained representations for dense multi-class segmentation, highlighting the importance of architectural compatibility and competitive semantic reasoning.

5 Conclusion

In this paper, we propose the **D**ense **C**ompetitive **R**epresentations **SAM** (**DCR-SAM**) framework to mitigate the inference latency and inherent semantic misclassifications encountered during the adaptation of SAM3 to multi-class segmentation. The proposed method decouples the feature fusion process to ensure optimal inference efficiency. Furthermore, the alignment of textual and visual representations within a dense competitive paradigm effectively suppresses overlapping predictions among conceptually similar categories. Extensive experiments demonstrate that the framework achieves SOTA performance across diverse benchmarks while delivering a 33× speedup. We hope our work paves the way for adapting foundation models to complex, real-world multi-class segmentation tasks.

Limitations DCR-SAM reduces text prompt sensitivity and inference latency, facilitating its potential application in autonomous systems and medical imaging. However, its reliance on large-scale foundation models incurs substantial computational costs and may inherit biases from pretraining data. Moreover, inter-class competition may impair the recognition of unseen concepts. Future work will focus on improving computational efficiency and generalization in open-vocabulary segmentation.

Acknowledgment

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDA0450000) and the National Natural Science Foundation of China (62372433).

References

[1] Nikita Araslanov and Stefan Roth. Single-stage semantic segmentation from image labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4253–4262, 2020.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

[3] Roni Blushtein-Livnon, Osher Rafaeli, David Ioffe, Amir Boger, Karen Sandberg Esquenazi, and Tal Svoray. On the effectiveness of textual prompting with lightweight fine-tuning for sam3 remote sensing segmentation. arXiv preprint arXiv:2512.15564, 2025.

[4] Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Bangalath, et al. Perception encoder: The best visual embeddings are not at the output of the network. Advances in Neural Information Processing Systems, 38:60884–60937, 2026.

[5] Sébastien Bubeck, Christian Coester, Ronen Eldan, Timothy Gowers, Yin Tat Lee, Alexandru Lupsasca, Mehtaab Sawhney, Robert Scherrer, Mark Sellke, Brian K Spears, et al. Early science acceleration experiments with gpt-5. arXiv preprint arXiv:2511.16072, 2025.

[6] Valay Bundele, Mehran Hosseinzadeh, and Hendrik Lensch. Memory-enhanced sam3 for occlusion-robust surgical instrument segmentation. In Medical Imaging with Deep Learning, 2025.

[7] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1209–1218, 2018.

[8] Han Cai, Junyan Li, Muyan Hu, Chuang Gan, and Song Han. Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction. In Proceedings of the IEEE/CVF international conference on computer vision, pages 17302–17313, 2023.

[9] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719, 2025.

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In Proceedings of the IEEE/CVF international conference on computer vision, pages 9650–9660, 2021.

[11] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587, 2017.

[12] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European conference on computer vision (ECCV), pages 801–818, 2018.

[13] Tianrun Chen, Runlong Cao, Xinda Yu, Lanyun Zhu, Chaotao Ding, Deyi Ji, Cheng Chen, Qi Zhu, Chunyan Xu, Papa Mao, et al. Sam3-adapter: Efficient adaptation of segment anything 3 for camouflage object segmentation, shadow detection, and medical image segmentation. arXiv preprint arXiv:2511.19425, 2025.

[14] Yanhui Chen, Baoyao Yang, Siqi Liu, and Jingchao Wang. Coco-sam3: Harnessing concept conflict in open-vocabulary semantic segmentation. arXiv preprint arXiv:2604.19648, 2026.

[15] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. Advances in neural information processing systems, 34: 17864–17875, 2021.

[16] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 1290–1299, 2022.

[17] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3213–3223, 2016.

[18] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 3146–3154, 2019.

[19] Yunxiang Fu, Meng Lou, and Yizhou Yu. Segman: Omni-scale context modeling with state space models and local attention for semantic segmentation. In Proceedings of the Computer Vision and Pattern Recognition Conference, pages 19077–19087, 2025.

[20] Jingda Guo, Xu Ma, Andrew Sansom, Mara McGuire, Andrew Kalaani, Qi Chen, Sihai Tang, Qing Yang, and Song Fu. Spanet: Spatial pyramid attention network for enhanced image recognition. In 2020 IEEE International Conference on Multimedia and Expo (ICME), 2020.

[21] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. Advances in neural information processing systems, 35:1140–1156, 2022.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[23] Yuanfeng Ji, Zhe Chen, Enze Xie, Lanqing Hong, Xihui Liu, Zhaoqiang Liu, Tong Lu, Zhenguo Li, and Ping Luo. Ddp: Diffusion model for dense visual prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 21741–21752, 2023.

[24] 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 International conference on machine learning, pages 4904–4916. PMLR, 2021.

[25] Tommie Kerssies, Niccolo Cavagnero, Alexander Hermans, Narges Norouzi, Giuseppe Averta, Bastian Leibe, Gijs Dubbelman, and Daan De Geus. Your vit is secretly an image segmentation model. In Proceedings of the computer vision and pattern recognition conference, pages 25303–25313, 2025.

[26] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6399–6408, 2019.

[27] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In Proceedings of the IEEE/CVF international conference on computer vision, pages 4015–4026, 2023.

[28] Earl J Kirkland. Bilinear interpolation. In Advanced computing in electron microscopy, pages 261–263. Springer, 2010.

[29] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. Advances in neural information processing systems, 24, 2011.

[30] Jingjing Li, Yue Feng, Yuchen Guo, Jincai Huang, Yongri Piao, Qi Bi, Miao Zhang, Xiaoqi Zhao, Qiang Chen, Shihao Zou, et al. Sam3-i: Segment anything with instructions. arXiv preprint arXiv:2512.04585, 2025.

[31] Kaiyu Li, Shengqi Zhang, Yupeng Deng, Zhi Wang, Deyu Meng, and Xiangyong Cao. Segearth-ov3: Exploring sam 3 for open-vocabulary semantic segmentation in remote sensing images. arXiv preprint arXiv:2512.08730, 2025.

[32] Seoha Lim, Jinmyeong Kim, Jieun Kim, and Sung-Bae Cho. Hierarchical prototype learning for semantic segmentation. In The Fourteenth International Conference on Learning Representations, 2025.

[33] Yida Lin, Bing Xue, Mengjie Zhang, Sam Schofield, and Richard Green. Progressive per-branch depth optimization for defom-stereo and sam3 joint analysis in uav forestry applications. arXiv preprint arXiv:2602.20539, 2026.

[34] Yuhe Liu, Chuanjian Liu, Kai Han, Quan Tang, and Zengchang Qin. Boosting semantic segmentation from the perspective of explicit class embeddings. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 821–831, 2023.

[35] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440, 2015.

[36] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

[37] Xiaowen Ma, Rongrong Lian, Zhenkai Wu, Renxiang Guan, Tingfeng Hong, Mengjiao Zhao, Mengting Ma, Jiangtao Nie, Zhenhong Du, Siyang Song, et al. A novel scene coupling semantic mask network for remote sensing image segmentation. ISPRS Journal of Photogrammetry and Remote Sensing, 221:44–63, 2025.

[38] Xiaowen Ma, Rongrong Lian, Zhenkai Wu, Hongbo Guo, Fan Yang, Mengting Ma, Sensen Wu, Zhenhong Du, Wei Zhang, and Siyang Song. Logcan++: Adaptive local-global class-aware network for semantic segmentation of remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 2025.

[39] ITC Markus Gerke. Use of the stair vision library within the isprs 2d semantic labeling benchmark (vaihingen), 2014.

[40] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 891–898, 2014.

[41] Diederick C Niehorster and Marcus Nyström. Eye image segmentation using visual and concept prompts with segment anything model 3 (sam3). arXiv preprint arXiv:2603.17715, 2026.

[42] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193, 2023.

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems, 32, 2019.

[44] Gensheng Pei, Xiruo Jiang, Yazhou Yao, Xiangbo Shu, Fumin Shen, and Byeungwoo Jeon. Taming sam3 in the wild: A concept bank for open-vocabulary segmentation. arXiv preprint arXiv:2602.06333, 2026.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pages 8748–8763. PmLR, 2021.

[46] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714, 2024.

[47] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[48] Franz Rottensteiner, Gunho Sohn, Markus Gerke, and Jan D Wegner. Isprs semantic labeling contest. ISPRS: Leopoldshöhe, Germany, 1(4):4, 2014.

[49] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. arXiv preprint arXiv:2508.10104, 2025.

[50] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 7262–7272, 2021.

[51] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. arXiv preprint arXiv:1904.04514, 2019.

[52] Quan Tang, Bowen Zhang, Jiajun Liu, Fagui Liu, and Yifan Liu. Dynamic token pruning in plain vision transformers for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 777–786, 2023.

[53] Quan Tang, Chuanjian Liu, Fagui Liu, Jun Jiang, Bowen Zhang, CL Philip Chen, Kai Han, and Yunhe Wang. Rethinking feature reconstruction via category prototype in semantic segmentation. IEEE Transactions on Image Processing, 34:1036–1047, 2025.

[54] Zhenran Tang, Rohan Nagabhirava, and Changliu Liu. Cad-prompted sam3: Geometry-conditioned instance segmentation for industrial objects. arXiv preprint arXiv:2602.20551, 2026.

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

[56] Junjue Wang, Zhuo Zheng, Ailong Ma, Xiaoyan Lu, and Yanfei Zhong. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. arXiv preprint arXiv:2110.08733, 2021.

[57] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 14408–14419, 2023.

[58] Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan, Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. isaid: A large-scale dataset for instance segmentation in aerial images. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pages 28–37, 2019.

[59] Monika Wysoczańska, Antonin Vobecky, Amaia Cardiel, Tomasz Trzciński, Renaud Marlet, Andrei Bursuc, and Oriane Siméoni. Test-time contrastive concepts for open-world semantic segmentation with vision-language models. arXiv preprint arXiv:2407.05061, 2024.

[60] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In European conference on computer vision, pages 432–448. Springer, 2018.

[61] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems, 34:12077–12090, 2021.

[62] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In European conference on computer vision, pages 173–190. Springer, 2020.

[63] Chengxi Zeng, Yuxuan Jiang, and Aaron Zhang. Efficientsam3: Progressive hierarchical distillation for video concept segmentation from sam1, 2, and 3. arXiv preprint arXiv:2511.15833, 2025.

[64] Chengxi Zeng, Yuxuan Jiang, Ge Gao, Shuai Wang, Duolikun Danier, Bin Zhu, Stevan Rudinac, David Bull, and Fan Zhang. Sam3-litetext: An anatomical study of the sam3 text encoder for efficient vision-language segmentation. arXiv preprint arXiv:2602.12173, 2026.

[65] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In Proceedings of the IEEE/CVF international conference on computer vision, pages 11975–11986, 2023.

[66] Bowen Zhang, Zhi Tian, Quan Tang, Xiangxiang Chu, Xiaolin Wei, Chunhua Shen, et al. Segvit: Semantic segmentation with plain vision transformers. Advances in Neural Information Processing Systems, 35:4971–4982, 2022.

[67] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2881–2890, 2017.

[68] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 6881–6890, 2021.

[69] Zhuo Zheng, Yanfei Zhong, Junjue Wang, and Ailong Ma. Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4096–4105, 2020.

[70] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 633–641, 2017.

[71] Hao Zhu, Shuo Jin, Wenbin Liao, Jiayu Xiao, Yan Zhu, Siyue Yu, and Feng Dai. Vip: Visual-guided prompt evolution for efficient dense vision-language inference. arXiv preprint arXiv:2605.12325, 2026.

[72] Hao Zhu, Heyu Sun, Shuo Jin, Jiayu Xiao, and Feng Dai. Tag: Calibrating sam3 for universal image segmentation without dense supervision. arXiv preprint arXiv:2610.14651, 2026.

A Implementation and Architectural Details **16**
A.1 Detailed Implementation Setup 16
A.2 Expanded Architectural Details of DCR-SAM 17

B Additional Experimental Analysis and Discussion **18**
B.1 Further Experimental Analysis 18

C Expanded Qualitative Analysis **20**
C.1 Cross-Modal Activation Map Visualizations 20
C.2 Extended Qualitative Comparisons 20

A Implementation and Architectural Details

A.1 Detailed Implementation Setup

Training and Evaluation Protocols During the training phase, we strictly adhere to the data processing protocols established for the SAM3 [9] model. To maintain batch consistency across varying image scales, all input images are randomly resized to dimensions within the range of 480 to 1008 pixels utilizing a stride of 32, and subsequently padded to a unified spatial resolution of 1008×1008. In addition, standard data augmentation techniques, including random horizontal flipping and color jittering, are uniformly applied. During the inference phase, all images are processed using the standard 1008×1008 input resolution natively required by the SAM3 architecture. Furthermore, to ensure fairness in the evaluation, all experimental results are derived directly from the raw outputs of the model without the application of any post-processing techniques (*e.g.*, PAMR [1] or DenseCRF [29]).

Measurement of Inference Latency For a fair and rigorous evaluation of inference efficiency as reported in Figure 2 and Table 2, all inference latency measurements are conducted on a single NVIDIA RTX 4090 GPU. We report the per-image processing time configured with a batch size of 1. To eliminate hardware initialization and memory allocation overhead, we perform 50 warm-up iterations prior to measurement, and the final latency is calculated as the average execution time of the subsequent forward passes.

Configurations for SAM3 Baselines To thoroughly evaluate our method, we fine-tune the pre-trained SAM3 model under two distinct paradigms. As shown in Figure 7, for both baselines, the image encoder and the text encoder are strictly frozen to preserve broad generic priors, while only the fusion encoder and the segmentation heads are fully optimized. For foreground categories, we directly utilize the original class names provided by the dataset, whereas a generic text prompt, *i.e.*, "background", is employed to represent the background category. The detailed settings are as follows:

I) *Native-Binary* (**SAM3**†): This baseline strictly adheres to the official strategy of SAM3. The fusion encoder processes each text prompt independently in a serial $\mathcal{O}(n)$ manner. The model is optimized using isolated binary prediction heads, supervised by a combination of the binary cross-entropy loss, dice loss, and presence loss for each individual concept. To construct the training data for this binary paradigm, we decouple the dense ground-truth annotations into independent binary masks based on the target categories present in the current image. Furthermore, we randomly introduce negative samples (*i.e.*, unrepresented categories with completely empty ground-truth masks) during training to enhance the model's semantic discriminability and effectively suppress false positive predictions.

II) *Native-Multi-Class* (**SAM3**‡): To adapt the native SAM3 architecture for a conventional multi-class setting, we concatenate the independently generated logits along the category dimension. A standard `softmax` activation is then applied to explicitly calibrate probabilities across the N classes, followed by optimization via the cross-entropy loss and dice loss. To prevent Out-of-Memory (OOM)

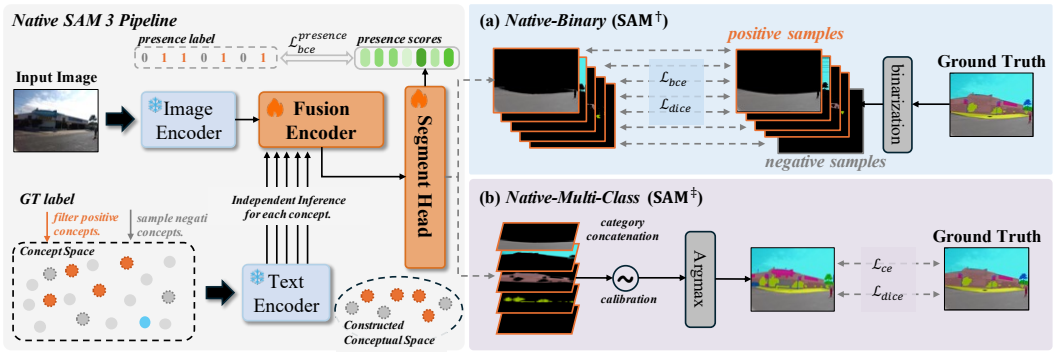


Figure 7: **Overview of two SAM3 baseline models for multi-class semantic segmentation**. We detail the construction of the concept space and the differences in training settings between the two baselines.

errors caused by processing an excessive number of categories simultaneously, we employ a dynamic prompt sampling strategy during training: we sample all ground-truth categories present in the current scene alongside a dynamically varying number of negative categories, ensuring a stable memory footprint per forward pass.

Hard-Coding Background Strategy Details In the component ablation study presented in Table 2, we evaluated a hard-coding background strategy as a comparative baseline against our proposed dynamic background token ($\mathcal{T}_{bg}$). To implement this strategy without extensive manual annotation, we employed large language models (LLMs) to automatically generate a comprehensive, dataset-specific list of negative concepts (*e.g.*, "sky", "road", and "cloud") that frequently appear in scenes but belong to the background class. Specifically, we utilized GPT-5 [5] to generate K diverse background concepts tailored to the domain of each benchmark. To prevent excessive semantic overlap with existing foreground categories, we applied the filtering pipeline of previous work [59] to these background concepts. First, we discarded visually uninformative and abstract concepts, such as "image" and "view". Subsequently, we used the CLIP [45] model to compute the semantic similarity between the generated background concepts and the predefined foreground categories. Any background concept exhibiting a similarity score exceeding a threshold of 0.8 was removed. The specific prompts used are detailed below:

> You are an expert in computer vision and semantic segmentation. I need you to generate {K} background or negative concept words that typically appear within the scenes of {dataset name} dataset:
> {dataset scene type, foreground categories, and additional related information}.
> These concepts should encompass a wide range of background elements, environmental descriptors, or non-target objects to ensure high diversity.
> examples: "asphalt road", "cloudy sky", "dense forest" and "bare soil".
> All generated terms must be in lowercase English. You are required to output exclusively a Python list containing {K} strings, formatted similarly to ["sky", "road", "water"], and to omit any additional explanations or code block markers.

A.2 Expanded Architectural Details of DCR-SAM

To complement the methodological overview (§3.2) in of DCR-SAM, we detail the micro-architectural designs and precise mathematical formulations of our proposed modules, ensuring complete structural transparency.

Vision-Conditioned Text Refinement (VCTR) In practice, the *Vision-Conditioned Text Refinement (VCTR)* module is implemented using the standard `nn.MultiheadAttention` function provided by the `PyTorch` framework. Specifically, the text tokens $p_i \in \mathbb{R}^{l \times d}$ serves as the *Query* (Q), while the globally contextualized visual representation $\mathcal{V}^* \in \mathbb{R}^{hw \times d}$ acts simultaneously as both the *Key* (K) and the *Value* (V). To ensure optimization stability, this process is naturally accompanied by standard

Layer Normalization [2] (LN) and residual connections [22]. The entire module relies exclusively on highly optimized matrix operations, thereby introducing minimal computational latency to the overall inference pipeline (please see Table 2 for more details).

Vision-Guided Text Pooling (VGTP) Regarding the dimensional broadcasting in Eq. 7 and Eq. 8, the global visual embedding $\mathcal{V}_g \in \mathbb{R}^{1\times d}$ is implicitly broadcasted along the category dimension N to match the shape of the aligned text sequences. Crucially, the attention computation is strictly performed per class rather than being shared. The dot product between $\mathcal{V}_g$ and $\tilde{\mathcal{P}}$ yields an attention matrix $\mathcal{A} \in \mathbb{R}^{N\times l}$, where the softmax activation normalizes the weights strictly along the sequence length l for each specific category. This ensures that the aggregated text anchors $\mathcal{T}_{anchor}$ capture class-specific discriminative cues without inter-class sequence interference.

Dense Competitive Head After obtaining the dense logits map $\mathcal{M}^* \in \mathbb{R}^{(N+1)\times hw}$, we subsequently reshape it back to the 2D spatial dimensions of $\mathbb{R}^{(N+1)\times h\times w}$. A bilinear upsampling algorithm [28] is then applied to align the resolution with that of the original input image. Following this, an `argmax` operation is performed along the category dimension to produce the final segmentation masks.

B Additional Experimental Analysis and Discussion

B.1 Further Experimental Analysis

Impact of Concept Space Scale on Performance As observed in Table 1, DCR-SAM achieves more pronounced performance gains on remote sensing benchmarks compared to natural scene benchmarks. We hypothesize that this phenomenon is fundamentally driven by the scale of the concept space: the massive number of fine-grained categories in natural scenes exacerbates feature space crowding and semantic competition. To validate this hypothesis, we conducted a progressive subsetting experiment on the highly complex ADE20K [70] dataset ($N = 150$). Specifically, we establish a base set comprising the Top-30 most frequent categories and progressively expand the target category count $N \in \{30, 60, 90, 150\}$. For subsets where $N < 150$, all pixels belonging to unselected categories are uniformly absorbed by our dynamic background token ($\mathcal{T}_{bg}$). Crucially, we isolate and track the performance of the identical Base-30 classes across all experimental configurations.

As shown in Table 6, As the number of semantic categories increases ($30 \rightarrow 150$), the overall performance declines due to the introduction of difficult long-tail classes. Furthermore, the isolated **Base-30** class also suffers a substantial degradation, dropping from 79.4% to 70.3%. This indicates that even the most well-represented foundational categories are negatively impacted by the severe feature competition and representational crowding triggered by a larger concept space. Consequently, in remote sensing scenarios where the concept space is typically constrained (*e.g.*, $N \leq 10$), the dense competitive head can establish highly stable and non-overlapping semantic anchors. This effectively unlocks the full discriminative potential of the model architecture, thereby yielding superior performance gains.

Table 6: **Impact** of the concept space scale on the ADE20K dataset.

Target Category Count	Semantic Space Composition	Time↓	Overall mIoU (%)	Base-30 mIoU (%)
$N = 30$	Top-30 frequent classes only	236	**79.4**	**79.4**
$N = 60$	Base-30 + Next 30 frequent classes	238	75.6↓3.8	77.5↓1.9
$N = 90$	Base-30 + Next 60 frequent classes	241	71.5↓7.9	73.9↓5.5
$N = 150$	Full concept space	244	68.7↓10.7	70.3↓9.1

Impact of Concept Space Scale on Inference Efficiency We further investigate the impact of the concept space scale on inference latency. Specifically, we evaluate the DCR-SAM framework on the ADE20K dataset with 150 categories (**A-150**) alongside its full version comprising 847 categories (**A-847**), as well as the PASCAL Context dataset with 59 categories (**PC-59**) and its full version comprising 459 categories (**PC-459**). As illustrated in Figure 8, expanding the target categories from 150 to 847 on ADE20K introduces a manageable latency overhead of +369 ms (244 ms → 613 ms).

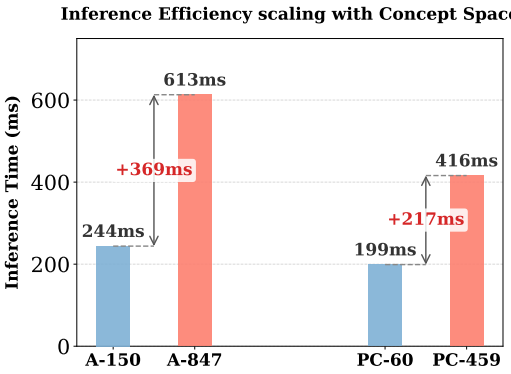


Figure 8: **Impact** of the concept space scale on inference efficiency.

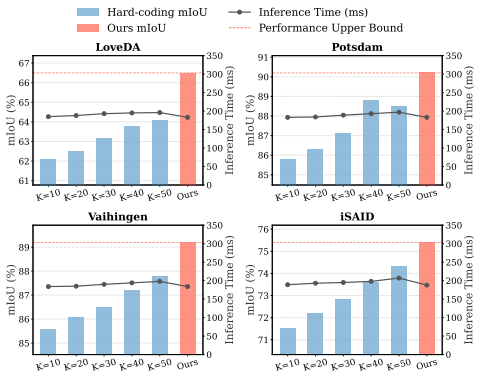


Figure 9: **Impact** of hard-coding background concept number K.

Similarly, scaling from 59 to 459 categories on PASCAL Context incurs a mere $+217$ ms increase (199 ms $\rightarrow$ 416 ms). Compared to the staggering 8038 ms required by the native SAM3 on the **A-150** benchmark (please see Table 2), these fractional increments are practically negligible. While our core cross-modal feature fusion process remains strictly category-agnostic, the lightweight matrix operations within the *VCTR* module and the dense competitive head naturally introduce a sub-linear computational overhead as the number of target concepts N grows. Nevertheless, achieving high-fidelity semantic segmentation for nearly a thousand categories in approximately 600 ms thoroughly demonstrates the extreme efficiency and scalability of our architecture, effectively bypassing the severe $\mathcal{O}(N)$ complexity bottleneck inherent to the native SAM3 baseline.

Investigating Spatial-Preserving Pooling Strategies Our *VGTP* module employs Global Average Pooling (GAP) on the visual features $\mathcal{V}^*$ to generate a global visual embedding $\mathcal{V}_g$ (*cf.* Eq.6). A reasonable concern is whether this GAP operation collapses fine-grained spatial cues, which are often deemed critical for resolving small objects in dense segmentation tasks. To investigate this, we conducted an ablation study comparing our GAP-based *VGTP* against a Spatial-Preserving Attention (SPA) [20] variant. Specifically, we bypassed the GAP operation and flattened the spatial dimensions, computing a dense similarity matrix $\mathcal{S}$ directly between the aligned text sequence $\tilde{\mathcal{P}} \in \mathbb{R}^{N\times l\times d}$ and the full spatial visual features $\mathcal{V}^* \in \mathbb{R}^{hw\times d}$. The spatial-aware attention weights $\mathcal{A}_{spa} \in \mathbb{R}^{N\times l}$ are obtained by marginalizing this dense similarity matrix over the spatial dimensions:

$$\mathcal{A}_{spa} = \texttt{softmax}\left(\frac{1}{hw}\sum_{i=1}^{hw}\mathcal{S}_{:,:,i}\right), \quad \text{where } S = \frac{\tilde{\mathcal{P}}(\mathcal{V}^*)^T}{\sqrt{d}},\ \mathcal{S} \in \mathbb{R}^{N\times l\times hw}. \tag{13}$$

The final semantic anchors are subsequently aggregated using these spatial-aware weights, formulated as $\mathcal{T}_{anchor} = \sum_{i=1}^{l}(\mathcal{A}_{spa})_{:,i} \odot \tilde{\mathcal{P}}_{:,i}$.

As presented in Table 7, the incorporation of the full spatial resolution in the SPA variant elevates the computational complexity of the attention mechanism. This introduces a slight inference latency and a concurrent performance decline (*e.g.*, a 4.8% mIoU drop on the ADE20K dataset). We argue that the retention of the full spatial resolution during the anchor generation phase compels the model to overfit to localized, noisy visual artifacts. In contrast, the GAP operation serves as an effective information bottleneck. It distills robust global representations from the features to reliably guide the aggregation of the text sequence, while delegating the high-fidelity spatial delineation to the final Dense Competitive Head.

Impact of Hard-Coding Background Concept Number To thoroughly investigate the efficacy and potential limitations of the hard-coding strategy, we conducted an ablation study varying the number of generated background concepts, denoted as K. Using the filtered background concepts (please see Appendix A.1), we reported the overall mIoU and inference time on the remote sensing benchmarks by progressively scaling the hard-coding concept number, *i.e.*, $K \in \{10, 20, 30, 40, 50\}$. During the inference phase, we appended these filtered concepts to the target category list to explicitly

Table 7: **Ablation study** on the text sequence pooling strategy across four diverse datasets.

Pooling Strategy	ADE20K		Cityscapes		LoveDA		Potsdam	
	Time.↓	mIoU	Time.↓	mIoU	Time.↓	mIoU	Time.↓	mIoU
SPA (Spatial-Preserving)	248	63.9	184	81.5	187	59.6	185	85.8
***VGTP* (GAP-based)**	**244**	**68.7**	**183**	**89.1**	**183**	**66.5**	**183**	**90.2**

absorb non-target pixels. As shown in Figure 9, due to the highly efficient, decoupled architecture of DCR-SAM, processing these additional concepts introduces only a marginal computational overhead rather than a severe latency bottleneck. However, the primary limitation emerges in the semantic performance, while scaling up K initially improves background absorption, the performance gains rapidly saturate, even with a large set of explicitly designed negative concepts, the rigid textual anchors struggle to encompass all unconstrained background variations, resulting in suboptimal performance compared to our proposed dynamic background strategy ($\mathcal{T}_{bg}$), which consistently achieves superior performance, demonstrating that it provides a more elegant and adaptable solution for robust semantic masking without the necessity of exhaustive prompt engineering.

Compared with batched inference mode To further evaluate inference efficiency, we compare DCR-SAM with native SAM 3 under different batch sizes on ADE20K, which contains 150 categories. As shown in Table 8, increasing the batch size from 1 to 50 reduces the inference latency of SAM 3 from 8038 ms to 5687 ms per image, but increases memory consumption from 9010 MB to 29080 MB. In contrast, DCR-SAM achieves a substantially lower latency of 244 ms with only 6318 MB of memory.

The limited acceleration of batched inference stems from the architecture of SAM 3, which involves backbone inference, feature fusion, and mask decoding. Although batching reduces redundant backbone computations by sharing visual features across categories, feature fusion and mask decoding must still be performed separately for each category, resulting in increasing computational overhead as the number of categories grows. In contrast, DCR-SAM performs backbone inference and feature fusion only once, with negligible additional overhead from the lightweight matrix operations in the VCTR module. These results demonstrate the superior computational efficiency and memory utilization of DCR-SAM in large-scale multi-class segmentation.

Table 8: **Efficiency comparison** between SAM 3 and DCR-SAM across different batch sizes.

Inference Strategy	Backbone Passes	Time (ms)↓	Memory (MB)↓
SAM 3 (Sequential, bs=1)	150	8038	9010
SAM 3 (Batched, bs=15)	10	6392	12960
SAM 3 (Batched, bs=25)	6	6067	17488
SAM 3 (Batched, bs=30)	5	5966	19816
SAM 3 (Batched, bs=50)	3	5687	29080
***DCR-SAM* (Ours)**	**1**	**244**	**6318**

C Expanded Qualitative Analysis

C.1 Cross-Modal Activation Map Visualizations

To verify the effectiveness of our method in mitigating overlapping predictions, we conduct a visual analysis of the cross-modal activation maps. As illustrated in Figure 10, it can be observed that directly aggregating the masks obtained from the independent inference of the native SAM3 results in overlapping predictions within the same regions. In contrast, our method effectively alleviates this issue through explicit inter-class competition.

C.2 Extended Qualitative Comparisons

To further substantiate the qualitative superiority of our proposed framework, we present an extended set of visual comparisons across diverse natural scene and remote sensing benchmarks, as shown in

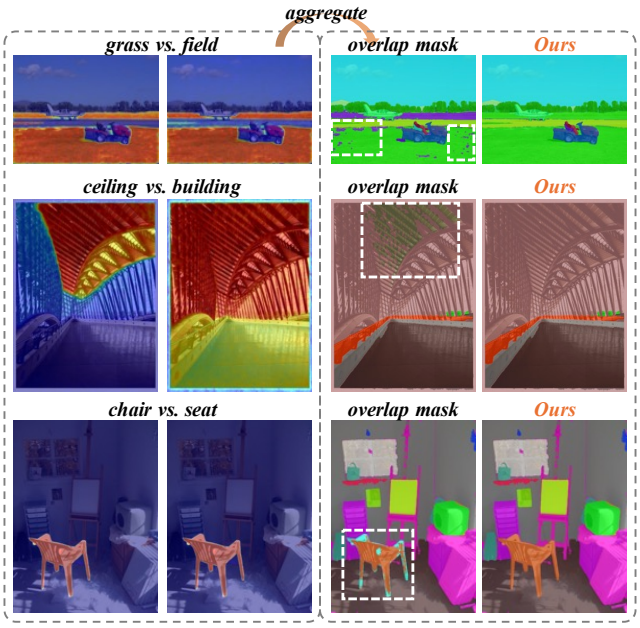


Figure 10: **Cross-modal semantic activation maps.** Aggregation in the original SAM3 leads to overlapping masks, whereas our method mitigates this issue through explicit inter-class competition.

Figure 11-18. These comprehensive visualizations demonstrate that the proposed method robustly delineates high-fidelity spatial boundaries while mitigating the severe semantic fragmentation prevalent in baseline models.

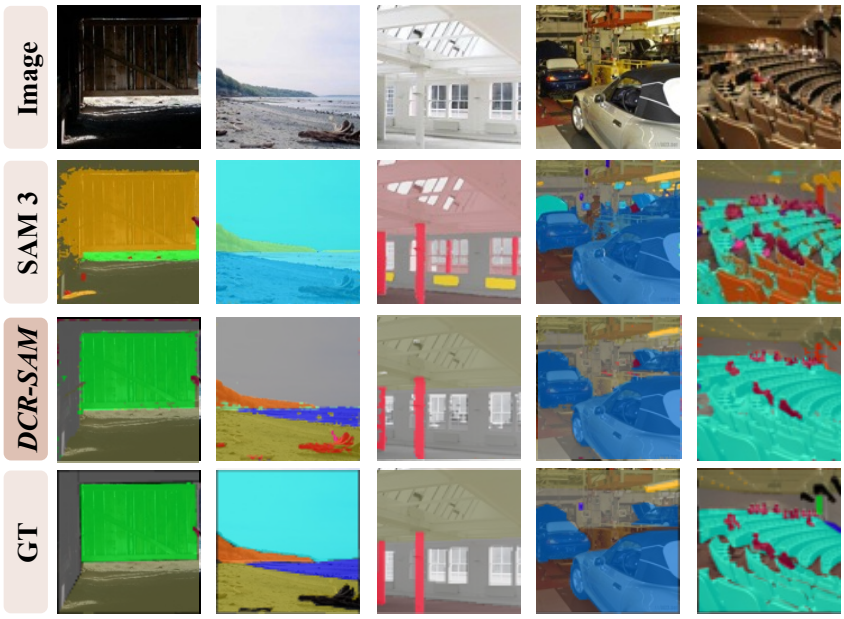


Figure 11: **Qualitative comparison** results on ADE20k dataset.

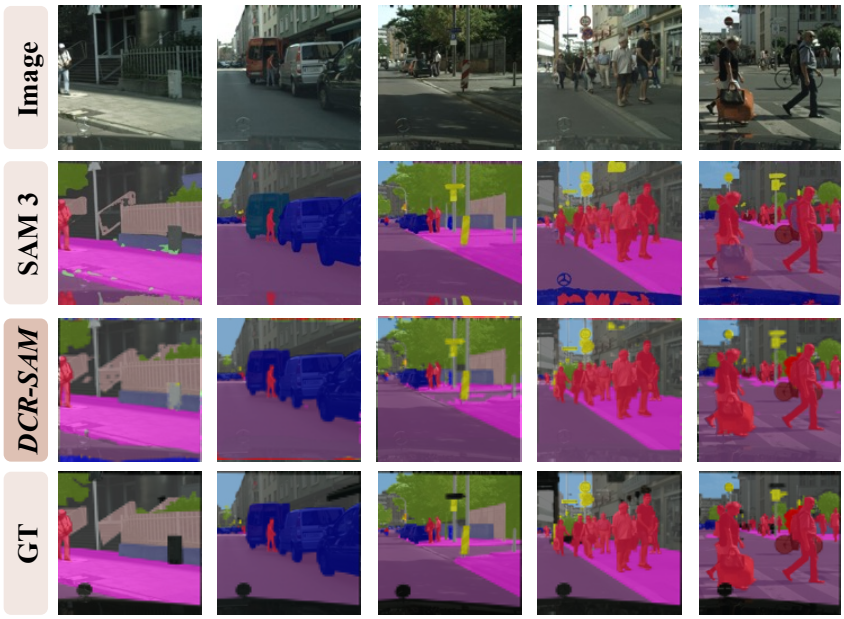


Figure 12: **Qualitative comparison** results on Cityscapes dataset.



Figure 13: **Qualitative comparison** results on COCO-Stuff dataset.

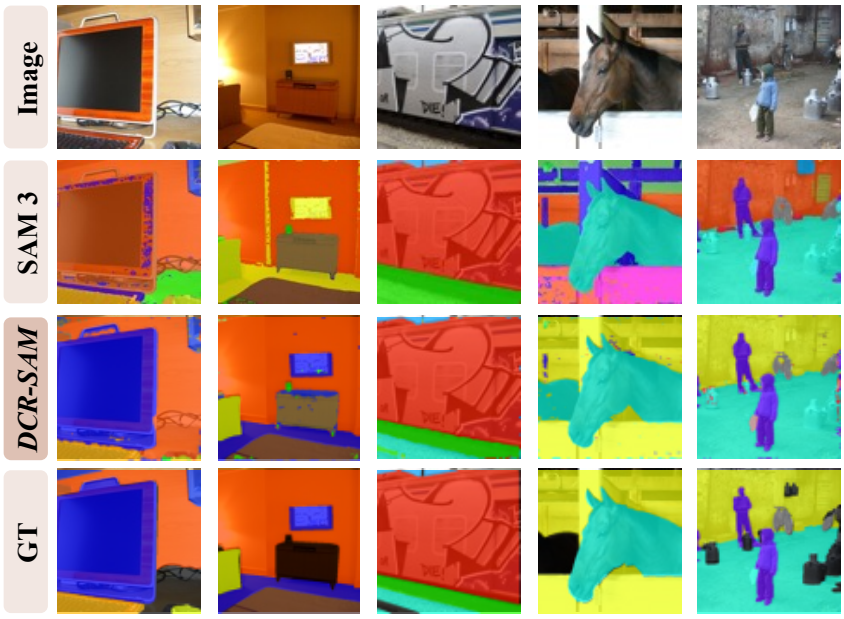


Figure 14: **Qualitative comparison** results on Context59 dataset.

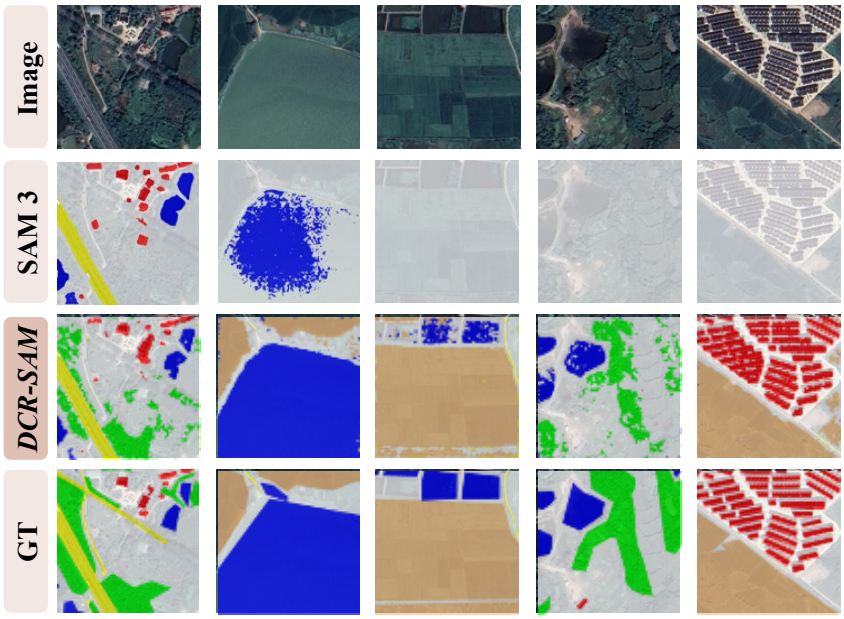


Figure 15: **Qualitative comparison** results on LoveDA dataset.

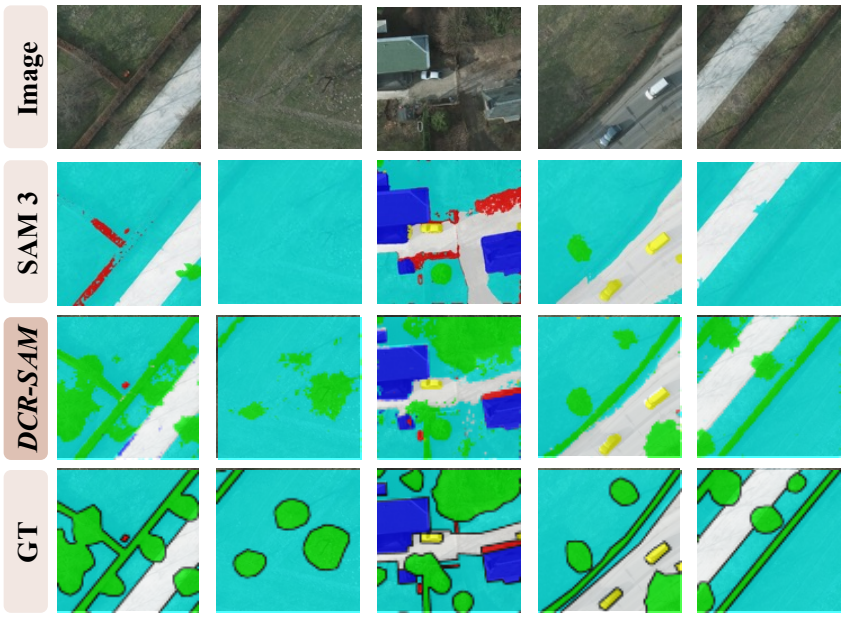


Figure 16: **Qualitative comparison** results on Potsdam dataset.

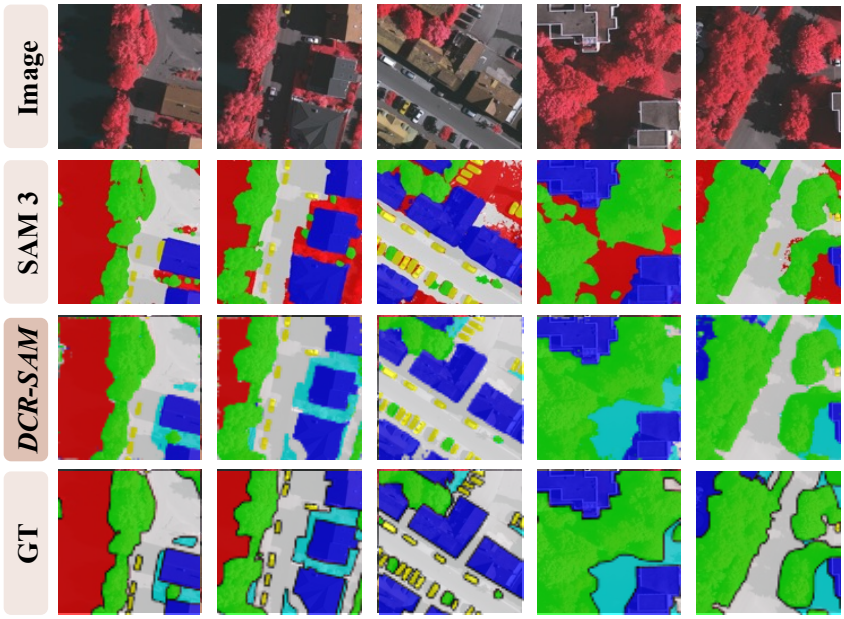


Figure 17: **Qualitative comparison** results on Vaihingen dataset.

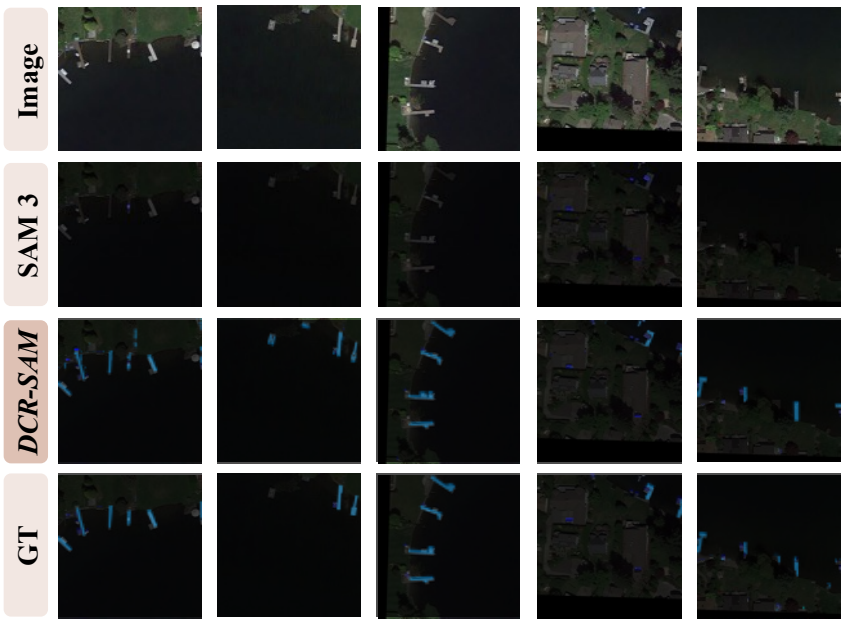


Figure 18: **Qualitative comparison** results on iSAID dataset.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have clearly stated this in the introduction section.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed limitations in the appendix B.2 of the paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We have provided this in the methodology section.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have provided implementation detail in the experiments section and appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have provided the data and code in supplemental material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided implementation detail in the experiments section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to limitation in computational resource, we did not conduct multiple iterations of the same experiment to calculate error.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided the necessary computer resource information for reproducing the experiment in the implementation details of the experiments section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We adhere to the NeurIPS Code of Ethics in every respect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have mentioned the broader impacts in the appendix B.2.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We strictly adhered to the relevant licensing agreements and terms of use for various assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: Our work does not involve any new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: This study did not incorporate LLMs as any significant component.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.