Early Semantic Grounding in Image Editing Models for Zero-Shot Referring Image Segmentation

Jingxuan He[1] **Xiyu Wang**[1] **Yunke Wang**[1] **Mengyu Zheng**[2] **Chang Xu**[1,*]

[1]The University of Sydney [2]Beijing Jiaotong University

Abstract

Instruction-based image editing (IIE) models have recently demonstrated strong capability in modifying specific image regions according to natural language instructions, which implicitly requires identifying where an edit should be applied. This indicates that such models inherently perform language-conditioned visual semantic grounding. In this work, we investigate whether this implicit grounding can be leveraged for zero-shot referring image segmentation (RIS), a task that requires pixel-level localization of objects described by natural language expressions. Through systematic analysis, we reveal that strong foreground-background separability is already present in the internal representations of these models at the earliest denoising timestep, well before any visible image transformation occurs. Building on this insight, we propose a training-free framework that repurposes pretrained image editing models for RIS by exploiting their intermediate representations. Our approach decomposes localization into two complementary components: attention-based spatial priors that estimate where to focus, and feature-based semantic discrimination that determines what to segment. By leveraging feature-space separability, the framework produces accurate segmentation masks using only a single denoising step, without requiring full image synthesis. Extensive experiments on RefCOCO, RefCOCO+, and RefCOCOg demonstrate that our method achieves superior performance over existing zero-shot baselines. Project page: `https://jessie459.github.io/EarlySemanticGrounding/`.

1 Introduction

Instruction-based Image Editing (IIE) has recently witnessed significant progress. By jointly modeling textual and visual signals, modern image editing frameworks [57, 48, 47, 21, 23, 46] demonstrate strong capability in modifying specific regions in accordance with natural language instructions. This ability implicitly requires the model to identify *where* the edit should be applied, indicating that image editing models inherently perform language-conditioned visual semantic grounding. In this work, we investigate whether this implicit grounding can be directly extracted and repurposed for Referring Image Segmentation (RIS) [17], a task that requires pixel-level localization of objects described by natural language expressions.

Despite recent progress in fully and weakly supervised RIS, these methods [9, 55, 6, 5] rely on costly pixel-level annotations. Zero-shot RIS provides a more practical alternative, yet existing frameworks typically adopt a multi-stage pipeline that combines the Segment Anything Model (SAM) [18, 4] for mask proposal and the CLIP [31] model for mask selection. This design suffers from two fundamental limitations. First, SAM is primarily geometry-driven and lacks an intrinsic vision-language alignment ability to interpret complex referring expressions. Second, recent RIS methods, such as HybridGL [25] and RefAM [19], employ intricate linguistic preprocessing to improve localization accuracy by

*Corresponding author.

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

decomposing referring expressions into noun phrases and explicitly modeling spatial relationships, which introduces additional complexity and is prone to error propagation.

In contrast, instruction-based image editing models provide a unified framework that jointly models language understanding and visual reasoning. We therefore take a fundamentally different perspective by *recasting zero-shot RIS as an image editing problem*. However, this reformulation raises a critical question: *in what form, and at what stage, does an image editing model establish the localization of the target object?*

Through systematic analysis of latent representations, we uncover a key and previously underexplored property of modern image editing models [21, 23]. We find that *object-level semantics are encoded in the feature space of clean image tokens*, and more importantly, that *strong foreground-background separability is already present at the earliest denoising timestep*, well before any visible image transformation occurs. This suggests that semantic grounding is largely established before visible synthesis, and that subsequent denoising primarily refines visual appearance rather than object-level discrimination. This insight also implies that localization signals can be extracted directly from early representations without requiring full image synthesis.

Building on these observations, we propose a training-free framework that repurposes pretrained image editing models for zero-shot referring image segmentation. Rather than performing iterative denoising, which remains costly even with optimized sampling schedules [45], we extract internal representations from a single timestep and leverage them for localization. Our approach decomposes the localization process into two complementary components. First, we derive a coarse spatial prior from cross-attention, which captures instruction-conditioned relevance. To improve spatial coherence, we introduce an affinity-based propagation mechanism that diffuses attention responses across semantically related regions. Second, we exploit the strong semantic separability of latent features by constructing foreground and background prototypes, and perform pixel-wise classification via cosine similarity to obtain the final segmentation result. This design explicitly decouples *where* to look (attention) from *what* to segment (feature semantics), leading to accurate and robust localization. Extensive experiments on RefCOCO [28], RefCOCO+ [28], and RefCOCOg [27] demonstrate that our method consistently outperforms existing zero-shot baselines.

The contributions of this work are summarized as follows:

- We reformulate referring image segmentation as an instruction-based image editing problem, showing that generative models inherently support discriminative visual grounding without requiring task-specific supervision.
- We reveal that modern image editing models exhibit strong foreground-background separability in their internal representations at the earliest denoising timestep.
- We propose a training-free framework that decomposes localization into two complementary components: attention-based spatial priors that estimate *where* to focus, and feature-based semantic discrimination that determines *what* to segment, resulting in precise and robust localization.
- Extensive experiments on RefCOCO, RefCOCO+, and RefCOCOg demonstrate that our method consistently outperforms existing zero-shot baselines, supported by comprehensive ablation studies that validate the effectiveness of each component.

2 Related Work

Instruction-based Image Editing. Instruction-based image editing aims to modify images according to natural language instructions while preserving content outside the edited regions. Early methods built upon Stable Diffusion [16, 32] learn to translate textual instructions into targeted visual modifications, as exemplified by InstructPix2Pix [2], MagicBrush [56], UltraEdit [58], and AnyEdit [52]. More recent approaches, such as ICEdit [57], improve synthesis fidelity by fine-tuning stronger Diffusion Transformer (DiT) backbones [30]. To further enhance instruction following, several state-of-the-art frameworks [23, 46, 3, 40] incorporate multi-modal large language models to strengthen cross-modal alignment between textual and visual representations. While these models are primarily designed for generative tasks, their ability to interpret editing instructions implies an inherent capacity for visual grounding. Recent work further shows that edit regions can be localized inside DiT-based editors from attention-guided feature centroids, which improves content consistency

outside the edited regions [14]. Building on this intuition, we repurpose instruction-based image editing models for referring image segmentation in a zero-shot manner.

Referring Image Segmentation. Referring image segmentation (RIS) is a visual grounding task that aims to segment objects specified by natural language expressions, which requires models to establish fine-grained associations between visual content and linguistic cues. While fully supervised [7, 24, 35, 36, 9, 55] and weakly supervised [10, 50, 6, 5] RIS methods have achieved strong performance, they remain heavily reliant on dense pixel-level annotations. Zero-shot referring image segmentation [53, 29, 38, 37, 44, 25, 19] offers a more scalable alternative by leveraging pretrained foundation models. Global-Local [53] is an early representative that exploits pretrained FreeSOLO [43] and CLIP [31] for mask prediction. CaR [37] improves upon this by iteratively refining masks through a CLIP-based recurrent framework. TAS [38] enhances segmentation performance with descriptive captions generated by an auxiliary captioner. More recently, HybridGL [25] introduces a hybrid global-local feature extraction strategy with spatial guidance to mitigate segmentation ambiguity, and RefAM [19] augments referring expressions with stop words to suppress attention sinks. Despite these advances, most existing zero-shot RIS methods remain dependent on SAM [18] for mask proposals and thus inherit its class-agnostic limitations in resolving referring expressions. In contrast, our framework directly harnesses the semantic grounding capability of pretrained image editing models for referring image segmentation.

Segmentation with Attention and Diffusion Features. Attention and affinity have been widely used to capture spatial relationships for segmentation. Ru et al. [33] and Xu et al. [49] learn affinities from transformer attention for weakly supervised segmentation, while DiffSeg [41] and DiffCut [8] exploit self-attention maps and diffusion features, respectively, for unsupervised segmentation. More recently, ConceptAttention [15] localizes concepts in text-to-image MM-DiTs through similarities between image representations and separately introduced concept representations. RefAM [19] uses DiT cross-attention to localize referring expressions and obtains segmentation masks with SAM. Our work instead examines clean reference-image tokens in instruction-based editing models and shows that they already exhibit strong foreground-background separability at the earliest evaluated denoising step. Building on this finding, our training-free framework uses attention-derived spatial priors to construct foreground and background prototypes, then predicts referring masks through feature-space classification in a single denoising step.

3 Method

3.1 Preliminaries: Instruction-based Image Editing

Recent multi-modal image editing models [20, 21, 23, 51] typically adopt a Large Language Model (LLM) [1] to encode editing instructions and a Multi-Modal Diffusion Transformer (MM-DiT) [30, 11] as a unified generator that jointly models textual and visual signals. Given an editing instruction P and a reference image $I \in \mathbb{R}^{H\times W\times 3}$, the LLM encodes P (and I in multi-modal settings) into a sequence of prompt embeddings $\mathbf{Z}_p \in \mathbb{R}^{N_p\times D}$, where H and W denote the height and width of the reference image, N_p is the number of prompt tokens, and D is the embedding dimension. For notational simplicity, we use "tokens" to refer to token embeddings throughout this work. The MM-DiT operates on three types of tokens: prompt tokens, clean image tokens $\mathbf{Z}_v \in \mathbb{R}^{N_v\times D}$, and noisy image tokens $\hat{\mathbf{Z}}_v^{(t)} \in \mathbb{R}^{N_v\times D}$ at diffusion timestep t, where N_v is the number of image tokens. These tokens are concatenated along the sequence dimension [39] and passed through a stack of transformer blocks, where the Multi-Modal Self-Attention (MMSA) module enables information exchange across modalities. Specifically, we denote the input to MMSA within the l-th DiT block at timestep t as $\mathbf{Z}^{(t,l)} = [\mathbf{Z}_p^{(l)}; \hat{\mathbf{Z}}_v^{(t,l)}; \mathbf{Z}_v^{(l)}] \in \mathbb{R}^{(N_p+2N_v)\times D}$. This unified token sequence $\mathbf{Z}^{(t,l)}$ is then transformed into queries $\mathbf{Q}^{(t,l)}$, keys $\mathbf{K}^{(t,l)}$, and values $\mathbf{V}^{(t,l)}$ through linear projection. The MMSA module is formulated as:

$$\text{MMSA}\left(\mathbf{Z}^{(t,l)}\right) = \mathbf{A}^{(t,l)}\mathbf{V}^{(t,l)}, \text{ where } \mathbf{A}^{(t,l)} = \text{Softmax}\left(\frac{\mathbf{Q}^{(t,l)}{\mathbf{K}^{(t,l)}}^\top}{\sqrt{D}}\right). \tag{1}$$

We denote the full attention matrix as $\mathbf{A}^{(t,l)} \in \mathbb{R}^{(N_p+2N_v)\times(N_p+2N_v)}$. Starting from a Gaussian noise, the noisy image tokens $\hat{\mathbf{Z}}_v^{(t)}$ are iteratively denoised across diffusion timesteps, and the fully denoised tokens are decoded to produce the final edited image.

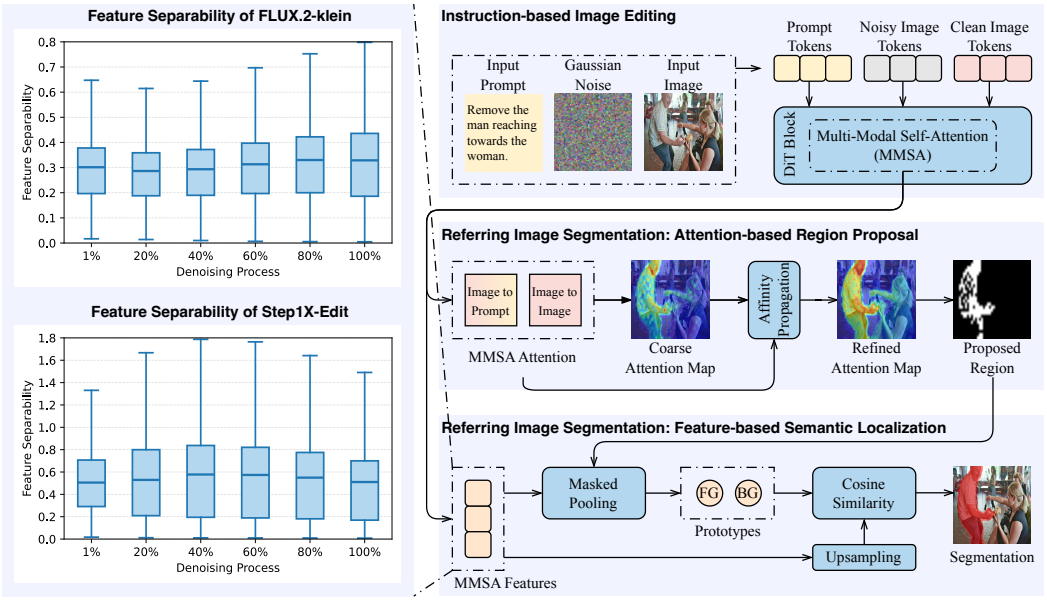

Figure 1: **Overview of our framework.** We repurpose pretrained instruction-based image editing models for referring image segmentation. In the module of attention-based region proposal, we derive coarse spatial priors through cross-attention and affinity propagation. In the module of feature-based semantic localization, the semantic separability of MMSA output features is exploited under the guidance of the proposed region to produce the final segmentation mask. The quantitative results in boxplots on the left panel show that strong foreground-background discriminability is already present at the earliest evaluated timestep (denoted as 1%) and remains relatively stable throughout the denoising process.

3.2 Referring Image Segmentation

Given an input image I and a referring expression T_{ref}, the goal of referring image segmentation is to produce a binary mask that identifies the referred object in the image. We reformulate this task as an instance of instruction-based image editing, where the referring expression T_{ref} is converted into an editing instruction T by prepending "remove" to the referring expression. We further refine the instructions using the built-in instruction-refinement pipeline of Step1X-Edit [51], which preserves the original meaning and retains instructions that are already clear. Details and the effect of refinement are provided in Appendix A3, and the robustness to alternative instruction templates is analyzed in Section 4.4.

Although image editing models are primarily designed for image synthesis, their denoising process inherently requires identifying regions relevant to the editing instruction. This suggests that intermediate representations may already encode object-level localization signals. Motivated by this insight, we first analyze the foreground-background separability of MMSA features extracted from clean image tokens, and then develop a two-stage framework that combines attention-based spatial priors with feature-based semantic discrimination for precise localization.

Feature Separability Analysis. Let $\mathbf{F}^{(t,l)} \in \mathbb{R}^{N_v \times D}$ denote MMSA output features of clean image tokens within the l-th DiT block at timestep t, and let Ω_{fg} and Ω_{bg} denote the sets of foreground and background spatial locations, respectively. We compute class-wise means $\mu_{\text{fg}}^{(t,l)}, \mu_{\text{bg}}^{(t,l)}$ and variances $\sigma_{\text{fg}}^{2(t,l)}, \sigma_{\text{bg}}^{2(t,l)}$ from these features. Inspired by the Fisher discriminant criterion [12], we define the separability score as:

$$\mathcal{J}^{(t,l)} = \frac{\left\| \mu_{\text{fg}}^{(t,l)} - \mu_{\text{bg}}^{(t,l)} \right\|^2}{\sigma_{\text{fg}}^{2(t,l)} + \sigma_{\text{bg}}^{2(t,l)}}. \tag{2}$$

This metric measures the ratio between inter-class distance and intra-class variance, where larger values indicate stronger feature discriminability. Detailed equations and rationale for this metric are provided in Appendix A1.

We evaluate $\mathcal{J}^{(t,l)}$ on FLUX.2-klein [21] and Step1X-Edit [23] across a range of timesteps t. The DiT block l is set to approximately two-thirds of the model depth. As illustrated in the left panel of Fig. 1, both models exhibit strong feature separability as early as the first denoising timestep, and the separability remains relatively stable throughout the denoising process. Quantitative results across three benchmarks further support this observation (Table A1). This indicates that *semantic grounding is already established at early timesteps*, while subsequent timesteps may focus on fine-grained visual synthesis rather than improving object-level discrimination.

Attention-based Region Proposal (ARP). Although MMSA features exhibit strong discriminative capability at early timesteps, they do not explicitly capture the spatial extent of the referred object. We therefore first derive a coarse estimate of the target region from attention, which serves as a spatial prior for subsequent feature-based localization.

As shown in the top right panel of Fig. 1, the l-th DiT block takes as input the prompt tokens $\mathbf{Z}_p^{(l)}$, the clean image tokens $\mathbf{Z}_v^{(l)}$, and the noisy image tokens $\hat{\mathbf{Z}}_v^{(t,l)}$ at timestep t. After token-wise concatenation, the MMSA module processes the unified token sequence by projecting it into queries $\mathbf{Q}^{(t,l)}$, keys $\mathbf{K}^{(t,l)}$, and values $\mathbf{V}^{(t,l)}$, respectively, followed by the standard self-attention operation defined in Eq. (1). Next, as illustrated in the middle right panel of Fig. 1, we extract the image-to-prompt submatrix $\mathbf{A}_{v\to p}^{(t,l)} \in \mathbb{R}^{N_v \times N_p}$ from the full attention matrix. This submatrix is aggregated across N_p prompt tokens and N_l DiT blocks, resulting in the Coarse Attention Map (CAM) $\mathbf{A}_{\text{CAM}}^{(t)} \in \mathbb{R}^{N_h \times N_w}$:

$$\mathbf{A}_{\text{CAM}}^{(t)} = \mathcal{N}\left(\sum_{l=1}^{N_l} \mathcal{F}^{-1}\left(\mathbf{A}_{v\to p}^{(t,l)} \mathbf{1}_{N_p}\right)\right), \tag{3}$$

where $\mathbf{1}_{N_p}$ denotes an all-ones vector of length N_p, $\mathcal{F}^{-1}(\cdot)$ reshapes a flattened sequence of length N_v into a spatial map of size $N_h \times N_w$, and $\mathcal{N}(\cdot)$ denotes min-max normalization.

Although the coarse attention map identifies an approximate location of the referred object, it often exhibits fragmented or noisy activations due to the lack of spatial coherence in cross-attention. To alleviate this issue, inspired by [49, 33], we introduce an affinity-based propagation mechanism that leverages self-attention to propagate cross-attention responses across semantically related regions. Specifically, we extract the image-to-image submatrix $\mathbf{A}_{v\to v}^{(t,l)} \in \mathbb{R}^{N_v \times N_v}$ from the full attention matrix, and interpret it as an affinity matrix that captures pairwise relationships among clean image tokens. We perform row-wise normalization to obtain the stochastic transition matrix $\mathbf{T}^{(t,l)} \in \mathbb{R}^{N_v \times N_v}$, which is used to compute the Refined Attention Map (RAM) $\mathbf{A}_{\text{RAM}}^{(t)} \in \mathbb{R}^{N_h \times N_w}$:

$$\mathbf{A}_{\text{RAM}}^{(t)} = \mathcal{N}\left(\sum_{l=1}^{N_l} \mathcal{F}^{-1}\left(\mathbf{T}^{(t,l)} \mathbf{A}_{v\to p}^{(t,l)} \mathbf{1}_{N_p}\right)\right), \text{ where } \mathbf{T}^{(t,l)} = \mathcal{D}\left(\mathbf{A}_{v\to v}^{(t,l)} \mathbf{1}_{N_v}\right)^{-1} \mathbf{A}_{v\to v}^{(t,l)}. \tag{4}$$

Here $\mathcal{D}(\cdot)$ constructs a diagonal matrix from its input. This attention propagation design can be interpreted as a random walk [26] over image tokens, where the stochastic transition matrix defines a Markov process based on pairwise affinities. Since each affinity is determined by the query-key similarity between two clean-image tokens, tokens within the same semantic region, which share appearance and context, generally exhibit stronger affinities. Attention is therefore diffused to semantically coherent regions, resulting in improved spatial continuity and better alignment with object boundaries.

Feature-based Semantic Localization (FSL). While attention-based region proposal provides necessary spatial priors, it remains inherently imprecise due to the diffuse nature of attention. In contrast, our early analysis shows that MMSA features exhibit strong foreground-background separability, suggesting that object-level semantics are already well captured in the feature space. We therefore leverage the rich semantics of these features to transform the initial proposal into an accurate localization of the referred object.

As depicted in the bottom right panel of Fig. 1, we extract MMSA features of clean image tokens at timestep t' and DiT block l'. These features are reshaped into a spatial feature map and ℓ_2-normalized to lie on the unit hypersphere, yielding $\hat{\mathbf{F}}^{(t',l')} \in \mathbb{R}^{D \times N_h \times N_w}$. We then obtain a binary mask $\mathbf{M}^{(t)} = \mathbf{1}[\mathbf{A}_{\text{RAM}}^{(t)} > \tau]$ by thresholding the refined attention map, which selects high-confidence regions of the referred object for prototype construction, where τ is a predefined threshold and $\mathbf{1}[\cdot]$

denotes the indicator function. Foreground and background semantic prototypes, $\mathbf{P}_{\text{fg}} \in \mathbb{R}^D$ and $\mathbf{P}_{\text{bg}} \in \mathbb{R}^D$, are constructed via masked average pooling over the normalized features:

$$\begin{aligned}\mathbf{P}_{\text{fg}} &= \frac{\sum_{h=1}^{N_h}\sum_{w=1}^{N_w}\mathbf{M}_{h,w}^{(t)}\hat{\mathbf{F}}_{:,h,w}^{(t',l')}}{\sum_{h=1}^{N_h}\sum_{w=1}^{N_w}\mathbf{M}_{h,w}^{(t)}},\\ \mathbf{P}_{\text{bg}} &= \frac{\sum_{h=1}^{N_h}\sum_{w=1}^{N_w}\left(1-\mathbf{M}_{h,w}^{(t)}\right)\hat{\mathbf{F}}_{:,h,w}^{(t',l')}}{\sum_{h=1}^{N_h}\sum_{w=1}^{N_w}\left(1-\mathbf{M}_{h,w}^{(t)}\right)}.\end{aligned} \tag{5}$$

The feature map $\hat{\mathbf{F}}^{(t',l')}$ is subsequently upsampled to the original image resolution, yielding $\hat{\mathbf{F}}_{\text{up}}^{(t',l')} \in \mathbb{R}^{D\times H\times W}$. We perform pixel-wise binary classification by computing cosine similarity between each prototype and the upsampled features. Each pixel is assigned to the class whose prototype has higher similarity. The final segmentation mask $\mathbf{S} \in \{0,1\}^{H\times W}$ is obtained as:

$$\mathbf{S}_{h,w} = \underset{c\in\{\text{fg},\text{bg}\}}{\arg\max}\,\mathbf{P}_c^{\top}\hat{\mathbf{F}}_{\text{up},:,h,w}^{(t',l')}. \tag{6}$$

We empirically find that performing attention estimation at $t = 0$ and feature extraction at $t' = 0$ within a deep DiT block yields the best performance on the validation sets (Section 4.4). This configuration provides both an accurate and efficient solution for referring image segmentation.

4 Experiments

4.1 Datasets and Metrics

Following previous works [38, 44, 25, 19], we conduct experiments on three standard Referring Image Segmentation (RIS) benchmarks: RefCOCO [28], RefCOCO+ [28], and RefCOCOg [27]. All images in these benchmarks are drawn from the MSCOCO dataset [22] and are paired with carefully annotated referring expressions that specify target objects. RefCOCO contains relatively concise expressions, many of which include positional cues such as "left" and "right", while such spatial terms are excluded in RefCOCO+. RefCOCOg is a more challenging benchmark that features longer and more complex expressions. We adopt two commonly used metrics to evaluate segmentation performance: overall Intersection over Union (oIoU) and mean Intersection over Union (mIoU). Specifically, oIoU computes intersections and unions over all samples before taking their ratio, thereby emphasizing performance on larger objects, while mIoU averages IoU scores across individual samples and reflects performance across varying object scales.

4.2 Implementation Details

We repurpose two state-of-the-art instruction-based image editing models, FLUX.2-klein [21] and Step1X-Edit [23], for zero-shot referring image segmentation. To obtain Coarse Attention Maps (CAMs) and Refined Attention Maps (RAMs), we perform a single denoising step at $t = 0$ and aggregate attention weights across all DiT blocks. For feature-based semantic localization, we extract MMSA output features at $t' = 0$. Due to differences in model depth, we select features from $l' = 20$ out of 32 DiT blocks for FLUX.2-klein and $l' = 40$ out of 57 DiT blocks for Step1X-Edit. All hyperparameters ($t = t' = 0$, $l' = 40$, and $\tau = 0.8$) are selected once using Step1X-Edit on the *val* splits (Section 4.4) and fixed for all datasets and test splits; for FLUX.2-klein, we transfer the selected relative depth (about two-thirds) to its 32 blocks. All experiments are conducted on one NVIDIA A100 GPU. A detailed analysis of inference efficiency, including latency and memory comparisons with baselines, is provided in Appendix A2.

4.3 Comparison with State-of-the-arts

Table 1 presents quantitative comparisons with state-of-the-art zero-shot referring image segmentation methods on RefCOCO, RefCOCO+, and RefCOCOg. Overall, both FLUX.2-klein and Step1X-Edit variants achieve competitive performance across all three benchmarks, with Step1X-Edit consistently attaining the best results across dataset splits under both oIoU and mIoU metrics. In terms of oIoU, the Step1X-Edit variant surpasses the strongest baseline RefAM [19], which uses a DiT of the same

Table 1: **Comparisons with state-of-the-art zero-shot RIS methods on RefCOCO, RefCOCO+, and RefCOCOg**. The best and second-best results are highlighted in **bold** and underlined, respectively. The notation of * indicates the use of additional data beyond the task-specific dataset. The number in parentheses to the right of the model name denotes the parameter count of the corresponding DiT model.

Metric	Method	Vision Backbone	Pretrained Model	RefCOCO			RefCOCO+			RefCOCOg	
				val	testA	testB	val	testA	testB	val	test
oIoU	*zero-shot methods w/ additional training*										
	Pseudo-RIS [54]	ViT-B	SAM, CoCa, CLIP	37.33	43.43	31.90	40.19	46.43	33.63	41.63	43.52
	VLM-VG [42]	R101	COCO*, VLM-VG*	45.40	48.00	41.40	37.00	40.70	30.50	42.80	44.10
	zero-shot methods w/o additional training										
	Grad-CAM [34]	R50	SAM, CLIP	23.44	23.91	21.60	26.67	27.20	24.84	23.00	23.91
	MaskCLIP [59]	R50	SAM, CLIP	20.18	20.52	21.30	22.06	22.43	24.61	23.05	23.41
	Global-Local [53]	R50	FreeSOLO, CLIP	24.58	23.38	24.35	25.87	24.61	25.61	30.07	29.83
	Global-Local [53]	R50	SAM, CLIP	24.55	26.00	21.03	26.62	29.99	22.23	28.92	30.48
	Global-Local [53]	ViT-B	SAM, CLIP	21.71	24.48	20.51	23.70	28.12	21.86	26.57	28.21
	Ref-Diff [29]	ViT-B	SAM, SD, CLIP	35.16	37.44	34.50	35.56	38.66	31.40	38.62	37.50
	TAS [38]	ViT-B	SAM, BLIP2, CLIP	29.53	30.26	28.24	33.21	38.77	28.01	35.84	36.16
	SAM 3 [4]	PE	SAM 3	38.77	37.43	41.80	31.62	32.44	30.70	35.04	35.81
	HybridGL [25]	ViT-B	SAM, CLIP	41.81	44.52	38.50	35.74	41.43	30.90	42.47	42.97
	RefAM [19]	DiT	SAM, FLUX.1-dev (12B)	46.91	52.30	43.88	38.57	42.66	34.90	45.53	44.45
	Ours	DiT	FLUX.2-klein (9B)	50.13	52.99	46.10	40.17	43.25	36.06	46.90	46.26
	Ours	DiT	Step1X-Edit (12B)	**53.94**	**58.10**	**49.60**	**44.38**	**49.30**	**40.06**	**51.23**	**50.86**
mIoU	*zero-shot methods w/ additional training*										
	Pseudo-RIS [54]	ViT-B	SAM, CoCa, CLIP	41.05	48.19	33.48	44.33	51.42	35.08	45.99	46.67
	VLM-VG [42]	R101	COCO*, VLM-VG*	49.90	53.10	46.70	42.70	47.30	36.20	48.00	48.50
	zero-shot methods w/o additional training										
	Grad-CAM [34]	R50	SAM, CLIP	30.22	31.90	27.17	33.96	25.66	32.29	33.05	32.50
	MaskCLIP [59]	R50	SAM, CLIP	25.62	26.66	25.17	27.49	28.49	30.47	30.13	30.15
	Global-Local [53]	R50	FreeSOLO, CLIP	26.70	24.99	26.48	28.22	26.54	27.86	33.02	33.12
	Global-Local [53]	R50	SAM, CLIP	31.83	32.93	28.64	34.97	37.11	30.61	40.66	40.94
	Global-Local [53]	ViT-B	SAM, CLIP	33.12	36.52	29.58	35.29	39.58	31.89	40.08	40.74
	Ref-Diff [29]	ViT-B	SAM, SD, CLIP	37.21	38.40	37.19	37.29	40.51	33.01	44.02	44.51
	TAS [38]	ViT-B	SAM, BLIP2, CLIP	39.84	41.08	36.24	43.63	49.13	36.54	46.62	46.80
	CaR [37]	ViT-B and ViT-L	CLIP	33.57	35.36	30.51	34.22	36.03	31.02	36.67	36.57
	SAM 3 [4]	PE	SAM 3	34.80	33.36	38.91	26.83	28.30	25.56	32.14	32.08
	HybridGL [25]	ViT-B	SAM, CLIP	49.48	53.37	45.19	43.40	49.13	37.17	51.25	51.59
	RefAM [19]	DiT	SAM, FLUX.1-dev (12B)	57.24	59.78	53.32	43.59	47.28	38.77	47.11	48.35
	Ours	DiT	FLUX.2-klein (9B)	54.02	56.09	50.22	44.15	46.72	40.25	50.62	50.47
	Ours	DiT	Step1X-Edit (12B)	**57.56**	**60.34**	**53.76**	**48.50**	**52.37**	**44.28**	**54.52**	**54.25**

Table 2: **Ablation of modules within our framework**. "CAMs", "RAMs", and "FSL" denote coarse attention maps, refined attention maps, and feature-based semantic localization, respectively.

	CAMs	RAMs	FSL	RefCOCO	RefCOCO+	RefCOCOg
oIoU	✓			33.02	26.83	32.82
	✓	✓		49.56	40.23	47.24
	✓	✓	✓	**53.94**	**44.38**	**51.23**
mIoU	✓			39.44	32.03	40.11
	✓	✓		53.36	43.82	51.04
	✓	✓	✓	**57.56**	**48.50**	**54.52**

Table 3: **Ablation of feature extraction location**. "Block", "AdaLN", and "MMSA" denote output features of a diffusion transformer block, adaptive layer normalization, and multi-modal self-attention, respectively.

Feature	RefCOCO		RefCOCO+		RefCOCOg	
	oIoU	mIoU	oIoU	mIoU	oIoU	mIoU
Block	52.07	56.67	41.25	45.83	47.93	53.25
AdaLN	53.63	57.34	43.24	47.42	49.89	53.82
MMSA	**53.94**	**57.56**	**44.38**	**48.50**	**51.23**	**54.52**

parameter scale (12B), by a clear margin of approximately 5-7 points, while the FLUX.2-klein (9B) variant consistently achieves the second-best results. For mIoU, the Step1X-Edit variant outperforms existing approaches across all three benchmarks, indicating more robust segmentation across objects of varying scales. Notably, although our framework requires no training, the Step1X-Edit variant also outperforms Pseudo-RIS [54] and VLM-VG [42], which rely on additional training, on all dataset splits under both metrics. Unlike prior methods that rely on explicit positional modeling or directional bias (e.g., Ref-Diff [29], MaskCLIP [59], Global-Local [53], HybridGL [25], RefAM [19]), our approach directly leverages the semantic grounding capability of pretrained image editing models. This also explains the relatively smaller mIoU gains of our method on RefCOCO, where expressions frequently depend on directional cues. Although SAM 3 [4] is specifically designed for segmentation, it performs suboptimally on the RIS task due to its class-agnostic nature. In particular, it often misses the referred object entirely or segments all instances of the same category, regardless of the spatial relations specified in the expression. Additional analysis and qualitative examples of SAM 3 are provided in Appendix A4.

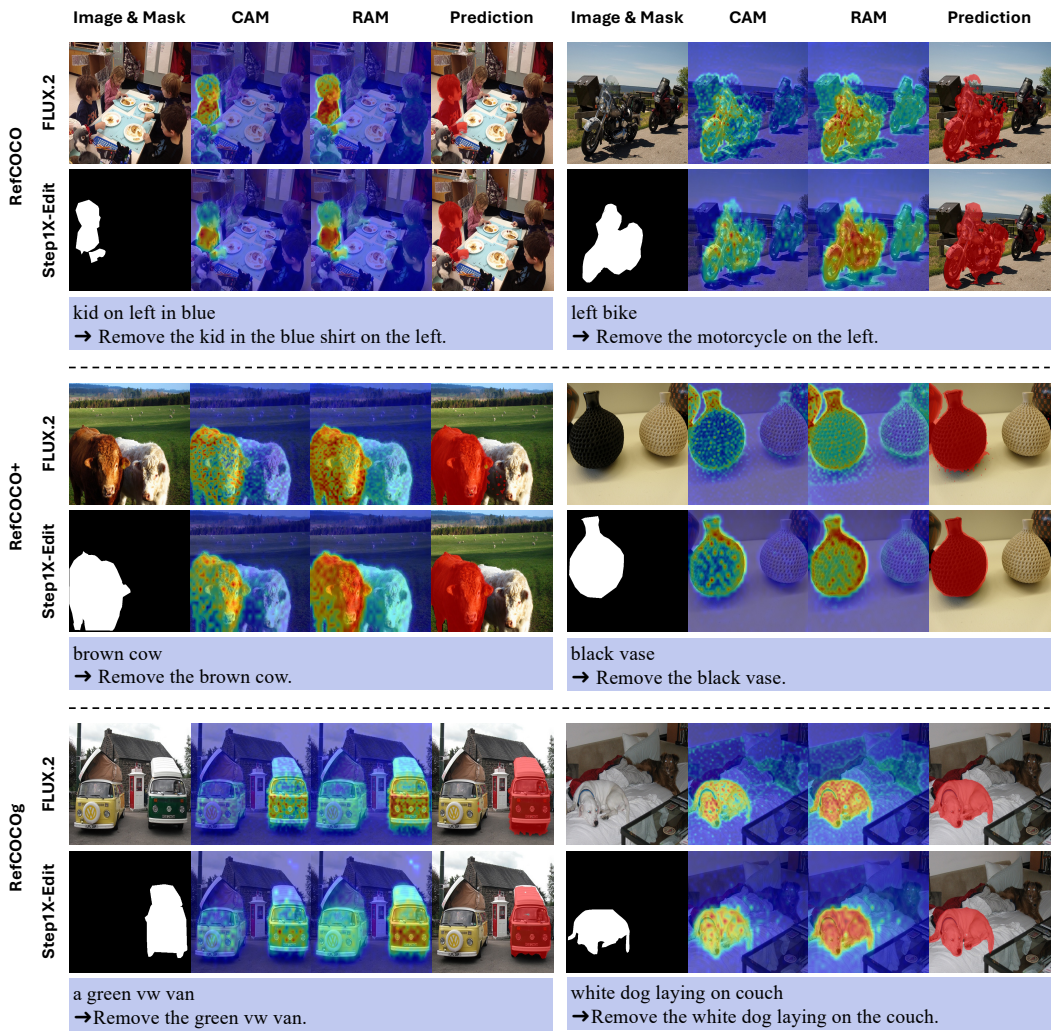

Figure 2: **Qualitative examples from RefCOCO, RefCOCO+, and RefCOCOg datasets.** "CAM" denotes Coarse Attention Map, "RAM" denotes Refined Attention Map, and "GT" denotes Ground Truth. The first row of text presents the original referring expressions, and the second row presents the corresponding transformed editing instructions.

4.4 Ablation Studies

To assess the contribution of each module in our framework and the impact of hyperparameters, we conduct comprehensive ablation studies using the Step1X-Edit variant on the *val* splits of RefCOCO, RefCOCO+, and RefCOCOg.

Quantitative ablation of modules. We evaluate the contributions of Attention-based Region Proposal (ARP) and Feature-based Semantic Localization (FSL). For ablation experiments, attention maps are binarized into segmentation masks using a fixed threshold of 0.5. As shown in Table 2, CAMs alone yield limited performance, indicating that coarse spatial priors are insufficient for accurate segmentation. Introducing RAMs leads to substantial improvements across all datasets (e.g., around 16-point oIoU gain on RefCOCO). This shows that affinity propagation effectively enhances spatial coherence. Incorporating FSL further improves oIoU and mIoU, which highlights the importance of leveraging semantically separable latent features for precise foreground-background discrimination. Overall, these results demonstrate that accurate segmentation arises from the complementary roles of attention-based spatial priors and feature-based semantic localization.

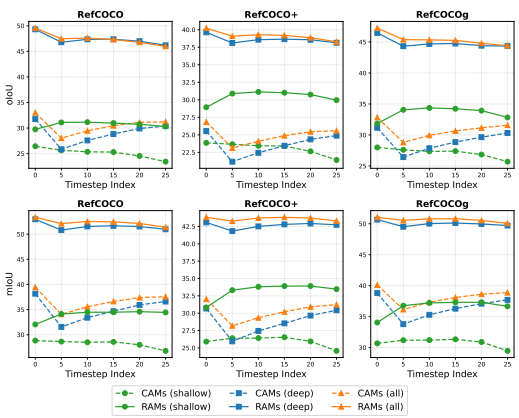

Figure 3: **Impact of timesteps and DiT blocks within the ARP module.** "shallow" and "deep" refer to accumulating attention weights over double-stream and single-stream DiT blocks, respectively, while "all" signifies attention accumulation over all DiT blocks.

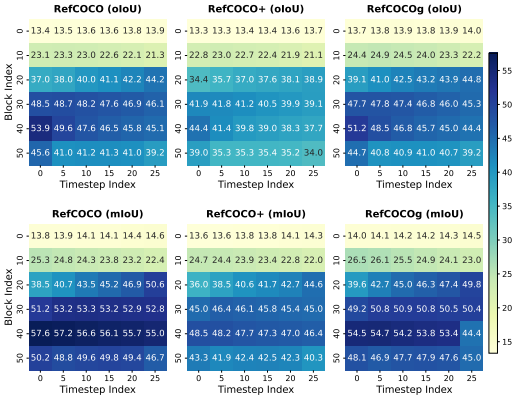

Figure 4: **Impact of timesteps and DiT blocks within the FSL module.** Heatmaps show oIoU (1st row) and mIoU (2nd row) on RefCOCO, RefCOCO+, and RefCOCOg datasets. Each cell represents segmentation performance when extracting features from a specific block at a given timestep.

Qualitative ablation of modules. We present qualitative examples in Fig. 2. As illustrated, CAMs provide initial but spatially incomplete object localization, while RAMs expand the coverage to more coherent regions via affinity propagation. Guided by the proposed region, FSL produces segmentation masks that accurately delineate the full extent of the referred object. Notably, when attention maps activate only on salient parts (e.g., the blue shirt of the child in the first row), FSL is able to recover the complete object. In more challenging cases, where attention maps highlight multiple similar instances or exhibit noisy activations (as exemplified in the last row), FSL correctly identifies the referred "green van" between the two candidates and suppresses spurious responses outside the "dog" region. These examples corroborate the quantitative findings: attention provides a coarse spatial prior, while precise localization relies on semantic feature discrimination.

Impact of denoising timesteps and DiT blocks. For the ARP module, we examine the impact of denoising timesteps and DiT blocks on the quality of CAMs and RAMs. Specifically, we evaluate oIoU and mIoU across denoising timesteps $t \in \{0, 5, 10, 15, 20, 25\}$, and analyze the contribution of different DiT components by accumulating attention weights over double-stream (i.e., shallow), single-stream (i.e., deep), and all blocks. The experimental results illustrated in Fig. 3 lead to several key observations. First, RAMs (solid lines) consistently outperform CAMs (dotted lines) across all configurations, which corroborates the effectiveness of affinity propagation observed in Table 2. Second, from an architectural perspective, aggregating attention from shallow blocks (green lines) yields substantially lower performance than using deep blocks (blue lines) or all blocks (orange lines). Notably, accumulation over deep blocks achieves comparable performance with that over all blocks, indicating that semantic cues critical for referring object comprehension are mainly captured in deep layers. Finally, we observe that the initial denoising step ($t = 0$) yields peak performance across all benchmarks in terms of oIoU and mIoU. This finding supports our design choice of single-step inference without a full iterative denoising process.

For the FSL module, we conduct experiments over timesteps $t' \in \{0, 5, 10, 15, 20, 25\}$ and blocks $l' \in \{0, 10, 20, 30, 40, 50\}$ to study the effect of feature extraction configurations. As shown in Fig. 4, the combination of $t' = 0$ and $l' = 40$ consistently yields the peak segmentation performance across all benchmarks in terms of oIoU and mIoU. These empirical findings resonate with our previous observations: First, the initial timestep ($t' = 0$) already provides sufficiently discriminative features for referred object recognition, thereby eliminating the need for subsequent denoising iterations; Second, features extracted from deep layers (approximately two-thirds of the model depth) possess strong semantics that can be leveraged for accurate segmentation of the referred object.

Ablation of feature extraction location. To determine an effective feature extraction location for the FSL module, we compare representations extracted from three locations within a DiT block: the outputs of Multi-Modal Self-Attention (MMSA), Adaptive Layer Normalization (AdaLN), and the block output. As shown in Table 3, MMSA features consistently achieve the best performance

Table 4: **Sensitivity to the instruction template.** We report changes in oIoU / mIoU on the *val* splits relative to the default "Remove" instruction.

Model	Instruction	RefCOCO	RefCOCO+	RefCOCOg
FLUX.2-klein (4B)	Erase	$+1.12/+1.17$	$+0.28/+0.39$	$+0.11/+0.41$
	Delete	$+0.97/+0.77$	$+0.53/+0.50$	$+0.10/+0.29$
	Replace	$+0.27/+1.18$	$-1.13/-0.75$	$-1.09/-0.26$
Step1X-Edit (12B)	Erase	$-1.17/-0.80$	$-0.49/-0.44$	$-0.90/-0.65$
	Delete	$-1.53/-1.00$	$-1.03/-0.90$	$-0.94/-0.71$
	Replace	$-4.72/-3.81$	$-3.60/-3.72$	$-5.36/-4.54$

across all benchmarks, outperforming both AdaLN and block features by a clear margin. While prior work [13] suggests that post-AdaLN features exhibit strong semantic correspondences, our results indicate that such features are less effective for referring image segmentation where precise cross-modal grounding is required. We attribute the advantage of MMSA to its explicit modeling of interactions between textual and visual tokens, which yields more discriminative and semantically aligned representations for the referred object.

Impact of thresholding. We study the effect of the threshold τ within the FSL module in Fig. 5. As τ increases from 0.1 to 0.8, both oIoU and mIoU scores increase steadily, suggesting that higher thresholds help emphasize more discriminative features and promote coherent clustering of semantically similar regions. However, a slight performance degradation occurs at $\tau = 0.9$. This indicates that overly aggressive thresholding may discard useful semantic cues. Based on this trade-off, we adopt $\tau = 0.8$ for feature-based semantic localization.

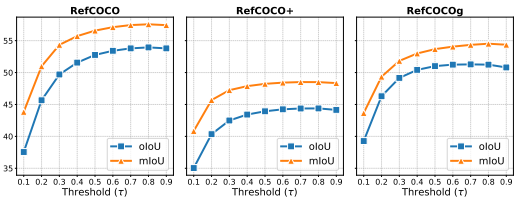

Figure 5: **Impact of thresholding within the FSL module**. The performance exhibits a steady increase until $\tau = 0.8$, then a slight performance degradation occurs at $\tau = 0.9$.

Sensitivity to the instruction template. Since our framework converts each referring expression into a removal instruction, we examine how sensitive segmentation performance is to the choice of editing verb. On the full *val* splits, we replace "Remove" with "Erase", "Delete", or "Replace" while keeping the rest of each instruction unchanged. As shown in Table 4, the two removal synonyms, "Erase" and "Delete", yield modest changes of at most 1.53 oIoU points across both image editing models. The effect of "Replace" varies more across models: performance changes by at most 1.13 oIoU points for FLUX.2-klein, whereas Step1X-Edit drops by 3.60-5.36 points. One possible explanation is that removal encourages localization of the entire target, while replacement may emphasize only the regions that need modification. Even with "Replace", Step1X-Edit achieves 49.22, 40.78, and 45.87 oIoU on RefCOCO, RefCOCO+, and RefCOCOg, respectively, outperforming RefAM's corresponding scores of 46.91, 38.57, and 45.53. Overall, these results indicate limited sensitivity to the tested removal synonyms, but also show that segmentation performance depends partly on the operation specified and the underlying image editing model.

5 Conclusion

We introduce a fundamentally different perspective on zero-shot referring image segmentation by reformulating it as an instruction-based image editing task. Our analysis reveals that semantically separable representations already emerge at early denoising steps. Building on this insight, we propose a simple yet effective training-free framework that combines attention-derived spatial priors with feature-level semantic separation to directly predict segmentation masks. Extensive experiments demonstrate that our approach achieves state-of-the-art performance while remaining computationally efficient. Our findings challenge the conventional view that object localization in RIS requires explicit spatial modeling, and instead suggest that it can arise naturally from the semantic grounding capability in generative models. We hope this work motivates further exploration of generative representations as a unified foundation for both synthesis and discriminative vision-language tasks.

Acknowledgments

This work was supported in part by the Australian Research Council under Projects DP240101848 and FT230100549.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[3] Huanqia Cai, Sihan Cao, Ruoyi Du, Peng Gao, Steven Hoi, Zhaohui Hou, Shijie Huang, Dengyang Jiang, Xin Jin, Liangchen Li, et al. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

[4] 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.

[5] Xiaofu Chen, Yaxin Luo, Gen Luo, Jiayi Ji, Henghui Ding, and Yiyi Zhou. Dvin: Dynamic visual routing network for weakly supervised referring expression comprehension. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14347–14357, 2025.

[6] Silin Cheng, Yang Liu, Xinwei He, Sebastien Ourselin, Lei Tan, and Gen Luo. Weakmcn: Multi-task collaborative network for weakly supervised referring expression comprehension and segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9175–9185, 2025.

[7] Yong Xien Chng, Henry Zheng, Yizeng Han, Xuchong Qiu, and Gao Huang. Mask grounding for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26573–26583, 2024.

[8] Paul Couairon, Mustafa Shukor, Jean-Emmanuel Haugeard, Matthieu Cord, and Nicolas Thome. Diffcut: Catalyzing zero-shot semantic segmentation with diffusion features and recursive normalized cut. In *Advances in Neural Information Processing Systems*, 2024.

[9] Ming Dai, Wenxuan Cheng, Jiang-jiang Liu, Sen Yang, Wenxiao Cai, Yanpeng Sun, and Wankou Yang. Deris: Decoupling perception and cognition for enhanced referring image segmentation through loopback synergy. *arXiv preprint arXiv:2507.01738*, 2025.

[10] Qiyuan Dai and Sibei Yang. Curriculum point prompting for weakly-supervised referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13711–13722, 2024.

[11] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[12] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2): 179–188, 1936.

[13] Chaofan Gan, Yuanpeng Tu, Xi Chen, Tieyuan Chen, Yuxi Li, Mehrtash Harandi, and Weiyao Lin. Unleashing diffusion transformers for visual correspondence by modulating massive activations. *arXiv preprint arXiv:2505.18584*, 2025.

[14] Jingxuan He, Xiyu Wang, Mengyu Zheng, Xiangyu Zeng, Yunke Wang, and Chang Xu. Rethinking where to edit: Task-aware localization for instruction-based image editing. *arXiv preprint arXiv:2604.20258*, 2026.

[15] Alec Helbling, Tuna Han Salih Meral, Benjamin Hoover, Pinar Yanardag, and Duen Horng Chau. Conceptattention: Diffusion transformers learn highly interpretable features. In *Proceedings of the 42nd International Conference on Machine Learning*, pages 22946–22963, 2025.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[17] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *European conference on computer vision*, pages 108–124. Springer, 2016.

[18] 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.

[19] Anna Kukleva, Enis Simsar, Alessio Tonioni, Muhammad Ferjad Naeem, Federico Tombari, Jan Eric Lenssen, and Bernt Schiele. Refam: Attention magnets for zero-shot referral segmentation. *arXiv preprint arXiv:2509.22650*, 2025.

[20] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024.

[21] Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[23] Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025.

[24] Sihan Liu, Yiwei Ma, Xiaoqing Zhang, Haowei Wang, Jiayi Ji, Xiaoshuai Sun, and Rongrong Ji. Rotated multi-scale interaction network for referring remote sensing image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26658–26668, 2024.

[25] Ting Liu and Siyuan Li. Hybrid global-local representation with augmented spatial guidance for zero-shot referring image segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29634–29643, 2025.

[26] László Lovász. Random walks on graphs. *Combinatorics, Paul erdos is eighty*, 2(1-46):4, 1993.

[27] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.

[28] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*, pages 792–807. Springer, 2016.

[29] Minheng Ni, Yabo Zhang, Kailai Feng, Xiaoming Li, Yiwen Guo, and Wangmeng Zuo. Ref-diff: Zero-shot referring image segmentation with generative models. *arXiv preprint arXiv:2308.16777*, 2023.

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[31] 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.

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[33] Lixiang Ru, Yibing Zhan, Baosheng Yu, and Bo Du. Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16846–16855, 2022.

[34] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[35] Nisarg A Shah, Vibashan VS, and Vishal M Patel. Lqmformer: Language-aware query mask transformer for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12903–12913, 2024.

[36] Chao Shang, Zichen Song, Heqian Qiu, Lanxiao Wang, Fanman Meng, and Hongliang Li. Prompt-driven referring image segmentation with instance contrasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4124–4134, 2024.

[37] Shuyang Sun, Runjia Li, Philip Torr, Xiuye Gu, and Siyang Li. Clip as rnn: Segment countless visual concepts without training endeavor. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13171–13182, 2024.

[38] Yucheng Suo, Linchao Zhu, and Yi Yang. Text augmented spatial-aware zero-shot referring image segmentation. *arXiv preprint arXiv:2310.18049*, 2023.

[39] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14940–14950, 2025.

[40] Meituan LongCat Team, Hanghang Ma, Haoxian Tan, Jiale Huang, Junqiang Wu, Jun-Yan He, Lishuai Gao, Songlin Xiao, Xiaoming Wei, Xiaoqi Ma, et al. Longcat-image technical report. *arXiv preprint arXiv:2512.07584*, 2025.

[41] Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3554–3563, 2024.

[42] Shijie Wang, Dahun Kim, Ali Taalimi, Chen Sun, and Weicheng Kuo. Learning visual grounding from generative vision and language model. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8057–8067. IEEE, 2025.

[43] Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M Alvarez. Freesolo: Learning to segment objects without annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14176–14186, 2022.

[44] Yuji Wang, Jingchen Ni, Yong Liu, Chun Yuan, and Yansong Tang. Iterprime: Zero-shot referring image segmentation with iterative grad-cam refinement and primary word emphasis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8159–8168, 2025.

[45] Yunke Wang, Xiyu Wang, Anh-Dung Dinh, Bo Du, and Chang Xu. Learning to schedule in diffusion probabilistic models. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2478–2488, 2023.

[46] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

[47] Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025.

[48] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13294–13304, 2025.

[49] Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4310–4319, 2022.

[50] Zaiquan Yang, Yuhao Liu, Jiaying Lin, Gerhard Hancke, and Rynson Lau. Boosting weakly supervised referring image segmentation via progressive comprehension. *Advances in Neural Information Processing Systems*, 37:93213–93239, 2024.

[51] Fukun Yin, Shiyu Liu, Yucheng Han, Zhibo Wang, Peng Xing, Rui Wang, Wei Cheng, Yingming Wang, Aojie Li, Zixin Yin, et al. Reasonedit: Towards reasoning-enhanced image editing models. *arXiv preprint arXiv:2511.22625*, 2025.

[52] Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26125–26135, 2025.

[53] Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. Zero-shot referring image segmentation with global-local context features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19456–19465, 2023.

[54] Seonghoon Yu, Paul Hongsuck Seo, and Jeany Son. Pseudo-ris: Distinctive pseudo-supervision generation for referring image segmentation. In *European Conference on Computer Vision*, pages 18–36. Springer, 2024.

[55] Seonghoon Yu, Joonbeom Hong, Joonseok Lee, and Jeany Son. Latent expression generation for referring image segmentation and grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21374–21383, 2025.

[56] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023.

[57] Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. Enabling instructional image editing with in-context generation in large scale diffusion transformer. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[58] Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems*, 37:3058–3093, 2024.

[59] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European conference on computer vision*, pages 696–712. Springer, 2022.

Appendix

A1 Feature Separability: Definition, Rationale, and Implementation

Definition. Let $\mathbf{F}^{(t,l)} \in \mathbb{R}^{N_v \times D}$ denote MMSA output features of clean image tokens within the l-th DiT block at timestep t. Given the ground-truth binary mask, we denote the sets of foreground and background token indices as Ω_{fg} and Ω_{bg}, respectively. The class-wise means are defined as:

$$\begin{aligned} \mu_{\text{fg}}^{(t,l)} &= \frac{1}{|\Omega_{\text{fg}}|} \sum_{i \in \Omega_{\text{fg}}} \mathbf{F}_i^{(t,l)}, \\ \mu_{\text{bg}}^{(t,l)} &= \frac{1}{|\Omega_{\text{bg}}|} \sum_{i \in \Omega_{\text{bg}}} \mathbf{F}_i^{(t,l)}. \end{aligned} \tag{A1}$$

The corresponding within-class variances are defined as:

$$\begin{aligned} \sigma_{\text{fg}}^{2(t,l)} &= \frac{1}{|\Omega_{\text{fg}}|} \sum_{i \in \Omega_{\text{fg}}} \left\| \mathbf{F}_i^{(t,l)} - \mu_{\text{fg}}^{(t,l)} \right\|^2, \\ \sigma_{\text{bg}}^{2(t,l)} &= \frac{1}{|\Omega_{\text{bg}}|} \sum_{i \in \Omega_{\text{bg}}} \left\| \mathbf{F}_i^{(t,l)} - \mu_{\text{bg}}^{(t,l)} \right\|^2. \end{aligned} \tag{A2}$$

The feature separability is formulated as:

$$\mathcal{J}^{(t,l)} = \frac{\left\| \mu_{\text{fg}}^{(t,l)} - \mu_{\text{bg}}^{(t,l)} \right\|^2}{\sigma_{\text{fg}}^{2(t,l)} + \sigma_{\text{bg}}^{2(t,l)}}. \tag{A3}$$

Connection to Fisher Discriminant. The metric $\mathcal{J}^{(t,l)}$ quantifies the ratio of inter-class separation to intra-class variance, evaluating how effectively foreground and background features are disentangled in the representation space. This formulation is closely related to the classical Fisher discriminant criterion for two classes in multivariate space:

$$\mathcal{J} = (\mu_{\text{fg}} - \mu_{\text{bg}})^\top (\Sigma_{\text{fg}} + \Sigma_{\text{bg}})^{-1} (\mu_{\text{fg}} - \mu_{\text{bg}}), \tag{A4}$$

where Σ_{fg} and Σ_{bg} denote the respective covariance matrices. Our formulation $\mathcal{J}^{(t,l)}$ can be viewed as a simplified isotropic variant. By approximating the covariance matrices as isotropic (i.e., $\Sigma \approx \sigma^{2(t,l)}\mathbf{I}$), the inverse covariance simplifies to a scalar division. This yields a tractable and numerically stable measure that is highly suitable for high-dimensional features.

Implementation. We evaluate the feature separability metric $\mathcal{J}^{(t,l)}$ on FLUX.2-klein [21] and Step1X-Edit [23] across six uniformly sampled timesteps. For both models, we extract features from the DiT block located at approximately two-thirds of the model depth ($l = 20$ of 32 for FLUX.2-klein and $l = 40$ of 57 for Step1X-Edit), where semantic representations are typically most pronounced. To obtain the spatial location sets Ω_{fg} and Ω_{bg}, we resize ground-truth segmentation masks to the spatial resolution of clean image tokens. All feature vectors are ℓ_2-normalized prior to computing $\mathcal{J}^{(t,l)}$ to ensure scale consistency across timesteps. For the boxplots in Fig. 1, we conduct this evaluation on the RefCOCO [28] *val* split.

Quantitative results across benchmarks. To quantify this trend beyond RefCOCO, we compute $\mathcal{J}^{(t,l)}$ on 500 randomly sampled examples from each *val* split of RefCOCO, RefCOCO+, and RefCOCOg, and report the median at six stages of the denoising trajectory in Table A1. For Step1X-Edit, the median separability is highest at the earliest evaluated stage on all three datasets. For FLUX.2-klein, the values at the earliest stage already reach 92.7%, 91.1%, and 94.7% of the maximum observed values on RefCOCO, RefCOCO+, and RefCOCOg, respectively. Thus, although FLUX.2-klein sometimes reaches its maximum later in the trajectory, a strong foreground-background discriminative signal is already present at the earliest evaluated timestep.

Table A1: **Quantitative feature separability across the denoising trajectory.** We report the median of $\mathcal{J}^{(t,l)}$ over 500 randomly sampled examples from each *val* split. The maximum of each row is in **bold**.

Model	Dataset	1%	20%	40%	60%	80%	100%
Step1X-Edit (12B)	RefCOCO	**0.403**	0.324	0.358	0.361	0.366	0.361
	RefCOCO+	**0.286**	0.212	0.208	0.206	0.200	0.182
	RefCOCOg	**0.407**	0.330	0.343	0.358	0.372	0.384
FLUX.2-klein (9B)	RefCOCO	0.318	0.296	0.305	0.322	0.342	**0.343**
	RefCOCO+	0.234	0.232	0.239	0.251	**0.257**	0.247
	RefCOCOg	0.301	0.294	0.299	0.314	**0.318**	0.293

Table A2: **Inference efficiency of the DiT backbone in our framework.** Latency and peak GPU memory are measured on a single NVIDIA A100 GPU with a batch size of 1. The size level follows the officially recommended settings.

Model	Size Level	Param.	Latency (ms)	Memory (GB)
FLUX.2-klein	1024	9B	1493.02	22.27
Step1X-Edit	512	12B	603.78	25.31

A2 Inference Efficiency Analysis

We evaluate the inference efficiency of our framework in terms of inference latency and memory usage. All measurements are conducted on a single NVIDIA A100 GPU with a batch size of 1. Following the officially recommended settings, we use a size level of 512 for Step1X-Edit and 1024 for FLUX.2-klein. As shown in Table A2, the DiT backbone of Step1X-Edit achieves an average latency of 603.78 ms with a peak memory usage of 25.31 GB, while FLUX.2-klein requires 1493.02 ms and 22.27 GB. The higher latency of FLUX.2-klein is mainly due to its larger input resolution.

We further compare the complete pipeline with representative baselines in Table A3. Note that Table A2 reports the DiT backbone only, whereas Table A3 measures end-to-end inference. Owing to single-step inference, our method achieves the lowest end-to-end latency, running $1.1\times$, $1.4\times$, and $2.8\times$ faster than HybridGL, Global-Local, and RefAM, respectively. However, the 12B configuration reaches a peak memory of 40.8 GB, most of which is attributable to the underlying editor: a Step1X-Edit forward pass alone requires 39.4 GB.

A3 Impact of Editing Instruction Refinement

We investigate the impact of editing instruction refinement on segmentation performance to understand how instruction clarity influences the model's internal priors. Each referring expression is first converted to "Remove {expression}" and then processed together with the source image by the built-in refinement pipeline of Step1X-Edit [51]. The refiner is instructed to preserve the original meaning, retain instructions that are already clear, and otherwise produce a concrete and actionable editing command. For a fair comparison, we also evaluate Global-Local [53] and the recent state-of-the-art RIS method HybridGL [25] using refined expressions with only the editing directive (e.g., "remove") discarded, since these methods take referring expressions rather than editing commands. As shown in Table A4, refinement generally improves all methods. We attribute this improvement to improved distributional alignment, as the refined prompts more closely match the data distribution seen during training of the image editing model. Although the baselines also benefit from clearer prompts, our method maintains a substantial performance margin. Without refinement, our method outperforms HybridGL by 10.10, 5.32, and 6.45 oIoU on RefCOCO, RefCOCO+, and RefCOCOg, respectively, and under the same refinement pipeline the margins remain 9.80, 5.85, and 7.49 oIoU. Thus, instruction refinement is beneficial but does not explain the advantage of our method.

A4 Qualitative Results of SAM 3

The Segment Anything Model (SAM) 3 [4] takes concept prompts to retrieve segmentation masks for all matching instances in an image. While SAM 3 demonstrates strong capability in producing

Table A3: **End-to-end efficiency comparison with baselines.** All methods are evaluated on the same NVIDIA A100 GPU with 5 warm-up runs, averaging over 500 samples from the RefCOCO *val* split.

Method	End-to-end latency (ms)	Peak memory (GB)
Global-Local (SAM + CLIP) [53]	1203	3.3
HybridGL [25]	942	6.9
RefAM (FLUX.1-dev) [19]	2367	37.0
Ours (Step1X-Edit)	**853**	40.8

Table A4: **Impact of editing instruction refinement**. We compare our method against Global-Local [53] and HybridGL [25] using both the original and refined referring expressions.

Refinement	Method	RefCOCO		RefCOCO+		RefCOCOg	
		oIoU	mIoU	oIoU	mIoU	oIoU	mIoU
×	Global-Local	24.55	31.83	26.62	34.97	28.92	40.66
	HybridGL	41.81	49.48	35.74	43.40	42.47	51.25
	Ours (Step1X-Edit)	**51.91**	**55.79**	**41.06**	**44.39**	**48.92**	**52.43**
✓	Global-Local	24.58	33.44	27.14	36.69	29.36	41.54
	HybridGL	44.14	52.55	38.53	46.78	43.74	52.78
	Ours (Step1X-Edit)	**53.94**	**57.56**	**44.38**	**48.50**	**51.23**	**54.52**

high-quality object boundaries, we observe that it exhibits two fundamental limitations when applied to referring image segmentation. First, SAM 3 often fails to predict the presence of the target object conditioned on the referring expression, resulting in a low recall rate. Second, even when the target object is successfully detected, the model tends to segment all instances belonging to the same semantic category, without adhering to instance-level constraints such as spatial relations specified in the referring expression. We present illustrative examples in Fig. A1 to demonstrate these failure modes. As can be seen, SAM 3 frequently misses the target object entirely, resulting in a high proportion of zero-IoU predictions. In the last example, even when the prompt specifies the position of the "giraffe", the model still produces masks covering all giraffes in the scene.

A5 Limitations and Discussions

We identify three main types of failure cases, as illustrated in Fig. A2. The examples in the first row show that the image editing model struggles with fine-grained object boundaries, leading to imprecise segmentation along detailed contours. The second row demonstrates cases where the model fails to distinguish the referred instance and instead activates on all objects of the same category (e.g., all people or all zebras in the image). The third row highlights situations in which the model cannot capture the complete extent of the target object, resulting in partial segmentation. Importantly, *these limitations primarily stem from the inherent capabilities of the underlying image editing model rather than our framework design*, as the predicted masks remain consistent with the edited regions.

To quantify these failure modes, we randomly sample 1,000 *val* expressions from each dataset and group them using the referring expressions and ground-truth masks: (1) *overlapping instances*, where the target spatially overlaps another instance of the same category; (2) *spatial relations*, where the target requires relational cues; (3) *thin edges*, where the target contains narrow structures that span very few spatial tokens in the latent space; (4) *occlusions*, where the visible target is separated into disconnected regions or has substantial missing regions in its silhouette; and (5) *partial objects*, where the target is truncated by the image boundary or explicitly described as only part of an object. As shown in Table A5, the largest and most consistent degradation occurs for thin edges, which we attribute primarily to the limited spatial resolution of the editor's latent features. One possible extension is to use our prediction as a semantic prompt for SAM to recover finer boundaries; we leave this to future work, as the present framework is intentionally designed to remain segmenter-free.

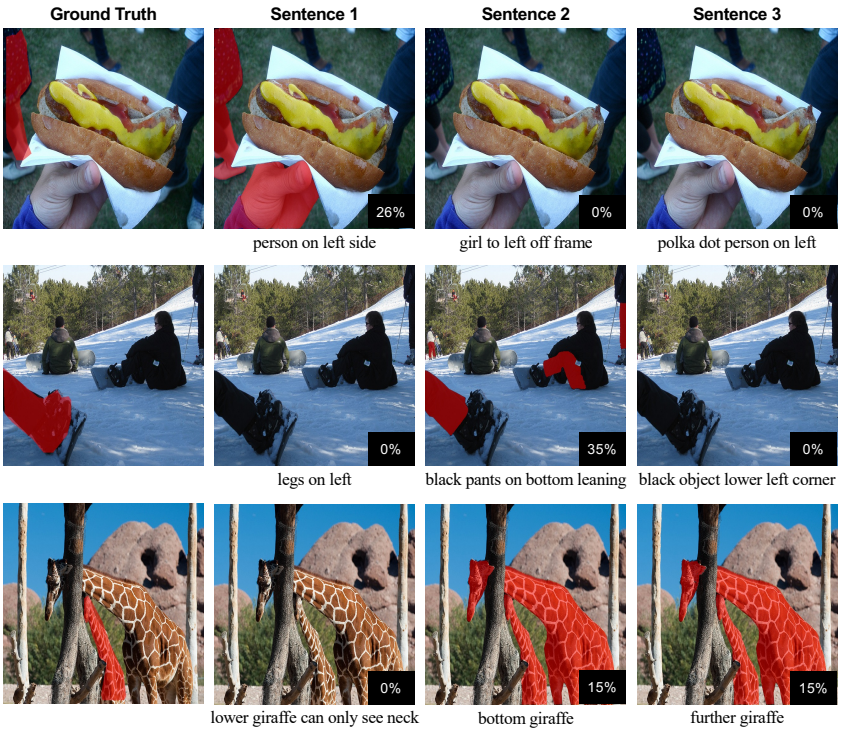

Figure A1: **Qualitative results of SAM 3**. The referring expression is shown at the bottom of each image. The corresponding IoU score is displayed at the bottom-right corner in each image.

Table A5: **Quantitative failure-case analysis.** We report the ratio of each category among 1,000 randomly sampled *val* expressions per dataset and the corresponding oIoU change (Ratio / ΔoIoU). Categories are not mutually exclusive.

Failure case	RefCOCO	RefCOCO+	RefCOCOg
Overlapping instances	$37.6\%/-02.54$	$40.5\%/-01.85$	$27.6\%/-08.52$
Spatial relations	$67.7\%/-04.18$	$17.2\%/-06.61$	$40.5\%/-06.32$
Thin edges	$13.2\%/-16.33$	$13.0\%/-11.49$	$21.4\%/-15.98$
Occlusions	$11.0\%/-12.59$	$12.7\%/-03.26$	$15.5\%/-11.67$
Partial objects	$25.3\%/-02.53$	$24.6\%/-02.72$	$24.3\%/-02.25$

A6 Broader Impact

This work advances zero-shot referring image segmentation by leveraging the implicit localization capability of instruction-based image editing models. Such capability may benefit applications in visual understanding where language-driven object grounding is required. On the other hand, the underlying image editing models repurposed in this work are capable of synthesizing realistic image manipulations, which raises concerns about potential misuse for generating misleading or harmful visual content. However, our framework exclusively extracts internal representations from these models for localization purposes and does not involve any image generation or modification. We therefore do not introduce additional risks beyond those already associated with the pretrained models themselves.



Figure A2: **Examples of failure cases**. The limitations primarily arise from the inherent capabilities of the underlying image editing model, as the predicted masks closely follow the edited regions.

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: The main claims are clearly stated in the abstract and introduction, including the reformulation of referring image segmentation as instruction-based image editing and the effectiveness of our training-free framework. See Abstract and Section 1 for details.

   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: The paper discusses the limitations in Appendix A5, including qualitative and quantitative failure analysis.

   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: [N/A]

Justification: The paper does not include theoretical results.

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 provide implementation details, datasets, metrics, and hyperparameters in Section 4.

   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 will release code upon publication. All datasets used in this paper are publicly available.

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: Datasets, evaluation metrics, and implementation details are specified in Section 4.

   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: We do not report statistical tests, following standard practice with deterministic benchmark evaluation.

   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: Compute setup, latency, and peak GPU memory are described in Appendix A2.

   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.

   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: The paper discusses both potential positive societal impacts and negative societal impacts in Appendix A6.

    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: The paper poses no such risks as it does not involve release of data or models.

    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: All datasets and models are publicly available and used in accordance with their licenses.

    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: The paper does not release 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: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    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.