COMPOSING COMPLEMENTARY DENSE REPRESENTATIONS FOR COST AGGREGATION IN OPEN-VOCABULARY SEMANTIC SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Open-vocabulary semantic segmentation enables dense scene understanding beyond a fixed training vocabulary. Existing cost-aggregation methods, however, typically reuse a single dense visual feature for both cost map construction and spatial guidance. Coupling these objectives within a shared dense feature space induces competition between semantic discrimination and spatial coherence, and simultaneously constrains the diversity of visual representations available across open-vocabulary scenarios. To address these limitations, we introduce **Co**mplementary **D**ense **R**epresentations (CoDR), a novel framework for constructing and composing dense representations for cost aggregation. Specifically, we first introduce Cost Spatial Decoupling, which assigns cost map construction and spatial guidance to distinct representation spaces, freeing both from the constraints of a shared feature. Building on this decoupling, Representation Diversification constructs multiple complementary representations within each space to preserve distinct semantic and spatial cues. Furthermore, we design Scene-Adaptive Representation Composition to hierarchically compose structurally distinct representations for each input scene, replacing a fixed global preference with scene-specific composition. Extensive experiments on five standard benchmarks demonstrate the strong performance of CoDR, which achieves 19.8 mIoU on ADE847. We will release all code and models upon acceptance.

1 INTRODUCTION

Open-vocabulary semantic segmentation (OVS) (Li et al., 2022; Ghiasi et al., 2022) extends pixel-level recognition beyond a closed category set by transferring image–text knowledge to categories absent from mask-supervised training. Vision–language models provide the foundation for this transfer. CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021) established scalable image–language alignment, while LiT (Zhai et al., 2022), SigLIP (Zhai et al., 2023), and EVA-CLIP (Sun et al., 2023) improved zero-shot transfer through new objectives and training strategies. Early OVS systems apply this alignment to dense prediction via pixel embeddings (Li et al., 2022), image-level supervision (Ghiasi et al., 2022), or class-agnostic masks with zero-shot classification (Ding et al., 2022; Xu et al., 2022b). Later works explore dense CLIP adaptation and visual grouping (Rao et al., 2022; Xu et al., 2022a), mask-level training (Liang et al., 2023; Ding et al., 2023), and prompts or side adapters (Zhou et al., 2023; Xu et al., 2023b). These advances highlight the importance of dense visual representations for OVS.

Among these developments, CAT-Seg (Cho et al., 2024) introduced cost aggregation, which constructs image–text cost maps from dense visual representations and progressively refines them with spatial and category context. The resulting class-conditioned cost volume preserves semantic matching for subsequent spatial and category reasoning. Related methods further strengthen this paradigm through embedding balancing and hyperspherical geometry (Shan et al., 2024; Peng et al., 2025), mask adaptation and decoding (Xie et al., 2024; Li et al., 2025), and convolutional features or additional spatial knowledge (Yu et al., 2023; Wang et al., 2025).

Despite these advances, a representation bottleneck in cost aggregation remains underexplored. As shown in Figure 1(a), existing methods commonly reuse a dense visual feature to compute

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

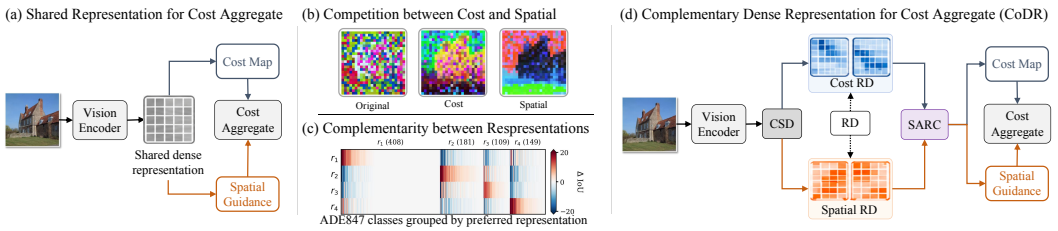


Figure 1: Motivation of CoDR. (a) The shared representation serves both cost construction and spatial guidance. (b) Cost-only and spatial-only training change the shared representation differently, exposing competing demands. (See Appendix A.2 for gradient-based diagnostics.) (c) The cost representations r_1–r_4 in Eq. (4) favor different categories. Color is their per-category IoU minus the four-representation mean, $\Delta\mathrm{IoU}_{k,c} = \mathrm{IoU}_{k,c} - \frac{1}{4}\sum_{j=1}^{4}\mathrm{IoU}_{j,c}$. (d) CoDR uses Cost Spatial Decoupling (CSD) to separate cost construction from spatial guidance, Representation Diversification (RD) to construct complementary representations in both spaces, and Scene-Adaptive Representation Composition (SARC) to compose them for each scene.

cost maps and guide their spatial aggregation. This design has two limitations: **(1) Competing objectives in shared representations.** A shared dense representation must provide text-aligned, category-discriminative responses for cost construction while preserving spatial structure for spatial guidance. Figure 1(b) shows that optimizing the two objectives separately produces distinct feature changes. Updates that improve text-aligned category discrimination can disrupt local spatial structure, whereas those that enforce spatial coherence can weaken semantic distinctions, producing conflicting optimization directions. **(2) Category-dependent representation preference.** A fixed representation favors some open-vocabulary categories during cost aggregation. The four dense cost representations examined in Figure 1(c) reveal this specificity: they favor different categories, and no single representation consistently dominates across the vocabulary. This leads us to investigate a core question: *can cost aggregation construct complementary representations that mitigate competition while exploiting the nonredundant information preserved by different representations?*

To address this issue, we introduce **Co**mplementary **D**ense **R**epresentations (CoDR), a framework for constructing and composing dense representations for cost aggregation. Figure 1(d) organizes the two observations into a unified process: separate cost construction from spatial guidance, diversify the representations available to each, and compose them for the input scene.

Specifically, we first introduce Cost Spatial Decoupling (CSD), which establishes distinct cost and spatial representation spaces through asymmetric token flow. This directed information flow allows text-aligned semantic correlation and local spatial structure to specialize without forcing both objectives into a shared dense feature, while their outputs still interact during cost aggregation. Built on this separation, Representation Diversification (RD) constructs multiple complementary representations within both spaces, preserving semantic correspondences and spatial structures that a single representation may suppress. Scene-Adaptive Representation Composition (SARC) then combines them using scene-specific coefficients. Lower-loss preference supervision gives more weight to the representation requiring less task-specific adjustment for each input instead of imposing one global choice. With a ViT-L/14 backbone, CoDR achieves 19.8 mIoU on ADE847, a **+2.2 mIoU** gain over the strongest prior result. Our contributions are threefold:

- We introduce CoDR, a complementary-representation framework for cost aggregation, which employs CSD to establish separate cost and spatial representation spaces through asymmetric token flow, thereby alleviating the competition caused by a shared dense feature.
- We develop RD and SARC to preserve complementary cost and spatial representations and compose them for each input scene using lower-loss preference supervision.
- Extensive experiments demonstrate that CoDR achieves state-of-the-art performance across five standard OVS benchmarks, with particularly strong gains on large-vocabulary datasets.

2 Related Work

Open-Vocabulary Segmentation. Open-vocabulary segmentation connects image–language pretraining to dense prediction through several interfaces. Some methods learn language-aligned pixel embeddings (Li et al., 2022); others transfer image-level supervision (Ghiasi et al., 2022) or classify class-agnostic masks with text-conditioned classifiers (Ding et al., 2022; Xu et al., 2022b). Mask-based approaches build on architectures that separate region formation from recognition (Cheng et al., 2021; 2022); universal variants extend this interface across segmentation tasks (Jain et al., 2023; Li et al., 2023). Later variants adapt CLIP through mask supervision, prompts, side networks, or universal decoders (Liang et al., 2023; Ding et al., 2023; Zhou et al., 2023; Xu et al., 2023b; Qin et al., 2023). Other methods use image–text pairs and generated or category-agnostic cues to reduce dense annotation (Zou et al., 2023; Cha et al., 2023; He et al., 2023; Chen et al., 2023). Cost aggregation offers another interface. CAT-Seg (Cho et al., 2024) constructs a cost volume from image–text similarities and refines it through spatial and class aggregation. PCA-Seg (Yin et al., 2026) uses parallel spatial and class aggregation to reduce interference between semantics and spatial context. Other related works improve embeddings (Shan et al., 2024; Peng et al., 2025), decoders and adaptation (Xie et al., 2024; Li et al., 2025), or introduce spatial priors (Wang et al., 2025). CoDR establishes separate representation spaces for cost construction and spatial guidance and composes multiple complementary representations.

Dense Representations for OVS. CLIP's image-level pretraining does not guarantee spatially coherent patch features. DenseCLIP (Rao et al., 2022), GroupViT (Xu et al., 2022a), and CLIP-S^4 (He et al., 2023) address this limitation through contextual prompting, visual grouping, or self-supervision. Training-free methods instead modify internal patch relations: SCLIP (Wang et al., 2024) uses query–query and key–key correlations, ProxyCLIP (Lan et al., 2024) imports proxy attention, CorrCLIP (Zhang et al., 2025) reconstructs patch correlations, and CLIPer (Sun et al., 2025) aggregates spatial features hierarchically. Analyses confirm that attention construction affects dense localization (Shao et al., 2024). Self-supervised DINO models provide another source of patch-level structure (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2025). Unlike approaches seeking one stronger dense feature, CoDR retains and composes multiple representations in separate cost and spatial spaces.

3 Method

3.1 Problem Formulation and Overview

Given an image I and category vocabulary $\mathcal{V} = \{v_k\}_{k=1}^{K}$, open-vocabulary semantic segmentation predicts a pixel-wise label map over K categories. First, an L-layer CLIP (Radford et al., 2021) visual Transformer (Dosovitskiy et al., 2021) maps I to visual tokens $H = [H_{\text{cls}}, H_p]$, where H_{cls} is the CLS token and H_p contains N patch hidden states. The CLIP head f_{head} then transforms H_p into the dense representation $X_c = f_{\text{head}}(H_p) \in \mathbb{R}^{N\times d}$ used for image–text matching, while the CLIP text encoder maps $\mathcal{V}$ to category embeddings $T \in \mathbb{R}^{K\times d}$.

Cost-aggregation methods row-wise ℓ_2-normalize X_c and T and compute the class-conditioned cost map $C = \widehat{X}_c\widehat{T}^{\top}$. The same X_c is mapped to the spatial representation $X_s = \mathcal{G}(X_c)$. After cost embedding Φ, the cost map is refined by spatial aggregation $\mathcal{A}_{\text{spa}}$ and class aggregation $\mathcal{A}_{\text{cat}}$, and is finally decoded by f_{dec}:

$$Y_{\text{shared}} = f_{\text{dec}}(\mathcal{A}_{\text{cat}}(\mathcal{A}_{\text{spa}}(\Phi(C), X_s), T)). \tag{1}$$

Appendix A.1 details these CAT-Seg operators. Both C and X_s are derived from the same X_c, forcing one representation to support both text-aligned category discrimination and local spatial coherence. As shown in Figure 2, CoDR addresses this bottleneck in three stages. Cost Spatial Decoupling (CSD) separates cost construction from spatial guidance through token flow and late representation paths; Representation Diversification (RD) constructs complementary representations in both spaces; and Scene-Adaptive Representation Composition (SARC) composes the resulting representations for each scene.

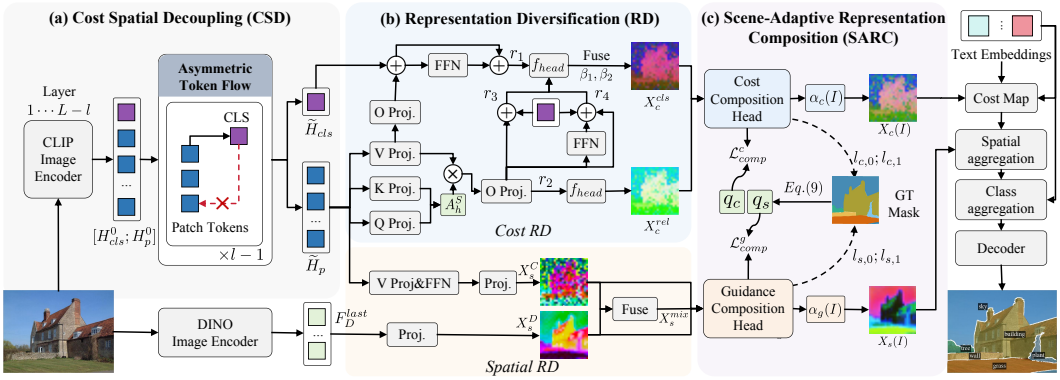


Figure 2: Overview of CoDR. CSD separates cost construction from spatial guidance through asymmetric token flow, RD constructs complementary representations within both spaces, and SARC composes them with scene-conditioned weights learned from lower-loss preference supervision.

3.2 Cost Spatial Decoupling

CLIP applies its image-level contrastive objective to the CLS token, causing late patch tokens to increasingly support CLS aggregation and cross-modal alignment and thus weakening their independence as local representations. CSD therefore progressively decouples the last l layers. The first $l-1$ layers specialize the tokens: CLS focuses on semantic alignment, while patches retain spatially independent representations; the last layer maps them to the cost and spatial paths.

Asymmetric token flow. To let CLS focus on semantic alignment while patch tokens retain the local structure and spatial variation required by dense prediction, CSD preserves patch-to-CLS aggregation while blocking the reverse CLS-to-patch broadcast. Specifically, layers $j = L-l+1, \ldots, L-1$ use the following directed attention mask:

$$A_h^j = \mathrm{softmax}\left(\frac{Q_h^j {K_h^j}^\top}{\sqrt{d_h}} + M^j\right), \qquad M_{p,q}^j = \begin{cases} -\infty, & p > 1,\ q = 1, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

Here, h indexes attention heads, and p, q index token positions. Position 1 denotes CLS and the remaining positions denote patch tokens. When $p > 1$ and $q = 1$, the corresponding attention logit is set to $-\infty$, blocking information transfer from CLS to patch while CLS can still attend to all patch tokens. These asymmetric layers produce $\widetilde{H} = [\widetilde{H}_{\mathrm{cls}}, \widetilde{H}_p]$.

Distinct cost and spatial paths. In the last layer, the cost path receives both $\widetilde{H}_{\mathrm{cls}}$ and $\widetilde{H}_p$ to construct cost representations for image–text matching. The CLIP spatial path receives only $\widetilde{H}_p$ and uses an independent value projection, residual FFN, and dense projection initialized from their CLIP counterparts. The two paths no longer share the same late dense representation and interact only during cost aggregation.

3.3 Representation Diversification

Although CSD separates cost construction from spatial guidance, it does not expand the available information: the cost space still uses one construction and the spatial space one visual source. RD diversifies cost representations before image–text correlation and complements CLIP spatial structure with an independent visual source.

Complementary cost representations. Figure 1(c) shows category-dependent strengths among independently constructed cost representations, motivating RD to retain complementary alternatives. Building on this CLS–patch specialization, RD varies two factors: patch information organized as local content or self-similarity relations, and aligned semantics introduced by a CLS residual and FFN refinement. Specifically, r_1 combines local token content, CLS semantics, and FFN refinement; r_2 models self-correlation over the complete CLS–patch sequence; r_3 explicitly injects CLS

semantics into r_2; and r_4 re-parameterizes the fused relational and semantic information through the FFN. Appendix A.3 analyzes the rationale for the cost-representation design.

Let Q_h, K_h, and V_h be the projections of $\mathrm{Norm}_1(\widetilde{H})$ in attention head h, and let $\delta(\cdot)$ denote row-wise softmax. The similarity-attention representation is defined as:

$$A_h^S = \delta(Q_h Q_h^\top/\sqrt{d_h}) + \delta(K_h K_h^\top/\sqrt{d_h}), \qquad S(\widetilde{H}) = P_o\big(\mathrm{Concat}_h(A_h^S V_h)\big), \tag{3}$$

where P_o includes output projection, layer scaling, and DropPath. $S(\widetilde{H})$ models relations over the complete CLS–patch sequence, following evidence that CLIP self-similarity supports dense prediction (Wang et al., 2024; Shao et al., 2024). Let B_{cls} be the broadcast CLS residual and $\mathcal{F}(Z) = Z + \mathrm{DropPath}(\mathrm{FFN}(\mathrm{Norm}_2(Z)))$ the residual FFN. The four token-level cost constructions are:

$$\begin{aligned} r_1(\widetilde{H}) &= \mathcal{F}\Big(\widetilde{H} + P_o(\mathrm{Concat}_h\, V_h) + B_{\mathrm{cls}}\Big), & r_2(\widetilde{H}) &= S(\widetilde{H}), \\ r_3(\widetilde{H}) &= S(\widetilde{H}) + B_{\mathrm{cls}}, & r_4(\widetilde{H}) &= \mathcal{F}\Big(S(\widetilde{H}) + B_{\mathrm{cls}}\Big). \end{aligned} \tag{4}$$

The constructions are finally passed through the CLIP head to obtain dense cost representations. Let Π_p retain patch positions. The i-th representation is $X_c^i = f_{\mathrm{head}}(\Pi_p(r_i(\widetilde{H}))),\ i \in \{1,2,3,4\}$. RD then forms a relation-centric cost representation and a CLS-enhanced cost representation:

$$X_c^{\mathrm{rel}} = X_c^2, \qquad X_c^{\mathrm{cls}} = X_c^1 + \beta_1 X_c^3 + \beta_2 X_c^4. \tag{5}$$

Here, β_1 and β_2 are learnable parameters. X_c^{rel} emphasizes relations modeled over the complete CLS–patch sequence, whereas X_c^{cls} combines local token content, explicit CLS semantics, and FFN-refined outputs, forming two complementary cost representations.

Complementary spatial representations. The spatial space combines task-aligned CLIP and independently pretrained DINOv3 (Siméoni et al., 2025) structures. We use the final-layer DINOv3 patch feature as the complementary spatial source. After spatial alignment, independent 3×3 projections followed by ReLU map both sources to $d_s = 128$ channels, yielding $X_s^C = p_C(X_{s,\mathrm{raw}}^C)$ and $X_s^D = p_D(F_D^{\mathrm{last}}(I))$. Here, $F_D^{\mathrm{last}}(I)$ denotes the patch feature from the final DINOv3 Transformer layer. The projections preserve source-specific statistics while aligning their dimensions. The task-optimized CLIP path carries category semantics, whereas DINOv3 supplies independently learned region and boundary structure only to the spatial path. Their complementarity is therefore spatial; category scores remain defined by CLIP cost representations and text embeddings.

3.4 Scene-Adaptive Representation Composition

RD preserves complementary representations, but a fixed composition cannot exploit their respective strengths as scenes vary. SARC therefore uses lightweight cost and spatial heads h_c and h_s to infer scene-conditioned weights. Specifically, for the normalized cost representations:

$$\begin{aligned} \alpha_c(I) &= \sigma\Big(h_c\Big([\mathrm{GAP}(\hat{X}_c^{\mathrm{cls}}); \mathrm{GAP}(\hat{X}_c^{\mathrm{rel}})]\Big)\Big), \\ X_c(I) &= [1-\alpha_c(I)]X_c^{\mathrm{cls}} + \alpha_c(I) X_c^{\mathrm{rel}}. \end{aligned} \tag{6}$$

The predicted $\alpha_c(I)$ weights the CLS-enhanced and relation-centric representations for each scene. Hats denote row-wise ℓ_2 normalization. Here, GAP denotes global average pooling, $[u; v]$ denotes channel-wise concatenation, and σ is sigmoid.

The spatial representation follows the same principle. It first forms a channel-calibrated mix and then predicts its scene-conditioned interpolation with the DINO representation:

$$\begin{aligned} X_s^{\mathrm{mix}} &= w_C \odot X_s^C + w_D \odot X_s^D, \\ \alpha_s(I) &= \sigma\big(h_s\big([\mathrm{GAP}(X_s^C); \mathrm{GAP}(X_s^D)]\big)\big), \\ X_s(I) &= [1-\alpha_s(I)]X_s^{\mathrm{mix}} + \alpha_s(I) X_s^D. \end{aligned} \tag{7}$$

Here, $w_C, w_D \in \mathbb{R}^{d_s}$ are channel-wise softmax weights satisfying $w_C + w_D = \mathbf{1}$. The symbol $\odot$ denotes element-wise multiplication: each channel weight is broadcast across all spatial positions.

As in the cost space, the scalar $\alpha_s(I)$ is shared across all positions and channels. The spatial head pools X_s^C and X_s^D to predict this scalar, which then interpolates between X_s^{mix} and X_s^D.

To determine the preferable representation, we regard the one producing a lower per-image segmentation loss as better matched to the current scene and thus assign it a higher composition weight. Accordingly, we convert the relative losses into soft targets for both heads. Let $m \in \{c, s\}$ index the cost or spatial space. For image i, $\ell_{m,0}^i$ and $\ell_{m,1}^i$ denote the segmentation losses from the two choices. They correspond to $(X_c^{\text{cls}}, X_c^{\text{rel}})$ for $m = c$ and $(X_s^{\text{mix}}, X_s^D)$ for $m = s$. The soft preference and composition loss are defined as:

$$
\begin{aligned}
q_m^i &= \sigma\left(\frac{\ell_{m,0}^i - \ell_{m,1}^i}{T_m\left[\frac{1}{2}(\ell_{m,0}^i + \ell_{m,1}^i) + \epsilon\right]}\right), \\
\mathcal{L}_{\text{comp}}^m &= \frac{1}{B}\sum_i \text{BCE}\big(\text{logit}(\alpha_m(I_i)), \text{sg}(q_m^i)\big)\,.
\end{aligned}
\tag{8}
$$

Here, B is the batch size, T_m is the preference temperature, and ϵ stabilizes the normalized loss difference; sg denotes stop-gradient, and BCE is binary cross-entropy applied to logits. Implementation details are provided in Appendix A.1.

3.5 Training and Inference

The composed cost representation $X_c(I)$ yields $C(I,T)$; $C(I,T)$ and the composed spatial representation $X_s(I)$ then replace C and X_s, respectively, in Eq. (1). The complete training objective is $\mathcal{L} = \mathcal{L}_{\text{seg}} + \lambda_s\mathcal{L}_{\text{comp}}^s + \lambda_c\mathcal{L}_{\text{comp}}^c$. Optimization is staged according to its gradient paths. We first train CSD, RD, aggregation, and decoding with $\mathcal{L}_{\text{seg}}$. After freezing this representation model, each composition head is optimized with the segmentation loss and its corresponding preference term. The additional per-representation passes are removed at inference.

4 Experiments

4.1 Experimental Setup

Datasets and evaluation. We train CoDR and all ablated variants on COCO-Stuff (Lin et al., 2014; Caesar et al., 2018). We evaluate our method on widely used OVS benchmarks, including ADE20K with 847 classes (ADE847) and 150 classes (ADE150) (Zhou et al., 2019), Pascal Context with 459 classes (PC459) and 59 classes (PC59) (Mottaghi et al., 2014), and Pascal VOC with 20 foreground classes (VOC20) (Everingham et al., 2010). Performance is measured by mean Intersection over Union (mIoU).

Implementation details. Unless stated otherwise, ablations pair EVA02-CLIP-L/14@336 (Fang et al., 2023; Sun et al., 2023) with DINOv3-L (Siméoni et al., 2025) spatial features and apply sliding-window inference. We implement all models in PyTorch (Paszke et al., 2019) on Detectron2 (Wu et al., 2019). Optimization uses four L40S GPUs, AdamW (Loshchilov & Hutter, 2019), a learning rate of 2×10^{-4}, weight decay of 10^{-4}, cosine decay, 80k iterations, and a total batch size of 8. Data augmentation consists of 384×384 random crops, SSD color augmentation, and horizontal flips. We train with dense BCE and the single prompt "A photo of a $\{\}$ in the scene". Inference timings are reported in Appendix A.8.

4.2 Comparison with Prior Methods

Table 1 compares CoDR with representative OVS methods across base- and large-scale visual backbones. With ViT-B/16, CoDR achieves the best results among the compared methods on all five benchmarks. With ViT-L/14, CoDR reaches 19.8 mIoU on ADE847 (+2.2 mIoU over the strongest prior result) and attains the highest mIoU among the compared methods on each of the remaining four benchmarks. DeCLIP is the most closely related baseline because it likewise introduces DINO spatial knowledge into dense CLIP representations. Compared with DeCLIP, CoDR achieves respective gains of +2.2, +1.4, +1.4, +0.3, and +0.2 mIoU on ADE847, ADE150, PC459, PC59, and

Table 1: Comparison results across five open-vocabulary segmentation benchmarks (mIoU).

Method	Backbone	Dataset	ADE847	ADE150	PC459	PC59	VOC20
ZSSeg (Xu et al., 2022b)	ViT-B/16	COCO-Stuff	7.0	20.5	–	47.7	88.4
EBSeg (Shan et al., 2024)	ViT-B/16	COCO-Stuff	11.1	30.0	17.3	56.7	94.6
CAT-Seg (Cho et al., 2024)	ViT-B/16	COCO-Stuff	12.0	31.8	19.0	57.5	94.6
DeCLIP (Wang et al., 2025)	ViT-B/16	COCO-Stuff	15.3	36.3	21.4	60.6	96.6
Mask-Adapter (Li et al., 2025)	ConvNeXt-B	COCO-Stuff	14.2	35.6	17.9	58.4	95.1
CoDR (Ours)	ViT-B/16	COCO-Stuff	**15.8**	**36.8**	**22.6**	**61.5**	**97.1**
ODISE (Xu et al., 2023a)	ViT-L/14	COCO-Panoptic	11.1	29.9	14.5	57.3	–
FC-CLIP (Yu et al., 2023)	ConvNeXt-L	COCO-Panoptic	14.8	34.1	18.2	58.4	95.4
EBSeg (Shan et al., 2024)	ViT-L/14	COCO-Stuff	13.7	32.8	21.0	60.2	96.4
CAT-Seg (Cho et al., 2024)	ViT-L/14	COCO-Stuff	16.0	37.9	23.8	63.3	97.0
DeCLIP (Wang et al., 2025)	ViT-L/14	COCO-Stuff	17.6	40.7	25.9	63.9	97.7
Mask-Adapter (Li et al., 2025)	ConvNeXt-L	COCO-Stuff	16.2	38.2	22.7	60.4	95.8
H-CLIP (Peng et al., 2025)	ViT-L/14	COCO-Stuff	16.5	38.4	24.2	64.1	97.7
CoDR (Ours)	ViT-L/14	COCO-Stuff	**19.8**	**42.1**	**27.3**	**64.2**	**97.9**

VOC20. The larger gains on ADE847, ADE150, and PC459 suggest that complementary representations are particularly beneficial for category discrimination in larger label spaces.

4.3 Component Ablation

Table 2 evaluates the components of CoDR. The shared EVA02-CLIP+DINOv3-L reference in row B reaches 18.5 mIoU on ADE847 but remains 1.3 mIoU below the complete model, showing that the gain cannot be attributed to DINO alone. Rows A and C–F form the cumulative CoDR path. CSD improves all five benchmarks, including gains of 0.5 mIoU on ADE847 and 1.3 mIoU on ADE150, supporting the separation of cost construction from spatial guidance. Cost RD further raises ADE847 from 18.4 to 19.0 using mean composition without DINO, showing that representation diversification provides gains independently of the auxiliary spatial encoder. Spatial RD improves four of five benchmarks, with gains of 0.6 mIoU on ADE150 and 0.8 mIoU on PC59, demonstrating the value of complementary spatial representations beyond the cost space. Finally, SARC raises ADE847 to 19.8 mIoU and improves ADE150, PC459, and PC59 through scene-conditioned composition. Table 5 provides a controlled comparison between mean composition, SARC, and the per-image Oracle with fixed representations.

Table 2: Component ablation of CoDR. B provides a shared EVA02-CLIP+DINOv3-L reference.

ID	Setting	ADE847	ADE150	PC459	PC59	VOC20
A	Shared (EVA02-CLIP)	17.9	40.3	25.8	63.5	97.8
B	A + DINOv3-L	18.5	41.5	26.7	63.8	97.8
C	A + CSD	18.4	41.6	26.6	63.7	97.9
D	C + Cost RD	19.0	41.4	26.6	63.3	**98.1**
E	D + Spatial RD	19.3	42.0	27.1	64.1	98.0
F	E + SARC (CoDR)	**19.8**	**42.1**	**27.3**	**64.2**	97.9

4.4 Analysis and Diagnostics

Effectiveness of CSD. CSD's asymmetric topology blocks direct CLS-to-patch transfer while preserving patch-to-CLS aggregation. Table 3 reports patch drift, the cosine distance between adapted and pretrained patch features, and CLS-patch cosine similarity. Both decrease with asymmetric token flow, indicating less deviation from pretrained patch features and lower similarity between CLS and patch features. These observations support the role of asymmetric token flow in reducing CLS-patch coupling while preserving pretrained patch representations. Figure 3 visualizes feature distributions in the shared, spatial, and cost spaces. Compared with the shared representation, the separated spatial and cost spaces develop distinct class structures with clearer separation, consistent

Table 3: Effect of asymmetric token flow on ADE150. Patch drift and CLS-patch cosine similarity decrease with asymmetric token flow.

Asymmetric token flow	Patch drift↓	CLS-patch cosine↓
×	0.27	0.57
✓	**0.21**	**0.47**

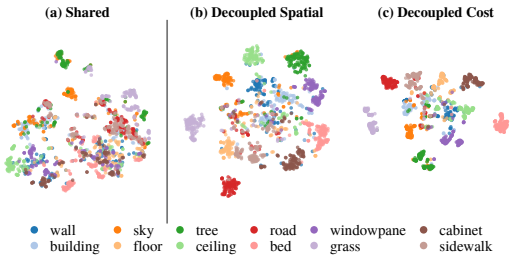


Figure 3: t-SNE comparison of patch representations on ADE150. Compared with the shared representation, CSD allows the spatial and cost spaces to develop distinct feature distributions with clearer class separation.

Table 4: Preference rates (%) of jointly trained cost and spatial representations on ADE847, excluding exact ties.

Cost space				Spatial space	
X_c^1	X_c^2	X_c^3	X_c^4	X_s^D	X_s^{mix}
12.8	59.0	16.8	11.5	49.1	50.9

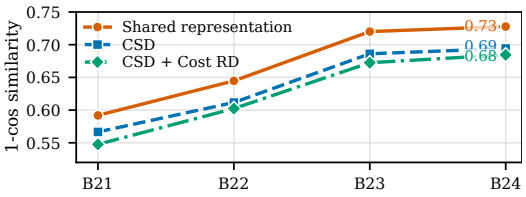


Figure 4: Late-layer CLIP patch-feature deviation from the pretrained encoder over layers 21–24, measured by 1−cosine similarity. Relative to the shared baseline, CSD and Cost RD progressively reduce feature displacement, indicating better preservation of pretrained CLIP knowledge.

with their different optimization objectives. Table 2 separately shows that adding CSD improves segmentation performance across all five benchmarks.

Figure 4 further examines patch-feature drift across late CLIP layers. The shared model deviates most from pretrained patch geometry, whereas CSD substantially reduces this displacement. This suggests that CSD reduces the need for one representation to accommodate competing objectives. Cost RD brings the features closer still, consistent with complementary representations offering alternative adaptation paths for different inputs instead of reshaping a single shared feature. The smaller displacement better preserves pretrained geometry for open-vocabulary generalization.

Analysis of RD complementarity. The observation in Section 3.3 shows that different classes favor different independently trained cost representations. Table 4 examines whether these distinctions persist after joint training. In the cost space, the four representations are each preferred on 11.5–59.0% of the cases, indicating that the model does not concentrate on a single construction. To test whether each remains independently predictive after joint training, we evaluate the same model four times, retaining one cost representation at a time and setting the other three composition weights to zero. The resulting scores range from 18.3 to 18.8 mIoU on ADE847, all below the 19.8 mIoU of joint composition. The individual representations are therefore effective and provide complementary gains when combined. In the spatial space, X_s^D and X_s^{mix} are preferred on 49.1% and 50.9% of ADE847 images, respectively. Both spaces thus retain complementary representations after joint training. We present complementary local predictions and distinct patch structures in Appendices A.5 and A.6, respectively, providing further evidence of representation complementarity.

Scene adaptivity of SARC. We compare SARC with scene-independent mean composition while keeping the representations fixed. As shown in Table 5, SARC improves ADE847 from 19.0 to 19.2 mIoU with only Cost RD, from 19.1 to 19.4 mIoU with only Spatial RD, and from 19.4 to 19.8 mIoU when both spaces are enabled.

These consistent gains show that input-conditioned composition benefits both representation spaces, with the largest improvement when they are used jointly. The corresponding per-image endpoint Oracle results reach 21.1, 20.5, and 22.1 mIoU. Because the Oracle uses test ground truth to select between endpoints, these results serve only as a diagnostic of the remaining potential for scene-specific composition.

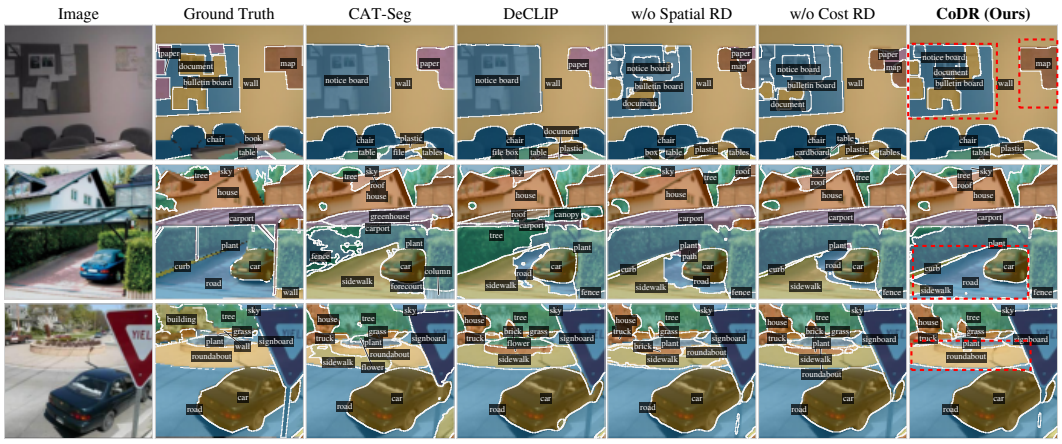


Figure 5: Qualitative ADE847 comparison with Cost RD and Spatial RD ablations. Red boxes highlight regions of difference. Additional results are in Appendix A.9.

Table 5: Comparison of mean, SARC, and per-image Oracle composition on ADE847 (mIoU). † The Oracle uses test ground truth to select the best representation.

Composition	Cost	Spatial	Both
Mean	19.0	19.1	19.4
+SARC	**19.2**	**19.4**	**19.8**
Oracle†	21.1	20.5	22.1

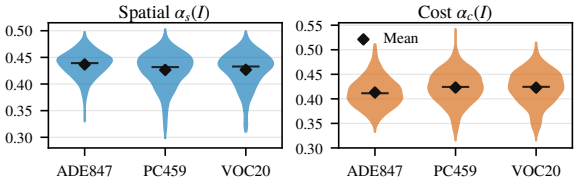


Figure 6: Input-conditioned SARC coefficients across three visual domains. $\alpha_s(I)$ composes X_s^D and X_s^{mix}, while $\alpha_c(I)$ composes X_c^{rel} and X_c^{cls}. Diamonds and bars indicate means and medians.

Figure 6 further examines whether the learned coefficients reduce to fixed global preferences. Across the three visual domains, both coefficients vary across inputs rather than collapsing to constants. Together with the performance gains in Tables 2 and 5, these distributions support the scene adaptivity of SARC. Appendix A.4 compares additional cost-composition strategies.

4.5 Qualitative Results

Figure 5 compares CoDR with CAT-Seg, DeCLIP, and two RD ablations on ADE847. Compared with CAT-Seg and DeCLIP, CoDR more reliably distinguishes visually similar categories while preserving coherent regions and object boundaries. Removing Cost RD primarily introduces errors in fine-grained category assignments, whereas removing Spatial RD reduces region continuity and boundary accuracy. The complete model addresses both types of error and produces predictions that more closely match the ground truth.

5 Conclusion

We presented CoDR, a framework that replaces the single shared representation in cost aggregation with separated and complementary dense representations. In CoDR, CSD separates cost construction from spatial guidance through asymmetric token flow, RD diversifies the representations within both spaces, and SARC composes them according to each input scene. Across five OVS benchmarks, CoDR achieves strong performance, including 19.8 mIoU on ADE847, with larger gains on datasets containing broader label spaces. The jointly trained representations retain nonredundant strengths, supporting their adaptive composition. These findings establish complementary dense representations as an effective alternative to a single shared feature for cost aggregation.

AI USE STATEMENT

We used generative AI tools to assist with translation and language polishing, including correcting grammatical errors, refining sentence structure, and improving the clarity and readability of the manuscript. We reviewed all AI-assisted revisions before incorporating them into the manuscript. These tools were not used to design experiments, implement the proposed method, process data, or generate experimental results. Experimental results were interpreted independently by the authors. The research contributions and conclusions are the work of the authors, who take responsibility for the final text, claims, and reported results.

REPRODUCIBILITY STATEMENT

Section 3 specifies the method, while Section 4 and Appendix A.1 provide the training and evaluation settings. Additional diagnostic and ablation protocols are documented in Appendix Sections A.2–A.7. We will release code and models upon acceptance.

REFERENCES

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*, pp. 1209–1218, 2018.

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*, 2021.

Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Jun Chen, Deyao Zhu, Guocheng Qian, Bernard Ghanem, Zhicheng Yan, Chenchen Zhu, Fanyi Xiao, Sean Chang Culatana, and Mohamed Elhoseiny. Exploring open-vocabulary semantic segmentation from clip vision encoder distillation only. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Advances in Neural Information Processing Systems*, 2021.

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*, 2022.

Seokju Cho, Heeseong Shin, Sunghwan Hong, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Zheng Ding, Jieke Wang, and Zhuowen Tu. Open-vocabulary universal image segmentation with maskclip. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010. doi: 10.1007/s11263-009-0275-4.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, 2022.

Wenbin He, Suphanut Jamonnak, Liang Gou, and Liu Ren. CLIP-S^4: Language-guided self-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Mengcheng Lan, Chaofeng Chen, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Proxyclip: Proxy attention improves clip for open-vocabulary segmentation. In *European Conference on Computer Vision*, 2024.

Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022.

Feng Li, Hao Zhang, Huaizhe Xu, Shilong Liu, Lei Zhang, Lionel M. Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Yongkang Li, Tianheng Cheng, Bin Feng, Wenyu Liu, and Xinggang Wang. Mask-adapter: The devil is in the masks for open-vocabulary segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

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*, pp. 891–898, 2014.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Zelin Peng, Zhengqin Xu, Zhilin Zeng, Yu Huang, Yaoming Wang, and Wei Shen. Parameter-efficient fine-tuning in hyperspherical space for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Jie Qin, Jie Wu, Pengxiang Yan, Ming Li, Yuxi Ren, Xuefeng Xiao, Yitong Wang, Rui Wang, Shilei Wen, Xin Pan, and Xingang Wang. Freeseg: Unified, universal and open-vocabulary image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Xiangheng Shan, Dongyue Wu, Guilin Zhu, Yuanjie Shao, Nong Sang, and Changxin Gao. Open-vocabulary semantic segmentation with image embedding balancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Tong Shao, Zhuotao Tian, Hang Zhao, and Jingyong Su. Explore the potential of clip for training-free open vocabulary semantic segmentation. In *European Conference on Computer Vision*, 2024.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. arXiv preprint arXiv:2508.10104, 2025.

Lin Sun, Jiale Cao, Jin Xie, Xiaoheng Jiang, and Yanwei Pang. Cliper: Hierarchically improving spatial representation of clip for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

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

Feng Wang, Jieru Mei, and Alan Yuille. Sclip: Rethinking self-attention for dense vision-language inference. In *European Conference on Computer Vision*, 2024.

Junjie Wang, Bin Chen, Yulin Li, Bin Kang, Yichi Chen, and Zhuotao Tian. Declip: Decoupled learning for open-vocabulary dense perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

Bin Xie, Jiale Cao, Jin Xie, Fahad Shahbaz Khan, and Yanwei Pang. Sed: A simple encoder-decoder for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022a.

Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023a.

Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, 2022b.

Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b.

Jianjian Yin, Tao Chen, Yi Chen, Gensheng Pei, Xiangbo Shu, Yazhou Yao, and Fumin Shen. PCA-Seg: Revisiting cost aggregation for open-vocabulary semantic and part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Convolutions die hard: Open-vocabulary segmentation with single frozen convolutional clip. In *Advances in Neural Information Processing Systems*, 2023.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

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*, 2023.

Dengke Zhang, Fagui Liu, and Quan Tang. Corrclip: Reconstructing patch correlations in clip for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019. doi: 10.1007/s11263-018-1140-0.

Ziqin Zhou, Yinjie Lei, Bowen Zhang, Lingqiao Liu, and Yifan Liu. Zegclip: Towards adapting clip for zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, Nanyun Peng, Lijuan Wang, Yong Jae Lee, and Jianfeng Gao. Generalized decoding for pixel, image, and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

A Appendix

A.1 Additional Implementation Details

Table 6 complements the settings in the main paper. We set the CSD depth to $l = 4$: the first $l - 1$ layers apply asymmetric token flow, and the last layer produces separate cost and spatial representations. The CLIP and auxiliary-encoder attention parameters use a 0.01 learning-rate multiplier, while projections, aggregation modules, and composition heads use the base rate. Training uses SSD color augmentation and horizontal flipping. Evaluation resizes the short edge to 640 and uses the same 384×384 window size as training.

Cost-aggregation operators. Equation (1) summarizes the CAT-Seg aggregation pipeline (Cho et al., 2024). For an $h \times w$ patch grid and K candidate categories, $C \in \mathbb{R}^{hw \times K}$ stores patch–category cosine similarities. The following describes the modules used in our ViT-L configuration.

$\mathcal{G}$ reshapes visual tokens onto the patch grid and applies a 3×3 convolution followed by ReLU to obtain spatial guidance. CoDR uses separate projections for the CLIP and auxiliary spatial sources, then composes them into $X_s(I)$.

Φ applies a shared 7×7 convolution to each category's single-channel similarity map, producing 128-channel cost features while preserving spatial resolution. The resulting tensor has shape $h \times w \times K \times 128$, with categories kept as a separate axis.

$\mathcal{A}_{\mathrm{spa}}$ contains a window-attention block followed by a shifted-window block, using 12×12 windows and a shift of 6 positions. Spatial guidance is concatenated with normalized cost features to form queries and keys; values are projected from cost features alone. Each block includes a two-layer feed-forward network with GELU, layer normalization, and residual connections.

$\mathcal{A}_{\mathrm{cat}}$ applies linear attention across categories at each spatial position. Text embeddings are projected by a linear layer and ReLU, then concatenated with cost features to guide queries and keys; values again come from cost features. Attention uses the positive feature map $\varphi(x) = \mathrm{ELU}(x) + 1$, followed by a two-layer ReLU feed-forward network with normalization and residual connections. Our 1×1 pooling setting preserves spatial resolution. Two spatial–class aggregation stages are applied in sequence.

f_{dec} uses two shared upsampling stages for each category. Each stage doubles spatial resolution with a 2×2 transposed convolution, concatenates higher-resolution visual guidance, and applies two 3×3 convolutions with GroupNorm and ReLU. A final 3×3 convolution produces one logit map per category, which is bilinearly resized to the output resolution.

Trainable parameters. During representation training, we update the q/v projections in EVA02-CLIP visual attention, the input projections in CLIP text attention, and the attention parameters of DINOv3. Within the copied CLIP spatial branch, its value projection, copied FFN and normalization parameters, and output head are also trainable. All remaining pretrained encoder parameters are frozen, whereas the representation projections, RD weights, cost aggregation modules, and decoder are optimized from their task-specific initialization. During each SARC stage, the complete representation model is frozen and only the corresponding composition head is updated. The pooled head inputs and the endpoint losses used to construct preference targets are detached.

Composition heads and optimization. The cost head receives the globally pooled normalized representations $[\mathrm{GAP}(\hat{X}_c^{\mathrm{cls}}); \mathrm{GAP}(\hat{X}_c^{\mathrm{rel}})]$, yielding a 1,536-dimensional descriptor. It uses LN–$\mathrm{FC}_{1536\to32}$–GELU– $\mathrm{FC}_{32\to1}$. Its output layer is zero-initialized and its bias is set to $\mathrm{logit}(0.3)$, giving an initial relation-centric weight $\alpha_c = 0.3$. The complementary CLS-enhanced group retains the learned within-group weights for (r_1, r_3, r_4), so the initial cost composition reproduces the preceding four-representation mixture. The spatial head receives the pooled $[\mathrm{GAP}(X_s^C); \mathrm{GAP}(X_s^D)]$ descriptor and uses LN–$\mathrm{FC}_{256\to32}$–GELU– $\mathrm{FC}_{32\to1}$. Its output weights are zero-initialized and its bias is set to $\mathrm{logit}(0.25)$, yielding $\alpha_s = 0.25$ at initialization.

We first optimize CSD, RD, the spatial projections, cost aggregation modules, and decoder with $\mathcal{L}_{\mathrm{seg}}$. These modules are then frozen while the spatial composition head is trained, followed by the cost composition head. The corresponding objectives are $\mathcal{L}_{\mathrm{seg}} + \lambda_s \mathcal{L}^s_{\mathrm{comp}}$ and $\mathcal{L}_{\mathrm{seg}} + \lambda_c \mathcal{L}^c_{\mathrm{comp}}$,

Table 6: Training and evaluation hyperparameters for the ViT-B and ViT-L configurations.

Item	Setting
CLIP encoder (ViT-B)	EVA02-CLIP-B/16
CLIP encoder (ViT-L)	EVA02-CLIP-L/14@336
Spatial encoder (ViT-B)	DINOv3-B/16
Spatial encoder (ViT-L)	DINOv3-L/16
CSD depth	$l = 4$
Training crop	384×384
Optimizer	AdamW
New-module LR	2×10^{-4}
Pretrained-attention LR	2×10^{-6}
Weight decay	10^{-4}
Schedule	Cosine, 80k iterations
Global batch size	8 on 4 L40S GPUs
Sliding-window crop	384×384

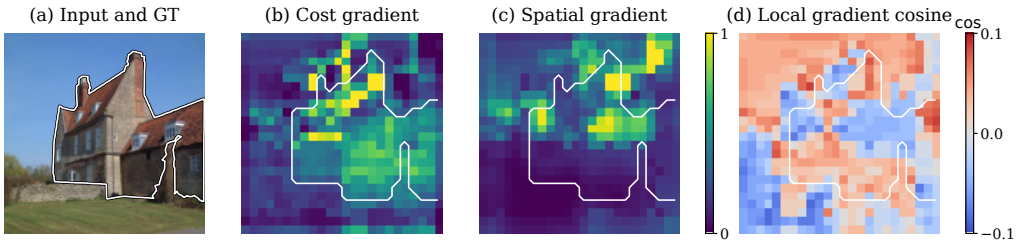


Figure 7: Spatial localization of cost–spatial gradient interaction on the shared r_2 representation. (a) Input with the ADE150 building region outlined. (b,c) Cost and spatial gradient magnitudes, independently normalized to $[0, 1]$ and shown with a shared color scale. (d) Patch-wise gradient cosine, where blue denotes opposing updates; its color scale spans $[-0.1, 0.1]$. White contours mark the ground-truth region.

with $\lambda_s = \lambda_c = 0.01$, $T_s = T_c = 0.1$, and $\epsilon = 10^{-6}$. Per-representation losses are detached when constructing preference targets. They are needed only during composition-head training; inference directly predicts one coefficient pair per evaluated view.

A.2 Gradient Interaction Between Cost and Spatial Objectives

We directly examine spatially localized gradient disagreement at the shared dense representation. We probe the r_1 and r_2 constructions in Eq. (4) before decoupling, using 500 images from ADE150 and 500 images from ADE847. For each image, the two largest ground-truth classes form image–class pairs, giving 1,000 pairs on each benchmark. For each pair, the negative-patch rate computes the fraction of spatial positions whose local gradient cosine is below zero. The target rate restricts this statistic to the ground-truth region, and the cost-gradient share is $\|g_c\|/(\|g_c\| + \|g_s\|)$.

Table 7 shows that 34.7–55.1% of spatial positions and 32.9–57.8% of target-region positions receive opposing local updates, revealing substantial patch-level disagreement. The cost-gradient share ranges from 63.8% to 84.7%, indicating that cost construction supplies the stronger update at this

Table 7: Source-level cost–spatial gradient interaction for two shared dense representations. Values are averaged over image–class pairs.

Metric	r_1		r_2	
	ADE150	ADE847	ADE150	ADE847
Negative patches (%)	51.4	55.1	53.3	34.7
Target negative patches (%)	55.5	57.8	50.5	32.9
Cost-gradient share (%)	84.7	79.0	68.6	63.8

Table 8: Single-representation cost-construction comparison with ViT-L/14. All rows use the same no-DINO scalar aggregation protocol.

Dense cost construction	ADE847	ADE150	PC459	PC59	VOC20
Self-similarity attn. (r_2)	18.1	40.5	25.6	63.8	97.9
Value proj. + FFN	17.9	41.3	**26.8**	63.5	97.9
Value proj. + CLS + FFN	**18.6**	**41.5**	26.6	63.8	97.8
Final-layer patch tokens	18.4	40.6	26.6	62.9	**98.0**
Six-layer shared CLS	17.5	40.6	26.2	63.7	97.7
Pixel-top-k CLS	18.1	41.2	26.4	63.6	97.8

Table 9: Four- and six-construction mean composition under the same no-DINO training protocol.

Cost set	ADE847	ADE150	PC459	PC59	VOC20
r_1–r_4	**19.2**	**41.4**	26.6	63.3	**98.2**
r_1–r_4 + Q–Q/K–K	18.6	41.3	**26.7**	**63.5**	97.9

shared source grid. These results support localized and spatially heterogeneous competition between the two objectives.

Figure 7 localizes the aggregate statistics in the first ADE150 example. Cost construction and spatial guidance emphasize different regions, with an IoU of 0.20 between their top-20% gradient locations. Moreover, 42.7% of all patches and 44.3% of target-region patches have negative local cosine, further showing that the two objectives produce spatially heterogeneous updates.

A.3 Cost Representation Design Space

Before fixing r_1–r_4, we compared several ways to obtain a dense cost representation from the same EVA02-CLIP-L/14 encoder. Table 8 reports independently trained no-DINO models under the same scalar cost-aggregation protocol. Standard final-layer patch tokens are competitive but not uniformly best. Incorporating CLS semantics after the local projection improves ADE847 and ADE150, whereas sharing one CLS across six late layers or retaining only pixel-top-k CLS responses does not provide a consistent gain. These results motivate preserving multiple constructions instead of selecting one globally.

We also tested whether simply adding more relation operators improves RD. Starting from the four constructions underlying r_1–r_4, we add query–query (Q–Q) and key–key (K–K) relation branches and retain mean composition. Q–Q forms its relation matrix from pairwise similarities between query tokens, whereas K–K uses pairwise similarities between key tokens; each matrix is normalized and used to aggregate the value tokens. Table 9 shows that the six-branch model lowers ADE847 by 0.65 mIoU without consistent gains on the remaining benchmarks. The compact four-construction set is therefore not an arbitrary truncation: the two additional relation variants increase redundancy without broad improvements.

A.4 Cost Composition Strategy Comparison

We next keep the four cost constructions and all remaining modules fixed while changing only their composition rule after branch-wise cost embedding. Uniform mean directly averages the four representations. Global softmax maintains one learnable scalar logit per representation and normalizes the four logits with softmax. Patch-conditioned softmax computes the mean, maximum, and root mean square of each representation over the hidden and class dimensions at every spatial position. A 1×1 convolution maps the resulting 12-channel local descriptor to four patch-wise logits. Text-conditioned softmax instead mean-pools and normalizes the prompt embeddings of each class, then applies a linear layer to obtain four class-specific logits. The learnable logits, convolution, and linear layer are zero-initialized, so all three learned variants start from uniform composition.

Table 10: Cost-composition strategies with matched representations, training schedule, and no auxiliary spatial encoder.

Composition rule	ADE847	ADE150	PC459	PC59	VOC20	Avg.
Uniform mean	**19.0**	41.4	26.6	63.3	98.1	49.7
Global softmax	18.9	**41.9**	**27.3**	**63.9**	97.9	**50.0**
Patch-wise softmax	18.7	41.4	26.6	63.7	**98.3**	49.7
Text-cond. softmax	18.6	40.9	26.3	63.4	98.2	49.5

Global softmax gives the highest average among the fixed composition rules, but its ADE847 result remains below SARC under the matched cost-only setting in Table 5. This comparison shows that a fixed global composition remains less effective than scene-adaptive composition.

A.5 Representation Complementarity

The main paper measures representation preference after joint CoDR training. We further examine whether these differences survive the complete prediction pipeline. Using the final ViT-L checkpoint, we bypass the cost α-head and intervene on the same jointly trained model by assigning one-hot weights to r_1–r_4; the spatial representations, aggregation modules, and decoder remain unchanged. The fused prediction uses the learned static four-way composition. Figure 8 shows two representative ADE847 image–category cases for each representation, where its target-class IoU is clearly higher than those of the other three.

The prediction differences remain after the shared cost embedding, aggregation, and mask decoding stages rather than being confined to raw correlation maps. Moreover, the static composition generally retains the reliable regions of the best individual representation. The independently trained constructions in Table 8 provide a complementary dataset-level view: their ordering varies across benchmarks. The category-level analysis in Figure 10 extends this observation across all evaluation datasets, where many classes retain a substantial IoU range among the separately trained constructions. Together, the controlled one-hot interventions and broader category statistics show that the dense cost constructions remain nonredundant at both local and aggregate levels.

A.6 CLS–Patch Structure and Text Alignment

Figure 9 compares the category geometry of r_1–r_4 using the same ADE150 patch samples. Their different class distributions qualitatively indicate that the four constructions form distinct category organizations. To further isolate the effects of explicit CLS injection and FFN refinement, Table 11 reports full-dimensional metrics for r_2–r_4 on the complete ADE150 validation set.

Compared with r_2, explicit CLS injection in r_3 increases the present-versus-absent text margin from 0.16 to 0.17 while maintaining comparable pixel AUROC. Applying the frozen output FFN in r_4 raises within-region cosine to 0.75 while reducing the text margin to 0.15. The FFN therefore changes the balance between class-aligned responses and local coherence rather than uniformly increasing visual–text similarity. Together with the independent t-SNE projections, these measurements support the interpretation that the CLS-enhanced construction produces a more category-discriminative dense distribution. We rely on the full-dimensional metrics for this conclusion rather than cross-panel distances in independently fitted t-SNE projections.

A.7 Alternative Spatial Foundation Model

To test whether Spatial RD depends specifically on DINO, we replace DINOv3-L/16 with SAM2.1-Hiera-L while retaining EVA02-CLIP-L/14, the four cost constructions, independent spatial projections, the channel-wise spatial composition, the 80k schedule, and the evaluation protocol. SAM uses only its image encoder; its prompt encoder and mask decoder are disabled.

Table 12 shows that SAM and DINO obtain nearly identical mean performance while favoring different datasets. DINO is slightly stronger on ADE847, PC459, and PC59, whereas SAM is stronger on ADE150 and VOC20. The result indicates that Spatial RD is not tied to a particular auxiliary en-

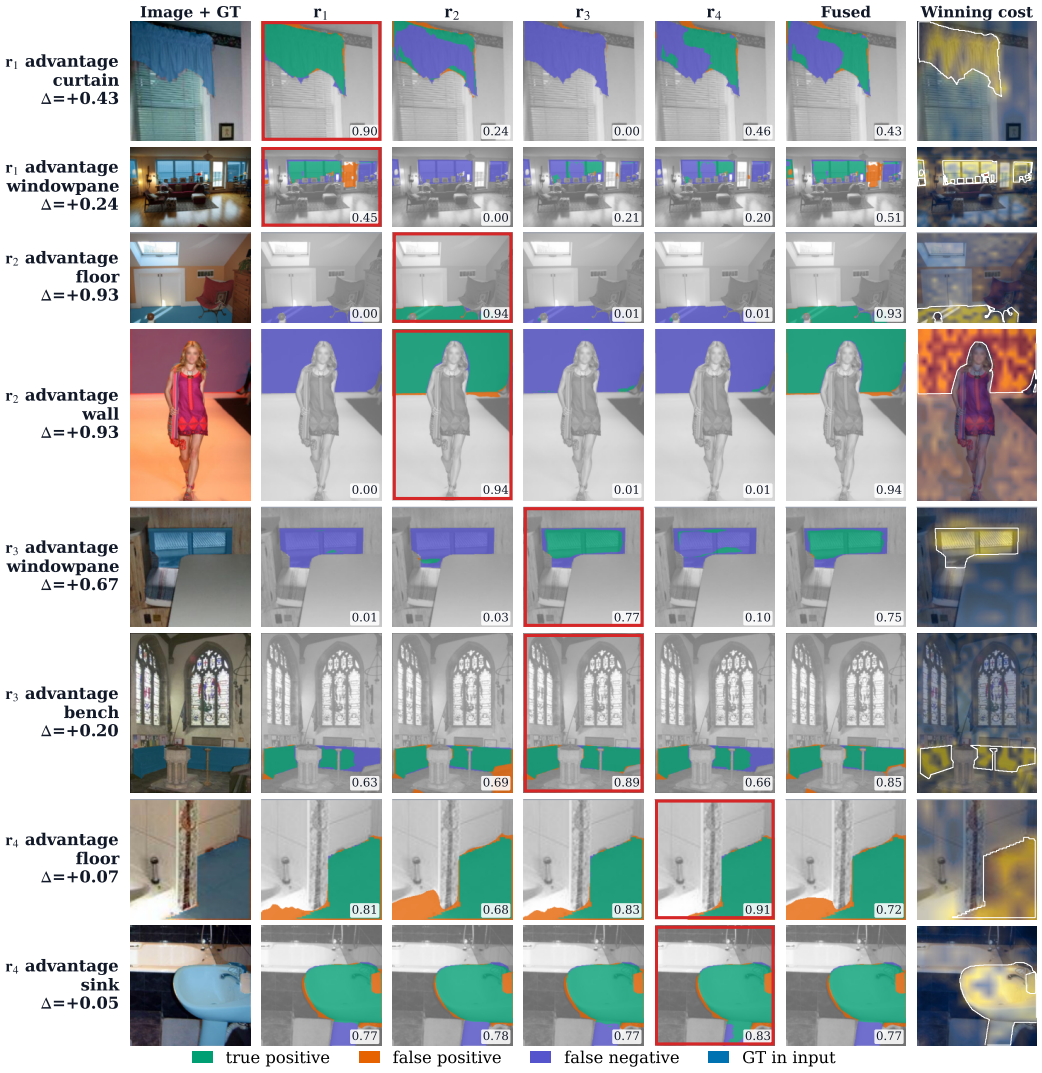


Figure 8: Final-mask complementarity of r_1–r_4 on ADE847. Each two-row group shows cases where the indicated cost representation gives the highest target-class IoU among four one-hot interventions. The r_1–r_4 columns show final decoder masks, while Fused uses the learned static composition. Green, orange, and purple denote true positives, false positives, and false negatives; the last column shows the winning representation's class-cost response. Red boxes mark the winning representation.

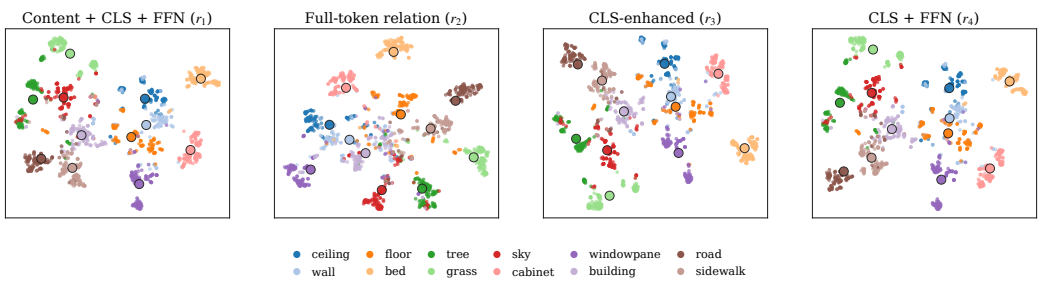


Figure 9: Independent t-SNE projections of matched ADE150 patches for the four cost representations r_1–r_4. Colors denote ground-truth categories.

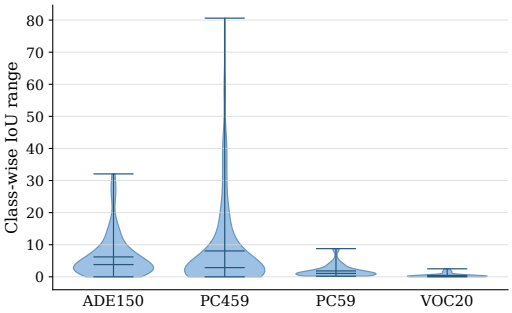


Figure 10: Per-category IoU range among four independently trained cost representations. Larger ranges indicate stronger sensitivity to the representation construction.

Table 11: Dense–text alignment and patch-structure metrics on ADE150. Text margin is the similarity gap between present and absent classes; region cosine measures within-region patch coherence.

Representation	Pixel AUROC	Text margin	Region cosine
r_2	0.92	0.16	0.71
r_3 (+CLS)	0.92	**0.17**	0.74
r_4 (+CLS+FFN)	0.92	0.15	**0.75**

Table 12: Replacing the auxiliary spatial encoder under the same CLIP–auxiliary composition design.

Spatial source	ADE847	ADE150	PC459	PC59	VOC20
DINOv3-L	**19.4**	42.0	**27.2**	**64.3**	98.0
SAM2.1-Hiera-L	19.2	**42.5**	27.1	64.2	**98.1**

coder; it benefits from a complementary spatial source whose information is aligned and composed with the CLIP spatial representation.

A.8 Inference Efficiency

Table 13 compares inference time under identical hardware. CAT-Seg+DINO serves as an intermediate configuration to separate the overhead of the DINO spatial branch from that of RD and SARC. Compared with CAT-Seg, the DINO spatial branch adds 35–42 ms per image. CoDR adds 13–24 ms over CAT-Seg+DINO, corresponding to an increase of 4.7–5.2%. Thus, the additional inference cost introduced by CoDR remains moderate.

A.9 Additional Qualitative Results

Scope of the qualitative evidence. The examples in Figures 11 and 12 are selected to expose differences between methods and are not used to estimate average accuracy. They show that the gains are not restricted to one scene type: CoDR improves small object assignments, visually similar categories, coherent stuff regions, and object boundaries. Remaining errors are concentrated in very small regions, ambiguous annotation boundaries, and categories with nearly synonymous text labels. The quantitative comparison in the main paper remains the basis for the overall performance claim.

Following Figure 5, we select nontrivial cases where CoDR achieves both high absolute accuracy and a clear advantage over the stronger prediction from CAT-Seg and DeCLIP. Each image contains at least five annotated classes, no class occupies more than 75% of valid pixels, and the CoDR margin is at least 10 mIoU points.

Table 13: Inference time (ms/img) measured with batch size 1 on a single NVIDIA RTX 6000 Ada Generation GPU. All methods use identical sliding-window settings.

Method	ADE847	ADE150	PC59
CAT-Seg	476	321	209
CAT-Seg+DINO	511	361	251
CoDR (Ours)	535	379	264

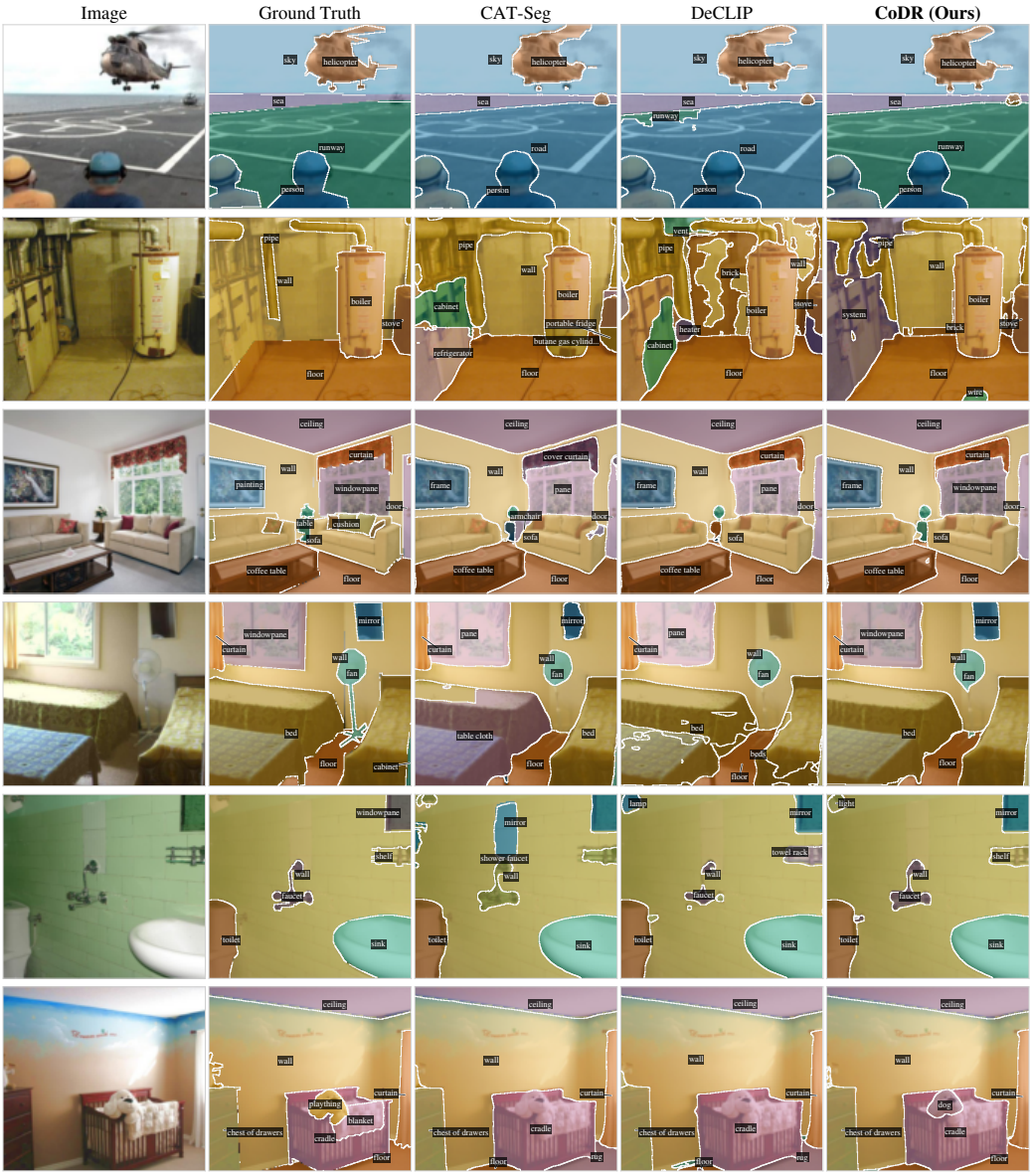


Figure 11: Additional qualitative comparison on ADE847.

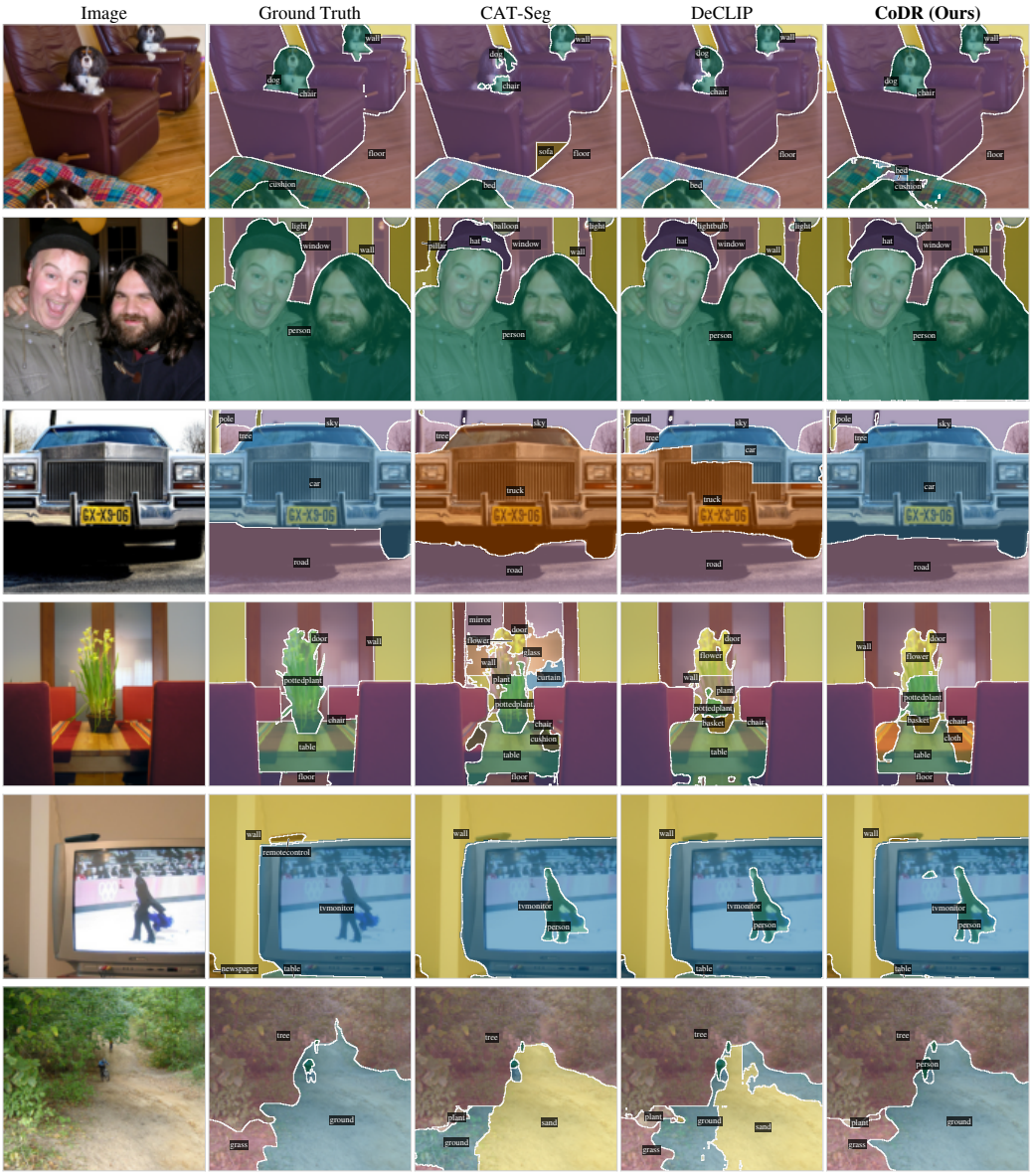


Figure 12: Additional qualitative comparison on PC459.