UniRefiner: Teaching Pre-trained ViTs to Self-Dispose Dross via Contrastive Register

Supplementary Material

This supplementary material is organized as follows:

- In Sec. 1, we compare UniRefiner with denoising-based refinement methods on large ViTs and demonstrate the advantages of our approach.
- In Sec. 2, we present a comprehensive analysis of spurious tokens, including additional visualizations and a discussion that connects our observations to prior work on analyzing artifact tokens.
- In Sec. 3, we report further ablation studies that examine the influence of different UniRefiner components.
- In Sec. 4, we provide implementation details for UniRefiner, as well as for the dense prediction tasks and their evaluation protocols.
- In Sec. 5, we report additional evaluations on ImageNet-1K classification, image-text retrieval, and the Hummingbird benchmark to assess global integrity and concept-level dense capability beyond standard linear probing.
- In Sec. 6, we supply additional qualitative results that illustrate how UniRefiner improves the spatial representations of large pre-trained ViTs.

1. Comparison with Denoising-based Refinement Methods

In the main paper, we motivated UniRefiner by identifying a critical scaling issue: spurious tokens in large-scale foundation models (e.g., EVA-CLIP-8B) are pervasive and therefore require an explicit mechanism for their characterization and mitigation. These tokens are not sparse outliers but can constitute a substantial fraction of the representation—frequently exceeding 40% (see the Introduction). In this section, we provide a detailed comparison and discussion of prior denoising-based refinement methods [3, 16], which typically target a single class of artifacts or do not explicitly model spurious tokens.

Comparison with DVT [16]. Denoising Vision Transformer (DVT) models spurious tokens as global patterns correlated with positional embeddings and attempts to separate them from regular image tokens by employing two independent neural fields together with a content-aware residual network. We apply DVT to two large ViTs—EVA-CLIP-8B and SigLIP2-So400M—using the authors' official codebase. The results are shown in Fig. 1, which visualize DVT-denoised and UniRefiner-refined features for direct comparison. Across both models and diverse spurious patterns, DVT does not reliably disentangle spurious tokens from semantically meaningful image tokens. Instead of isolating these tokens, DVT primarily smooths the features and therefore induces substantial feature collapse. Consequently, PCA projections and clustering outputs remain disordered, with object-related features heavily entangled with spurious signals. These observations indicate that DVT's assumption of predominantly position-correlated artifacts is insufficient, where spurious tokens are more heterogeneous and pervasive.

Table 1. **Training Time Efficiency Comparison**. We compare the training time with single GPU of different refinement methods on DINOv2 ViT-B backbone. Results of DVT and PH-Reg are reported from [3].

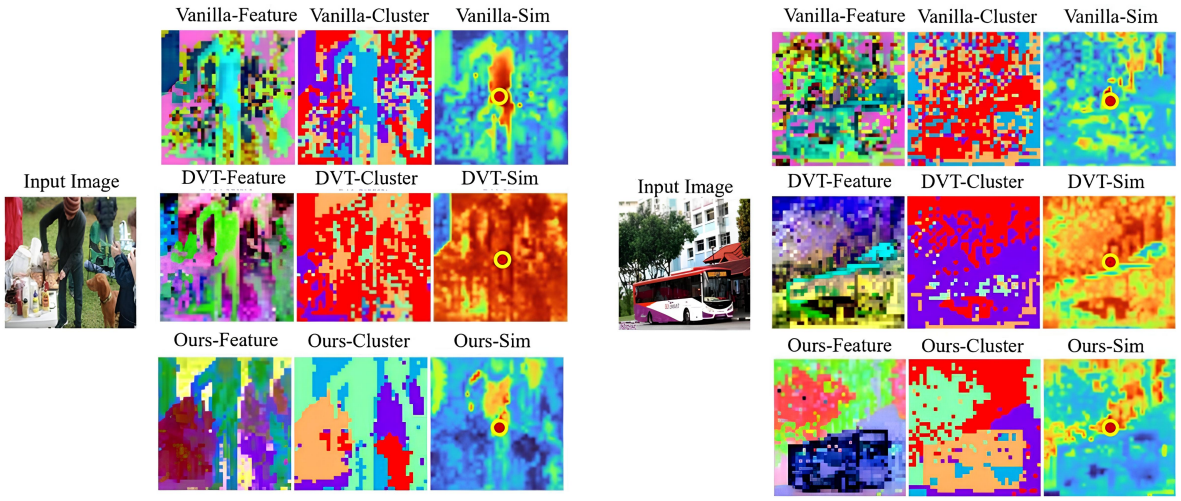
Method	Stage 1 Extraction	Stage 1 Distillation	Stage 2 Training	Total
DVT	2998 min	18340 min	570 min	21908 min (365.1 h)
PH-Reg	-	-	-	9000 min (150 h)
UniRefiner	-	-	-	5 min

PH-Reg [3]. PH-Reg is a direct successor to DVT that omits the neural-field components for computational efficiency and instead obtains denoised features by simply averaging noisy feature maps. As reported in the quantitative comparison in the main paper, PH-Reg likewise fails to improve the quality of dense representations in large ViTs. Crucially, without the explicit contrastive supervision employed by UniRefiner—which actively enforces a separation between regular and spurious tokens—passive registers do not receive sufficient gradient signal to absorb and reroute the large volume of spurious tokens in large-scale ViT models.

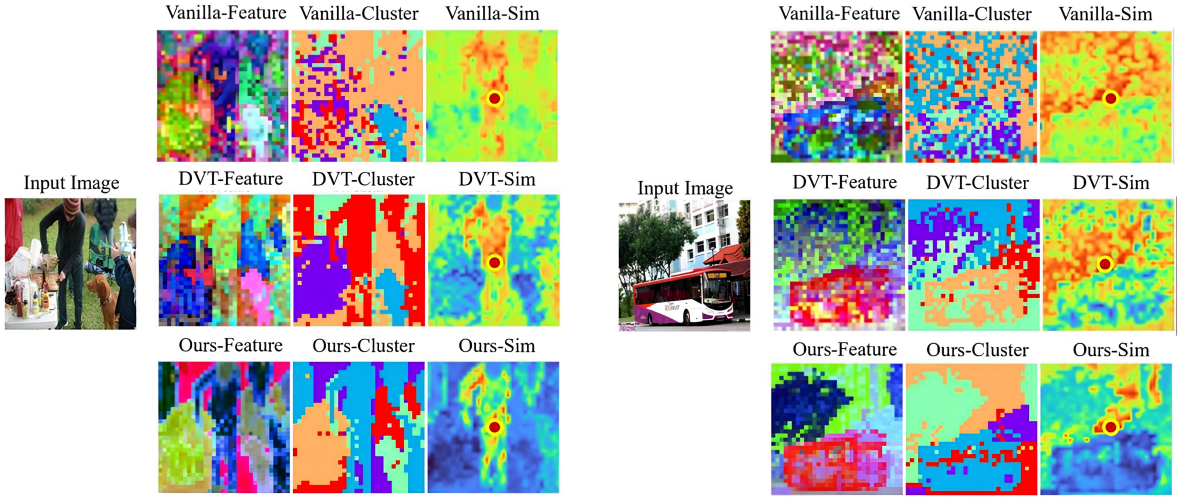
Training Time Efficiency. In Tab. 1 we compare the training-time efficiency of different feature refinement methods. DVT requires multi-stage training and auxiliary neural-field networks, imposing substantial computational overhead. PH-Reg is more parameter-efficient than DVT, but its unconstrained register design leads to slow convergence in practice. By contrast, UniRefiner leverages explicit constraints on the register tokens and a highly compatible adapter design, enabling rapid convergence—five minutes on a single GPU for ViT-base backbones—making it practical for large-scale vision foundation models.

2. Spurious Tokens

This section presents a more comprehensive analysis of spurious tokens. In Sec. 2.1, we provide additional examples across various models, supporting the generality of our findings. In Sec. 2.2, we relate the spurious token categories identified in this work to previous findings, including high-norm tokens [5, 14] and position-correlated artifacts [16].



(a) EVA-CLIP-8B



(b) SigLIP2-So400M

Figure 1. **Qualitative comparison with DVT on EVA-CLIP-8B and SigLIP2-So400M.** We visualize the PCA components of the feature maps (Feature), K-Means clustering results (Cluster), and patch-wise cosine similarity maps (Sim) with respect to the query point marked by a red dot. The **Vanilla** baseline is heavily contaminated by spurious tokens. **DVT** tends to simply smooth the features, producing severe entanglement between regular and spurious tokens, inducing feature collapse and blurring object boundaries. **Ours (UniRefiner)** effectively removes spurious tokens, yielding semantically cleaner features and improved object boundaries.

2.1. Spurious Token in Different Models

We present additional visualizations of the spurious tokens identified in several pre-trained ViTs, including CLIP-Giant, DINOv2-Giant, EVA-CLIP-8B, SigLIP2-So400M, and InternViT-6B. Examples are shown in Fig. 2, Fig. 3, and Fig. 4, which respectively illustrate Fixed Pattern (FP) tokens, Global Proxy (GP) tokens, and Attention Hijackee (AH) tokens.

2.2. Connection with Prior Works

2.2.1. High Norm Tokens

In large-scale pre-trained ViTs, prior work has reported the presence of abnormal tokens exhibiting extremely high channel activations [5, 14]. These high-norm tokens frequently dominate attention maps in the final transformer layers. Darcet et al. [5] characterize them as a specialized group of image tokens that encode global information, whereas Sun et al. [14] interpret them as functional tokens that balance attention computation and do not necessarily correspond to meaningful visual content.

In this work, we observe a related phenomenon: a subset

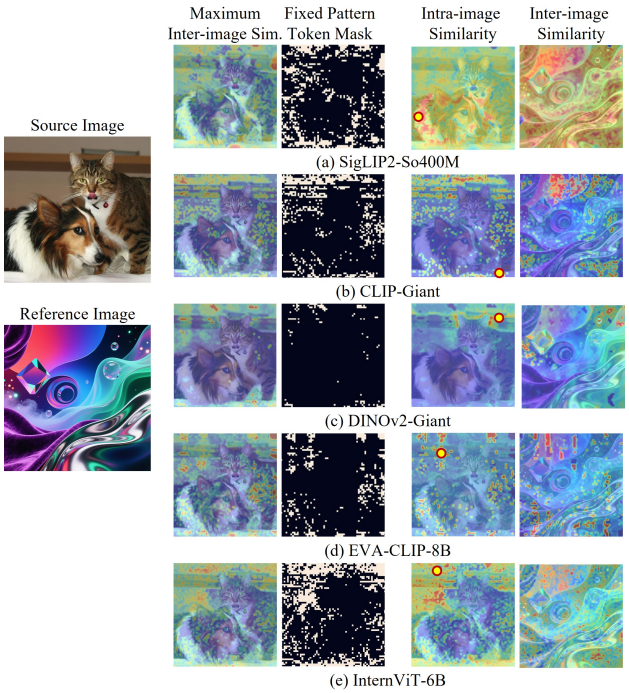


Figure 2. **Visualization of Fixed Pattern (FP) Tokens in Different Pre-trained ViTs.** From left to right: **Maximum Inter-image Similarity** — for each token in the source image, we compute its maximum cosine similarity with all tokens from a reference image that characterizes FP tokens; **FP Token Mask** — binary mask of FP tokens obtained by thresholding the similarity map; **Intra-image Similarity** — cosine similarity between a randomly sampled FP token and all tokens within the source image; **Inter-image Similarity** — cosine similarity between the sampled FP token and all tokens in the reference image. The sampled FP token is highlighted with a yellow dot.

of the Fixed Pattern (FP) tokens coincides with high-norm tokens. To quantify this overlap, for each token i at layer l we compute its maximum channel activation

$$v_i^{(l)} = \max_c \boldsymbol{Z}^{(l)}[i, c], \quad c \in \{1, \ldots, C\}, \tag{1}$$

where c indexes the feature channels. We then compare the distribution of $v_i^{(l)}$ against the FP detections to assess the degree of correspondence between high-norm and Fixed Pattern tokens.

Observation. As shown in Fig. 5, some Fixed Pattern (FP) tokens exhibit abnormally high channel activations relative to other tokens, indicating a partial overlap between FP tokens and high-norm tokens. However, many FP tokens do not present elevated channel activations; this suggests that channel norm alone is an insufficient criterion for detecting FP tokens, and that a more general numerical characterization is required to identify these spurious tokens reliably.

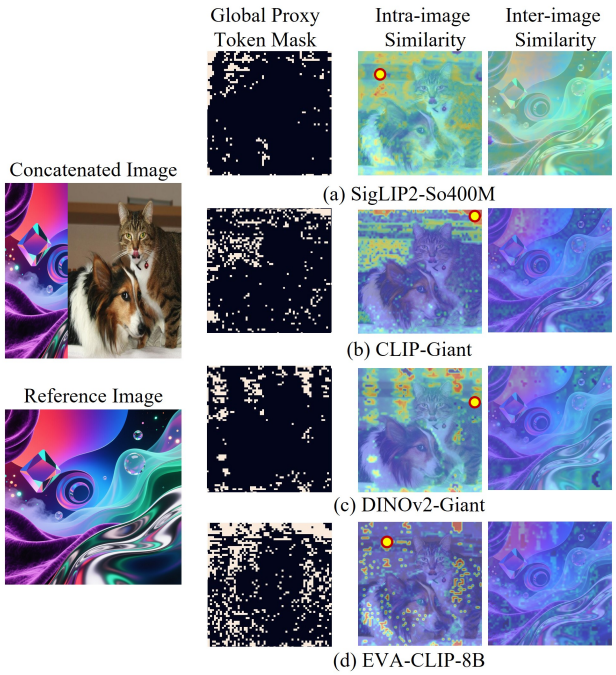


Figure 3. **Visualization of Global Proxy (GP) Tokens in Different Pre-trained ViTs.** From left to right: **GP Token Mask** — a binary mask indicating detected GP tokens obtained by thresholding the similarity map; **Intra-image Similarity** — the cosine similarity between a randomly sampled GP token and all tokens within the same source image; **Inter-image Similarity** — the cosine similarity between the sampled GP token and all tokens from the concatenated irrelevant reference images. For the similarity visualizations, we sample one GP token per image, marked with a yellow dot.

2.2.2. Position-correlated Tokens

Previous work [16] has shown that these artifacts correlate strongly with positional embeddings, suggesting they originate from fixed spatial cues. To investigate the relationship between such position-related artifacts and our detected tokens, we analyze the spatial distribution of the union of Fixed Pattern (FP) and Global Proxy (GP) tokens. Both FP and GP tokens are identified using cosine-similarity criteria (see main paper); hence, joint FP–GP occurrences are defined as locations exhibiting high similarity to either a reference image or to concatenated irrelevant images. Figure 6 presents occurrence heatmaps of the joint FP–GP tokens under two input conditions — a zeroed input and sampled natural images — following the evaluation protocol of [16]. We forward both inputs through the ViTs to compare the resulting spatial patterns.

Observation. As shown in the qualitative results, the occurrences of FP and GP tokens are strongly correlated with positional embeddings in DINOv2-Giant and EVA-CLIP-8B, whereas in CLIP-Giant and SigLIP2-So400M their distribution is primarily content-dependent. Moreover, since

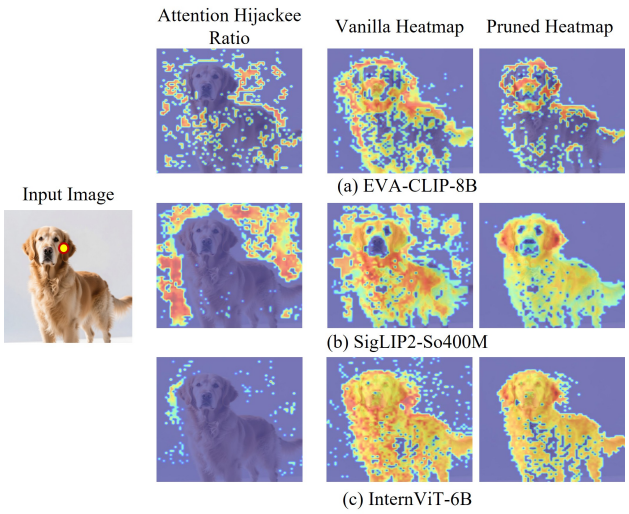


Figure 4. **Visualization of Attention Hijackee Tokens in Different Pre-trained ViTs.** From left to right: **Attention Hijackee Score** — the computed Attention Hijack score that tracks intermediate attention flow; **Vanilla similarity heatmap** — the cosine similarity between a sampled Attention Hijacker token and all image tokens, illustrating how its semantics propagate to Attention Hijackee tokens; **Pruned similarity heatmap** — the cosine similarity map after removing the detected Attention Hijackee tokens, removing these tokens yields substantially cleaner salient regions and object boundaries. For the similarity visualizations, we sample one Attention Hijacker token per image, marked with a yellow dot.

Yang et al. [16] provide mainly qualitative analyses of these artifacts, we introduce a quantitative measurement method that explicitly localizes them within ViTs.

3. Ablation Studies

We present additional ablation studies to analyze the influence of different UniRefiner components.

3.1. Ablation on Spurious Token Filter

Three thresholds control the Spurious Token Filter: $\tau_{fp\text{-}gp}$, which thresholds cosine similarity to jointly detect FP and GP tokens; τ_{reg}, which thresholds cosine similarity using the updated register tokens that act as adaptive detectors; and τ_{ah}, which thresholds the Attention Hijackee score interval to identify AH tokens. Lower values of $\tau_{fp\text{-}gp}$ or τ_{reg} make the criterion stricter (identifying more spurious tokens), whereas a higher τ_{ah} yields a looser criterion for AH detection.

We ablate these thresholds on the SigLIP2-So400M backbone; results are reported in Tab. 2. Because the updated register tokens are adaptive detectors, the filtering pipeline is robust to different choices of $\tau_{fp\text{-}gp}$ when τ_{reg} is set to a stricter value. τ_{ah} also influences performance, with a moderate setting of -0.5 producing the best results in our experiments. Identifying AH tokens requires tracking intermediate attention weights rather than relying solely on cosine similarity, so developing an adaptive thresholding strategy for τ_{ah} is non-trivial and left to future work.

Table 2. **Ablation on Spurious Token Filter thresholds.** We evaluate the three thresholds of the Spurious Token Filter to study their effect on UniRefiner's learning dynamics. Experiments use the SigLIP2-So400M backbone and report results on the ADE20K semantic segmentation task. Symbols indicate direction: ↓ (lower = stricter) and ↑ (higher = stricter).

τ_{reg} ↓	τ_{fp-gp} ↓	τ_{ah} ↑	ADE20K mIoU
0.55	0.7	-0.5	48.8
0.55	0.6	-0.5	49.2
0.55	0.55	-0.5	49.8
0.55	0.5	-0.5	48.2
0.55	0.4	-0.5	47.5
0.7	0.7	-0.5	46.9
0.7	0.55	-0.5	47.5
0.55	0.55	-1.0	48.0
0.55	0.55	0.0	49.3

Table 3. **Ablation on Alignment Objective.** We evaluate different alignment objectives for UniRefiner training. Experiments use the SigLIP2-So400M backbone and report results on the ADE20K semantic segmentation task.

Alignment Objective	ADE20K mIoU
InfoNCE	49.8
MSE	42.1
Cosine Similarity	43.0

3.2. Alignment Objective

We adopt the InfoNCE loss [10] as our default alignment objective. This contrastive loss aligns image tokens and register tokens with their corresponding regular and spurious tokens by treating matched pairs as positives and all other tokens as negatives. For comparison, we also evaluate two non-contrastive alternatives: mean squared error (MSE) and cosine similarity losses; results are reported in Tab. 3. We find that objectives that do not explicitly regularize negative samples suffer from severe feature collapse and yield substantially worse performance. We hypothesize that the refinement process is intrinsically noisy, and that contrastive supervision provides a stabilizing inductive bias that prevents collapse.

4. Implementation Details

4.1. Training Details

To train UniRefiner, we employ the AdamW optimizer [8] with a base learning rate of 1×10^{-4}, a batch size of 16, and a weight decay of 0.1. We do not apply learning-rate warm-up or a cosine-decay schedule. Input resolutions are 448×448 for patch size 14 and 512×512 for patch size 16, both

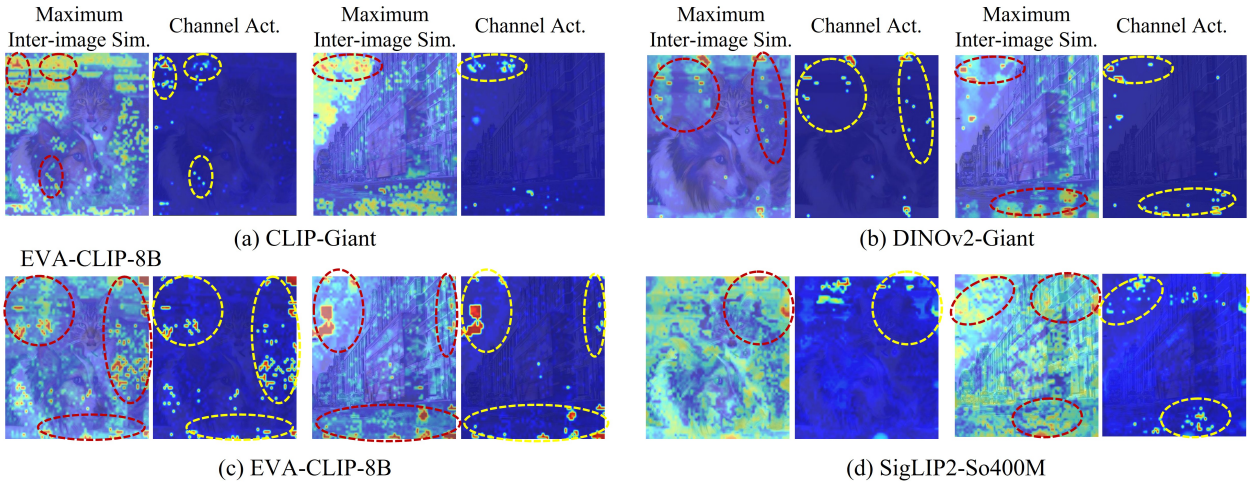


Figure 5. **Comparison between Fixed-Pattern and High-Norm Tokens.** We visualize (**left**) the maximum cosine similarity to a reference image that identifies Fixed-Pattern (FP) tokens, and (**right**) the maximum channel activation per token for image tokens extracted from CLIP-Giant, DINOv2-Giant, EVA-CLIP-8B, and SigLIP2-So400M. The red and yellow dashed lines denote tokens that simultaneously exhibit high similarity to the reference image and high channel activation; thus, high-norm artifacts typically form a subset of FP tokens.

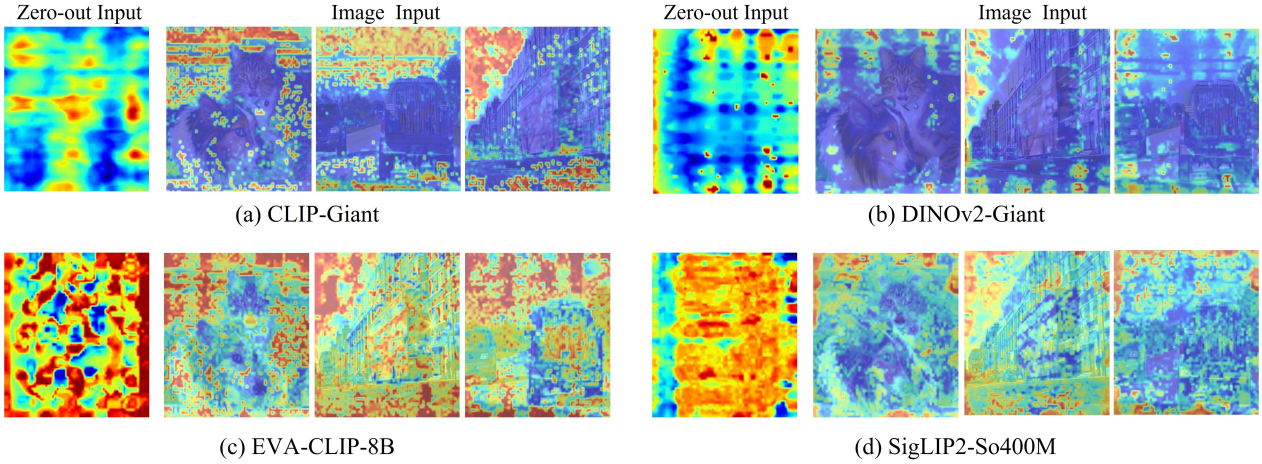


Figure 6. **Comparison between FP–GP tokens and position-embedding-related artifacts.** We visualize occurrence heatmaps of joint FP–GP tokens under two input conditions: a zeroed input (left) and a natural image input (right). In DINOv2-Giant and EVA-CLIP-8B, FP and GP occurrences align strongly with positional embeddings, whereas in CLIP-Giant and SigLIP2-So400M their distribution is primarily content-dependent.

yielding 32×32 feature maps. Training is conducted for two epochs on a 5k-image subset of Conceptual Captions (CC3M) [12]. The exact training split will be released.

UniRefiner-specific optimization. Unless otherwise specified, we use a fixed set of loss weights for all backbones, with λ_{spu}=0.1 and λ_{uni}=0.5, without model-specific tuning. We adopt an asymmetric InfoNCE objective because the teacher branch is frozen. Concretely, for $\mathcal{L}_{\text{regu}}$, all tokens within the same image serve as negatives, while for $\mathcal{L}_{\text{spu}}$, all registers within the same image act as negatives. We set the temperature to τ=0.2 for all token-level contrastive terms.

Register injection. Our default setting injects registers along the left, right, upper, and lower boundaries of the input image, which expands the resulting 32×32 feature map to 34×34. This default "LRUD" configuration corresponds to N_{reg}=132 register tokens. We implement these registers as Gaussian-noise patches injected before patch embedding. This design naturally scales the number of registers with the input resolution, and the injected registers receive the same positional encoding as standard image patches. We also evaluated zero padding and a shared learnable bias as alternative register variants; while all are effective, Gaussian-noise registers achieve the best performance in our experiments.

4.2. Dense Prediction

We evaluate the quality of spatial representations produced by UniRefiner-adapted backbones across two distinct paradigms: vision-centric linear probing (semantic segmentation and depth estimation) and vision-language zero-shot inference.

4.2.1. Vision-centric Tasks: Linear Probing

For vision-centric evaluations, we strictly follow the linear probing protocol defined in DINOv2 [11]. In these experiments, the backbone (with integrated UniRefiner adapters) remains frozen.

Semantic Segmentation.

- **Model:** We train a simple linear classifier (a 1×1 convolutional layer) on top of the frozen patch tokens from the last ViT layer to predict pixel-wise class logits. The class token ([CLS]) is discarded.
- **Datasets:** We evaluate on ADE20K [17] (150 classes), Cityscapes [4] (19 classes), and PASCAL VOC 2012 [6] (20 foreground classes + background).
- **Training & Evaluation:** Following DINOv2, we use the SGD optimizer (momentum 0.9, learning rate $1e^{-3}$) and a "poly" learning rate schedule (power 0.9). We do not apply weight decay. Consistent across all datasets, we train for 40k iterations with a batch size of 16. Data augmentation includes random resizing (0.5-2.0), cropping (512×512), and horizontal flipping. We report mIoU and mAcc.

Monocular Depth Estimation.

- **Model:** We adopt the **"lin. 1"** configuration from DINOv2 [11]. A simple linear layer regresses the depth map directly from the patch tokens of the final transformer block. Predictions are upsampled by a factor of 4 to match the input resolution.
- **Datasets:** We train and evaluate on the NYUv2-Depth dataset [13] using the official splits.
- **Training & Evaluation:** Optimization is performed using AdamW (learning rate $3e^{-4}$, weight decay $1e^{-4}$). We train for 38,400 iterations with a batch size of 16. We report the Root Mean Squared Error (RMSE), Absolute Relative error (Abs Rel), and the threshold accuracy (δ_1).

4.2.2. Vision-language Tasks: Zero-shot Segmentation

To evaluate open-vocabulary capabilities, we employ a training-free approach based on the DeCLIP [15] codebase.

- **Model:** We follow MaskCLIP [18] for training-free modification, which modifies the attention mechanism of the CLIP image encoder's final block to extract dense feature maps directly from the value projection. These maps are matched with text embeddings with an ensemble of 85 prompt templates (e.g., "a photo of a {class}").
- **Datasets:** We evaluate different models on 8 benchmarks. *With Background:* PASCAL VOC 2012 (21 classes, denoted as VOC21) [6], PASCAL Context (60 classes, Context60) [9], and COCO-Object (81 classes, COCO-Obj) [7]. *Without Background:* PASCAL VOC 2012 (20 classes, VOC20) [6], Cityscapes (19 classes) [4], PASCAL Context (59 classes, Context59) [9], ADE20K (150 classes) [17], and COCO-Stuff164k (171 classes) [2].
- **Inference & Evaluation:** Since this is a zero-shot setting, no training is involved. Input images are resized such that the shorter side is 448 pixels. We perform pixel-wise classification by computing the cosine similarity between dense image features and text embeddings. The predictions are bilinearly upsampled to the original image resolution for evaluation. No test-time augmentation is applied. We report mIoU.

Table 4. **Additional evaluations on global integrity.** We report ImageNet-1K classification Top-1 accuracy and Flickr30K image-text retrieval Recall@1. UniRefiner preserves image-level semantics and cross-modal alignment.

Task	Model	Vanilla	+ UniRefiner
IN-1K Top-1	DINOv2-G	86.5	86.4
IN-1K Top-1	EVA-CLIP-8B	87.1	87.1
Flickr30K I2T R@1	EVA-CLIP-8B	95.6	95.4
Flickr30K T2I R@1	EVA-CLIP-8B	80.8	81.2

5. Additional Evaluations

We additionally report results on ImageNet-1K classification, image-text retrieval, and the Hummingbird benchmark [1]. The first two tasks test whether UniRefiner preserves global image semantics and cross-modal alignment, while Hummingbird probes concept-level dense capability.

5.1. Global Integrity: IN-1K and Image-Text Retrieval

ImageNet-1K classification. We additionally evaluate image-level classification on ImageNet-1K. As shown in Tab. 4, UniRefiner preserves the original models' global recognition ability: DINOv2-G changes only marginally from 86.5 to 86.4 Top-1 accuracy, and EVA-CLIP-8B remains unchanged at 87.1.

Image-text retrieval. We further evaluate the EVA-CLIP-8B backbone on image-to-text and text-to-image retrieval. Results in Tab. 4 show that UniRefiner largely preserves cross-modal alignment, with only a marginal change on image-to-text retrieval and a slight improvement on text-to-image retrieval.

5.2. Concept-level Dense Capability: Hummingbird

We additionally evaluate refined backbones on the Hummingbird benchmark [1], which assesses concept-level dense capability beyond the linear probing tasks emphasized in the main paper. As reported in Tab. 5, UniRe-

Table 5. **Evaluation on Hummingbird.** UniRefiner consistently improves concept-level dense capability on both vision-centric and vision-language backbones.

Model	Vanilla	+ UniRefiner
DINOv2-G	73.3	74.5
SigLIP2-So400M	64.6	72.6

finer consistently improves the benchmark score on both DINOv2-G and SigLIP2-So400M, with especially large gains on the vision-language backbone SigLIP2-So400M.

6. Visualization Results

We provide additional qualitative results that demonstrate UniRefiner's effectiveness in improving the spatial representations of large pre-trained ViTs. The vision-centric visualizations are presented in Fig. 7; these compare cosine-similarity heatmaps between a selected query token and all image tokens before and after integrating UniRefiner. The vision–language visualizations are shown in Fig. 8; these depict similarity maps between language prompts and visual tokens. Across models and tasks, UniRefiner consistently yields cleaner and more precise similarity maps while preserving the models' vision–language alignment.

References

[1] Ivana Balažević, David Steiner, Nikhil Parthasarathy, Relja Arandjelović, and Olivier J. Hénaff. Towards in-context scene understanding. In *Advances in Neural Information Processing Systems*, 2023. 6

[2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 6

[3] Yinjie Chen, Zipeng Yan, Chong Zhou, Bo Dai, and Andrew F Luo. Vision transformers with self-distilled registers. *arXiv preprint arXiv:2505.21501*, 2025. 1

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6

[5] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*, 2023. 1, 2

[6] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 6

[7] 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. 6

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 4

[9] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014. 6

[10] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4

[11] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 6

[12] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 5

[13] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 6

[14] Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*, 2024. 1, 2

[15] 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 Computer Vision and Pattern Recognition Conference*, pages 14824–14834, 2025. 6

[16] Jiawei Yang, Katie Z Luo, Jiefeng Li, Congyue Deng, Leonidas Guibas, Dilip Krishnan, Kilian Q Weinberger, Yonglong Tian, and Yue Wang. Denoising vision transformers. In *European Conference on Computer Vision*, pages 453–469. Springer, 2024. 1, 3, 4

[17] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 6

[18] 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. 6

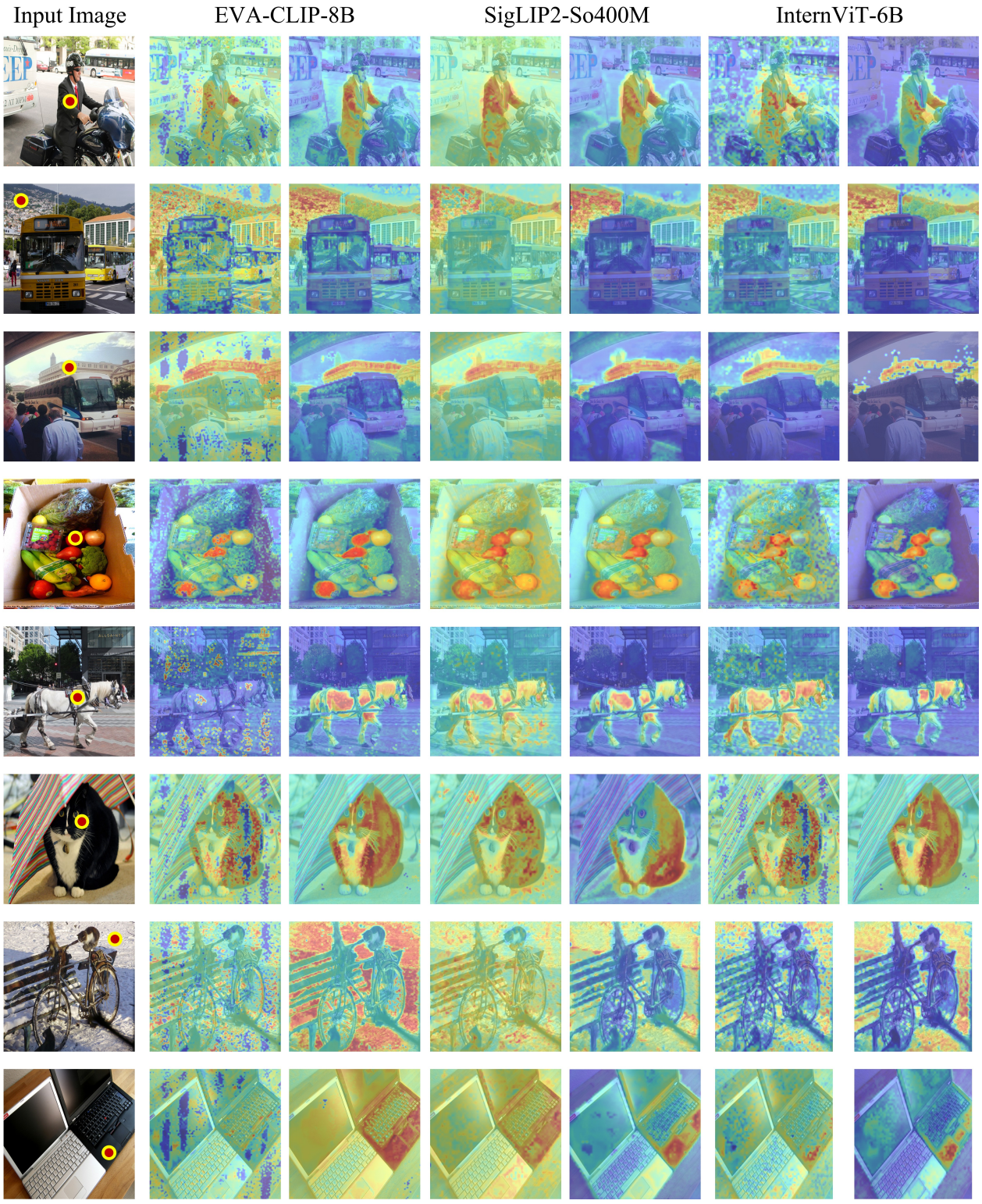


Figure 7. **Vision-centric visualization results.** We present additional qualitative comparisons of cosine-similarity heatmaps between a selected query token (indicated by a yellow dot) and all image tokens. For each row and each model, the left column shows the output of the original ViT backbone, while the right column shows the result after integrating UniRefiner. UniRefiner consistently improves the spatial quality of the representations, yielding cleaner and more precise similarity maps.

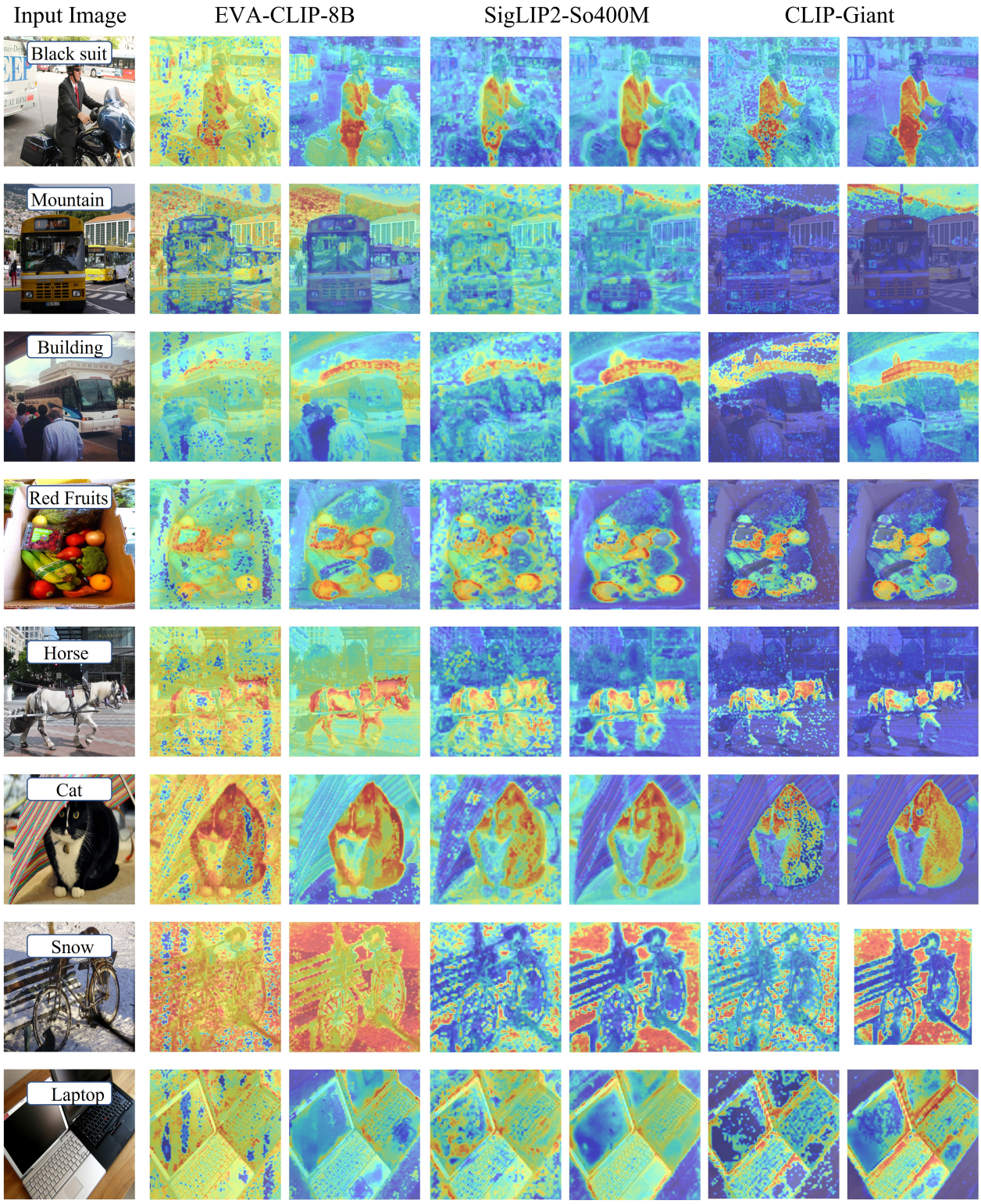


Figure 8. **Vision–language visualization results.** Cosine-similarity heatmaps between language prompts and visual tokens. For each row, outputs from the original ViT backbone (left) are compared with results after integrating UniRefiner (right). UniRefiner improves spatial integrity while preserving the model's vision–language alignment.