INSID3: Training-Free In-Context Segmentation with DINOv3

Supplementary Material

Claudia Cuttano [*1,2] Gabriele Trivigno [*1] Christoph Reich [2,3,5,6]
Daniel Cremers [3,5,6] Carlo Masone [1] Stefan Roth [2,4,5]

[1]Politecnico di Torino [2]TU Darmstadt [3]TU Munich [4]hessian.AI [5]ELIZA [6]MCML [*]equal contribution

https://visinf.github.io/INSID3

In this appendix, we provide additional analyses, implementation details, and experiments for INSID3.

Specifically:

- **Positional bias**. In Sec. A, we further analyze the issue of positional bias in DINOv3, including comparisons with DINOv2 and additional debiasing studies.
- **Implementation details**. In Sec. B, we report the hyperparameters used throughout the paper. The same set of hyperparameters is fixed across all datasets and tasks.
- **Additional experiments**. In Sec. C, we present further experiments, including the 5-shot setting, backbone comparisons, evaluation against SAM 3, and the empty-mask corner case.
- **Computational cost**. In Sec. D, we analyze the computational cost of our method and compare it against training-free and fine-tuned baselines.
- **Qualitative examples**. In Sec. E, we report additional qualitative comparisons and examples of our method.
- **Limitations and future directions**. In Sec. F, we discuss current limitations of INSID3 and outline possible extensions.

A. On the Positional Bias

A.1. DINOv2 vs. DINOv3: Positional bias

To assess how strongly positional bias is encoded in DINOv3 [56] versus DINOv2 [47], we analyze the similarity maps from both encoders, computed as in Sec. 3.1 of the main paper. Given a reference image with one annotated keypoint and a target image, we extract dense patch embeddings $\mathbf{F}^r$ and $\mathbf{F}^t$ from the respective backbone, let j denote the patch index of the keypoint, and form the reference prototype

$$\mathbf{p}^r = \mathbf{F}^r_j. \tag{15}$$

We then compute the target similarity map as

$$\mathrm{sim}(i) = \langle \mathbf{F}^t_i, \mathbf{p}^r \rangle, \qquad i \in \{1, 2, \ldots, P\}. \tag{16}$$

For each backbone, we visualize *(i)* the similarity map obtained from the original features, and *(ii)* the similarity map obtained after applying our debiasing projection from Eqs. (3) and (4) from the main paper. These similarities are visualized in Fig. 8. The maps show a notable difference between the two models. With the original *DINOv3* features, we observe pronounced coordinate-

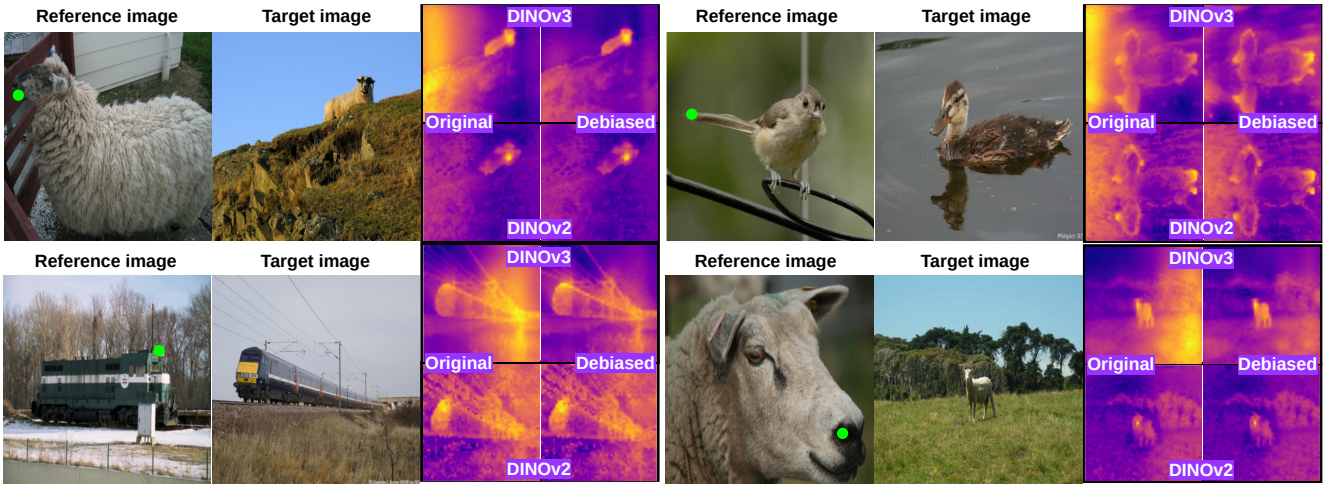

Figure 8. **Positional bias in DINOv2 *vs*. DINOv3.** We visualize the similarity map between a reference keypoint and all target patches using Eq. (16). Darker purple regions indicate low similarity, while brighter yellow regions indicate high similarity. DINOv3 exhibits strong coordinate-aligned artifacts, with bright responses appearing at the same absolute spatial location as the reference keypoint. Our debiasing projection suppresses these spurious activations. We observe that DINOv2 shows much weaker positional bias.

Table 4. **Semantic correspondence on SPair-71k.** PCK@T (%, ↑) for DINOv2 and DINOv3 using original and debiased features. Debiasing produces marginal improvements for DINOv2, but yields substantial gains for DINOv3.

Backbone	Features	PCK@0.05	PCK@0.10	PCK@0.20
DINOv2	original	34.4	50.5	66.8
	debias	**34.7** (+0.3)	**50.8** (+0.3)	**67.1** (+0.3)
DINOv3	original	29.2	45.0	59.8
	debias	**32.3** (+3.1)	**50.0** (+5.0)	**66.4** (+6.6)

aligned artifacts: patches located at the same absolute position as the reference keypoint consistently exhibit strong responses. This phenomenon is more prominent in background regions, where semantic content is scarce or absent, exposing the positional component, which then emerges as spurious activations. After applying the debiasing projection, these position-driven responses are suppressed while the similarity patterns on the object remain stable. This shows that our debiasing suppresses positional bias without degrading the semantic structure encoded in the features. **DINOv2**, in comparison, exhibits weaker positional structure even without any debiasing. The coordinate-aligned patterns characteristic of DINOv3 are mostly absent, and debiasing leads to minor modifications. This contrast suggests that DINOv3 encodes more positional information. We suspect that this could arise as a by-product of its training objective, where the local-consistency constraints and Gram-anchoring may inadvertently amplify absolute spatial correlations in regions with weak semantic cues. The different positional encoding strategy used in DINOv3 compared to DINOv2 may further contribute to this effect.

To quantitatively analyze this behavior, we evaluate semantic correspondence on SPair-71k [42], following the protocol described in Sec. 4.2 of the main paper. This benchmark isolates the *correspondence* component of the representation by measuring cross-image alignment directly on the backbone features, without the overhead of segmentation. As shown in Tab. 4, debiasing yields only small gains for **DINOv2** (all improvements within +0.3 PCK), consistent with its limited positional coupling. In sharp contrast, **DINOv3** benefits substantially, with improvements of +3.1 PCK at $T = 0.05$, +5.0 PCK at $T = 0.10$, and +6.6 PCK at $T = 0.20$. These gains are significant at all thresholds and reflect the removal of positional components that are detrimental for tasks involving feature matching across images.

A.2. Alternative debiasing strategies

We compare our debiasing scheme against training-free alternatives on COCO-20^i, reported in Tab. 5. As a baseline, we propose to reduce positional sensitivity by averaging features over multiple augmented views of each image.

Table 5. **Different debiasing strategies.** Comparison of training-free strategies for feature debiasing on COCO-20^i using mIoU (in %, ↑). Augmentations are a viable alternative but require *multiple forward passes*. In contrast our method adds no overhead, except for a single matrix multiplication, since the debiasing projection is pre-computed offline and stored.

Method	mIoU
No debiasing	54.5
Augmentations, 4 views	55.7
Augmentations, 12 views	56.5
Ours (SVD from 1 noise image)	57.6
SVD from 5 noise images	57.7
SVD from 10 noise images	57.7

Concretely, for each image, we sample $n \in \{4, 12\}$ views using random horizontal and vertical flips as well as homographies. We feed each view independently to DINOv3 and average the corresponding patch features before performing matching and segmentation. This simple strategy already improves over the non-debiased baseline (from 54.5 % to 55.7 % and 56.5 % mIoU for $n = 4$ and $n = 12$, respectively), but *requires n forward passes per each image* at inference time. In contrast, our projection-based debiasing (SVD from a single noise image) achieves a larger gain (57.6 % mIoU) while adding virtually no overhead beyond a single matrix multiplication at inference time. Importantly, our debiasing projection is computed once and stored offline. To test the stability of the estimated positional subspace, we alternatively perform SVD on the features from a small pool of images with *low semantic content*: black, white, horizontal gradient, vertical gradient, and Gaussian noise patterns. Using 5 or 10 such images yields virtually identical results (57.7 % mIoU), confirming that the positional subspace is stable and can be captured reliably with a single noise realization.

A.3. Semantics in debiased feature space

To further examine whether our debiasing projection removes primarily positional variance while preserving the semantic information of the representation, we analyze how semantic information is distributed across principal components in the original and debiased feature spaces. The key idea is simple: in the original DINOv3 representation, positional components account for a substantial amount of variance, causing PCA to allocate some of the leading directions to positional rather than semantic structure. After applying the projection (*cf*. Eq. (4) of the main paper), this positional subspace is removed, reducing the effective rank of the feature representation: although the channel dimensionality remains unchanged, the directions associated with positional variation are collapsed. Therefore, if debiasing removes positional directions, the remaining variance no

longer reflects positional structure, and the resulting principal components predominantly encode semantic variability.

We test this hypothesis by measuring cross-image semantic correspondence accuracy as a function of the retained dimensionality d. The PCA of original features should degrade earlier when reducing d, since its components contain a mixture of semantic and positional signals. This experiment complements the analysis in Fig. 7 of the main paper, which focuses on in-context segmentation, by directly probing the effect of debiasing on cross-image feature matching.

The experimental protocol is as follows. For both the *original* and the *debiased* feature spaces, we compute PCA using a separate set of training images, obtaining two orthonormal bases $\mathbf{U}_{\text{orig}}$ and $\mathbf{U}_{\text{debias}}$. This allows us to study how semantic information is distributed when the feature representation is constrained to d dimensions. For each d we evaluate:

- **PCA-original.** We compute the PCA basis $\mathbf{U}_{\text{orig}}$ on original DINOv3 features extracted from COCO-20^i training images. At test time, target features $\mathbf{F}$ are projected onto the top-d components:

$$\mathbf{F}^{(d)}_{\text{orig}} = \mathbf{F}\,\mathbf{U}_{\text{orig}}[:, 1{:}d]. \tag{17}$$

- **Debias+PCA.** We first apply our positional debiasing projection to the same COCO-20^i training features, then compute a PCA basis $\mathbf{U}_{\text{debias}}$ in this debiased space. At test time, features are first debiased, then projected onto the top-d PCA components:

$$\tilde{\mathbf{F}}^{(d)}_{\text{PCA}} = \tilde{\mathbf{F}}\,\mathbf{U}_{\text{debias}}[:, 1{:}d]. \tag{18}$$

Semantic correspondence is evaluated on SPair-71k, isolating cross-image alignment from segmentation, by using these two d-dimensional representations. Results are plotted in Fig. 9. Across all dimensionalities, the PCA-debiased curve remains consistently above the PCA-original. This shows that the variance removed by our projection does not encode meaningful semantic information. Once positional variance is eliminated, the remaining dimensions form a cleaner and more stable semantic subspace. In contrast, the leading components of the *original* features also capture structured positional signals, resulting in consistently lower correspondence accuracy. In particular, at $d = 32$, the plot shows a gap of +7.8 PCK@0.10. As d increases, both representations approach their full-dimensional capacity, and the gap converges to +2.8 PCK@0.10. While this experiment does not establish that *all* positional biases are removed, it provides clear evidence that the directions suppressed by our debiasing are not useful for identifying semantic correspondences, and that the resulting representation supports more reliable matching across a wide range of dimensionalities.

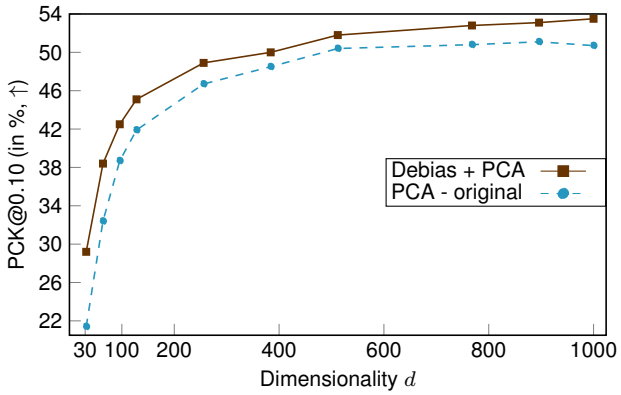

Figure 9. **Debiased features preserve semantic structure.** We compute PCA bases on COCO training images for *(i)* the original DINOv3 features and *(ii)* our debiased features. We then project both representations to d principal components and evaluate semantic correspondence (PCK@0.10) on SPair-71k. Across all d, Debias + PCA consistently outperforms PCA-original, indicating that the removed directions primarily encode positional bias. After debiasing, the remaining variance concentrates around semantic structure, yielding better results when compressed via PCA.

Table 6. **Hyperparameter overview.** INSID3 uses only three hyperparameters, all fixed across datasets, for both semantic, part, and personalized segmentation.

Hyperparameter		**Value**
τ	(clustering sensitivity)	0.6
α	(aggregation threshold)	0.2
s	(debias rank)	500

A.4. Discussion

We proposed a simple approach for removing positional biases from dense features of DINOv3. We demonstrate that debiased features aid accuracy for cross-image matching. Still, positional information remains a trade-off. While positional information can support approaching certain tasks, strong absolute positional biases can hamper cross-image correspondence. For example, in semantic segmentation, predicting sky can benefit from absolute positional information, whereas correspondence tasks require features that generalize across spatial positioning. Existing work in different domains also removed absolute positional information explicitly or implicitly from dense features by exploiting equivariance across augmented views [76, 79, 81] or multi-view consistency [32, 80, 82]. Different from these approaches, our approach does not require any training and provides a flexible, efficient, and simple decomposition between semantic features and absolute positional encoding.

Table 7. **Comparison of INSID3 (mIoU in %, ↑) on 5-shot semantic and part segmentation.** Models are provided with 5 contextual examples and tasked with segmenting the annotated concept in the target image. INSID3 scales effectively to multiple references, achieving robust performance across domains. All hyperparameters are reused from the 1-shot setting without any tuning, highlighting the versatility of our approach. Gray indicates the model was trained on the corresponding train split of the dataset; best results **bold**, 2nd best underlined. † denotes a GF-SAM variant using DINOv3 features.

			Semantic						Part		
Method	**Encoder**	**#Param**	**LVIS-92^i**	**COCO-20^i**	**ISIC**	**SUIM**	**iSAID**	**X-Ray**	**PASCAL**	**PACO**	**Avg**
Task-specific fine-tuning: *Semantic + mask supervision*											
SegGPT [64]	ViT	354 M	25.4	67.9	45.2	33.7	35.9	**89.1**	42.8	14.1	44.3
SINE [38]	DINOv2	373 M	35.5	66.1	28.6	54.8	40.5	40.6	36.4	25.4	41.0
DiffewS [74]	Stable Diffusion	890 M	35.4	72.2	32.7	49.8	48.0	45.1	39.7	26.1	43.6
Training free: *Mask-supervised pre-training*											
Matcher [39]	DINOv2 + SAM	945 M	40.0	60.7	35.0	50.6	34.3	71.2	45.8	33.6	46.4
GF-SAM [69]	DINOv2 + SAM	945 M	44.2	**66.8**	55.2	58.1	52.4	52.9	51.2	41.9	52.8
GF-SAM† [69]	DINOv3 + SAM	945 M	42.8	64.4	56.7	58.6	53.2	54.7	50.3	39.3	52.5
↳ + *our debias*	DINOv3 + SAM	945 M	43.6	64.6	58.2	59.2	54.1	59.1	51.4	40.2	53.8
Training free: *Unsupervised pre-training*											
INSID3 *(ours)*	DINOv3	**304 M**	**47.2**	65.1	**63.9**	**61.7**	**56.9**	80.1	**57.1**	**46.8**	**59.9**

B. Implementation details

INSID3 relies on only three scalar hyperparameters, summarized in Tab. 6. All hyperparameters are selected once on the COCO-20^i training split using a k-fold cross-validation procedure (with $k = 3$), and are kept *fixed across all datasets, domains, and tasks* (semantic, part, and personalized segmentation). No dataset-specific or task-specific tuning is performed. The clustering sensitivity τ controls the granularity of the agglomerative clustering step. We choose a value ($\tau = 0.6$) that provides fine-grained clusters at part level, and rely on aggregation based on similarity with the in-context example to merge clusters to the desired granularity. The aggregation threshold α determines how strictly clusters must agree semantically and structurally with the seed region; the debiasing rank s specifies the number of positional directions removed by our projection. Despite their simplicity, these three values generalize well across all experiments, highlighting the stability of the method.

C. Additional Experiments

Here, we provide additional experimental results, complementing Sec. 4 of the main paper.

C.1. k-shot segmentation

Although our main experiments evaluate INSID3 under the 1-shot segmentation setting, our approach naturally extends to multiple reference examples. Let $\{(\mathbf{I}^{r_m}, \mathbf{M}^{r_m})\}_{m=1}^{k}$ denote the k reference images and masks of the k-shot setting. The clustering of the target image (Sec. 3.2) and the aggregation procedure (Sec. 3.4) remain unchanged. Only the correspondence stage (Sec. 3.3) requires adaptation.

For each target patch i, we compute its nearest neighbor in every reference image using the debiased features:

$$\mathrm{NN}_m(i) = \arg\max_{j \in \Omega} \langle \tilde{\mathbf{F}}_i^t, \tilde{\mathbf{F}}_j^{r_m} \rangle. \tag{15}$$

A patch is retained as a valid candidate if its nearest neighbor in the reference lies within the mask for a majority of the reference images, *i.e.*,

$$\left| \{\, m \mid \mathbf{M}^{r_m}_{\mathrm{NN}_m(i)} = 1 \,\} \right| \;\geq\; \left\lceil \frac{k}{2} \right\rceil. \tag{19}$$

This majority-vote filtering keeps only correspondences consistently supported across references. For prototype construction, we compute one debiased prototype per reference, and then average them:

$$\tilde{\mathbf{p}}^r = \frac{1}{k} \sum_{m=1}^{k} \tilde{\mathbf{p}}^{r_m}. \tag{20}$$

The aggregated prototype is plugged into Eq. (10) to compute cross-image similarity for each target cluster, which drives seed selection and subsequent aggregation. We follow standard few-shot segmentation protocols and evaluate the case $k = 5$ in Tab. 7.

Discussion. We observe that INSID3 benefits consistently from additional reference examples. Moving from the 1-shot to the 5-shot setting, our method yields substantial gains across all benchmarks (*cf*. Tabs. 1 & 7), with improvements in the range of +1.3–9.5 % pts. mIoU. Overall, INSID3 achieves the best average score with a gain over the best competitor of +6.1 % points. In particular, INSID3 outperforms GF-SAM by +3.0 % pts. mIoU on the challenging LVIS-92^i, and by 5.4 % pts. on average on Part segmentation. On challenging datasets from the medical domain, the

Table 8. **Backbone analysis for INSID3.** We report mIoU (in %, ↑) on COCO-20^i of INSID3 with different backbone features. DINOv3 features lead to the best segmentation accuracy.

DINOv3	DINOv2	Franca	Perception Enc.	Stable Diff.
57.6	45.1	39.2	48.4	33.2

gain is +8.7 % and +27.2 % pts. on ISIC and X-Ray, respectively. On SUIM the performance boost is +3.6 % pts., and +4.5 % pts. mIoU on the remote-sensing domain of iSAID. Only on COCO, INSID3 incurs in a small gap of −1.7 % pts. Notably, our proposed debiasing strategy also improves the baseline of GF-SAM where DINOv3 is used as encoder, providing an improvement of +1.0 % pts. over the original method with DINOv2. Overall, the table reveals a trend consistent with the 1-shot setting: fine-tuned methods struggle to generalize when the test distribution deviates from the training distribution. Even when excluding the challenging medical and remote-sensing datasets, models fine-tuned on COCO exhibit a substantial accuracy drop on LVIS, whose long-tailed taxonomy differs from that of COCO. In contrast, training-free approaches avoid these pitfalls, offering stronger generalization across tasks and datasets without requiring domain-specific adaptation. Among them, our method stands out by leveraging only self-supervised DINOv3 features, yet achieving results that are competitive with or superior to methods relying on explicit mask-level supervision. Importantly, these 5-shot results are obtained *without* introducing any new components or tuning additional hyperparameters: we reuse exactly the same encoder, clustering configuration, and aggregation thresholds as in the 1-shot experiments, and only replace the correspondence stage with the simple majority-vote and prototype averaging scheme in Sec. C.1. This shows that INSID3 scales to multiple references in a strictly plug-and-play manner, and that its training-free design can effectively integrate complementary in-context signals without dataset- or shot-specific adaptation.

C.2. Dense representations matter

A central motivation of our work is that earlier ICS methods had to *compensate* for the limited spatial structure of existing VFM features, either by training segmentation decoders [41], fine-tuning diffusion models [74], or coupling DINOv2 with SAM for mask generation [39, 69]. These components were introduced because prior VFMs did not simultaneously provide *(i)* reliable semantic correspondence across images and *(ii)* sufficiently dense, part-aware structure within a single image. We validate this argument directly by replacing DINOv3 with other VFMs. In particular, we use features from DINOv2 [47], Franca [78], Perception Encoder [2], and Stable Diffusion 2.1 [52]. We apply INSID3 without any algorithmic changes, and tune hyperparameters for each backbone (*cf*. Tab. 8). Segmentation accuracy of all other VFMs drops significantly on COCO-20^i over DINOv3. These results support a key observation: DINOv3 self-supervised representations jointly exhibit strong semantic structure and spatially localized, dense features, which together are sufficient for ICS to arise from a single frozen backbone without decoders, fine-tuning, or external models.

Table 9. **Comparison with SAM3.** We compare INSID3 to Segment Anything 3 (SAM 3) on COCO-20^i using mIoU (in %, ↑). We evaluate two adaptations of SAM 3: *(i)* a *video-style* formulation, where reference and target are treated as consecutive frames and the mask is propagated, and *(ii)* an *image concatenation* strategy, where images are combined horizontally and prompted.

Method	mIoU
Training free: *Mask-supervised pre-training*	
Segment Anything 3 (video-style propagation)	26.7
Segment Anything 3 (image concatenation)	52.9
Training free: *Unsupervised pre-training*	
INSID3 *(ours)*	**57.6**

C.3. Comparison with SAM 3

We further compare INSID3 with the recent Segment Anything 3 (SAM 3) model [75], a foundation model for promptable segmentation that supports multiple input modalities, including points, masks, and text prompts. Among its capabilities, SAM 3 enables *visual prompting*, where a mask provided on an object can be used to segment other instances of the same concept *within the same image*. In this experiment, we investigate whether such visual prompting can generalize across *different* images, *i.e*. from a reference to a target, to solve in-context segmentation. Since SAM 3 is not natively designed for this setting, we consider two adaptations. First, following prior work [12], we adopt a *video-style* formulation, where the reference and target images are treated as consecutive frames and the mask is propagated from the reference (first frame) to the target. Second, we propose an *image concatenation* strategy, where the reference and target are stacked horizontally into a single image and the mask is provided on the reference region. Results on COCO-20^i are reported in Tab. 9. We observe that the image concatenation strategy significantly outperforms the video-style propagation, achieving 52.9 % mIoU compared to a lower accuracy of 26.7 %, respectively. Notably, INSID3 attains 57.6 % mIoU, outperforming SAM3 by 4.7 percentage points despite not relying on any segmentation supervision.

C.4. Corner case: Empty mask

The standard in-context segmentation formulation assumes that the reference concept is present in the target image.

Table 10. **Absence of the reference concept.** Percentage of correct empty-mask predictions (%, ↑) when the target image does not contain the prompted object (COCO, 4 000 pairs). *n.a.*: methods that always return a non-empty mask (*i.e.*, 0 % in practice).

Method	Correct empty predictions (in %) ↑
Task-specific fine-tuning: *Semantic + mask supervision*	
SegGPT [64]	79
SegIC [41]	36
Training free: *Mask-supervised pre-training*	
PerSAM [72]	*n.a.*
Matcher [39]	*n.a.*
GF-SAM [69]	*n.a.*
Training free: *Unsupervised pre-training*	
INSID3 *(ours)*	**85**

Table 11. **Computational analysis** of INSID3 *vs*. previous methods in milliseconds (ms, ↓). Inference time is measured for a single in-context example (*i.e.*, example and target image). Runtime is measured on a single RTX 4090.

Method	Backbone	Resolution	Runtime ↓
Task-specific fine-tuning: *Semantic + mask supervision*			
SegGPT [64]	ViT	640^2	110 ms
SegIC [41]	DINOv2	896^2	301 ms
Training free: *Mask-supervised pre-training*			
PerSAM [72]	SAM	1024^2	504 ms
Matcher [39]	DINOv2 + SAM	1024^2	9 000 ms
GF-SAM [69]	DINOv2 + SAM	1024^2	1 030 ms
Training free: *Unsupervised pre-training*			
INSID3 *(ours)*	DINOv3	1024^2	**302 ms**

However, a practical use case arises when the target image does not contain the object specified by the reference prompt, in which case the desired output is an empty mask. INSID3 can naturally accommodate this case by also subjecting the seed cluster to the aggregation criterion (*cf*. Eq. 12). When the concept is absent, cross-image similarity scores remain uniformly low, and no cluster satisfies the selection criterion, resulting in an empty prediction.

To evaluate this setting, we construct 4 000 reference–target pairs from COCO (80 classes), where the target image does not contain the reference concept. Table 10 reports the percentage of correct empty predictions. INSID3 correctly predicts empty masks in 85% of cases. Methods based on SAM [39, 69, 72] cannot produce empty outputs by design and therefore always return a non-empty mask. Compared to supervised approaches, INSID3 outperforms SegIC [41] and SegGPT [64] despite requiring no task-specific training.

Table 12. **Detailed computational analysis of INSID3.** We report inference time of each component in milliseconds (ms, ↓). Inference time is measured for a single in-context example. Runtime measured on a single RTX 4090.

Component	Runtime ↓
Encoder forward + debiasing	78 ms
Compute similarity + cluster scoring	3 ms
Clustering	166 ms
CRF refinement	55 ms
Total inference time	**302 ms**

Table 13. **Computational analysis for different resolutions.** We report inference time of INSID3 for different resolution in milliseconds (ms, ↓). Inference time is measured for a single in-context example. Runtime measured on a single RTX 3090.

Low resolution ← **Runtime of INSID3** → *High resolution*						
512 px	**720 px**	**896 px**	**1 024 px**	**1 440 px**	**1 600 px**	**1 760 px**
93 ms	180 ms	230 ms	302 ms	689 ms	930 ms	1 102 ms

D. Computational Cost of INSID3

Table 11 reports the inference speed of INSID3 compared to training-free and fined-tuned baselines. INSID3 provides a significantly faster inference runtime than existing training-free approaches. This efficiency follows directly from our *single-backbone design*. In contrast, SAM-based pipelines separate the problem into two stages: DINOv2 is used to compute semantic correspondences, and SAM is prompted to produce masks. Because these two models operate in different feature spaces, semantic matching and mask generation are decoupled and must be coordinated through additional steps (*e.g.*, point selection, prompt engineering, mask filtering, and scoring). This multi-stage interaction results in significant inference overhead.

In contrast, INSID3 performs both semantic alignment and mask generation directly in the DINOv3 feature space. On a single RTX 4090, INSID3 runs at 302 ms, compared to 1 030 ms for GF-SAM, the most competitive baseline w.r.t. downstream accuracy. The detailed runtime breakdown (*cf*. Tab. 12) shows that the dominant cost of INSID3 is agglomerative clustering (166 ms), with the DINOv3 forward pass and debiasing accounting for 78 ms. Similarity computation and cluster scoring are negligible (3 ms). We use a GPU-accelerated dense CRF [77] for refinement (55 ms). Although agglomerative clustering has quadratic complexity in the number of tokens, INSID3 nevertheless scales well in practice to higher image resolutions (*cf*. Table 13). Notably, INSID3 at $1\,760 \times 1\,760$ remains comparable in runtime to GF-SAM at $1\,024 \times 1\,024$. Overall, INSID3 remains computationally efficient by operating entirely within a single feature space, without relying on auxiliary segmenta-

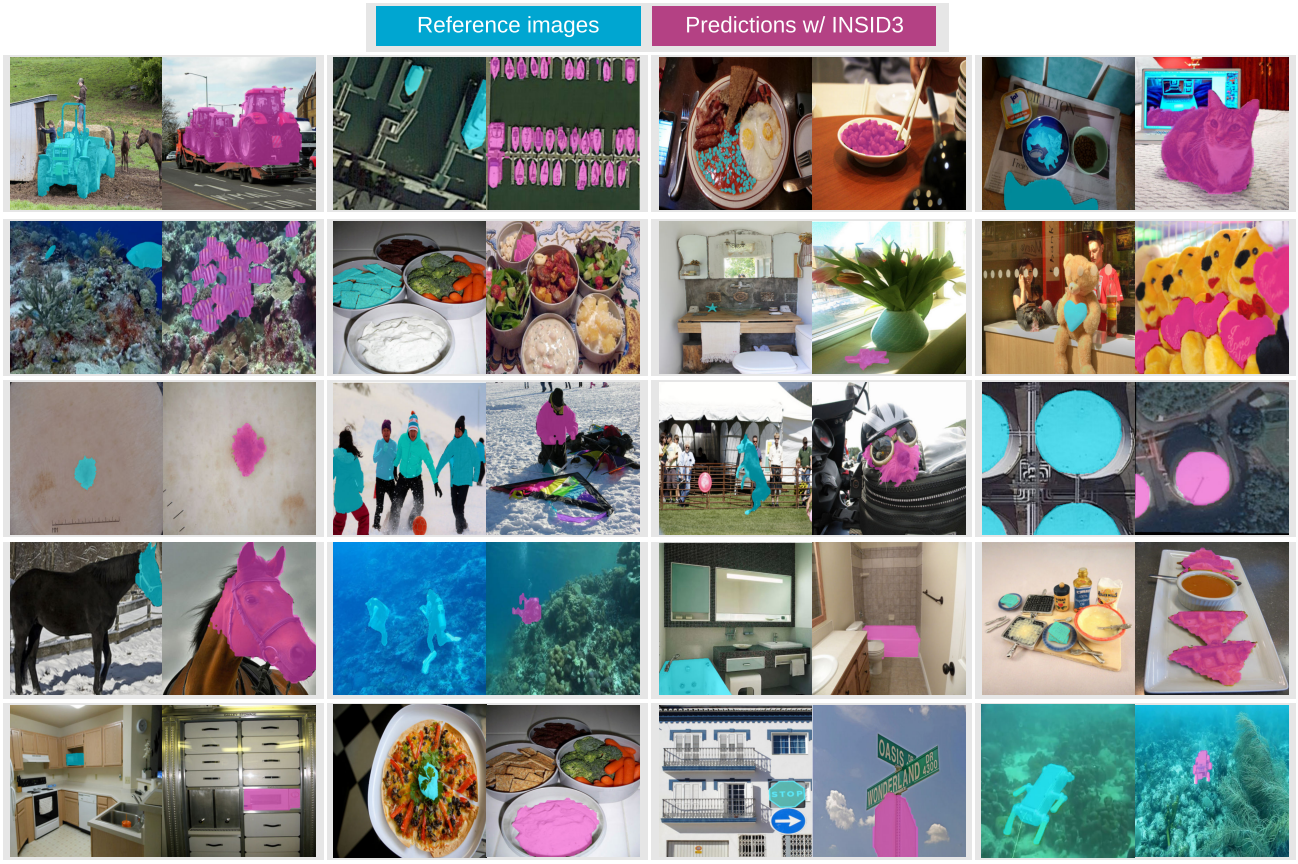

Figure 10. **Qualitatives results with INSID3**. Each pair shows the reference image with its annotated region *(light blue mask)* and the predicted mask on the target *(purple)*. INSID3 handles a wide range of semantic granularities, from full objects to fine-grained parts, and generalizes robustly across diverse domains, including aerial, marine, medical, and everyday scenes.

tion models such as SAM. We emphasize that all reported runtimes are measured in 32-bit precision and averaged over 100 examples. Additional speedups could be obtained through standard optimizations such as half-precision inference.

E. Qualitative Examples

Figure 10 presents additional qualitative results across a wide range of scenarios. Each example shows the reference image with its annotated region *(light blue)* alongside the predicted mask on the target image *(purple)*. Across a wide range of visual concepts, spanning different semantic granularities (from full objects to fine-grained parts) and diverse visual domains (aerial imagery, marine scenes, medical scans, and everyday images), INSID3 consistently produces coherent, accurate masks. It reliably identifies the correct concept even in the presence of distractor objects, similar classes, or large appearance changes, illustrating the consistency and versatility of INSID3.

F. Limitations and Future Work

INSID3 demonstrates that strong in-context segmentation capabilities can emerge directly from frozen self-supervised representations, without task-specific training or architectural modifications. While this simple formulation already achieves competitive results across diverse domains and granularities, some limitations remain. First, INSID3 currently handles one target concept at a time, requiring separate reference prompts when multiple concepts are present in the target image. Extending the method to jointly reason about multiple concepts within a single inference pass would be an interesting direction for future work. Second, INSID3 relies on *masks* to define the target concept. In contrast, models such as Segment Anything can be prompted using lighter spatial annotations such as points or bounding boxes. These forms of annotation are typically cheaper and faster to collect, which could facilitate practical deployment and interactive use; incorporating lighter prompt modalities would therefore broaden the applicability of the approach. A related future direction concerns instance-level reasoning. While our focus is *semantic* in-context segmen-

tation, one could instead provide an example object in the reference image and ask the model to segment the corresponding instances in the target image separately, rather than merging them into a single semantic mask. In our observations, clusters belonging to the same instance often exhibit stronger mutual similarity than clusters from different instances. Exploiting these affinities to iteratively recover multiple instances, for example by expanding a seed with its most similar clusters and then re-seeding on the remaining regions, is an interesting direction for future work. Finally, INSID3 relies on the semantic structure encoded in frozen self-supervised features of DINOv3. Although our debiasing improves cross-image matching, the quality of the final segmentation still depends on the representational properties of the underlying backbone, suggesting that future advances in self-supervised representations could further strengthen this paradigm.

References

[75] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath et al. SAM 3: Segment anything with concepts. *arXiv:2511.16719 [cs.CV]*, 2025. v

[76] Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. FeatUp: A model-agnostic framework for features at any resolution. In *ICLR*, 2024. iii

[77] Đ.Khuê Lê-Huu and Karteek Alahari. Regularized Frank-Wolfe for dense CRFs: Generalizing mean field and beyond. In *NeurIPS*, pages 1453–1467, 2021. vi

[78] Shashanka Venkataramanan, Valentinos Pariza, Mohammadreza Salehi, Lukas Knobel, Spyros Gidaris, Elias Ramzi, Andrei Bursuc, and Yuki M. Asano. Franca: Nested Matryoshka clustering for scalable visual representation learning. *arXiv:2507.14137 [cs.CV]*, 2025. v

[79] Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. AnyUp: Universal feature upsampling. In *ICLR*, 2026. iii

[80] Jiawei Yang, Boris Ivanovic, Or Litany, Xinshuo Weng, Seung Wook Kim, Boyi Li, Tong Che, Danfei Xu, Sanja Fidler, Marco Pavone, and Yue Wang. EmerNeRF: Emergent spatial-temporal scene decomposition via self-supervision. In *ICLR*, 2024. iii

[81] 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 *ECCV*, volume 85, pages 453–469, 2024 iii

[82] Yuanwen Yue, Anurag Das, Francis Engelmann, Siyu Tang, and Jan Eric Lenssen. Improving 2D feature representations by 3D-aware fine-tuning. In *ECCV*, volume 2, pages 57–74, 2024 iii