TOKEN-LEVEL DECISIONS FOR TRAINING-FREE FEW-SHOT SEGMENTATION OF OVERHEAD IMAGERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Few-shot segmentation asks a model to delineate a class in a query image from a handful of annotated examples. The need is most acute in remote sensing, where labels are scarce and the imagery departs from the photographs vision backbones are pretrained on. A growing family of methods answers it without training by reading the prediction off a frozen self-supervised encoder, and its design effort has concentrated on the class model that turns support features into a score. We argue that accuracy is governed largely by what surrounds the class model, first by how finely the decision is committed in space and then by how well the feature geometry is refined on the support. FROST is a training-free segmenter built on this position. It projects a positional component out of the frozen DINOv3 tokens, whitens a within-class scatter estimated from the support under heavy shrinkage, scores every token by a density ratio whose threshold the Bayes rule fixes at zero, and commits a label per token rather than per cluster, with nothing tuned per dataset. A component study attributes 13.5 mIoU to the granularity of the decision, more than every other component together, and 5.1 to the feature refinement. An oracle labelling of every decision unit shows the first to be a bound rather than a scoring failure, since coarsening the unit costs almost exactly the accuracy it forecloses, and the effect is not ours alone, since moving the decision of a competing method to the token raises it by 2.3 mIoU. Across seventeen remote-sensing benchmarks FROST surpasses both training-free and learning-based methods, leading by 5.5 mIoU from a single annotated example and by a wider margin at every larger support set, while remaining among the smallest models compared.

1 INTRODUCTION

Every new remote-sensing segmentation task begins with dense pixel labels drawn by hand, an effort that grows with the number of scenes and the expertise the classes demand and recurs whenever the sensor, the region or the class set changes. Label efficiency is therefore a precondition for the field, and the obstacle is sharper than in natural imagery, since a model pretrained on everyday photographs meets an unfamiliar viewpoint, scale and spectral makeup overhead, so competence on one collection seldom survives the move to another without new labels and training.

Few-shot segmentation answers this constraint by asking a model to reproduce a class from a few reference masks. The established route learns to do so from labelled episodes, but the parameters it fits carry the imprint of the classes and scenes they saw, so a new domain reopens the need for annotation and retraining (Min et al., 2021; Hong et al., 2022). A newer route drops the optimization and reads the answer off a frozen self-supervised encoder, whose feature neighbourhoods already place semantically related regions together having seen no segmentation label (Siméoni et al., 2025). Within this route the design effort has gone into the class model, the function that turns support features into a foreground score, offered as a few prototypes (Zakir & Ho, 2026) or as a discrete partition of the scene (Cuttano et al., 2026). Two choices surrounding that class model have drawn far less attention. One is the geometry in which the comparison runs, reshaped so far only by subtracting a direction fixed once rather than read from the support. The other is the region over which one decision is committed, which no class model can subdivide once a whole cluster shares a label. Figure 1 makes the second concrete, since widening the region that carries one label from a single token to a block of eight tokens on a side costs 11.7 mIoU, more than every other component of the pipeline together.

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.

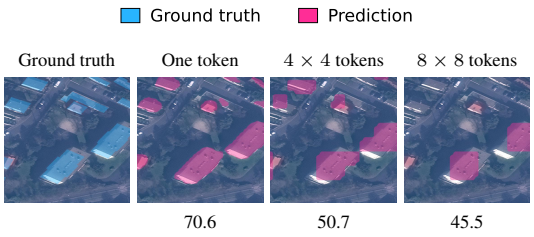
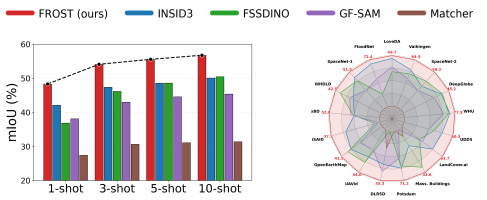


Figure 1: With every stage before the decision held fixed, widening the region that carries one label costs more accuracy than every other component of FROST together. Each value is the foreground IoU of that prediction on one SpaceNet-2 episode, of which the panels show a detail.



(a) Per support size (b) Per benchmark

Figure 2: On the seventeen remote-sensing benchmarks, FROST leads the strongest training-free few-shot segmenters in mean foreground mIoU at every support size $k \in \{1, 3, 5, 10\}$, reported per support size on the left and per benchmark on the right.

Our position is that what frozen-feature segmentation still lacks is not more training but a better setting for whichever class model is used, first a decision committed at the finest granularity the encoder allows, which bounds what any class model can reach, and then a feature geometry refined on the support before any comparison is made. We introduce FROST, a training-free few-shot segmenter that realizes both. It projects a positional component out of the frozen DINOv3 tokens and whitens the within-class scatter of the reference tokens under heavy shrinkage, scores every token by a density ratio whose threshold the Bayes rule fixes at zero under equal priors, and commits a label for every token rather than for a cluster as a whole. The whitening transform, the bandwidth and a spatial gate are all read from the support alone, so nothing is calibrated per dataset and no threshold, prototype count or cluster count has to be chosen. Overhead imagery is where these choices should matter most. A class there is a scatter of many small instances threaded through background that shares their built appearance, so a region large enough to be grouped reliably tends to hold both and a single label for it is wrong on one of them, while the foreground fills enough of a tile for the equal-prior threshold to be the right one. Both properties hold far less often in object-centric natural images, so we expect, and later observe, the advantage of FROST to be largest overhead.

Our contributions are threefold. First, we identify the granularity of the decision as the dominant source of accuracy in this setting and the feature refinement as the next largest. A component study that removes each stage in turn establishes the attribution, an oracle labelling of every decision unit explains it, since what a granularity forecloses accounts for nearly all of what coarsening it costs, and the same labelling applied to five decisions shows that a fine unit creates room rather than accuracy, the loosest bound of the five belonging to the least accurate method. Second, we present FROST, the training-free segmenter this attribution is read from, which reads a high-resolution mask off a frozen backbone from support statistics alone and calibrates nothing per dataset. Neither the per-token label, which FSSDINO also commits, nor the shrinkage whitening, a classical estimator, is new on its own, and we claim neither as a primitive. Third, we evaluate FROST on seventeen remote-sensing benchmarks, where it overtakes both training-free and learning-based methods from a single mask and by a wider margin at every larger support set, and on natural-image and cross-domain benchmarks, where it overtakes the strongest training-free methods as references accumulate. Figure 2 summarizes this outcome, with FROST leading the mean at every support size, and Table 6 places it among the smallest models compared.

2 Related Work

Few-shot segmentation. The task hands a model a handful of labelled references and asks it to segment the same class elsewhere, and most solutions learn how. Prototype methods average the support foreground into a few vectors and match query pixels against them (Wang et al., 2019; Tian et al., 2020), correlation methods learn dense affinity or hypercorrelation between support and query (Min et al., 2021; Hong et al., 2022), and generalist in-context models predict a mask from a visual example (Wang et al., 2023). A distinct group leaves a promptable foundation model untouched and feeds it prompts derived from the references, so its mask decoder produces the output (Kirillov et al., 2023; Zhang et al., 2024b; Liu et al., 2023). Each design pays for its accuracy with a learning stage

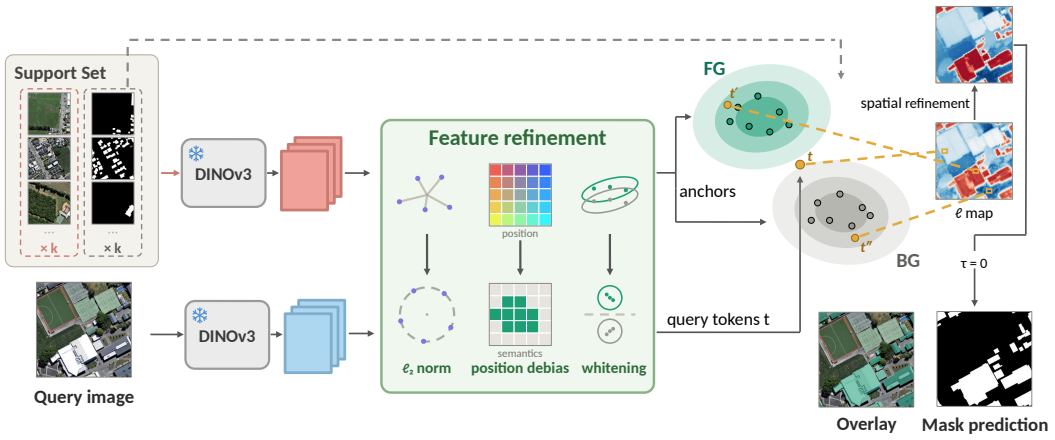


Figure 3: The FROST pipeline, from frozen DINOv3 tokens to a mask through feature refinement, a density ratio over foreground and background anchors, spatial smoothing and gating, and upsampling.

on segmentation data or with a decoder already trained on masks, and because that stage sees natural images, the imprint it leaves transfers poorly to overhead imagery. FROST keeps neither and reads its prediction from frozen features and reference statistics alone.

Segmentation from frozen self-supervised features. Self-supervised vision transformers expose dense features whose neighbourhoods respect object extent and cross-image correspondence, which has supported unsupervised discovery and matching with no fine-tuning (Caron et al., 2021; Siméoni et al., 2021; Amir et al., 2021), and the localized features of DINOv3 carry segmentation from a single frozen model (Siméoni et al., 2025). INSID3 segments in context from these features by grouping a scene into discrete regions and transferring labels through correspondence, after cancelling a positional bias it finds in the similarities (Cuttano et al., 2026). FSSDINO probes the few-shot capacity of the same features with cluster prototypes refined by Gram matrices (Zakir & Ho, 2026). FROST is training-free in the same sense and adopts the positional debiasing of INSID3, but it departs from both in the two stages that surround the class model. Where INSID3 and FSSDINO compare tokens in a geometry fixed once the encoder has run, FROST also whitens a within-class scatter estimated from the reference masks of the episode, so the geometry adapts to the support. Where INSID3 assigns one label to a whole cluster, FROST labels every token on its own, as FSSDINO also does.

Regularized and nonparametric statistics. FROST is assembled from established statistical tools, and deliberately so, since attributing accuracy to the stages that surround the class model is credible only when the class model itself is as standard as the literature offers. Pulling a noisy covariance estimate toward a structured target is the classical remedy when the dimension is large relative to the sample (Friedman, 1989; Ledoit & Wolf, 2004), and deciding between classes by competing kernel density estimates, with the bandwidth read from the data, is standard nonparametric practice (Silverman, 2018). FROST carries this machinery onto the dense features of a foundation model, where the anchors of one episode outnumber the feature dimension yet are neighbouring tokens of a handful of scenes rather than independent draws, so shrinkage plays exactly the role the theory anticipates. The samples are moreover tokens on a sphere rather than tabular observations, a geometry that motivates the spherical kernel developed next.

3 Method

FROST assembles a few-shot segmenter from a frozen encoder and statistics computed on the support set alone, with no training. We fix the setting and notation first, then describe its three stages in turn, the feature refinement that reshapes the geometry the comparison runs in, the class model that scores each query token by a density ratio, and the spatial refinement that commits a label for every token, a granularity Section 5 finds dominant, and carries it to full resolution.

3.1 SETTING AND OVERVIEW

Given a query image and k support images with binary masks for a target class, we predict that class in the query. At our 1024×1024 input, a frozen DINOv3 ViT-L/16 maps each image to a 64×64 grid of $C = 1024$ dimensional patch tokens, and horizontal flips double the support. FROST refines the tokens, splits the support into foreground and background anchors, scores each query token by their density ratio, smooths and gates the field, and commits a label for every token by thresholding it once upsampled to full resolution (Figure 3). The whitening transform, kernel bandwidth, and gate radius are read from the support, while the projection rank and shrinkage intensity are fixed across benchmarks. Throughout, j indexes token-grid positions, m feature directions and ν propagation steps.

3.2 FEATURE REFINEMENT

A query token reaches a decision only through its similarities with the anchors, so the geometry of the feature space bounds what any class model can recover, and we reshape it in three steps.

Normalization. We first place every token on the unit sphere by L^2 normalization, which removes the variation in token magnitude across images so the comparisons below depend on direction alone, which is what the von Mises–Fisher kernel of Section 3.3 reads.

Positional debiasing. We then apply the positional debiasing of INSID3 (Cuttano et al., 2026), which finds that the cross-image similarity of frozen tokens is driven partly by absolute position rather than content. Being shared across images, this component raises the similarity between unrelated tokens at the same location and would pull the score toward position rather than appearance, so following INSID3 we estimate it by passing a content-free Gaussian image $X_{\text{noise}} \sim \mathcal{N}(0, I)$ through the encoder, taking the r leading directions $u_1, \dots, u_r$ of its token covariance, and projecting them out of every token by $v \mapsto v - \sum_{m=1}^{r} (v^\top u_m)\, u_m$ before renormalizing, with $r = 250$.

Shrinkage whitening. The reference tokens vary along a few high-variance directions tracking global scene appearance rather than the target class, and we suppress them so the kernel of Section 3.3 responds to class-discriminative variation. We use the within-class scatter rather than the total covariance, since centring each class on its own mean keeps the foreground-to-background mean difference intact, which a total covariance would absorb into its leading directions and the whitening would then erase. Splitting the support tokens by the reference masks into a foreground group $\mathcal{S}_{\text{FG}}$ and a background group $\mathcal{S}_{\text{BG}}$ of sizes N_{FG} and N_{BG} with means $\bar{a}_{\text{FG}}$ and $\bar{a}_{\text{BG}}$, we form

$$\Sigma_W = \frac{1}{N_{\text{FG}} + N_{\text{BG}} - 2} \Big(\sum_{a \in \mathcal{S}_{\text{FG}}} (a - \bar{a}_{\text{FG}})(a - \bar{a}_{\text{FG}})^\top + \sum_{a \in \mathcal{S}_{\text{BG}}} (a - \bar{a}_{\text{BG}})(a - \bar{a}_{\text{BG}})^\top \Big). \quad (1)$$

One support view already supplies 64×64 tokens, so the anchor count exceeds C, but these are neighbouring tokens of a few scenes whose strong correlation leaves far fewer effectively independent observations, and Σ_W rests on too little independent evidence to invert as it stands. We regularize it by heavy shrinkage toward a scaled identity, forming $\Sigma_{W,\lambda} = (1 - \lambda)\, \Sigma_W + \lambda\, \mu_W\, I_C$ with the mean eigenvalue $\mu_W = \frac{1}{C} \operatorname{tr}(\Sigma_W)$ and intensity $\lambda = 0.95$, so the empirical scatter retains only the weight $1 - \lambda = 0.05$ (Ledoit & Wolf, 2004). Appendix A.4 reports the response to λ, a broad optimum under heavy shrinkage, together with the intensity a closed-form rule selects on these anchors, which exceeds what complete shrinkage allows. We whiten every token by $\Sigma_{W,\lambda}^{-1/2}$ and keep the whitened scale rather than renormalizing, since the bilateral term of Section 3.4 reads the whitened token directly, while the density ratio and the forward test of the same section compare directions through cosine similarity and need no unit norm. We write t_j for the whitened token at position j.

Anchors. The reference masks, downsampled to the token grid, label each support token, so after whitening the foreground tokens form the anchor set $\mathcal{A}_{\text{FG}}$ of size N_{FG} and the background tokens form $\mathcal{A}_{\text{BG}}$ of size N_{BG}, and the horizontal flips of the support enlarge both. The flip is applied to the reference image and its mask together, so the flipped tokens carry the same labels and simply double the anchor count, at the price of one further encoder pass over each reference. Neither set is subsampled or summarized, so the density ratio of Section 3.3 reads every anchor and has no prototype count to choose.

3.3 DENSITY RATIO CLASSIFICATION

Density ratio. We treat each anchor set as a sample from a class-conditional density on the sphere and estimate it with the exponential of the cosine similarity $\cos(u, v) = \frac{u^\top v}{\|u\|\,\|v\|}$, the von Mises–Fisher kernel with concentration $1/\sigma$, so that for a query token t_j the two estimates are

$$\hat{p}(t_j \mid \mathrm{FG}) = \frac{1}{N_{\mathrm{FG}}} \sum_{a \in \mathcal{A}_{\mathrm{FG}}} \frac{1}{Z_\sigma} \exp\Big(\frac{\cos(t_j, a)}{\sigma}\Big), \qquad \hat{p}(t_j \mid \mathrm{BG}) = \frac{1}{N_{\mathrm{BG}}} \sum_{a \in \mathcal{A}_{\mathrm{BG}}} \frac{1}{Z_\sigma} \exp\Big(\frac{\cos(t_j, a)}{\sigma}\Big), \quad (2)$$

where σ is the kernel bandwidth and Z_σ its normalizing constant. Because each estimate is formed from the anchors themselves, it fixes neither a number of prototypes nor a number of clusters in advance, so the stage introduces no quantity that would have to be chosen for each dataset. We label the query token by the log ratio of the two densities,

$$\ell_j = \log \hat{p}(t_j \mid \mathrm{FG}) - \log \hat{p}(t_j \mid \mathrm{BG}), \quad (3)$$

in which Z_σ cancels, and we assign the token to the foreground when ℓ_j exceeds a threshold τ, which the Bayes rule places at $\tau = 0$ under equal class priors rather than at a value that would have to be calibrated. Every quantity this rule depends on is estimated on the support set, namely the two anchor sets, the whitening transform of Section 3.2 that produced the tokens t_j, and the bandwidth σ fixed below, so each added reference mask enters all of them at once.

Bandwidth selection. We fix σ from the support set before any query is scored, by a leave-one-out class margin over a small grid,

$$M(\sigma) = \frac{1}{N_{\mathrm{FG}} + N_{\mathrm{BG}}} \sum_a s_a\, \ell_\sigma^{\setminus a}(a), \quad (4)$$

where $s_a = +1$ for a foreground anchor and -1 for a background one and $\ell_\sigma^{\setminus a}(a)$ is the log ratio of Equation 3 at a with a removed from its own class set, and we keep the bandwidth maximizing this signed margin, rather than by held-out likelihood, because the threshold acts on the margin.

3.4 SPATIAL REFINEMENT

The class model so far scores each token on its own, and three inexpensive operations turn the resulting field into a mask while keeping the decision as fine as the encoder permits.

Bilateral propagation. A token mislabelled in isolation is usually surrounded by similar-looking neighbours labelled correctly, so we let the score field exchange evidence among them through a bilateral affinity coupling feature and colour similarity within a window,

$$B_{jj'} = \exp\Big(\frac{t_j^\top t_{j'}}{\gamma_f}\Big)\, \exp\Big(-\frac{\|c_j - c_{j'}\|^2}{\gamma_c}\Big), \qquad \|g_j - g_{j'}\|_\infty \le d_{\max}, \quad (5)$$

where t_j is the whitened token at position j, c_j the mean colour of its patch, g_j its grid coordinate, γ_f and γ_c feature and colour temperatures, and the affinity vanishes outside the window of radius $d_{\max}$. We row-normalize B into a row-stochastic matrix P and iterate $\ell_j^{(\nu+1)} = (1-\beta)\,\ell_j + \beta \sum_{j'} P_{jj'}\, \ell_{j'}^{(\nu)}$ from $\ell^{(0)} = \ell$ for a fixed number of steps with mixing weight $\beta = 0.70$, writing $\tilde{\ell}$ for the smoothed field, where the colour term keeps the propagation from bleeding across object boundaries.

Forward and backward candidate gating. A dense overhead scene contains structures that resemble the target locally and raise isolated false positives, which we suppress with a forward and a backward test that demand support-query agreement and mark which positions are admissible. The forward test admits a position only when its token has positive cosine similarity to the foreground prototype, the mean of the foreground anchors, and this prototype serves as a coarse pre-filter rather than the class model, which remains the density ratio of Equation 3. The backward test, adapted from the reciprocal matching of INSID3 (Cuttano et al., 2026), keeps a position only when it lies among the k_b nearest query positions of the anchor it matches, with $k_b = 3$, while a spatial gate of radius $\rho = \lfloor \rho_{\max}\, \pi_{\mathrm{FG}} \rceil$, with $\pi_{\mathrm{FG}} = N_{\mathrm{FG}}/(N_{\mathrm{FG}} + N_{\mathrm{BG}})$ and $\rho_{\max} = 4$, restricts the agreement to a neighbourhood that widens for a class occupying more of the scene. The foreground fraction therefore enters only through the extent of the admissible region and not through the per-token threshold, which the equal-prior assumption keeps at $\tau = 0$. The two tests are computed on the token grid and intersected into a candidate mask.

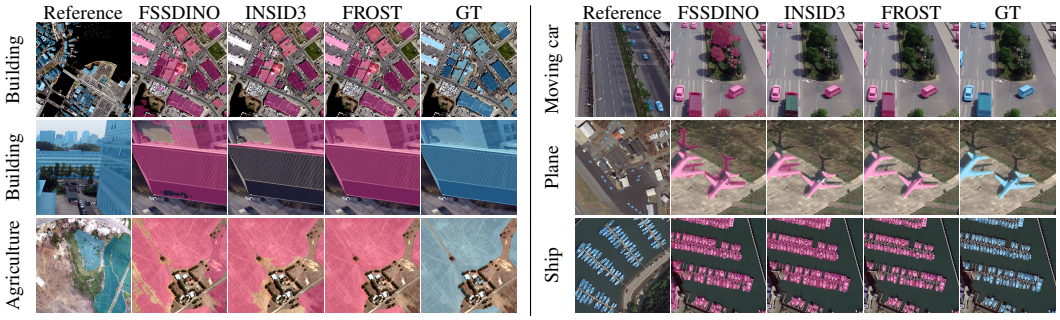


Figure 4: Qualitative results for the building, agriculture, moving car, plane, and ship classes, where GT is the ground truth.

Upsampling and the granularity of the decision. We upsample the scalar field rather than the prediction, interpolating $\tilde{\ell}$ bilinearly to 1024×1024 and thresholding afterwards, and intersect the result with the candidate mask at the same resolution to yield the final mask. Only the field and the candidate mask are upsampled, never the tokens, which keeps the step inexpensive. A label is therefore committed for every token on its own rather than shared across a cluster, which matters most where a structure spans only a few patches, since a single label covering several patches is wrong on part of what it covers, an effect Section 5 measures.

4 Experiments

We ask whether a training-free segmenter built on refined frozen features and a decision committed per token matches or surpasses methods that train, across the breadth of remote-sensing imagery. We first describe the benchmarks, the protocol, the baselines and the implementation, then report the main comparison, study how each method scales as reference masks accumulate, and test the same configuration beyond overhead imagery, leaving to Section 5 the question of which stage the accuracy rests on.

4.1 Setup

Benchmarks. We develop and primarily evaluate FROST on seventeen remote-sensing benchmarks spanning high-altitude satellite collections and lower-altitude drone views, grouped by the task they pose. The building and footprint group contains WHU Building (Ji et al., 2018), the two SpaceNet challenges (Van Etten et al., 2018), Massachusetts Buildings (Mnih, 2013), and the damaged-building benchmark xBD (Gupta et al., 2019), in each of which the target is a field of many small and visually varied structures. The land-cover group contains LoveDA (Wang et al., 2021), the ISPRS Potsdam and Vaihingen labelling sets (Rottensteiner et al., 2012), LandCover.ai (Boguszewski et al., 2021), OpenEarthMap (Xia et al., 2023), DeepGlobe (Demir et al., 2018), the drone benchmarks UAVid (Lyu et al., 2020) and UDD5 (Chen et al., 2018), iSAID (Zamir et al., 2019), DLRSD (Shao et al., 2018), and WHDLD (Shao et al., 2020), and the flood group contains the drone benchmark FloodNet (Rahnemoonfar et al., 2021). In all of them a class appears as a scatter of small instances rather than one dominant object, the regime FROST is designed for. To test how the same fixed configuration transfers beyond overhead imagery, we add six standard few-shot segmentation benchmarks. Four are natural-image benchmarks, PASCAL-5^i (Shaban et al., 2017), COCO-20^i (Nguyen & Todorovic, 2019), and LVIS-92^i (Liu et al., 2023) together with the part benchmark PACO-Part (Liu et al., 2023), whose targets are large object-centric or part-centric instances rather than a scatter of small ones. The other two leave natural imagery entirely, the dermoscopy benchmark ISIC (Codella et al., 2019) and the underwater benchmark SUIM (Islam et al., 2020), which probe imaging conditions unseen during pretraining.

Evaluation protocol. We evaluate on all publicly available evaluation data from each benchmark, yielding a large collection of episodes. For every query image and target class, we form a separate few-shot episode by randomly selecting k annotated examples of that class as the support set, with

Table 1: Foreground mean intersection over union (mIoU) on the remote-sensing building and footprint benchmarks together with the land-cover benchmark DeepGlobe and the flood benchmark FloodNet, with each entry the mean and the standard deviation over the four support sizes $k \in \{1, 3, 5, 10\}$, the best value in each column in bold and the second best underlined.

Method	WHU Building	SpaceNet-1	SpaceNet-2	Mass. Build.	xBD	DeepGlobe	FloodNet
Learning-based							
SegGPT	68.9 ± 5.8	41.3 ± 2.2	48.1 ± 4.3	11.2 ± 0.2	<u>35.5</u> ± 5.0	20.6 ± 1.8	45.9 ± 1.9
SegIC	56.5 ± 8.3	48.0 ± 4.0	46.9 ± 2.9	22.4 ± 3.2	31.2 ± 3.8	40.0 ± 0.7	65.1 ± 2.9
DiffewS	63.1 ± 1.7	45.5 ± 0.6	36.0 ± 3.7	18.3 ± 0.4	17.3 ± 3.3	39.7 ± 2.6	67.2 ± 5.9
SINE	22.2 ± 5.5	29.2 ± 6.7	15.8 ± 4.9	18.0 ± 0.9	11.3 ± 5.1	30.7 ± 3.7	39.2 ± 4.5
Training-free							
PerSAM	2.6 ± 0.1	2.4 ± 1.7	2.9 ± 0.3	0.6 ± 0.5	2.3 ± 0.5	22.9 ± 0.3	34.0 ± 3.9
Matcher	30.7 ± 3.0	32.5 ± 1.8	19.1 ± 1.5	15.2 ± 0.3	11.6 ± 1.7	38.2 ± 3.1	55.7 ± 3.2
GF-SAM	67.2 ± 1.8	42.1 ± 0.6	39.5 ± 3.8	11.6 ± 0.3	35.4 ± 4.9	43.0 ± 4.7	65.6 ± 7.3
FSSDINO	72.2 ± 2.0	47.0 ± 4.3	<u>49.4</u> ± 4.6	<u>31.0</u> ± 1.7	33.0 ± 7.0	<u>44.2</u> ± 8.0	60.2 ± 10.4
INSID3	<u>73.2</u> ± 0.4	<u>48.5</u> ± 1.8	44.0 ± 2.4	20.0 ± 0.7	32.1 ± 3.5	44.1 ± 4.8	<u>70.9</u> ± 5.2
FROST (ours)	**77.9** ± 0.4	**51.3** ± 2.3	**59.2** ± 2.3	**33.6** ± 0.4	**52.8** ± 4.2	**45.2** ± 4.3	**72.4** ± 8.5

Table 2: Foreground mIoU on the remote-sensing land-cover and drone benchmarks, with each entry the mean and standard deviation over the four support sizes $k \in \{1, 3, 5, 10\}$, the best in each column in bold and the second best underlined.

Method	LoveDA	Potsdam	Vaihingen	LandCover.ai	OpenEarthMap	UAVid	UDD5	iSAID	DLRSD	WHDLD
Learning-based										
SegGPT	45.3 ± 3.5	39.2 ± 0.5	50.7 ± 2.7	47.0 ± 1.5	20.6 ± 0.9	31.4 ± 1.4	27.4 ± 1.3	22.2 ± 1.6	20.1 ± 0.7	24.8 ± 2.1
SegIC	34.4 ± 0.5	61.2 ± 1.3	51.0 ± 2.8	56.4 ± 3.9	29.2 ± 0.6	53.0 ± 1.0	53.2 ± 2.6	23.5 ± 2.0	32.4 ± 0.7	34.1 ± 0.6
DiffewS	29.5 ± 1.6	52.3 ± 5.2	47.0 ± 1.5	49.3 ± 5.1	23.3 ± 0.9	**56.1** ± 2.5	44.6 ± 4.6	12.7 ± 1.4	31.3 ± 1.0	30.6 ± 1.6
SINE	15.3 ± 1.6	34.0 ± 3.6	23.1 ± 1.9	34.8 ± 1.8	20.6 ± 2.4	25.1 ± 3.6	27.2 ± 3.1	5.5 ± 1.7	16.2 ± 0.5	24.2 ± 1.4
Training-free										
PerSAM	14.3 ± 0.4	21.4 ± 1.8	9.2 ± 0.5	36.4 ± 1.2	6.8 ± 1.0	16.4 ± 1.5	17.9 ± 1.5	7.4 ± 1.4	20.7 ± 0.4	19.2 ± 0.6
Matcher	27.4 ± 0.8	43.1 ± 2.2	32.5 ± 1.8	43.4 ± 1.2	20.8 ± 1.2	35.9 ± 4.4	33.5 ± 1.4	19.8 ± 3.9	27.8 ± 1.6	24.9 ± 0.3
GF-SAM	44.7 ± 4.4	58.5 ± 2.3	51.6 ± 1.9	53.7 ± 3.6	32.2 ± 3.7	44.3 ± 3.9	47.1 ± 5.3	20.4 ± 3.0	<u>33.2</u> ± 2.2	36.9 ± 2.6
FSSDINO	42.5 ± 6.4	63.0 ± 7.5	55.0 ± 2.5	52.2 ± 11.6	<u>38.2</u> ± 6.5	49.9 ± 7.8	50.0 ± 9.3	21.5 ± 4.4	23.8 ± 3.9	<u>40.6</u> ± 5.2
INSID3	<u>48.4</u> ± 5.0	<u>64.2</u> ± 3.4	<u>57.7</u> ± 1.4	<u>58.7</u> ± 4.7	36.6 ± 5.8	47.5 ± 4.2	<u>54.5</u> ± 7.3	<u>32.0</u> ± 4.8	31.6 ± 2.3	35.5 ± 2.8
FROST (ours)	**49.7** ± 3.3	**71.2** ± 4.6	**64.5** ± 1.9	**63.7** ± 2.6	**41.5** ± 5.1	<u>54.0</u> ± 3.9	**60.3** ± 7.7	**37.3** ± 6.4	**35.3** ± 3.7	**42.5** ± 3.3

$k \in \{1, 3, 5, 10\}$, drawing them from the images containing the class after removing the query image, which is never part of its own support pool. Where images are tiled, the exclusion removes the whole source image, so no other tile of it can serve as a reference either. The support set is sampled anew for each query, and Appendix A.5 gives the pool construction in full. We report the foreground mean intersection over union, pooling the intersection and the union over the episodes of a class before taking the ratio and averaging over classes. Pixels annotated as background count as negatives and only regions a benchmark leaves unlabelled are excluded.

Baselines. We compare against training-free methods, namely the frozen-feature DINOv3 methods INSID3 (Cuttano et al., 2026) and FSSDINO (Zakir & Ho, 2026) together with the SAM-prompting methods PerSAM (Zhang et al., 2024b), Matcher (Liu et al., 2023), and GF-SAM (Zhang et al., 2024a), and against learning-based methods, namely SegIC (Meng et al., 2024), SINE (Liu et al., 2024), DiffewS (Zhu et al., 2024), and the in-context generalist SegGPT (Wang et al., 2023). FROST shares the frozen DINOv3 ViT-L backbone of INSID3 and FSSDINO, so their comparison isolates the pipeline rather than the backbone, and Table 6 lists the backbone and the regime of every method, while Appendix L reports what the same pipeline attains once that backbone is replaced.

Implementation details. We use a frozen DINOv3 ViT-L/16 backbone at 1024×1024 input and no test-time optimization. One global configuration is selected from one-shot performance on LoveDA, Potsdam, and iSAID, then frozen for every benchmark and support size, leaving fourteen benchmarks and the larger support sizes outside the selection, with the positional debiasing rank $r = 250$, the shrinkage intensity $\lambda = 0.95$, the bandwidth σ chosen per episode by the leave-one-out margin, the threshold $\tau = 0$, the bilateral propagation run for $T = 10$ steps with $\beta = 0.70$, $\gamma_f = 0.20$, $\gamma_c = 0.05$, and $d_{\max} = 16$, and the candidate gating using $k_b = 3$ and $\rho_{\max} = 4$. Appendix A.3 gives the selection and the final configuration.

4.2 Results on remote-sensing benchmarks

Overall comparison. Tables 1 and 2 report the foreground mIoU of every method, and FROST attains the best value on sixteen of the seventeen benchmarks, including every benchmark of the building and footprint group. Averaged over benchmarks and support sizes it reaches 53.7 mIoU, ahead of the strongest training-free baseline INSID3 by 6.6 and of the strongest learning-based baseline SegIC by 10.2 mIoU. The one exception is UAVid, where DiffewS leads narrowly and FROST remains the strongest training-free method. The column Mass. Build. abbreviates Massachusetts Buildings, and the spread beside each mean reports how sharply a method responds to more references rather than run-to-run noise. The advantage is largest on building and footprint benchmarks such as xBD and SpaceNet-2, where the foreground is many small and varied structures and the region over which one label is committed has to be small, precisely the regime FROST is built for. Figure 4 compares FROST with FSSDINO and INSID3 on representative episodes, where FROST recovers the separate instances that the coarser decision units of the two baselines merge. A signed-rank test on the seventeen paired differences finds the margin over each of the four strongest baselines improbable under equality at every support size, in Appendix C. The per-support-size results behind these tables appear in Appendix B.

Scaling with the support set. FROST leads every competing training-free method at each of the four support sizes, rising from 48.3 mIoU at one shot to 56.8 at ten, while the strongest competitor at one shot, INSID3, starts from 42.1 mIoU, as Figure 2 reports on the mean over the seventeen benchmarks. The rise is carried by the anchors themselves rather than by the statistics read from them, since holding the whitening transform, the bandwidth and the gate radius at their one-shot estimates while the density ratio still sees every anchor recovers 7.0 of the 8.5 mIoU, as Appendix K reports. Figure 5 gives the per-benchmark view, where FROST holds the outermost contour on all but five of the sixty-eight pairings of a benchmark with a support size, giving up FloodNet at one shot and LoveDA and DeepGlobe at five and at ten shots. Its margin over the strongest competing training-free method grows from 6.2 mIoU at one shot to 7.0 at five shots and stands at 6.3 at ten, and the number of benchmarks on which it records the best result grows from twelve of seventeen at one shot to fifteen at ten. The fine-tuned SegIC does not share this behaviour, peaking at three shots and then declining as further references conflict with its trained priors.

4.3 Generalization beyond overhead imagery

Applied without change to the six benchmarks of Section 4.1, the same configuration trails the strongest competitor by 2.1 mIoU at one shot and leads by 3.4 at ten on the average over the six, far less than the 6.7 mIoU it holds over INSID3 at ten shots overhead. It leads every training-free method at both support sizes on the cross-domain ISIC and SUIM, where no pretraining prior transfers, and reaches PACO-Part and LVIS-92^i by ten shots, while PASCAL-5^i and COCO-20^i stay with the methods carrying natural-image object priors. An object-centric photograph rarely presents the scatter of small instances that makes a fine decision unit decisive, which is the pattern Section 1 anticipates. Appendix D reports every method at every support size and Appendix E the masks.

5 Attributing the accuracy

Table 3 removes each component in turn and reports the one-shot mean, and the ordering it produces is the one our position predicts. The granularity of the decision is by far the largest term, since replacing the label committed for each token by one label per cluster of the query costs 13.5 mIoU, more than every other component combined. The feature refinement follows at 5.1 mIoU, its row removing the positional debiasing and the whitening together, more than the 2.4 and the 0.9 they cost separately, so its two steps are complementary. The horizontal flip adds 1.0 mIoU by doubling the anchors and the bilateral propagation 0.5.

Table 3: One-shot ablation on the seventeen remote-sensing benchmarks, where each row removes one component from the full method.

Variant	Mean mIoU	Difference
FROST (full)	48.3	—
− bilateral propagation	47.8	−0.5
− whitening	47.4	−0.9
− horizontal flip	47.3	−1.0
− positional debiasing	45.9	−2.4
− feature refinement	43.2	−5.1
− token-level decision	34.8	−13.5

The granularity row isolates the decision, since every earlier stage stays in place and only the region carrying one label changes. Its cost concentrates where the target is many small structures threaded through background of similar appearance, at 46.6 mIoU on SpaceNet-2 and 39.2 on xBD, and nearly vanishes where a class covers large connected regions, at 1.2 on FloodNet, as Appendix F shows. A term of this size invites three checks. Because that row also decides where the unit boundaries fall, we set the unit to a fixed block of tokens, which forms no grouping. On the ladder of Appendix G the cost grows monotonically with the side of the block, reaching 11.7 of the 13.5 mIoU at eight tokens, which places the term on the size of the unit rather than on the partition. A coarse unit also forecloses accuracy outright, and labelling every unit from the ground truth shows in Appendix H that this reachable value falls by 11.4 over the same step against the attained 11.7, so the cost is the bound the granularity imposes rather than a failure of the aggregated score. Nor is the effect confined to our pipeline, since moving the decision of INSID3 to the token raises it from 42.1 to 44.4 mIoU, as Appendix I reports.

That labelling reads only the partition, the gate and the ground truth, never the features or the score, so the ceiling it gives is a property of the decision a method commits to rather than of the method and can be measured for any segmenter that commits one. Table 4 accordingly splits the distance five decisions keep from a perfect segmentation into the ceiling each imposes and the shortfall from it. The ceiling orders none of the accuracy. FSSDINO commits a label at every token under no gate, which forecloses almost nothing and leaves it the loosest ceiling of the five at 81.0 mIoU, 17.2 above FROST, yet the lowest accuracy of the five. GF-SAM is bounded no more tightly than 62.5, within 1.3 of FROST, and attains 10.2 less, so the mask proposals of a segmentation model are not what limits it. What separates the five is the shortfall, from 11.0 to 44.1 mIoU. A fine decision unit therefore creates room rather than accuracy, and the class model and the feature refinement beneath it settle how much is taken, so the 13.5 is the cost of coarsening the decision inside FROST rather than a gain over prior work and a token-level decision alone guarantees little. A finer unit can only draw on the discrimination the features beneath it support, so the two stages surrounding the class model compound rather than substitute for one another.

Table 4: One-shot attained accuracy, the ceiling the decision of each method imposes, and the shortfall it keeps from that ceiling, over the seventeen remote-sensing benchmarks. Appendix J gives the decision unit of each method and the per-benchmark values.

Method	Attained	Ceiling	Shortfall
FROST	48.3	63.8	15.5
INSID3	42.1	62.2	20.1
INSID3 to token	44.4	55.4	11.0
FSSDINO	36.9	81.0	44.1
GF-SAM	38.1	62.5	24.4

6 Conclusion

We presented FROST, a training-free few-shot segmenter that refines frozen DINOv3 features on the support, scores every query token by a density ratio, and commits a label per token. Our component study places the dominant share of its accuracy on the granularity of that decision, worth 13.5 mIoU, and the next on the feature refinement, worth 5.1, while a decomposition of five decisions shows that a fine granularity opens room only the features beneath it can fill. Across seventeen remote-sensing benchmarks it surpasses both training-free and learning-based methods at every support size, with a lead at ten shots exceeding that at one. The same configuration transfers beyond overhead imagery, leading every training-free method on the two cross-domain benchmarks at one and ten shots and on the average of all six from three shots on. The room left in this family therefore lies in that granularity and in the geometry the decision runs in.

Limitations. The prediction inherits whatever the frozen encoder does not separate, so classes sharing an appearance stay confused however fine the granularity, as Appendix E displays. The threshold is placed at zero by the equal-prior Bayes rule, which suits an overhead tile whose foreground fills much of the scene but not an image whose target is small or absent, the largest such deficit being on COCO-20^i, and Appendix N measures the absent case. The density ratio compares every query token against every anchor, so inference grows with the support set, as Appendix M measures, and each class needs its own comparison though not its own encoding. Joint prediction over several classes and a prior read from the support remain open.

AI USE STATEMENT

Generative AI tools were used to assist with drafting and revising the manuscript. Specifically, they were used to improve the clarity, structure, and presentation of the text during the writing and editing process. The authors reviewed and verified all AI-assisted content and take full responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

FROST is training-free, so reproducing it calls for no optimization, no learned parameters and no random initialization. Section 3 specifies every stage of the pipeline together with the equations for the shrinkage whitening, the density ratio and the bilateral propagation, while Section 4.1 and Appendix A.3 give the single global configuration behind every number we report, selected once and then frozen for every benchmark and support size with nothing tuned per dataset. The backbone is a publicly released DINOv3 ViT-L/16 checkpoint, used frozen and unmodified, and Appendix A records the hardware and the software environment. Every benchmark is publicly available, and Appendix A.5 gives the episode construction in full, covering the deterministic seeding of the reference draw, the removal of the query from its own support pool and the tiling applied above the input resolution. All methods see the identical episodes under the protocol and the metric of Section 4.1, and the baselines run from their official public implementations at the backbones listed in Table 6. The reference draw is the only stochastic element, and Appendix C reports five draws for FROST and the two strongest baselines alongside a signed-rank test over the paired per-benchmark differences. Code will be released.

REFERENCES

Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021.

Adrian Boguszewski, Dominik Batorski, Natalia Ziemba-Jankowska, Tomasz Dziedzic, and Anna Zambrzycka. Landcover. ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1102–1110. IEEE, 2021.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Yu Chen, Yao Wang, Peng Lu, Yisong Chen, and Guoping Wang. Large-scale structure from motion with semantic constraints of aerial images. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pp. 347–359. Springer, 2018.

Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

Claudia Cuttano, Gabriele Trivigno, Christoph Reich, Daniel Cremers, Carlo Masone, and Stefan Roth. INSID3: Training-free in-context segmentation with DINOv3. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 172–17209. IEEE, 2018.

Jerome H Friedman. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston. xbd: A dataset for assessing building damage from satellite imagery. *arXiv preprint arXiv:1911.09296*, 2019.

Sunghwan Hong, Seokju Cho, Jisu Nam, Stephen Lin, and Seungryong Kim. Cost aggregation with 4d convolutional swin transformer for few-shot segmentation. In *European Conference on Computer Vision*, pp. 108–126. Springer, 2022.

Md Jahidul Islam, Chelsey Edge, Yuyang Xiao, Peigen Luo, Muntaqim Mehtaz, Christopher Morse, Sadman Sakib Enan, and Junaed Sattar. Semantic segmentation of underwater imagery: Dataset and benchmark. In *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 1769–1776. IEEE, 2020.

Shunping Ji, Shiqing Wei, and Meng Lu. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on geoscience and remote sensing*, 57(1):574–586, 2018.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

Yang Liu, Muzhi Zhu, Hengtao Li, Hao Chen, Xinlong Wang, and Chunhua Shen. Matcher: Segment anything with one shot using all-purpose feature matching. *arXiv preprint arXiv:2305.13310*, 2023.

Yang Liu, Chenchen Jing, Hengtao Li, Muzhi Zhu, Hao Chen, Xinlong Wang, and Chunhua Shen. A simple image segmentation framework via in-context examples. *Advances in Neural Information Processing Systems*, 37:25095–25119, 2024.

Ye Lyu, George Vosselman, Gui-Song Xia, Alper Yilmaz, and Michael Ying Yang. Uavid: A semantic segmentation dataset for uav imagery. *ISPRS journal of photogrammetry and remote sensing*, 165: 108–119, 2020.

Lingchen Meng, Shiyi Lan, Hengduo Li, Jose M Alvarez, Zuxuan Wu, and Yu-Gang Jiang. Segic: Unleashing the emergent correspondence for in-context segmentation. In *European Conference on Computer Vision*, pp. 203–220. Springer, 2024.

Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Volodymyr Mnih. *Machine learning for aerial image labeling*. University of Toronto (Canada), 2013.

Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *2019 IEEE/CVF international conference on computer vision (ICCV)*, pp. 622–631. IEEE, 2019.

Maryam Rahnemoonfar, Tashnim Chowdhury, Argho Sarkar, Debvrat Varshney, Masoud Yari, and Robin Roberson Murphy. Floodnet: A high resolution aerial imagery dataset for post flood scene understanding. *IEEE Access*, 9:89644–89654, 2021.

Franz Rottensteiner, Gunho Sohn, Jaewook Jung, Markus Gerke, Caroline Baillard, Sebastien Benitez, and Uwe Breitkopf. The isprs benchmark on urban object classification and 3d building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:293–298, 2012.

Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv preprint arXiv:1709.03410*, 2017.

Zhenfeng Shao, Ke Yang, and Weixun Zhou. Performance evaluation of single-label and multi-label remote sensing image retrieval using a dense labeling dataset. *Remote Sensing*, 10(6):964, 2018.

Zhenfeng Shao, Weixun Zhou, Xueqing Deng, Maoding Zhang, and Qimin Cheng. Multilabel remote sensing image retrieval based on fully convolutional network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:318–328, 2020.

Bernard W Silverman. *Density estimation for statistics and data analysis*. Routledge, 2018.

Oriane Siméoni, Gilles Puy, Huy V Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. *arXiv preprint arXiv:2109.14279*, 2021.

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

Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 44(2):1050–1065, 2020.

Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pp. 839–846. IEEE, 1998.

Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018.

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

Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9196–9205. IEEE, 2019.

Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. Seggpt: Towards segmenting everything in context. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1130–1140. IEEE, 2023.

Junshi Xia, Naoto Yokoya, Bruno Adriano, and Clifford Broni-Bediako. Openearthmap: A benchmark dataset for global high-resolution land cover mapping. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 6243–6253. IEEE, 2023.

Hussni Mohd Zakir and Eric Tatt Wei Ho. Revealing the semantic selection gap in dinov3 through training-free few-shot segmentation. *arXiv preprint arXiv:2602.07550*, 2026.

Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan, Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. isaid: A large-scale dataset for instance segmentation in aerial images. *arXiv preprint arXiv:1905.12886*, 2019.

Anqi Zhang, Guangyu Gao, Jianbo Jiao, Chi H Liu, and Yunchao Wei. Bridge the points: Graph-based few-shot segment anything semantically. *Advances in neural information processing systems*, 37: 33232–33261, 2024a.

Renrui Zhang, Zhengkai Jiang, Ziyu Guo, Shilin Yan, Junting Pan, Hao Dong, Yu Qiao, Gao Peng, and Hongsheng Li. Personalize segment anything model with one shot. In *International Conference on Learning Representations*, volume 2024, pp. 18250–18279, 2024b.

Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16, 2003.

Muzhi Zhu, Yang Liu, Zekai Luo, Chenchen Jing, Hao Chen, Guangkai Xu, Xinlong Wang, and Chunhua Shen. Unleashing the potential of the diffusion model in few-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 37:42672–42695, 2024.

A Implementation Details and Environment Setup

A.1 Hardware and software environment

All experiments run on a single workstation with $4\times$ NVIDIA RTX A6000 GPUs (46 GB each), an AMD Ryzen Threadripper PRO 5975WX (32 cores and 64 threads), and 503 GB of system memory, under Ubuntu 20.04.6 LTS with NVIDIA driver 535.183.01. Inference for a single method fits on one GPU, and the four GPUs are used only to evaluate different benchmarks and methods in parallel. The implementation uses Python 3.12, PyTorch 2.7.1 with CUDA 12.6 and cuDNN 9.0.5, and torchvision 0.22.1, together with NumPy 2.2.6, SciPy 1.16.0, Pillow 11.3.0, `timm` 1.0.26, `einops` 0.8.2, and scikit-learn 1.8.0. All models run in full `float32`, with no test-time augmentation, no mixed precision, and no `torch.compile`.

A.2 Backbone and features

The backbone is a frozen general-domain DINOv3 ViT-L/16 pretrained on LVD-1689M, with patch size 16 and about 304 M parameters. Every image is processed at a single 1024×1024 scale, which yields a 64×64 grid of 1024-dimensional last-layer patch tokens. Each reference is encoded under two views, the identity and its horizontal flip, with the same flip applied to the image and its mask, which doubles the support to $2k$ views and the anchor sets with it. Patch tokens are L^2 normalized before all subsequent operations. We do not ablate this normalization, because it is a premise of the pipeline rather than an additive component and fixes the feature geometry the later stages operate on. Without it the token covariance behind the positional debiasing is dominated by token magnitude rather than content, so the directions it projects out would no longer correspond to position, whereas the kernel compares directions through the cosine similarity and is unaffected by token scale. As Table 6 summarizes, the same frozen backbone with identical settings is shared by INSID3 and FSSDINO, the two training-free baselines on frozen DINOv3 features, so the comparison among the three isolates the pipeline built on top of the backbone, and FROST is among the smallest models compared.

A.3 Method hyperparameters

All global hyperparameter searches use only LoveDA, Potsdam, and iSAID as development data, evaluating candidate settings only in the one-shot regime and on a small subset of the episodes to choose a single global configuration. Since FROST is training-free, the episode protocol of Appendix A.5 defines no separate training split, so we hold out the episodes this selection reads and exclude them from the reported evaluation, to which the three benchmarks contribute their remaining episodes, while the fourteen remaining benchmarks and the three larger support sizes take no part in the selection. After selection, all global values and search grids are frozen and applied unchanged to every reported benchmark and support size, with no tuning per benchmark and no test-time optimization.

The final configuration follows the pipeline of Section 3 stage by stage. The feature refinement applies a positional debiasing that projects out the $r = 250$ leading directions of the token covariance, followed by a within-class Mahalanobis whitening whose scatter is shrunk toward a scaled identity with intensity $\lambda = 0.95$, so that the empirical scatter retains the weight $1 - \lambda = 0.05$, and a small floor $\epsilon = 10^{-4}$ is added to the diagonal before inversion for numerical stability. The whitened tokens keep their scale, with the bilateral term reading them directly and the density ratio comparing their directions through cosine similarity. The class-conditional density is a mixture of von Mises–Fisher kernels over all foreground or background anchors of the support set, the bandwidth σ is selected per episode by the leave-one-out class margin over the fixed grid $\{0.05, 0.1, 0.2, 0.5, 1.0\}$, and the log ratio is thresholded at $\tau = 0$ under equal class priors. Bilateral label propagation on the 64×64 grid, which couples a bilateral affinity (Tomasi & Manduchi, 1998) with the propagation iteration of label spreading (Zhou et al., 2003), runs $T = 10$ steps with mixing weight $\beta = 0.70$, feature and colour temperatures $\gamma_f = 0.20$ and $\gamma_c = 0.05$, and a spatial window radius $d_{\max} = 16$ grid positions. Candidate gating intersects a forward cosine test, which admits a position with positive cosine similarity to the foreground prototype, with a backward vote built from the three nearest reference patches per view, and the resulting region is dilated by a support-adaptive radius $\rho = \lfloor \rho_{\max}\, \pi_{\mathrm{FG}} \rceil$ with $\rho_{\max} = 4$, where $\pi_{\mathrm{FG}} = N_{\mathrm{FG}}/(N_{\mathrm{FG}} + N_{\mathrm{BG}})$ is the foreground fraction of the reference

Table 5: Sensitivity of the one-shot mean foreground mIoU to the shrinkage intensity λ, measured on the development data described in Appendix A.3. The first row applies a diagonal whitening instead of the shrunken full covariance. The selected setting is marked.

Whitening of the within-class scatter	Mean foreground mIoU
Diagonal only	47.34
Full covariance, $\lambda = 0.80$	46.52
Full covariance, $\lambda = 0.90$	47.81
Full covariance, $\lambda = 0.93$	48.07
Full covariance, $\lambda = 0.95$ (selected)	48.19
Full covariance, $\lambda = 0.97$	48.04
Full covariance, $\lambda = 0.98$	47.88

masks. Finally, the continuous propagated field is bilinearly upsampled to the image resolution and thresholded there, and the result is intersected with the candidate region carried to the same resolution.

A.4 Sensitivity to the shrinkage intensity

Table 5 reports how the one-shot mean foreground mIoU responds to the shrinkage intensity λ of Section 3.2, together with the diagonal whitening that ignores the correlation between feature directions and serves as the reference the full covariance has to beat.

The response has a broad plateau rather than a sharp peak, since every intensity from 0.90 to 0.98 falls within 0.4 mIoU of the selected $\lambda = 0.95$, so the configuration does not depend on locating a precise value. Accuracy falls away only once the empirical scatter is given appreciable weight, losing 1.7 mIoU at $\lambda = 0.80$, which is also the only intensity at which the full covariance drops below the diagonal reference. This is the pattern a small effective sample size predicts, since an anchor set carrying as much independent evidence as its raw count of more than eight thousand tokens would support a far more faithful estimate of Σ_W and would therefore favour a much smaller λ.

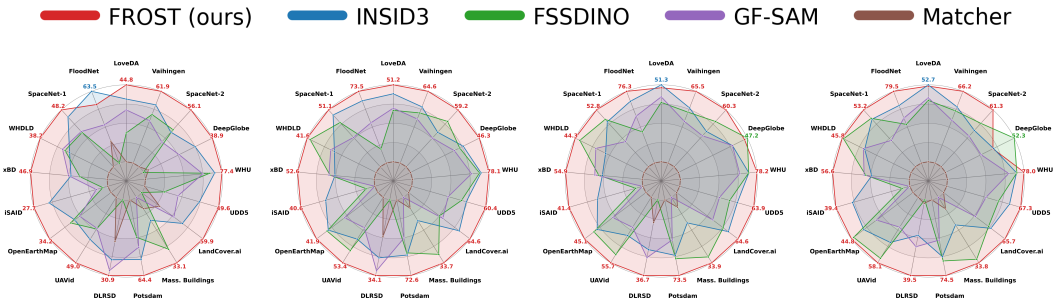
We can replace that prediction by a measurement, because the classical shrinkage rule (Ledoit & Wolf, 2004) states the optimal intensity in closed form as the ratio of the expected squared error of the sample covariance to the squared distance between that covariance and the shrinkage target, and the ratio is a usable intensity only while it stays below one. Evaluated on the anchors of one hundred and forty-nine episodes drawn from LoveDA, Potsdam, Vaihingen, SpaceNet-2 and UDD5, the ratio never falls below 7.9 and averages 16.5, so on every episode the rule asks for more shrinkage than complete shrinkage can supply. Eight thousand independent draws in a thousand dimensions would return a ratio far below one instead, which places the evidence the anchors carry at a small fraction of their count and settles the reading of Section 3.2. The selected $\lambda = 0.95$ therefore stops just short of the intensity the rule would impose, and the weight $1 - \lambda$ it leaves on the empirical scatter is worth 0.31 mIoU against the $\lambda = 0.98$ that lies nearer to the rule.

A.5 Episode construction and the support pool

Every reported number comes from episodes built by the same procedure, which we state here in full because the amount of overlap permitted between a query and its references governs how a few-shot result should be read. For each benchmark we form a single pool from all of its publicly available evaluation data and order it deterministically, then enumerate one episode for every pair of a query image and a class present in that image. Images larger than the 1024×1024 input are processed as tiles taken at the native resolution with a stride equal to the tile size, so that a large scene contributes one episode per tile and per class rather than being resized, and a tile that is almost entirely nodata is discarded. The support pool for a class consists of the images of the same benchmark that contain it, and the seed that selects the k references is derived from the benchmark name together with the position and class of the query, so the assignment is fixed across methods and reproducible across processes. All ten methods we compare consume exactly the same episodes with exactly the same references. The query is excluded from its own support pool, and the exclusion is applied to the

Table 6: Backbone and parameter count of each method, where M denotes millions of parameters from the released checkpoint. The upper block lists the learning-based methods and the lower block lists the training-free methods, each in the order of the main comparison tables. FROST, INSID3, and FSSDINO share the same frozen DINOv3 ViT-L backbone, so that the comparison among these three isolates the pipeline built on top of the backbone rather than the backbone itself, and FROST is among the smallest models. Every method runs forward only at inference with no test-time optimization. For the learning-based methods the count includes the trained components on top of the backbone, and for Matcher and GF-SAM it is the sum of a DINOv2 ViT-L and a SAM ViT-H encoder.

Method	Backbone	Parameters
Learning-based		
SegGPT	ViT-L	371M
SegIC	DINOv2 ViT-L	310M
DiffewS	Stable Diffusion 2.1	950M
SINE	DINOv2 ViT-L	323M
Training-free		
PerSAM	SAM ViT-H	641M
Matcher	DINOv2 ViT-L and SAM ViT-H	946M
GF-SAM	DINOv2 ViT-L and SAM ViT-H	946M
FSSDINO	DINOv3 ViT-L	304M
INSID3	DINOv3 ViT-L	304M
FROST (ours)	DINOv3 ViT-L	304M



(a) 1-shot (b) 3-shot (c) 5-shot (d) 10-shot

Figure 5: Per-benchmark foreground mIoU on the remote-sensing benchmarks for FROST and the four strongest training-free methods at each support size $k \in \{1, 3, 5, 10\}$, with the shared legend at the top.

source image rather than to the tile, so for a tiled benchmark no other tile of the query image can be drawn as a reference for it either. For LoveDA, Potsdam, and iSAID the episodes that selected the configuration of Appendix A.3 are removed from the pool, so no reported number is computed on them.

B PER-SUPPORT-SIZE RESULTS

Figure 5 together with Table 7 and Table 8 give the foreground mIoU of every method at each support size, the values that the main tables average, with the best value in each column in bold and the second best underlined within each support size. The overall accuracy of FROST increases monotonically with the number of reference masks, from 48.3 mIoU at one shot to 54.1, 55.6, and 56.8 mIoU at three, five, and ten shots, whereas the fine-tuned SegIC peaks at three shots and then declines. FROST records the best result of any method on twelve of the seventeen benchmarks at one shot and on fifteen at ten shots, so it is ahead on more benchmarks at ten shots than at one.

Table 7: Foreground mIoU on the remote-sensing building and footprint benchmarks together with DeepGlobe and FloodNet, reported at each of the four support sizes $k \in \{1, 3, 5, 10\}$, with the best value in each column in bold and the second best underlined within each support size.

Method	WHU Building	SpaceNet-1	SpaceNet-2	Mass. Build.	xBD	DeepGlobe	FloodNet
1-shot							
SegGPT	60.2	38.1	41.7	11.5	28.1	23.3	44.4
SegIC	65.3	**49.1**	45.9	26.0	25.6	**39.0**	**63.7**
DiffewS	61.2	46.1	30.7	17.7	12.6	35.9	58.5
SINE	14.1	20.0	8.8	16.7	4.2	25.3	32.5
PerSAM	2.7	4.9	2.6	1.3	1.5	22.6	28.5
Matcher	26.3	29.8	16.9	15.5	9.2	33.7	51.0
GF-SAM	64.6	41.2	34.0	11.8	28.2	36.2	55.1
FSSDINO	69.2	40.9	42.5	28.7	22.6	33.3	45.5
INSID3	72.6	45.9	40.8	19.3	27.0	37.5	63.5
FROST (ours)	**77.4**	48.2	**56.1**	**33.1**	**46.9**	38.9	60.2
3-shot							
SegGPT	70.6	42.1	49.4	11.2	36.8	19.7	48.6
SegIC	60.0	**51.9**	49.8	23.7	33.3	40.4	68.5
DiffewS	64.9	45.9	36.4	18.5	19.8	41.0	68.8
SINE	23.5	28.5	16.3	18.2	10.8	31.1	41.0
PerSAM	2.4	1.4	3.1	0.5	2.3	23.2	34.4
Matcher	31.9	33.1	20.0	15.4	11.8	38.6	57.6
GF-SAM	67.6	42.4	39.8	11.8	35.9	43.5	66.5
FSSDINO	72.4	47.1	51.3	30.9	35.3	44.1	60.7
INSID3	73.3	49.5	43.8	19.6	32.7	44.5	71.4
FROST (ours)	**78.1**	51.1	**59.2**	**33.7**	**52.6**	**46.3**	**73.5**
5-shot							
SegGPT	72.1	42.5	50.7	11.3	38.1	19.3	44.6
SegIC	54.7	48.8	48.5	21.2	34.0	40.7	66.4
DiffewS	63.9	45.4	38.5	18.6	19.6	40.8	71.3
SINE	25.3	33.5	18.5	18.4	14.7	32.8	41.5
PerSAM	2.7	1.4	3.2	0.4	2.7	23.1	36.0
Matcher	32.9	33.6	19.9	14.8	12.2	39.8	57.5
GF-SAM	68.0	42.1	41.9	11.4	38.6	45.6	69.8
FSSDINO	73.3	49.6	52.1	31.9	36.4	**47.2**	65.7
INSID3	73.4	48.8	45.0	20.2	34.4	45.6	73.7
FROST (ours)	**78.2**	**52.8**	**60.3**	**33.9**	**54.9**	46.9	**76.3**
10-shot							
SegGPT	72.6	42.7	50.7	11.0	39.1	20.0	45.8
SegIC	45.9	42.4	43.2	18.6	31.8	39.9	61.9
DiffewS	62.2	44.7	38.6	18.4	17.4	41.2	70.4
SINE	25.8	34.9	19.7	18.6	15.4	33.5	42.0
PerSAM	2.5	1.9	2.9	0.2	2.6	22.6	37.2
Matcher	31.7	33.6	19.7	15.3	13.2	40.7	56.7
GF-SAM	68.8	42.7	42.1	11.3	38.7	46.7	71.0
FSSDINO	73.7	50.3	51.5	32.6	37.8	**52.3**	69.0
INSID3	73.5	50.0	46.3	20.8	34.4	48.8	75.0
FROST (ours)	**78.0**	**53.2**	**61.3**	**33.8**	**56.6**	48.7	**79.5**

C CONSISTENCY OF THE MARGIN

The tables of Section 4.2 report means over benchmarks, and a large advantage on a few benchmarks would produce such a mean as readily as a smaller advantage on all of them, so we separate the two. For each baseline and each support size we take the seventeen paired differences in foreground mIoU, one per benchmark, and apply a two-sided Wilcoxon signed-rank test, which uses only the signs and the ranks of those differences and not their magnitudes, and we write p for the probability it assigns to a rank sum at least as extreme as the observed one were the two methods equal.

Table 9 reports the outcome. Against each of the four strongest baselines and at each of the four support sizes the test rejects equality at $p < 0.005$, and against the two that share the frozen backbone of FROST at $p < 10^{-3}$, so the margin rests on a broad advantage across the suite rather than on a handful of benchmarks. The single comparison that approaches the threshold is the one-shot

Table 8: Foreground mIoU on the remote-sensing land-cover and drone benchmarks, reported at each of the four support sizes $k \in \{1, 3, 5, 10\}$, with the best value in each column in bold and the second best underlined within each support size.

Method	LoveDA	Potsdam	Vaihingen	LandCover.ai	OpenEarthMap	UAVid	UDD5	iSAID	DLRSD	WHDLD
1-shot										
SegGPT	40.3	38.5	46.8	45.0	21.1	33.4	29.3	21.9	20.8	27.5
SegIC	34.1	59.6	52.9	50.6	28.3	51.7	49.4	20.7	**31.4**	33.7
DiffewS	27.4	44.6	45.6	45.7	22.4	**52.4**	37.8	10.7	29.9	28.2
SINE	13.5	28.6	20.3	32.2	17.1	20.0	22.5	3.2	15.6	22.2
PerSAM	13.9	19.3	9.5	34.7	6.8	14.2	15.7	5.6	20.5	18.7
Matcher	26.4	39.8	30.0	42.6	19.1	29.6	31.6	14.2	25.5	24.6
GF-SAM	38.8	55.2	48.8	48.3	26.8	38.8	39.7	16.3	30.1	33.8
FSSDINO	33.3	52.0	51.6	35.4	29.0	39.0	36.4	15.1	18.9	33.3
INSID3	41.4	59.2	55.9	51.7	28.0	42.1	44.9	24.9	28.4	32.4
FROST (ours)	**44.8**	**64.4**	**61.9**	**59.9**	**34.2**	49.0	**49.6**	**27.7**	30.9	**38.2**
3-shot										
SegGPT	45.7	39.5	51.2	47.0	21.6	31.2	27.3	22.7	20.5	25.2
SegIC	35.1	62.2	53.2	58.8	29.7	54.1	54.9	24.6	33.0	34.4
DiffewS	30.9	55.5	48.7	56.5	24.5	**57.4**	46.6	12.8	32.1	30.9
SINE	14.9	35.3	23.6	35.3	21.2	25.2	28.4	5.4	16.6	24.4
PerSAM	14.1	21.1	9.5	37.1	7.8	16.9	18.2	8.4	20.2	18.7
Matcher	27.1	44.5	32.7	42.9	22.0	36.2	33.8	20.1	28.4	24.9
GF-SAM	43.9	58.7	52.3	55.0	32.8	44.2	46.9	20.1	33.4	35.9
FSSDINO	43.2	64.2	55.0	53.8	38.5	50.2	51.7	22.5	23.0	40.9
INSID3	48.4	65.0	57.7	60.8	38.1	47.2	53.4	33.7	31.5	35.0
FROST (ours)	**51.2**	**72.6**	**64.6**	**64.6**	**41.9**	53.4	**60.4**	**40.6**	**34.1**	**41.6**
5-shot										
SegGPT	47.0	39.4	51.8	47.4	20.3	30.3	26.6	23.9	20.1	23.7
SegIC	34.1	62.3	50.7	58.5	29.4	53.6	54.9	25.1	32.9	34.8
DiffewS	30.7	55.1	47.8	49.4	23.5	**57.7**	47.6	13.1	31.9	32.0
SINE	15.6	35.9	24.2	35.4	21.8	26.9	29.0	6.2	16.7	25.0
PerSAM	14.8	21.7	9.2	36.9	5.5	17.4	18.3	8.6	20.8	19.7
Matcher	27.6	43.8	33.5	43.0	20.8	38.9	35.0	21.8	28.7	24.6
GF-SAM	47.6	59.5	52.1	55.7	33.9	46.6	50.5	21.9	34.0	38.4
FSSDINO	45.9	66.9	56.2	59.4	41.4	53.6	54.5	23.7	25.3	43.0
INSID3	**51.3**	65.6	58.1	61.2	40.1	48.7	57.5	33.8	32.8	35.6
FROST (ours)	50.4	**73.5**	**65.5**	**64.6**	**45.1**	55.7	**63.9**	**41.4**	**36.7**	**44.3**
10-shot										
SegGPT	48.3	39.5	52.9	48.5	19.5	30.6	26.3	20.2	19.1	22.6
SegIC	34.3	60.6	47.2	57.8	29.3	52.8	53.6	23.5	32.3	33.5
DiffewS	29.1	53.9	45.8	45.7	22.7	56.9	46.6	14.0	31.3	31.1
SINE	17.4	36.2	24.3	36.4	22.4	28.3	28.8	7.1	16.0	25.2
PerSAM	14.3	23.6	8.5	37.0	7.0	17.3	19.3	7.0	21.2	19.6
Matcher	28.4	44.4	33.9	45.2	21.2	38.8	33.5	23.0	28.7	25.3
GF-SAM	48.6	60.4	53.2	55.8	35.2	47.6	51.4	23.1	35.2	39.5
FSSDINO	47.8	68.7	57.4	60.2	44.0	56.8	57.2	24.7	28.2	45.4
INSID3	**52.7**	67.0	59.2	61.0	40.1	52.1	62.2	35.7	33.5	39.1
FROST (ours)	52.3	**74.5**	**66.2**	**65.7**	**44.8**	**58.1**	**67.3**	**39.4**	**39.5**	**45.8**

Table 9: Consistency of the margin of FROST over the four strongest baselines across the seventeen remote-sensing benchmarks. Each cell gives the number of benchmarks on which FROST is ahead and the two-sided p value of a Wilcoxon signed-rank test on the seventeen paired differences, that is the probability of a rank sum at least this extreme were the two methods equal, computed separately at each support size k.

Baseline	$k = 1$	$k = 3$	$k = 5$	$k = 10$
INSID3	16 of 17, 1.5×10^{-4}	17 of 17, 1.5×10^{-5}	16 of 17, 3.1×10^{-5}	15 of 17, 7.6×10^{-5}
FSSDINO	17 of 17, 1.5×10^{-5}	17 of 17, 1.5×10^{-5}	16 of 17, 3.1×10^{-5}	16 of 17, 2.1×10^{-4}
GF-SAM	17 of 17, 1.5×10^{-5}	17 of 17, 1.5×10^{-5}	17 of 17, 1.5×10^{-5}	17 of 17, 1.5×10^{-5}
SegIC	12 of 17, 4.6×10^{-3}	15 of 17, 7.6×10^{-5}	17 of 17, 1.5×10^{-5}	17 of 17, 1.5×10^{-5}

comparison with the meta-trained SegIC, which FROST leads on twelve of the seventeen benchmarks at that support size and on all seventeen from five shots onward.

That test holds the reference draw fixed, so it says nothing about how much of the margin the draw itself decides. Every episode of this paper selects its references from the images that contain the class under one seed, and a different seed hands every method a different support. We therefore re-run the protocol at five seeds on five benchmarks chosen to span the range of foreground scales, the building

Table 10: Sensitivity to the reference draw over five seeds on five of the seventeen remote-sensing benchmarks. The left block gives the standard deviation of each method's own foreground mIoU across the five draws, and the right block gives the mean and the standard deviation of FROST's lead, formed within each seed so that the three methods always compare on the same references, with the number of seeds at which that lead is positive in parentheses. The seed of every other table in this paper is one of the five, and its cells reproduce the values reported there.

		Standard deviation of the value			Lead of FROST	
Benchmark	Shots	FROST	INSID3	SegIC	Over INSID3	Over SegIC
SpaceNet-2	1	1.79	0.97	1.83	16.3 ± 1.2 (5/5)	10.5 ± 1.4 (5/5)
SpaceNet-2	10	0.36	0.19	0.44	15.3 ± 0.2 (5/5)	18.2 ± 0.2 (5/5)
xBD	1	3.00	2.64	3.18	18.4 ± 2.2 (5/5)	18.5 ± 3.7 (5/5)
xBD	10	0.52	0.76	0.89	20.5 ± 1.0 (5/5)	24.5 ± 0.6 (5/5)
LoveDA	1	0.48	0.46	0.61	3.6 ± 0.2 (5/5)	11.5 ± 0.9 (5/5)
LoveDA	10	1.06	1.69	1.01	0.6 ± 1.7 (3/5)	17.9 ± 1.1 (5/5)
iSAID	1	1.02	1.17	0.73	3.9 ± 1.0 (5/5)	7.2 ± 1.6 (5/5)
iSAID	10	2.32	2.27	1.08	5.3 ± 1.6 (5/5)	13.5 ± 2.6 (5/5)
FloodNet	1	1.17	1.07	2.49	-1.8 ± 1.2 (0/5)	-3.4 ± 2.5 (1/5)
FloodNet	10	1.32	1.42	2.02	3.8 ± 1.2 (5/5)	20.1 ± 3.2 (5/5)
Mean		**1.30**	**1.26**	**1.43**	$\mathbf{8.6 \pm 1.2}$	$\mathbf{13.8 \pm 1.8}$

footprints of SpaceNet-2, the disaster imagery of xBD, the land cover of LoveDA, the instance-dense iSAID, and the flood scenes of FloodNet, at one and at ten shots. FROST alone would not answer the question, because it would measure how far an absolute value moves while every claim in this paper is a comparison, so INSID3 and SegIC are re-run at each seed as well and each lead is formed inside a seed, where the three methods see the same references. One of the five seeds is the seed of every other table here, and its thirty cells reproduce the values reported there to within 0.001 mIoU, which is what licenses reading the other four.

Table 10 reports the outcome. A single value moves by 1.33 mIoU of standard deviation on average and by as much as 3.18, and the lead moves by a comparable amount, 1.2 against INSID3 and 1.8 against SegIC, so pairing the methods inside a seed does not cancel the variation, it only keeps the comparison honest. What survives is the ordering. FROST leads at all five seeds in seventeen of the twenty combinations of benchmark, support size and baseline, and the three exceptions are the places the main tables already report as close or adverse. FloodNet at one shot is a loss rather than a coin flip, since FROST trails INSID3 at five seeds of five and SegIC at four of five, and LoveDA at ten shots against INSID3 is genuinely marginal at 0.6 ± 1.7 and three wins of five, which is the same cell that Section 4.2 names among the pairings a competing method takes. So the reference draw does not decide the ordering except where the paper already declines to claim one.

D Full results beyond overhead imagery

Table 11 gives the foreground mIoU of every method on the six benchmarks at each support size, the values Section 4.3 summarizes, with the best value in each column in bold and the second best underlined within each support size. The accuracy of FROST rises with the number of reference masks as it does on the remote-sensing benchmarks, and among the training-free methods FROST moves from below the strongest competitor at one shot to the best overall accuracy at ten shots, while on the canonical object benchmarks PASCAL-5^i and COCO-20^i the lead stays with methods that carry natural-image object priors at every support size.

E Qualitative results

This appendix complements the quantitative comparison with the masks FROST produces. Figure 6 collects sixteen one-shot examples from the remote-sensing benchmarks, spanning the building and footprint, land-cover and urban-scene, and flood groups, so the reader can judge segmentation quality across the range of targets the benchmarks cover. Every example is a one-shot prediction from a

frozen DINOv3 backbone with no training, overlaid on the query image in the same colour as the reference mask. The reference column shows the one annotated image and mask pair that defines the target class, the ground-truth column shows the mask of the query that the prediction is compared against, and the column labelled FROST shows its prediction.

The examples illustrate the behaviour that the quantitative results summarize. On targets that recur as many small and visually distinct instances, such as the buildings and vehicles of the building and urban-scene benchmarks, FROST recovers the separate instances rather than merging them into a single blob, which is the regime for which a small decision unit matters most. On the large and continuous regions of the land-cover and flood benchmarks the predicted boundaries follow the query edges closely, a consequence of the bilateral propagation and the upsampling before thresholding. The failure cases that remain are concentrated on scenes in which several classes share a nearly identical appearance in the frozen features, so that the boundary between them is not expressed in the geometry FROST reads from.

Figure 7 repeats this inspection on the benchmarks beyond overhead imagery, with sixteen one-shot examples drawn from the natural-image and cross-domain benchmarks under the same protocol, the same frozen backbone, and the same overlay convention. The predictions confirm what the per-benchmark numbers report, since on the object-centric and part-centric targets FROST recovers the instance with boundaries that track the object, and on the cross-domain images its masks stay accurate even though the appearance is far from anything seen during the pretraining of the backbone. As in the overhead case, the residual gap on these benchmarks comes from harder scenes with several co-occurring instances, rather than from gross failures.



Figure 6: Qualitative results of FROST on the remote-sensing benchmarks. Each triplet shows, from left to right, the reference image with its mask, the query image with its mask, and the prediction of FROST, with two triplets per row separated by a gap. All examples are one-shot predictions obtained from a frozen DINOv3 backbone without any training.

Table 11: Foreground mIoU on the six benchmarks beyond overhead imagery at each of the four support sizes $k \in \{1, 3, 5, 10\}$, with the best value in each column in bold and the second best underlined within each support size. The learning-based methods SegGPT, SegIC, DiffewS, and SINE are trained on segmentation data that overlaps the natural-image benchmarks, so they are shown in grey and excluded from the best and second-best marking, which is computed over the training-free methods alone.

Method	PASCAL-5^i	COCO-20^i	LVIS-92^i	PACO-Part	ISIC	SUIM
1-shot						
SegGPT	81.5	53.9	18.3	17.7	32.4	38.7
SegIC	85.6	71.1	41.4	27.3	27.1	59.8
DiffewS	88.2	50.7	31.5	22.9	27.4	51.9
SINE	33.5	18.9	6.8	14.9	28.5	28.1
PerSAM	45.6	21.8	13.3	20.8	22.7	29.6
Matcher	68.2	54.6	36.5	34.3	33.4	46.7
GF-SAM	**71.9**	**59.2**	37.9	38.1	48.9	55.0
FSSDINO	65.5	39.9	22.2	34.3	52.3	51.3
INSID3	69.9	56.5	**40.9**	**39.6**	50.5	56.2
FROST (ours)	70.1	48.4	32.8	39.1	**52.6**	**58.0**
3-shot						
SegGPT	84.7	55.2	19.1	15.0	43.5	36.1
SegIC	87.0	70.3	41.4	29.7	24.6	61.6
DiffewS	89.0	57.8	33.1	24.8	25.8	62.5
SINE	36.0	23.5	7.9	16.1	31.1	36.6
PerSAM	51.0	30.0	16.1	23.9	23.6	36.6
Matcher	72.7	59.3	41.6	35.7	33.4	49.5
GF-SAM	**80.8**	**65.5**	44.6	41.7	53.1	**66.8**
FSSDINO	75.9	52.5	29.4	41.9	60.3	63.0
INSID3	77.9	64.3	**45.2**	44.4	62.4	**66.8**
FROST (ours)	80.5	59.5	43.5	**47.9**	**66.0**	66.2
5-shot						
SegGPT	85.6	55.6	19.7	14.6	43.7	40.8
SegIC	86.2	68.0	39.7	30.5	26.2	60.7
DiffewS	88.6	58.1	34.7	25.5	28.8	62.6
SINE	36.7	22.9	8.8	17.3	29.3	37.1
PerSAM	55.0	31.3	17.3	25.0	26.3	34.0
Matcher	73.9	57.1	41.2	33.3	34.7	53.2
GF-SAM	**83.1**	**65.8**	46.5	42.2	55.3	65.1
FSSDINO	78.9	53.8	32.4	45.6	62.8	62.7
INSID3	81.0	63.0	46.0	47.2	63.8	64.1
FROST (ours)	82.5	60.5	**47.1**	**52.1**	**66.3**	**68.8**
10-shot						
SegGPT	86.6	59.1	20.2	16.4	46.1	36.6
SegIC	85.9	66.6	37.7	31.3	26.2	56.4
DiffewS	86.8	60.4	33.4	27.0	29.6	57.8
SINE	37.4	21.6	10.6	18.5	32.8	37.3
PerSAM	55.0	29.9	16.4	27.1	26.9	35.9
Matcher	75.4	61.7	39.4	34.3	36.2	48.8
GF-SAM	**83.9**	**68.1**	48.1	45.4	60.7	63.2
FSSDINO	80.2	56.7	35.3	50.6	65.4	63.2
INSID3	81.5	64.0	46.8	50.3	67.0	67.6
FROST (ours)	83.7	64.3	**51.9**	**57.0**	**70.4**	**70.1**



Figure 7: Qualitative results of FROST on the benchmarks beyond overhead imagery. Each triplet shows, from left to right, the reference image with its mask, the query image with its mask, and the prediction of FROST, with two triplets per row separated by a gap. All examples are one-shot predictions obtained from a frozen DINOv3 backbone without any training.

F THE COST OF A COARSE DECISION

Section 5 reports a single mean for the variant that commits one label per cluster of the query rather than one per token, and because that variant carries the largest term of the ablation we give its effect on every benchmark in Table 12. The variant alters nothing before the decision, since the feature refinement, the whitening, the density ratio, the bilateral propagation and the candidate gate all run exactly as in the full method. The smoothed score is then averaged over the tokens of each cluster of the query and compared against the same threshold, so a cluster becomes foreground as a whole or not at all. The clusters come from the agglomerative grouping of INSID3 at the merging threshold published with that method, applied to the refined features of our own pipeline, and tokens the grouping leaves unassigned are labelled background as they are there. It is therefore not the partition INSID3 itself forms, so this row bounds what a coarse decision costs inside FROST rather than what it costs that method, which Appendix I measures directly.

The effect is far from uniform, and its shape is the reason a token-level decision matters in this domain. It is largest on the building and footprint benchmarks, where a class is many small structures threaded through background that shares their built appearance, so a region large enough to be grouped reliably tends to hold both and one label for it must be wrong on one of them. It is smallest where a class covers large connected regions, on FloodNet, UDD5 and DeepGlobe, which a coarse unit can fill without straddling a boundary. Object size alone does not order the table, since iSAID has the smallest targets of the suite yet an effect of 5.2 mIoU, well under the mean of 13.5, which we read as its compact objects standing against uniform tarmac, water or grass clearly enough that the grouping already isolates them. What the ordering follows is how far foreground and background resemble each other locally, and that resemblance is the ordinary condition of overhead imagery, where a target and the surfaces around it are built of the same materials and seen from the same distance under the same illumination.

Table 12: Effect of replacing the token-level decision by one label per cluster of the query, measured at one shot on each of the seventeen remote-sensing benchmarks and ordered by the size of the effect, where Mass. Build. abbreviates Massachusetts Buildings. Every stage before the decision is identical between the two columns, so the difference is attributable to the region over which one label is committed.

Benchmark	FROST	One label per cluster	Difference
SpaceNet-2	56.1	9.5	-46.6
xBD	46.9	7.7	-39.2
WHU Building	77.4	52.5	-24.9
Vaihingen	61.9	43.2	-18.7
Mass. Build.	33.1	17.6	-15.5
WHDLD	38.2	24.6	-13.6
SpaceNet-1	48.2	35.5	-12.7
OpenEarthMap	34.2	24.2	-10.0
LoveDA	44.8	35.1	-9.7
DLRSD	30.9	21.3	-9.6
LandCover.ai	59.9	52.7	-7.2
Potsdam	64.4	58.2	-6.2
iSAID	27.7	22.5	-5.2
UAVid	49.0	44.1	-4.9
DeepGlobe	38.9	35.8	-3.1
UDD5	49.6	48.3	-1.3
FloodNet	60.2	59.0	-1.2
Mean	**48.3**	**34.8**	$\mathbf{-13.5}$

G SEPARATING THE DECISION UNIT FROM THE PARTITION

The variant of Table 12 replaces a token by a cluster, which enlarges the region carrying one label but also decides where the boundaries of that region fall, so on its own it cannot say which of the two the cost belongs to. We separate them by repeating the measurement with the decision unit set to a fixed block of $S \times S$ tokens, where S is the side of the block in tokens, taking the mean of the propagated score over the block against the same threshold. No grouping is formed, so no partition can be constructed poorly, no token is left unassigned, and no continuous score is paired with a partition drawn up for a different one. The side of the block is then the only quantity that changes, and $S = 1$ recovers the full method.

Table 13 reports the ladder and Figure 8 shows it on one episode. The cost grows monotonically with the side of the block and reaches 11.7 mIoU at eight tokens on a side, against the 13.5 of the clustered variant, so most of that variant sits on the size of the decision unit and not on the way the query is partitioned. Only the remaining 1.8 is attributable to the grouping, which settles the reading of Section 5 in favour of the decision unit itself.

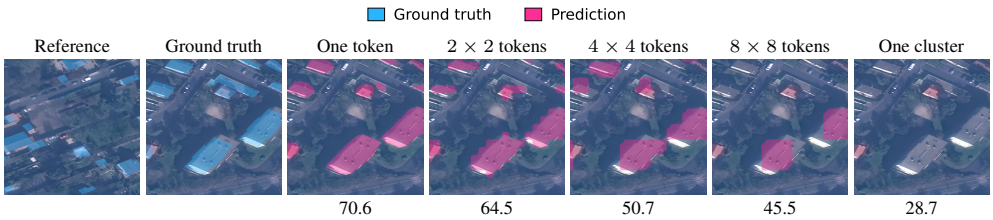


Figure 8: The ladder of Table 13 on one SpaceNet-2 episode, where the reference mask carries the same colour as the ground truth. Every stage before the decision is identical across the five predictions and only the region carrying one label changes, from a single token through blocks of tokens to a cluster of the query. Each value is the foreground IoU of that prediction over the whole episode, of which the panels show a detail, while the table gives the same comparison over the seventeen benchmarks.

H THE CEILING OF A DECISION GRANULARITY

A coarse decision can lose accuracy in two ways the ladder does not separate. The unit may be unable to express the target at all, or the score averaged over the unit may choose the wrong label for a unit that could have expressed it. We separate them by labelling every unit from the ground truth. Since the metric pools intersection and union over the episodes of a class, the labelling that maximizes it keeps every unit whose foreground fraction exceeds some value and drops the rest, so we accumulate the foreground and valid pixel counts of the units in a histogram over that fraction and afterwards take the cut that maximizes the pooled ratio, which is the most any score could extract at that granularity. The candidate gate of Section 3.4 is applied after the unit decision and is itself token-level, so a coarse variant is still cut at token resolution and the ceiling has to carry the same gate, without which it would not bound the variant.

Table 14 reports both quantities. Over the step from one token to a block of eight the reachable value falls by 11.4 mIoU while the attained value falls by 11.7, and the attained value stays 15.5, 15.8 and 14.7 below the reachable one at one token, at eight tokens and at a cluster. What a coarse decision costs is therefore the bound it imposes rather than a failure of the score aggregated over the unit, and the distance a fixed class model keeps from that bound is close to constant in the granularity. This is a statement about the mean and not a per-benchmark law, since xBD loses 35.6 mIoU against a bound that falls by only 11.0.

I MOVING THE DECISION OF A COMPETING METHOD

If the granularity of the decision governs accuracy in this setting then the effect should not be confined to our pipeline, so we test it on INSID3, whose cluster path assigns one label to each group of the

Table 13: Effect of the decision granularity at one shot on each of the seventeen remote-sensing benchmarks, where Mass. Build. abbreviates Massachusetts Buildings and the best entry of a row is set in bold. The last line gives the difference of each mean from the first column. Each column commits a single foreground or background label over the stated region of the query and leaves every earlier stage unchanged, and the four right-hand columns take the mean of the propagated score over the region against the same threshold. Across the three block columns the side of the block is the only quantity that changes, whereas the last column groups the query by the agglomerative clustering of INSID3 instead of by a fixed block and so alters the partition as well as the size of the unit.

Benchmark	One token	2×2 tokens	4×4 tokens	8×8 tokens	One cluster
WHU Building	**77.4**	68.6	57.6	44.8	52.5
SpaceNet-1	**48.2**	46.9	44.0	39.8	35.5
SpaceNet-2	**56.1**	51.7	43.0	28.2	9.5
Mass. Build.	**33.1**	27.9	24.5	21.9	17.6
xBD	**46.9**	37.9	23.4	11.3	7.7
DeepGlobe	**38.9**	38.8	38.6	38.0	35.8
FloodNet	**60.2**	59.7	58.5	56.1	59.0
LoveDA	**44.8**	44.0	41.9	38.6	35.1
Potsdam	**64.4**	62.6	58.7	51.3	58.2
Vaihingen	**61.9**	58.6	51.6	43.1	43.2
LandCover.ai	**59.9**	59.2	57.1	52.4	52.7
OpenEarthMap	**34.2**	32.3	29.9	27.4	24.2
UAVid	**49.0**	47.5	44.4	39.8	44.1
UDD5	**49.6**	49.3	48.2	45.4	48.3
iSAID	**27.7**	26.8	25.3	22.6	22.5
DLRSD	**30.9**	30.5	29.5	27.3	21.3
WHDLD	**38.2**	37.5	36.1	34.1	24.6
Mean	48.3	45.9	41.9	36.6	34.8
Difference	—	-2.4	-6.4	-11.7	-13.5

Table 14: Attained and reachable foreground mIoU at one shot for three decision granularities on each of the seventeen remote-sensing benchmarks, where Mass. Build. abbreviates Massachusetts Buildings. A reachable column labels every unit from the ground truth and reports the best value that labelling attains, so it is the most any score could extract at that granularity, and it carries the same candidate gate the attained variants carry.

Benchmark	One token	8×8	Cluster	One token	8×8	Cluster
SpaceNet-2	56.1	28.2	9.5	68.9	46.9	42.6
xBD	46.9	11.3	7.7	51.7	40.7	38.6
WHU Building	77.4	44.8	52.5	81.1	58.1	63.1
Vaihingen	61.9	43.1	43.2	76.4	57.7	55.5
Mass. Build.	33.1	21.9	17.6	39.7	29.0	27.3
WHDLD	38.2	34.1	24.6	60.3	48.0	40.7
SpaceNet-1	48.2	39.8	35.5	71.5	48.7	44.4
OpenEarthMap	34.2	27.4	24.2	51.7	40.9	37.8
LoveDA	44.8	38.6	35.1	60.2	54.1	49.2
DLRSD	30.9	27.3	21.3	41.5	37.6	35.1
LandCover.ai	59.9	52.4	52.7	72.6	65.7	63.2
Potsdam	64.4	51.3	58.2	79.7	67.5	66.8
iSAID	27.7	22.6	22.5	47.3	42.3	38.3
UAVid	49.0	39.8	44.1	63.7	58.3	56.2
DeepGlobe	38.9	38.0	35.8	63.9	58.9	51.5
UDD5	49.6	45.4	48.3	72.6	62.6	59.2
FloodNet	60.2	56.1	59.0	81.2	74.3	72.8
Mean	**48.3**	**36.6**	**34.8**	**63.8**	**52.4**	**49.5**

query. We leave its class model, its candidate gate and the merging threshold published with it untouched and change only where each factor is read, taking the cross-image similarity and the intra-image similarity to the seed prototype at the token instead of averaging each over a cluster.

Table 15: Foreground mIoU at one shot of the cluster path of INSID3 and of the same method with its decision committed one token at a time, on each of the seventeen remote-sensing benchmarks, where Mass. Build. abbreviates Massachusetts Buildings. The class model, the candidate gate and the merging threshold published with the method are untouched, and only the region carrying one label changes.

Benchmark	One cluster	One token	Difference
SpaceNet-2	40.8	48.3	$+7.5$
xBD	27.0	39.4	$+12.4$
WHU Building	72.6	74.0	$+1.4$
Vaihingen	55.9	57.4	$+1.5$
Mass. Build.	19.3	26.8	$+7.5$
WHDLD	32.4	33.3	$+0.9$
SpaceNet-1	45.9	44.3	-1.6
OpenEarthMap	28.0	31.7	$+3.7$
LoveDA	41.4	40.2	-1.2
DLRSD	28.4	30.7	$+2.3$
LandCover.ai	51.7	54.1	$+2.4$
Potsdam	59.2	61.2	$+2.0$
iSAID	24.9	29.9	$+5.0$
UAVid	42.1	41.7	-0.4
DeepGlobe	37.5	36.0	-1.5
UDD5	44.9	45.0	$+0.1$
FloodNet	63.5	60.4	-3.1
Mean	**42.1**	**44.4**	$\mathbf{+2.3}$

The coverage weight of a cluster is the fraction of it the candidate mask claims, whose token-level counterpart is membership of that mask. A check on two benchmarks over five values found the published threshold already the best of them, so the variant introduces no hyperparameter of its own.

Table 15 reports the outcome. The token-level decision raises INSID3 from 42.1 to 44.4 mIoU and helps on twelve of the seventeen benchmarks, by 12.4 on xBD and by 7.5 on both SpaceNet-2 and Massachusetts Buildings, which are the benchmarks whose targets are small structures threaded through background of a similar appearance. It costs accuracy where a class covers large connected regions, by 3.1 on FloodNet and by 1.5 on DeepGlobe. The gain is smaller than the 13.5 the same change is worth inside FROST, and the result remains 3.9 below FROST, so the granularity of the decision carries part of the distance between the two methods and the feature refinement carries the rest. That the same change is worth less to INSID3 is what our position anticipates, since a finer unit can only draw on the discrimination the features beneath it already support, so the two stages surrounding the class model compound rather than substitute for one another. FSSDINO makes the same point from the other side, since it commits a label per token from the start and still trails the cluster-level INSID3 by 5.2 mIoU at one shot, its features being compared in a geometry no support statistic has refined.

J THE SAME DECOMPOSITION ACROSS METHODS

The labelling that gives the reachable value of Appendix H, which we call the ceiling of a decision, reads only the partition of the query, the gate applied to it and the ground truth, and never the features or the score of the method that produced them. A ceiling is therefore a property of the decision a segmenter commits to rather than of the segmenter, and it can be measured for any method that commits one. It splits the distance a method keeps from a perfect segmentation into what its decision forecloses, which is the ceiling, and the shortfall the method keeps from that ceiling, which is what it fails to extract from what its decision leaves reachable. We re-ran three competing methods with the labelling switched on so that the attained value of each and the ceiling of its own decision come from the same run on the same episodes. The unit of INSID3 is one cluster of the query and its gate enters the score a cluster receives rather than cutting the output, so its ceiling carries no gate, while the token-level variant of Appendix I multiplies the score of a token by the gate and its ceiling carries one. FSSDINO commits a label at every token under no gate at all. GF-SAM returns a union of

Table 16: Attained foreground mIoU at one shot and the ceiling of the decision each method commits to, on each of the seventeen remote-sensing benchmarks, where Mass. Build. abbreviates Massachusetts Buildings and Token the variant of INSID3 whose decision is taken one token at a time. The shortfall of a method on a benchmark is the difference between the two blocks. Every ceiling is read from the same run as the attained value in the matching column, so the two are measured on the same episodes.

	Attained					Ceiling				
Benchmark	FROST	INSID3	Token	FSSDINO	GF-SAM	FROST	INSID3	Token	FSSDINO	GF-SAM
SpaceNet-2	56.1	40.8	48.3	42.5	34.0	68.9	49.1	56.6	82.0	49.6
xBD	46.9	27.0	39.4	22.6	28.2	51.7	42.0	47.9	67.0	55.4
WHU Building	77.4	72.6	74.0	69.2	64.6	81.1	73.5	76.2	83.5	69.2
Vaihingen	61.9	55.9	57.4	51.6	48.8	76.4	66.0	66.1	82.5	65.1
Mass. Build.	33.1	19.3	26.8	28.7	11.8	39.7	26.6	31.5	47.2	16.3
WHDLD	38.2	32.4	33.3	33.3	33.8	60.3	60.0	45.6	85.9	63.5
SpaceNet-1	48.2	45.9	44.3	40.9	41.2	71.5	52.2	54.1	82.7	49.8
OpenEarthMap	34.2	28.0	31.7	29.0	26.8	51.7	56.6	44.1	77.6	51.7
LoveDA	44.8	41.4	40.2	33.3	38.8	60.2	70.9	52.2	88.3	72.6
DLRSD	30.9	28.4	30.7	18.9	30.1	41.5	40.3	38.9	48.9	81.0
LandCover.ai	59.9	51.7	54.1	35.4	48.3	72.6	70.6	65.6	91.1	71.4
Potsdam	64.4	59.2	61.2	52.0	55.2	79.7	74.8	71.3	88.9	74.8
iSAID	27.7	24.9	29.9	15.1	16.3	47.3	50.7	44.2	76.6	57.5
UAVid	49.0	42.1	41.7	39.0	38.8	63.7	76.4	57.3	89.2	60.1
DeepGlobe	38.9	37.5	36.0	33.3	36.2	63.9	84.9	53.4	97.1	69.1
UDD5	49.6	44.9	45.0	36.4	39.7	72.6	76.3	61.9	92.5	70.6
FloodNet	60.2	63.5	60.4	45.5	55.1	81.2	87.3	75.1	95.5	84.7
Mean	**48.3**	**42.1**	**44.4**	**36.9**	**38.1**	**63.8**	**62.2**	**55.4**	**81.0**	**62.5**

overlapping mask proposals, for which we cut the proposals into the atoms of their arrangement, every union of proposals being a union of atoms, so the best labelling of the atoms bounds every selection the method could make and its ceiling is an upper bound rather than an attainable value.

Table 16 gives on every benchmark the decomposition Table 4 summarizes. Two readings the summary leaves out are worth recording. INSID3 reaches a ceiling of 62.2 mIoU, within 1.6 of the 63.8 of FROST, so a decision taken over clusters of the query forecloses barely more than a decision taken at the token once each carries the gate its own method applies, and it still attains 6.2 less. The shortfall of a method is also not a constant of the benchmark, since on DeepGlobe it runs from 17.4 mIoU for the token-level variant of INSID3 to 63.8 for FSSDINO while their ceilings are 53.4 and 97.1, so a method that forecloses little on an easy partition is not thereby close to it.

Read together with Appendix H the two measurements say different things and both are needed. Within one method, where the features, the class model and the gate are all held fixed and only the unit moves, coarsening the decision costs almost exactly what it forecloses, 11.7 mIoU attained against 11.4 reachable. Across methods nothing is held fixed and the ceiling accounts for none of the ordering. What the two support together is not that a finer unit is better but that no method can exceed what its unit permits and that every method measured here stays far short of that permission, by 15.5 mIoU in the case of FROST. The token-level variant of INSID3 shows both halves at once, since it gives up 6.8 mIoU of ceiling to a gate that cuts its output and still gains 2.3, the gate discarding more foreground the coarse unit could never have recovered than foreground it could.

K Where the gain from more references comes from

FROST reads the whitening transform, the bandwidth and the gate radius from the support, so accuracy could grow with the support either because those estimates improve or because the density ratio simply has more anchors to compare against. We separate them by holding those quantities at the estimates the first reference gives while the density ratio still sees the anchors of all ten, which leaves every other stage untouched, freezing first the whitening transform and the bandwidth and then the gate radius as well. With a single reference the frozen and the re-estimated settings coincide, and we confirmed that they agree to the last digit there.

Table 17 reports the result. Ten shots reach 56.8 mIoU against 48.3 at one shot, and holding all three quantities at their one-shot estimates still reaches 55.3, so 7.0 of the 8.5 mIoU the larger support

Table 17: Foreground mIoU at one shot and at ten shots, on each of the seventeen remote-sensing benchmarks, where Mass. Build. abbreviates Massachusetts Buildings. Of the three ten-shot columns, the first holds the whitening transform and the kernel bandwidth at the estimates the first reference gives, the second holds the radius of the spatial gate there as well, and the third re-estimates all three. The density ratio sees the anchors of every reference in every column.

Benchmark	One shot	Ten shots, transform and bandwidth frozen	Ten shots, gate frozen as well	Ten shots
SpaceNet-2	56.1	60.7	59.5	61.3
xBD	46.9	56.2	54.0	56.6
WHU Building	77.4	76.9	77.3	78.0
Vaihingen	61.9	65.9	65.6	66.2
Mass. Build.	33.1	31.6	29.7	33.8
WHDLD	38.2	44.4	43.2	45.8
SpaceNet-1	48.2	52.3	51.4	53.2
OpenEarthMap	34.2	44.9	41.9	44.8
LoveDA	44.8	53.3	52.4	52.3
DLRSD	30.9	38.3	37.2	39.5
LandCover.ai	59.9	65.2	65.2	65.7
Potsdam	64.4	74.3	74.4	74.5
iSAID	27.7	40.5	40.4	39.4
UAVid	49.0	58.0	57.2	58.1
DeepGlobe	38.9	49.0	47.6	48.7
UDD5	49.6	66.6	65.8	67.3
FloodNet	60.2	78.2	78.3	79.5
Mean	**48.3**	**56.3**	**55.3**	**56.8**

buys is bought by the anchors alone. Of the remaining 1.5, the gate radius accounts for 1.0 and the whitening transform and the bandwidth together for 0.5, so the extent the gate admits gains more from further references than the geometry the comparison runs in does. It is the nonparametric class model, rather than the refinement read from the support, that carries most of the behaviour of FROST as reference masks accumulate.

L Backbone ablation

FROST reads from whatever features its backbone provides, so a natural question is how much of its accuracy is owed to the specific frozen encoder we use and how much to the pipeline we build around it. To separate the two, we hold the entire FROST pipeline at its default configuration and replace only the backbone, comparing six frozen encoders that vary along three axes, the pretraining domain, the model scale, and the architecture family. We evaluate every encoder at one shot on five of the seventeen remote-sensing benchmarks, the land-cover benchmarks Potsdam, LoveDA, and Vaihingen together with the instance-dense iSAID and the drone benchmark UDD5, chosen to span the range of foreground scales the full suite covers, and Table 18 reports the foreground mIoU of each. The domain axis contrasts the default DINOv3 encoder, pretrained on general-domain imagery, with a DINOv3 variant pretrained on satellite imagery that we write DINOv3-SAT, the scale axis contrasts a ViT-L of about 0.3 billion parameters with a ViT-7B of about 6.7 billion, and the architecture axis contrasts the vision transformer of DINOv3 with its ConvNeXt-L variant and with the earlier DINOv2 ViT-L. The first row is the default DINOv3 ViT-L backbone of FROST, and the last column reports each encoder's change in mean foreground mIoU relative to it.

Three findings follow from the table. The first concerns the pretraining domain. The general-domain DINOv3 encoders surpass their satellite-pretrained DINOv3-SAT counterparts by a wide and scale-independent margin, the DINOv3 ViT-L leading the DINOv3-SAT ViT-L by 14.3 mean foreground mIoU and the DINOv3 ViT-7B leading the DINOv3-SAT ViT-7B by 14.4. Pretraining on satellite imagery therefore harms in-context segmentation on remote-sensing imagery at both scales rather than helping it, a counterintuitive result we read as evidence that the broad visual coverage of general-domain pretraining yields features that transfer to overhead scenes better than the narrower satellite distribution does, which the pipeline then exploits with no adaptation of its own.

The second finding concerns model scale. Growing the DINOv3 encoder from ViT-L to ViT-7B raises the mean by only 0.6 and wins on three of the five benchmarks, an improvement small against the

roughly twentyfold increase in parameters and the larger inference cost it carries, so the ViT-L sits at a favourable operating point at which accuracy has largely saturated.

The third finding concerns the architecture and the pretraining recipe at a fixed scale. The DINOv3 ViT-L exceeds the DINOv2 ViT-L by 6.1 and the DINOv3 ConvNeXt-L by 7.0, so the gain is not explained by the transformer architecture alone, since the ConvNeXt variant shares the DINOv3 recipe yet emits features on a native 32×32 grid at half the spatial resolution of the ViT and trails it on every benchmark. Together these results indicate that the accuracy of FROST rests on a general-domain transformer backbone of moderate scale rather than on a domain-specialized or larger one, and that its default DINOv3 ViT-L encoder is the configuration its pipeline is best served by.

Table 18: One-shot foreground mIoU of the FROST pipeline with the backbone replaced, on five remote-sensing benchmarks, where DINOv3 is pretrained on general-domain imagery and DINOv3-SAT on satellite imagery. The first row is the default backbone of FROST, the parameter column reports the approximate count in billions, the mean column reports the foreground mIoU averaged over the five benchmarks, and the last column reports the change in that mean relative to the default backbone.

Backbone	Parameters	Potsdam	iSAID	LoveDA	UDD5	Vaihingen	Mean	Difference
DINOv3 ViT-L (default)	0.3B	64.4	27.7	44.8	49.6	61.9	49.7	—
DINOv3 ViT-7B	6.7B	68.1	29.2	42.9	48.8	62.5	50.3	+0.6
DINOv2 ViT-L	0.3B	58.1	22.4	37.7	42.8	56.9	43.6	−6.1
DINOv3 ConvNeXt-L	0.2B	57.0	19.8	38.4	44.5	53.6	42.7	−7.0
DINOv3-SAT ViT-7B	6.7B	43.0	13.9	38.1	34.0	50.6	35.9	−13.8
DINOv3-SAT ViT-L	0.3B	43.6	12.5	37.9	33.5	49.7	35.4	−14.3

M INFERENCE COST

The parameter counts of Table 6 say what each method carries but not what it spends, so we measure the cost of a single episode directly. Table 19 reports it beside accuracy. Because a neighbouring job on the same machine can slow a run by an order of magnitude, the measurement is worth only as much as its control, and we therefore run one job at a time on one otherwise idle GPU, time the model alone from the references and the query going in to the mask coming out with the device synchronized at both ends, and discard the first three episodes of each run as warm-up. All eleven configurations see the same episodes of one benchmark, which is enough for a cost figure because every method processes a fixed input resolution and its cost is dominated by forward passes of fixed size, whereas accuracy still needs the full suite. We report medians rather than means, because the cost of GF-SAM and of Matcher follows the number of proposals SAM returns for an episode and varies by up to forty-six percent between the tenth and the ninetieth percentile of a run, where the cost of FROST, whose work per episode is fixed, varies by about one percent.

At one shot FROST costs between 1.25 and 1.31 times the two baselines that share its frozen backbone and returns between 6.2 and 11.4 mIoU more, while Matcher costs five times as much as FROST and returns 20.9 less. The gap widens with the support size, since FROST encodes every reference under both views and its cost is therefore linear in the number of references, where GF-SAM reads its proposals from one pass and barely moves. A cheaper operating point exists, in that removing the horizontal flip halves the encoder passes the support costs and buys back a factor of 1.52 at one shot and 2.09 at five shots for the 1.0 mIoU that Table 3 attributes to that view.

The encoder dominates this cost, which matters whenever a query carries several classes at once. Encoding the query takes 372 ms and every stage after it 110 ms per class, so a query with K known classes pays the encoder once rather than K times if the support is encoded offline and the query encoding is shared. Against the naive route that re-encodes for each class, the shared route takes 0.48 seconds instead of 1.22 at one class, 1.28 instead of 9.79 at eight, and 3.98 instead of 39.59 at thirty-two, a tenfold reduction that leaves the predicted masks bit-identical.

Table 19: Inference cost, measured on the same 26 episodes of LoveDA for every method, beside the mean foreground mIoU over the seventeen remote-sensing benchmarks. The second row removes the horizontal flip of the reference, and Table 6 gives the parameter count of every method.

	Seconds per episode		Mean foreground mIoU	
Method	One shot	Five shots	One shot	Five shots
Training-free				
FROST	1.21	5.71	48.3	55.6
FROST, single reference view	0.80	2.73	47.3	—
INSID3	0.97	2.50	42.1	48.6
FSSDINO	0.93	2.62	36.9	48.6
PerSAM	0.95	2.85	13.1	14.3
Matcher	6.12	6.87	27.4	31.1
GF-SAM	0.88	1.19	38.1	44.6
Learning-based				
SegIC	0.64	1.84	42.8	44.2
SINE	0.64	1.78	18.6	24.8
DiffewS	0.44	1.12	35.7	40.4
SegGPT	0.22	1.01	33.6	35.8

N QUERIES THAT DO NOT CONTAIN THE CLASS

Every episode of the standard protocol contains the class, so no table in this paper reports what a method predicts when the class is not there, although a segmenter that answers an absent query with a confident region is of little use. We therefore built a target-absent protocol on the same episodes. Each episode keeps its class and its reference masks and only the query is exchanged for the next image of the benchmark that holds no pixel of that class and is not itself a reference. The ground truth is then empty, the intersection over union is undefined, and what remains informative is the area a method predicts and how often it predicts nothing at all, for which we count an episode as empty when the prediction covers under one part in a hundred of the image. Benchmarks whose scenes exceed the input size are left out, since their episodes are tiles of a larger scene while the exchange is defined on whole images, which leaves six of the seventeen.

Table 20 reports both quantities for the three methods that share the frozen backbone. FROST predicts nothing on 63 per cent of the absent queries against the 35 of INSID3 and the 4 of FSSDINO, so a decision taken at every token and cut by a gate read from the support abstains far more readily than one taken over clusters of the query. Its mean predicted area is nevertheless 9.5 per cent against the 7.7 of INSID3, so on the episodes where FROST does answer it commits a larger region than the baseline does. The verdict splits between the two quantities and we report both halves rather than the one that favours us. FSSDINO loses on both, covering 21.4 per cent of the image on average and falling silent on 4 per cent of the queries, which is the account its ceiling already gives in Appendix J of a decision taken at every token with nothing suppressing it. The spread across benchmarks is wide, from 0.1 per cent of area with 98 per cent silent on xBD, whose targets are small buildings against a background the support separates cleanly, to 18.0 and 58 on LandCover.ai, whose classes cover large connected regions that resemble one another.

Table 20: Behaviour on queries that hold no pixel of the class, for the three methods that share the frozen backbone, at one shot. Every episode of the standard protocol keeps its class and its reference and only the query is exchanged for the next image of the benchmark without that class, so the ground truth is empty and the intersection over union is undefined. Area is the predicted foreground as a percentage of the valid pixels, averaged over the classes of the benchmark, and Empty is the percentage of episodes on which the prediction covers under one percent of the image. Lower area and higher Empty are better. Episodes counts the episodes of the benchmark, identical for the three methods. The benchmarks whose scenes exceed the input size are omitted, since their episodes are tiles of a larger scene and the exchange is defined on whole images.

Benchmark	Episodes	FROST		INSID3		FSSDINO	
		Area	Empty	Area	Empty	Area	Empty
LoveDA	300	10.2	55	9.7	30	20.7	2
UDD5	300	8.9	47	6.5	19	23.1	2
LandCover.ai	300	18.0	58	12.9	39	24.0	7
xBD	287	0.1	98	3.2	69	10.6	8
SpaceNet-1	125	3.8	78	5.1	39	24.1	2
WHDLD	300	15.8	43	8.5	16	25.9	3
Mean		**9.5**	**63**	**7.7**	**35**	**21.4**	**4**