TAMING SAM3 FOR FEW-SHOT SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Few-shot semantic segmentation (FSS) requires segmenting target objects using a few reference images and masks. Conventional methods learn cross-image correspondences through episodic training, incurring training costs and posing challenges for generalizing to novel categories. A recent vision foundation model (VFM), SAM3, invites a different view: once a visual prompt identifies a concept, the model can propagate it to corresponding instances. This behavior, also known as promptable concept segmentation (PCS), mirrors FSS yet exposes a spatial mismatch: the support visual evidence and the geometry prompts inhabit different image coordinates. To this end, we introduce Visual Concept Transportation for SAM3 (VCT-SAM3), which brings the support concept into the query coordinate system to exploit SAM3's native concept propagation. Rather than transferring the entire support scene, VCT-SAM3 extracts the annotated support object and places it into the query image. The inserted object then serves as a visual prompt, enabling SAM3 to discover and segment corresponding instances. Moreover, we explore a position-debiased semantic prior to protect potential target regions through the visual concept transportation. Our VCT-SAM3 uses a frozen SAM3, without auxiliary models or additional training. Experiments indicate that VCT-SAM3 achieves state-of-the-art (SOTA) performance (*e.g.* 7.9% mIoU gain on the COCO-20^i benchmark) with lower latency and memory consumption. Our codes will be released to foster future research.

1 INTRODUCTION

Few-shot semantic segmentation (FSS) aims to segment a target object in a query image given only a few annotated support examples. Most existing methods formulate FSS as a cross-image correspondence problem and primarily employ episodic training to learn how to match support and query features via prototypes or dense correlations (Wang et al., 2019; Tian et al., 2022; Min et al., 2021; Lang et al., 2022; Peng et al., 2023). Learning such a representation typically requires dedicated episodic training, which introduces additional training overhead and can limit generalization to distribution shifts between training and evaluation categories.

Promptable segmentation offers an alternative perspective: the SAM series generates object masks from provided cues. Specifically, PerSAM (Zhang et al., 2024b) computes the support-query similarity to derive positive and negative point prompts that guide SAM's mask prediction. FSSAM (Xu et al., 2025) encodes pseudo-mask prompts into the query memory and relies on fine-tuning the memory encoder for SAM2 prediction. However, in these cases, target discovery remains driven by cross-image correspondence that provides foreground localization prompts, while SAM series models are responsible only for mask prediction. Since the cross-image correspondence often relies on additional VFMs (Caron et al., 2021; Oquab et al., 2023) and is prone to inaccurate matching in complex scenes. It inspires us to explore how to discover the target object within the query image.

Recently, SAM3 (Carion et al., 2025) introduces promptable concept segmentation (PCS). Given a visual geometric prompt indicating an object in an image, SAM3 can use the prompted object as a reference to discover and segment other objects of the same concept within the same image. This capability provides a natural opportunity to extend SAM series beyond downstream segmentation, enabling the foundation model itself to perform both target discovery and segmentation through within-scene concept propagation. Yet a key bottleneck remains: the support annotations and the query image reside in separate images. A recent method, FSS-SAM3 (Tsai et al., 2026), spatially combines the support and query images into a unified image to directly use support annotations

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.

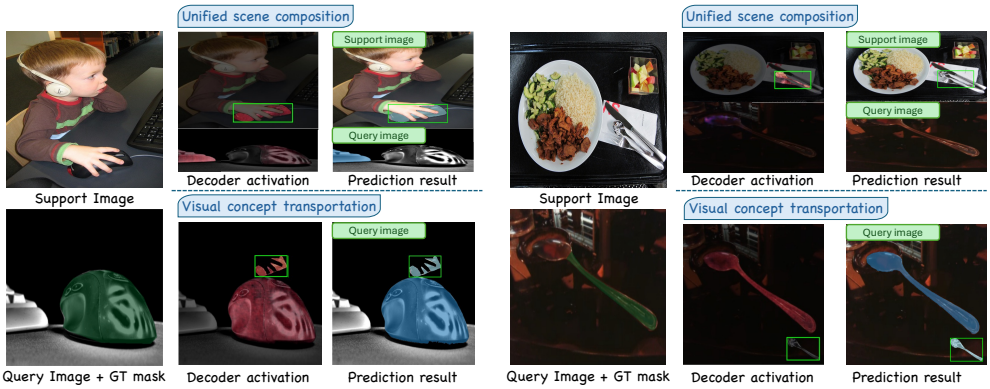


Figure 1: Comparison of unified scene composition and visual concept transportation for SAM3-based FSS. The green bounding box is employed as a visual prompt for SAM3.

as a visual prompt. However, it also introduces substantial unrelated context into the query. As shown in Fig. 1, unified scene composition can't yield a reliable concept propagation in the query image. In the left case, since the support box correctly specifies the *mouse*, SAM3 still produces incomplete activation on the query image and segments the incorrect object *keyboard*. A similar pattern is observed in the right case, where the query object *spoon* is weakly activated. In contrast, when support objects are transported into the query images, SAM3 produces cleaner activations that better cover the target object. This observation motivates us to explore a concept composition strategy that transfers only the visual support evidence while preserving the original query context.

Based on this insight, we propose Visual Concept Transportation for SAM3 (VCT-SAM3), which extracts each annotated support object and inserts it into the query image. To reliably transport this object, we examine the feature space of SAM3 and find that its patch representations exhibit positional bias similar to self-supervised and contrastive-supervised models (Cuttano et al., 2026; Yang et al., 2024), which can interfere with cross-image semantic matching. We therefore introduce Position-Debiased Semantic Matching (PDSM) to suppress position-dominated components and construct a more reliable semantic prior for visual concept transportation. It identifies suitable regions for transporting the support object into the query coordinate system, while the tight bounding box of the transported object serves as a visual prompt to enable SAM3 for target segmentation. VCT-SAM3 defines FSS as a concept transportation from the support image to the query coordinate system, while SAM3 itself performs both concept discovery and segmentation. Our contributions are concluded as:

- We reformulate FSS as the object-level visual concept transportation for PCS. The transported support object serves as a visual prompt to enable SAM3's native within-image concept propagation for query target discovery and segmentation.
- We introduce VCT-SAM3, a training-free visual concept transportation framework that combines position-debiased semantic matching with target-preserving placement for reliable concept transportation to the query image.
- Our VCT-SAM3 relies solely on a frozen SAM3 without additional models or training. Extensive experiments demonstrate state-of-the-art performance with high inference efficiency, outperforming existing approaches across diverse FSS benchmarks.

2 Related Work

2.1 Conventional Few-Shot Segmentation

Few-shot semantic segmentation (FSS) aims to segment target objects in a query image given only a few annotated support examples. Conventional approaches generally establish support-query correspondence through episodic training. Early prototype-based methods compress annotated support features into compact class representations for query prediction. PANet (Wang et al., 2019) intro-

duces bidirectional prototype alignment, while PFENet (Tian et al., 2022) enriches support prototypes with high-level priors and multi-scale features. BAM (Lang et al., 2022) further separates novel-foreground prediction from the suppression of base-category regions. To preserve finer spatial information, dense-correlation methods directly model local correspondence between support and query features. HSNet (Min et al., 2021) constructs multi-level 4D hypercorrelations, while HDMNet (Peng et al., 2023) progressively distills coarse semantic alignment into fine-grained correspondence. Despite different support representations and matching architectures, these methods share a common paradigm in which query prediction relies on cross-image correspondence learned from episodic training. This dependence introduces additional training costs and may limit generalization under distribution shifts between base and novel categories.

2.2 Promptable Semantic Segmentation

Promptable segmentation provides an alternative interface for FSS, where a model produces masks conditioned on external object cues. SAM (Kirillov et al., 2023) established this paradigm with geometric prompts such as points, boxes, and masks. Several studies adapt SAM to reference-guided segmentation by first converting support information into query-side prompts. PerSAM (Zhang et al., 2024b) derives positive and negative points from support-query feature similarities, Matcher (Liu et al., 2024b) combines general-purpose feature matching with SAM proposals and mask refinement, and GF-SAM (Zhang et al., 2024a) improves prompt selection through graph-based grouping. However, target localization in these approaches remains dependent on external support-query correspondence, making the final prediction sensitive to correspondence quality. SAM2 (Ravi et al., 2024) extends promptable segmentation to videos through a memory mechanism for object propagation across frames. FSSAM (Xu et al., 2025) adapts this memory interface to category-level FSS using pseudo prompts and memory refinement. More recently, SAM3 (Carion et al., 2025) introduces promptable concept segmentation (PCS), enabling concept-level detection and segmentation from visual prompts. This capability is particularly relevant to FSS: rather than requiring an external module to determine where the target lies in the query, a visual exemplar can directly specify the target concept for subsequent discovery. This motivates a different perspective on support-to-query transfer based on SAM3's native concept propagation capability.

2.3 In-context Visual Prompting

In-context visual prompting expresses task guidance directly through visual examples, providing an alternative to learned prompt embeddings. SegGPT (Wang et al., 2023a) represents segmentation prompts through visual coloring. SINE (Liu et al., 2024a) further uses context examples as visual references and explicitly models their interaction with the target image. These approaches demonstrate that visual examples can provide comprehensive semantic cues to localize the target object. With SAM3's visual prompting capability, reference evidence can instead be expressed directly in its native input space without additional prompt learning. FSS-SAM3 (Tsai et al., 2026) adopts a straightforward strategy by spatially composing complete support and query images, such that the annotated support instance serves as an in-context visual prompt. However, directly using the complete support scene introduces irrelevant visual context and scales poorly with multiple support examples. VCT-SAM3 instead constructs compact object-level visual prompts and transfers them into the query image, providing a more efficient interface for SAM3 concept propagation.

3 Method

In this section, we first formulate FSS through the concept interface of SAM3 and outline our framework. We then present visual concept transportation, Position-Debiased Semantic Matching (PDSM) for a reliable semantic prior, and within-image concept propagation, as illustrated in Fig. 2.

3.1 Problem Setup

An FSS episode comprises a query image I_q and a K-shot support set $\mathcal{S}$:

$$I_q \in \mathbb{R}^{H_q \times W_q \times 3}, \qquad \mathcal{S} = \left\{ \left(I_s^k, M_s^k\right) \,\middle|\, I_s^k \in \mathbb{R}^{H_k \times W_k \times 3},\ M_s^k \in \{0,1\}^{H_k \times W_k} \right\}_{k=1}^{K}, \quad (1)$$

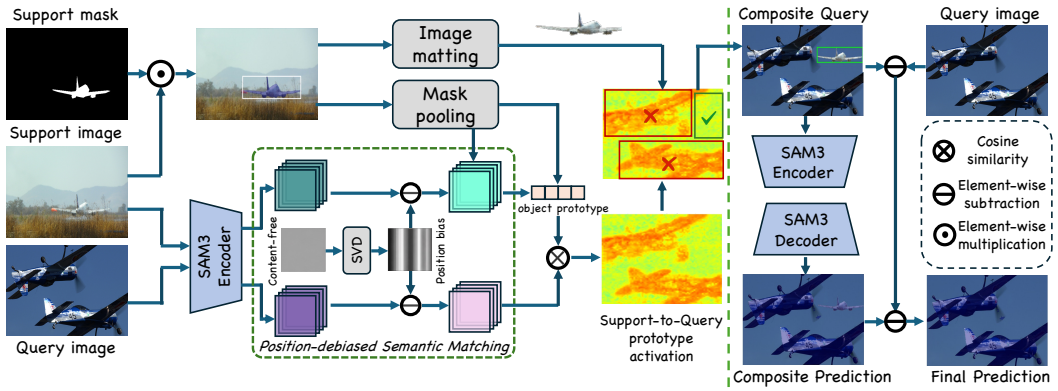


Figure 2: Overview of our VCT-SAM3 framework. The left part: visual concept transportation with position-debiased semantic matching. The right part: prediction of the composite query image. Our VCT-SAM3 converts the original cross-image support evidence into a valid visual prompt for SAM3 target discovery and segmentation.

where H_q and W_q denote the height and width of the query image, H_k and W_k denote the height and width of the k-th support image. Each support mask M_s^k identifies a foreground instance of the same semantic concept. The objective is to predict a binary mask M_q that segments instances of the support concept in I_q.

We then describe the reformulation of FSS through PCS. Assume a positive box $\mathcal{B}_q$ that encloses an exemplar instance of the target concept in the query image, the SAM3 decoder can predict the union of masks as:

$$\mathcal{G}_\theta(I_q; \mathcal{B}_q) = \mathcal{D}_{\text{SAM3}}\left(F_q, \Psi(\mathcal{B}_q)\right) \in \{0,1\}^{H_q \times W_q}, \tag{2}$$

where $\mathcal{D}_{\text{SAM3}}$ denotes the SAM3 decoder, F_q is the extracted feature of I_q and $\Psi(\cdot)$ denotes the prompt encoder. The key property is that a box enclosing a visible example can prompt a concept that $\mathcal{G}_\theta$ propagates to other matching instances within I_q. To facilitate this ability for FSS, we aim to construct a visual concept transportation $\mathcal{T}_{s\to q} : (I_q, I_s, M_s) \mapsto (\hat{I}_q, \hat{\mathcal{B}}_q)$, where $\hat{I}_q$ is the composite query image containing the transported support object and $\hat{\mathcal{B}}_q$ encloses this object. The prediction is then obtained as $\hat{M}_q = \mathcal{G}_\theta(\hat{I}_q; \hat{\mathcal{B}}_q)$.

3.2 Framework Overview

VCT-SAM3 realizes the visual concept transportation $\mathcal{T}_{s\to q}$ as illustrated in Fig. 2. For each support–query pair, we use semantic matching to identify candidate regions for placing the support object. Since positional bias in the SAM3 feature subspace introduces noise into semantic matching, we propose Position-Debiased Semantic Matching (PDSM) to mitigate it and yield more reliable candidate regions. Guided by these regions, we transport the support object into the query image, with the tight bounding box serving as a visual prompt. Finally, SAM3 predicts the segmentation mask on the composite image, and the transported support mask is then removed to obtain the final FSS prediction. For clarity, the framework is described in a one-shot setting. The details of the multi-shot setting are shown in Appendix A.2.

3.3 Semantic Prior for Concept Transportation

We transport the support object into the query image by constructing the visual concept transportation $\mathcal{T}_{s\to q}$, which converts the support evidence into a valid visual prompt. The key challenge is to determine a suitable location that introduces the support object without corrupting potential query targets. We first construct a semantic prior, then use it to guide target-preserving placement.

Let F_s and F_q denote the support and query features from I_s and I_q, respectively. We aggregate the annotated support object by masked average pooling, formulated as:

$$p_s = \text{Norm}\left(\frac{\sum_u M_s(u) F_s(u)}{\sum_u M_s(u) + \epsilon}\right), \tag{3}$$

where p_s represents the support concept and u denotes the u-th patch token. We then compute its cosine affinity with each query location that serves as the semantic prior for placement:

$$\mathcal{A}(u) = \cos\left(F_q(u), p_s\right), \tag{4}$$

where cos denotes the cosine similarity and $\mathcal{A}$ denotes the semantic prior over the query image. We partition this semantic prior into positive and negative regions according to their confidence scores and apply density-based clustering dbc (Kriegel et al., 2011) to the two candidate sets to identify spatially coherent regions and remove discrete noises, which are formulated as:

$$\mathcal{R}^+ = \mathrm{dbc}\left(\{u \mid \mathcal{A}(u) \geq \tau^+\}\right), \qquad \mathcal{R}^- = \mathrm{dbc}\left(\{u \mid \mathcal{A}(u) < \tau^-\}\right), \tag{5}$$

where τ^+ and τ^- denote the threshold for positive and negative regions, respectively. $\mathcal{R}^+$ indicates the region that contains potential query targets and should be preserved. $\mathcal{R}^-$ denotes regions with low affinity for the support concept and serves as admissible locations for placement. Together, they provide the candidate regions for subsequent target-preserving placement.

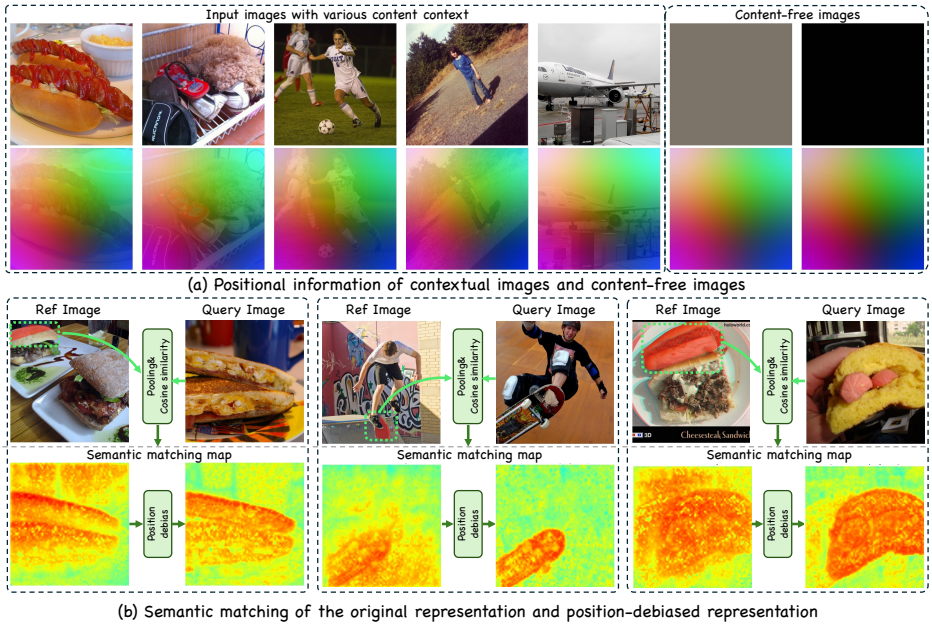


Figure 3: (a): Positional structure in SAM3 features under various contents and constant contents. (b): Semantic prior under SAM3 raw features and position-debiased features.

3.4 Position-Debiased Semantic Matching

The semantic prior (Eq. 4) for transportation depends on the quality of support–query matching. However, recent studies have shown that vision transformers pretrained with self-supervised or contrastive objectives (Oquab et al., 2023; Radford et al., 2021) exhibit position-biased feature representations (Darcet et al., 2024; Yang et al., 2024). Such positional bias can introduce spatially dependent noise, which may propagate through feature matching and ultimately pollute our semantic prior. Thus, we analyze this bias and propose Position-Debiased Semantic Matching (PDSM) to suppress position-dominated components to obtain reliable semantic priors.

First, we conduct a diagnostic analysis to explore the position-related subspace. In particular, we decompose the feature representations using Singular Value Decomposition (SVD) to isolate the position-related subspace, as illustrated in the left part of Fig. 3 (a). It reveals that the positional pattern remains consistent across images with diverse visual contents. To examine its dependence on image content, we consider two content-free inputs, including a uniform image filled with the ImageNet normalization mean (Deng et al., 2009) and a constant black image. As shown in the right part of Fig. 3 (a), these constant inputs also exhibit a similar pattern compared to diverse content images, suggesting the presence of content-independent positional components. This motivates us to estimate a position subspace from content-free features and remove it to mitigate positional bias.

To this end, we utilize a content-free image I^0 and extract its SAM3 feature $F^0 \in \mathbb{R}^{C\times N}$, where N is the patch token number. We extract the dominant directions to yield a position-related subspace. Specifically, we first center F^0 by subtracting the global pooled feature and then apply SVD, formulated as:

$$\overline{F}^0 = F^0 - \mathrm{GAP}(F^0), \qquad \overline{F}^0 = U\Sigma V^\top, \tag{6}$$

where $\mathrm{GAP}(\cdot)$ denotes global average pooling over spatial locations. U and V contain the left and right singular vectors, and Σ is the diagonal matrix of singular values. Subtracting the pooling feature removes the component shared across spatial locations, allowing SVD to capture the dominant directions of features. Since I^0 contains no visual contents, we attribute these dominant directions to position-related components and retain the leading r left singular vectors U_r as the estimated position-related subspace. Given an arbitrary SAM3 backbone feature map F, its projection $U_r U_r^\top F$ is the component aligned with this estimated subspace. We remove this component from the raw SAM3 features to obtain the position-debiased features $\widetilde{F}$:

$$\widetilde{F} = F - U_r U_r^\top F. \tag{7}$$

To obtain position-debiased semantic priors, we use $\widetilde{F}_s$ and $\widetilde{F}_q$ to replace F_s and F_q in Eq. 3 and Eq. 4 and obtain reliable positive and negative regions $(\widetilde{\mathcal{R}}^+, \widetilde{\mathcal{R}}^-)$. We also compare the semantic matching of raw SAM3 features and position-debiased features in Fig. 3 (b). The position-debiased features yield a cleaner semantic prior to identify positive and negative regions. We also analyze the effect of different content-free inputs for estimating the position-related subspace in Appendix C.1.

3.5 Support object placement for segmentation

Target-preserving placement. To determine $\mathcal{T}_{s\to q}$, we leverage the position-debiased positive and negative regions $(\widetilde{\mathcal{R}}^+, \widetilde{\mathcal{R}}^-)$. First, we search for placements in I_q that avoid overlapping $\widetilde{\mathcal{R}}^+$ while locating the centroid of the transported support object within $\widetilde{\mathcal{R}}^-$. We then search over a set of scales to align the transported object size with the potential target, yielding a more compatible transportation. Specifically, for a support mask transported to location z with scale α, denoted by $M_s^{z,\alpha}$, we first define the feasible set Ω as following:

$$\Omega = \left\{ (z,\alpha) \,\middle|\, M_s^{z,\alpha} \cap \widetilde{\mathcal{R}}^+ = \varnothing,\ c(M_s^{z,\alpha}) \in \widetilde{\mathcal{R}}^- \right\}, \tag{8}$$

where $c(\cdot)$ denotes taking the centroid of the mask. Among the feasible candidates, we then select scales where the support object size is closest to the estimated size of $\mathcal{R}^+$. For each scale α, we resize the support object and search for locations satisfying the constraints in Ω. We then select a feasible location-scale pair formulated as:

$$(z^\star, \alpha^\star) = \arg\min_{(z,\alpha)\in\Omega} \left| \ell(M_s^{z,\alpha}) - \ell(\widetilde{\mathcal{R}}^+) \right|, \tag{9}$$

where $\ell(\cdot)$ denotes the longer side of a region. This process yields the transportation parameter $(z^\star, \alpha^\star)$ that minimizes interference with potential targets. More details are shown in Appendix A.3. Using this parameter, we obtain the transported object $\hat{C}_s$ and its binary mask $\hat{M}_s$ as:

$$\hat{C}_s = \mathcal{T}_{obj}(I_s \odot M_s; z^\star, \alpha^\star), \qquad \hat{M}_s = \mathcal{T}_{mask}(M_s; z^\star, \alpha^\star), \tag{10}$$

where $\odot$ denotes element-wise multiplication, $\mathcal{T}_{obj}$ and $\mathcal{T}_{mask}$ respectively transform the support foreground and mask with scale $\alpha^\star$ and place them at $z^\star$ in the query coordinate. We then compose the transported object with the query image and yield its tight box as the visual prompt:

$$\hat{I}_q = (1 - \hat{M}_s) \odot I_q + \hat{C}_s, \qquad \hat{\mathcal{B}}_q = \mathrm{bbox}(\hat{M}_s), \tag{11}$$

where $\hat{I}_q$ is the composite query image after placement and $\hat{\mathcal{B}}_q$ is the tight bounding box of the transported object. These operations instantiate the visual concept transportation $\mathcal{T}_{s\to q}(I_q, I_s, M_s) = \{\hat{I}_q, \hat{\mathcal{B}}_q\}$. $\hat{\mathcal{B}}_q$ serves as the visual prompt for the SAM3 decoder.

Within-image concept propagation. After composition, the transported object and the original query content coexist within the same image coordinate system. We feed $\hat{\mathcal{B}}_q$ as the visual prompt to the SAM3 concept interface: $\hat{M}_q = \mathcal{G}_\theta(\hat{I}_q; \hat{\mathcal{B}}_q)$, defined in Eq. 2. Since $\hat{M}_q$ also contains the support object itself, we remove its transported mask $\hat{M}_s$ from the composite prediction:

$$M_q = \hat{M}_q \odot (1 - \hat{M}_s). \tag{12}$$

The remaining regions constitute the final FSS prediction on the query image.

Table 1: Performance comparison on PASCAL-5i and COCO-20i. † denotes the configuration without textual prompts. The gray oracle row uses a ground-truth (GT) query box only for target-preserving placement. The best result and second-best result are marked with **bold** and underline.

Method	PASCAL-5i				COCO-20i			
	1-shot		5-shot		1-shot		5-shot	
	mIoU	FB-IoU	mIoU	FB-IoU	mIoU	FB-IoU	mIoU	FB-IoU
Classical FSS methods								
HSNet (ICCV'21) (Min et al., 2021)	64.0	–	69.5	–	39.2	–	46.9	–
BAM (CVPR'22) (Lang et al., 2022)	67.8	79.7	70.9	82.2	46.2	–	51.2	–
SCCAN (ICCV'23) (Xu et al., 2023)	66.8	77.7	70.3	81.8	46.3	69.9	53.9	74.2
FPTrans (NeurIPS'22) (Zhang et al., 2022)	68.8	–	78.0	–	47.0	–	58.9	–
HDMNet (CVPR'23) (Peng et al., 2023)	69.4	–	71.8	–	50.0	72.2	56.0	77.7
PAM (AAAI'24) (Wang et al., 2024)	69.5	–	74.7	–	48.5	–	53.9	–
AENet (ECCV'24) (Xu et al., 2024a)	69.8	80.8	74.1	84.5	49.4	73.6	56.7	76.5
AMNet (NeurIPS'23) (Wang et al., 2023b)	70.1	–	73.2	–	51.0	72.9	57.3	78.8
HMNet (NeurIPS'24) (Xu et al., 2024b)	70.4	81.6	73.9	84.4	52.1	74.5	58.9	77.6
Training-based VFM-based FSS methods								
VRP-SAM (CVPR'24) (Sun et al., 2024)	71.9	–	–	–	53.9	–	–	–
FSSAM (ICML'25) (Xu et al., 2025)	81.0	89.4	85.4	91.9	62.3	77.3	69.3	82.9
Training-free VFM-based FSS methods								
Matcher (ICLR'24) (Liu et al., 2024b)	68.1	–	74.0	–	52.7	–	60.7	–
GF-SAM (NeurIPS'24) (Zhang et al., 2024a)	72.1	–	**82.6**	–	58.7	–	66.8	–
FSS-SAM3† (ArXiV'26) (Tsai et al., 2026)	79.6	88.3	81.2	88.9	66.6	82.9	71.8	84.9
VCT-SAM3 (Ours)	**80.8**	**89.2**	81.8	**89.4**	**74.5**	**86.3**	**76.6**	**86.8**
VCT-SAM3 + GT placement (Oracle)	82.4	90.4	83.9	91.3	75.7	87.8	77.1	88.4

4 Experiments

4.1 Experimental Setup

Datasets and evaluation metrics. We evaluate VCT-SAM3 on the standard FSS benchmarks, including PASCAL-5i (Everingham et al., 2010) and COCO-20i (Lin et al., 2014). PASCAL-5i divides 20 semantic classes into four disjoint folds of five classes, while COCO-20i partitions the 80 MS-COCO classes into four folds of 20 classes. We follow the standard four-fold cross-validation protocol and report results for both 1-shot and 5-shot episodes. Each fold contains 1,000 episodes. We take intersection over union as the evaluation metric, including mIoU and FB-IoU.

Implementation details. VCT-SAM3 uses the official SAM3 implementation with all model parameters frozen. We estimate the position subspace once from a content-free input. The threshold τ^+ and τ^- are set to 0.7 and 0.3, and the default rank is $r = 64$. The final mask is obtained by removing the transported foreground from the composite prediction. The whole inference pipeline uses neither task-specific training nor textual prompts or auxiliary segmentation models.

4.2 Comparison to SOTA Methods

4.2.1 Quantitative Comparison

We compare VCT-SAM3 with training-based and VFM-based FSS methods. As shown in Tab. 1, compared with the recent SOTA FSS-SAM3 (Tsai et al., 2026), VCT-SAM3 improves mIoU by 1.2%/0.6% mIoU on PASCAL-5i 1/5-shot and by 7.9%/4.8% mIoU on COCO-20i 1/5-shot, respectively. On PASCAL-5i 5-shot, VCT-SAM3 remains close to GF-SAM (Zhang et al., 2024a) while exceeding FSS-SAM3 (Tsai et al., 2026). The gain is more significant on COCO-20i, where our VCT-SAM3 also surpasses the training-based FSSAM (Xu et al., 2025) by 12.2% in the 1-shot setting. We also employ an oracle experiment that uses the query GT box only to enforce target-preserving placement, as marked by the gray row. This yields gains of 1.6%/2.1% mIoU on PASCAL-5i and 1.2%/0.5% on COCO-20i in the 1/5-shot settings, respectively. The oracle results

show only marginal improvements over VCT-SAM3, indicating that our position-debiased semantic prior already provides reliable guidance for target-preserving placement. The concrete results of each episodic fold are reported in Appendix B.2.

4.2.2 Qualitative Comparison

We compare our VCT-SAM3 against prior VFM-based FSS methods in Fig. 4. In the second and third cases, where the target objects occupy only a small portion of the support images, existing methods suffer from either over-segmentation or under-segmentation, whereas VCT-SAM3 consistently recovers the target regions with more complete and precise masks. The last case further reveals the ambiguity of cross-image support–query matching: existing methods tend to associate the support evidence with a *'bread'*-like appearance, thereby producing incomplete masks, whereas VCT-SAM3 correctly propagates the intended *'hot dog'* concept and captures the complete target instances. These results demonstrate that transporting annotated support objects serves as explicit within-image concept prompts, enabling more reliable concept identification and propagation.

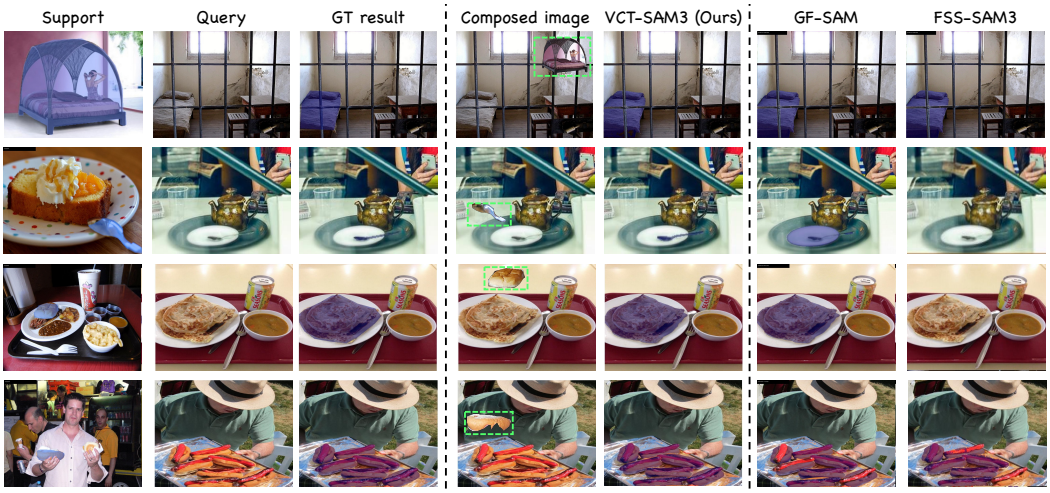


Figure 4: Visual comparisons of prediction masks among various SAM-based methods.

4.3 Ablation Study

We conduct comprehensive ablation experiments to validate the effect of the proposed components and verify the hyperparameters. More results are summarized in Appendix C. We also discuss the extremely failure cases and provide alternative solutions in Appendix B.3.

4.3.1 Component ablation.

We evaluate the component contributions of VCT-SAM3 in Tab. 2. The unified-scene composition variant spatially composes the complete annotated support image with the query image for SAM3 inference, instead of transporting the support object. Among other placement variants, random placement achieves 48.6% and 40.3% mIoU on PASCAL-5^i and COCO-20^i, which means it suffers from overlapping potential targets. Adding $\mathcal{R}^+$ preservation with raw affinity raises mIoU to 75.3% and 71.7%, highlighting the importance of protecting potential targets during transportation. Finally, replacing raw affinity with PDSM yields additional gains of 2.7% and 2.3% points, demonstrating the benefit of suppressing positional bias for more reliable placement.

4.3.2 Diagnosing the candidate regions

We evaluate the spatial quality of the positive region $\mathcal{R}^+$ using foreground and box mIoU (IoU^+_{fg} and $\text{IoU}^+_{\text{box}}$). Together, these metrics assess the quality of the positive-region prior used for target-preserving placement. Tab. 3 compares the semantical matching under raw features and positional debiased ones. It's observed that PDSM improves both metrics on the two benchmarks. On

Table 2: Component ablation (%) on 1-shot PASCAL-5i and COCO-20i.

Variant	PDSM	Transportation	$\mathcal{R}^+$	$\mathcal{R}^-$	PASCAL-5i mIoU	PASCAL-5i FB-IoU	COCO-20i mIoU	COCO-20i FB-IoU
unified-scene composition					78.7	87.6	65.4	82.3
VCT-SAM3 with random placement		✓			48.6	72.4	40.3	68.9
VCT-SAM3 with $\mathcal{R}^+$ guidance		✓	✓		75.3	86.9	71.7	83.4
VCT-SAM3 with $\mathcal{R}^+$ & $\mathcal{R}^-$ guidance		✓	✓	✓	78.1	87.6	72.2	84.9
VCT-SAM3	✓	✓	✓	✓	**80.8**	**89.2**	**74.5**	**86.3**

PASCAL-5i, IoU^+_{fg} increases from 67.4% to 72.1%, while $\text{IoU}^+_{\text{box}}$ rises from 70.5% to 73.9%. On COCO-20i, the corresponding gains are 3.5% and 4.6%. These improvements indicate that the PDSM-derived positive region better aligns with the target foreground, providing a more accurate spatial prior to preserve query regions during transportation.

Table 3: Positive-region diagnosis for 1-shot FSS. Metrics are computed per episode and averaged within each fold, followed by averaging across all folds.

Affinity feature	PASCAL-5i $\text{IoU}^+_{\text{fg}}\uparrow$	PASCAL-5i $\text{IoU}^+_{\text{box}}\uparrow$	COCO-20i $\text{IoU}^+_{\text{fg}}\uparrow$	COCO-20i $\text{IoU}^+_{\text{box}}\uparrow$
Raw SAM3 feature	67.4	70.5	62.3	61.8
PDSM feature	72.1	73.9	65.8	66.4

Table 4: Efficiency comparison. The unified-scene composition baseline directly concatenates the complete support and query images spatially and uses the default hyperparameters without resizing.

Method	1-shot Latency (s)	1-shot Memory (MB)	5-shot Latency (s)	5-shot Memory (MB)
Unified-scene composition	0.47	4765	1.98	18295
VCT-SAM3 (Ours)	0.35	3483	1.54	6572

4.3.3 Efficiency evaluation

We compare VCT-SAM3 with unified-scene composition in Tab. 4. VCT-SAM3 consistently reduces both latency and peak GPU memory, with the efficiency gain becoming more pronounced in the 5-shot setting. This advantage stems from the compact object transportation design of VCT: rather than concatenating complete support scenes with the query, VCT transports only the support objects required for concept propagation, substantially reducing the spatial size of the composite input. As the number of shots increases, this design avoids the rapid growth of redundant support content, leading to particularly substantial memory savings in the 5-shot setting.

5 Conclusion

In this paper, we introduced Visual Concept Transportation for SAM3 (VCT-SAM3) to bridge the support–query coordinate mismatch in FSS. Our framework transports object-level support evidence into the query image to enable SAM3's native concept propagation, while employing position-debiased semantic matching for target-preserving placement. Extensive experiments demonstrate its effectiveness and efficiency using only a frozen SAM3, without additional training or auxiliary models. These results establish object-level evidence transportation as an effective interface for adapting PCS models to cross-image FSS. We hope this work inspires further exploration.

AI USE STATEMENT

We use large language models (LLMs) to assist with language polishing and improve the clarity and readability of the manuscript. We have reviewed and verified all LLM-assisted revisions.

REPRODUCIBILITY STATEMENT

We provide detailed experimental settings and implementation details to ensure reproducibility. The core source codes are included in the supplementary material, with comments specifying the key configurations and procedures required to reproduce the reported results.

REFERENCES

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Dídac Surís, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pp. 9650–9660, 2021.

Claudia Cuttano, Gabriele Trivigno, Christoph Reich, Daniel Cremers, Carlo Masone, and Stefan Roth. Insid3: Training-free in-context segmentation with dinov3. In *CVPR*, pp. 21638–21648, 2026.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, pp. 2632–2652, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pp. 248–255, 2009.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *ICCV*, pp. 4015–4026, 2023.

Hans-Peter Kriegel, Peer Kröger, Jörg Sander, and Arthur Zimek. Density-based clustering. *WIREs Data Mining and Knowledge Discovery*, 1(3):231–240, 2011.

Chunbo Lang, Gong Cheng, Binfei Tu, and Junwei Han. Learning what not to segment: A new perspective on few-shot segmentation. In *CVPR*, pp. 8057–8067, 2022.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pp. 740–755, 2014.

Yang Liu, Chenchen Jing, Hengtao Li, Muzhi Zhu, Hao Chen, Xinlong Wang, and Chunhua Shen. A simple image segmentation framework via in-context examples. In *NeurIPS*, 2024a.

Yang Liu, Muzhi Zhu, Hengtao Li, Hao Chen, Xinlong Wang, and Chunhua Shen. Matcher: Segment anything with one shot using all-purpose feature matching. In *ICLR*, 2024b.

Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *ICCV*, pp. 6941–6952, 2021.

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

Bohao Peng, Zhuotao Tian, Xiaoyang Wu, Chengyao Wang, Shu Liu, Jingyong Su, and Jiaya Jia. Hierarchical dense correlation distillation for few-shot segmentation. In *CVPR*, pp. 23641–23651, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Yanpeng Sun, Jiahui Chen, Shan Zhang, Xinyu Zhang, Qiang Chen, Gang Zhang, Errui Ding, Jingdong Wang, and Zechao Li. VRP-SAM: SAM with visual reference prompt. In *CVPR*, pp. 23565–23574, 2024.

Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *IEEE T-PAMI*, 44(2):1050–1065, 2022.

Yi-Jen Tsai, Yen-Yu Lin, and Chien-Yao Wang. Few-shot semantic segmentation meets SAM3. *arXiv preprint arXiv:2604.05433*, 2026.

Jing Wang, Jiangyun Li, Chen Chen, Yisi Zhang, Haoran Shen, and Tianxiang Zhang. Adaptive FSS: A novel few-shot segmentation framework via prototype enhancement. In *AAAI*, pp. 5463–5471, 2024.

Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. PANet: Few-shot image semantic segmentation with prototype alignment. In *ICCV*, pp. 9197–9206, 2019.

Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. SegGPT: Towards segmenting everything in context. In *ICCV*, pp. 1130–1140, 2023a.

Yuan Wang, Naisong Luo, and Tianzhu Zhang. Focus on query: Adversarial mining transformer for few-shot segmentation. In *NeurIPS*, pp. 31524–31542, 2023b.

Qianxiong Xu, Wenting Zhao, Guosheng Lin, and Cheng Long. Self-calibrated cross attention network for few-shot segmentation. In *ICCV*, pp. 655–665, 2023.

Qianxiong Xu, Guosheng Lin, Chen Change Loy, Cheng Long, Ziyue Li, and Rui Zhao. Eliminating feature ambiguity for few-shot segmentation. In *ECCV*, pp. 416–433, 2024a.

Qianxiong Xu, Xuanyi Liu, Lanyun Zhu, Guosheng Lin, Cheng Long, Ziyue Li, and Rui Zhao. Hybrid mamba for few-shot segmentation. In *NeurIPS*, pp. 73858–73883, 2024b.

Qianxiong Xu, Lanyun Zhu, Xuanyi Liu, Guosheng Lin, Cheng Long, Ziyue Li, and Rui Zhao. Unlocking the power of SAM 2 for few-shot segmentation. In *ICML*, pp. 69950–69966, 2025.

Jiawei Yang, Katie Z Luo, Jiefeng Li, Congyue Deng, Leonidas Guibas, Dilip Krishnan, Kilian Q Weinberger, Yonglong Tian, and Yue Wang. Denoising vision transformers. In *ECCV*, pp. 453–469, 2024.

Anqi Zhang, Guangyu Gao, Jianbo Jiao, Chi Harold Liu, and Yunchao Wei. Bridge the points: Graph-based few-shot segment anything semantically. In *NeurIPS*, 2024a.

Jian-Wei Zhang, Yifan Sun, Yi Yang, and Wei Chen. Feature-proxy transformer for few-shot segmentation. In *NeurIPS*, pp. 6575–6588, 2022.

Renrui Zhang, Zhengkai Jiang, Ziyu Guo, Shilin Yan, Junting Pan, Hao Dong, Yu Qiao, Peng Gao, and Hongsheng Li. Personalize segment anything model with one shot. In *ICLR*, 2024b.

APPENDIX

This appendix provides implementation and placement details, diagnostic analyses of transported content and extreme cases, and additional ablations on the PDSM rank and feature source.

A Experimental Setup **12**

A.1 Implementation Details 12

A.2 Support Object Extraction and Multi-Shot Transportation 12

A.3 Placement Constraints 13

A.4 Efficiency Measurement 13

B Diagnostic Analysis **14**

B.1 Controlled Comparison of Transported Content 14

B.2 Fold-Level Comparison 15

B.3 Case Study 15

C Extra Ablation Studies **16**

C.1 Position-Related Subspace Controls 16

C.2 Sensitivity to the PDSM Rank 17

C.3 Feature Sources for Semantic Prior 17

C.4 Sensitivity to Positive and Negative Region Thresholds 17

C.5 Sensitivity to the Support Object Scale Range 18

A EXPERIMENTAL SETUP

A.1 IMPLEMENTATION DETAILS

Main experiment. The default PDSM rank is $r = 64$. The affinity thresholds are $\tau^+ = 0.7$ and $\tau^- = 0.3$. Candidate support object scales α are sampled from $[0.5, 1.5]$ with step $= 0.1$. The input image size is set by default. We estimate the basis of the position-related subspace once from a constant black input at the evaluation resolution and reuse it across episodes, following Eq. 6 and Eq. 7. The final prediction uses the frozen SAM3 concept interface on the composite query image. The default pipeline requires no textual prompts, task-specific training, or auxiliary feature extractor. The feature-source ablation explicitly replaces the features used for placement.

GT-assisted placement for oracle experiment. The oracle setting in Tab. 1 uses the query GT bounding box $\mathcal{B}_q^{\mathrm{GT}}$ for both target protection and scale matching. Specifically, the box area replaces $\widetilde{\mathcal{R}}^+$ in the non-overlap constraint of Eq. 8, and its long side $\ell(\mathcal{B}_q^{\mathrm{GT}})$ replaces $\ell(\widetilde{\mathcal{R}}^+)$ in the scale-selection objective of Eq. 9. The negative region $\widetilde{\mathcal{R}}^-$ remains the prediction from position-debiased semantic matching. The candidate scales, the placement search, and the subsequent segmentation procedure are unchanged. SAM3 is prompted with the tight boxes of the transported support objects. The query GT box is used only in the placement procedure and is not used as the visual prompt.

A.2 SUPPORT OBJECT EXTRACTION AND MULTI-SHOT TRANSPORTATION

One-shot Setting. Each support object is extracted from the support annotation and inserted into the query image using its transported binary mask. Its tight bounding box serves as a visual prompt. For a support annotation containing multiple disconnected foreground regions, we extract a separate support object from each region.

Multi-shot Setting. In the multi-shot setting ($K > 1$), the support set $\mathcal{S} = \{(I_s^k, M_s^k)\}_{k=1}^K$ provides several annotated examples of the same target concept. Collecting these objects across the K support images gives $L = \sum_{k=1}^K n_k$ object foregrounds. We index them by $j = 1, \ldots, L$, with $C_{s,j}$ and $M_{s,j}$ denoting the object foreground and its binary mask. Then, we transport these objects sequentially into a shared query image. For object j, its support–query match provides the position-debiased regions $(\widetilde{\mathcal{R}}_j^+, \widetilde{\mathcal{R}}_j^-)$. We select a location–scale pair $(z_j^\star, \alpha_j^\star)$ using the placement rule in Sec. A.3. In addition, the transported object mask must not overlap any previously inserted object masks. Following Eq. 10, we obtain the transported object $\hat{C}_{s,j}$ and mask $\hat{M}_{s,j}$.

To construct the composite query image, starting from $\hat{I}_q^{(0)} = I_q$, we insert one support object at a time, formulated as:

$$\hat{I}_q^{(j)} = (1 - \hat{M}_{s,j}) \odot \hat{I}_q^{(j-1)} + \hat{C}_{s,j}, \qquad j = 1, \ldots, L, \qquad \hat{I}_q = \hat{I}_q^{(L)}. \tag{13}$$

The non-overlap constraint preserves objects inserted in earlier steps. The final $\hat{I}_q$ therefore contains the query scene and all transported support objects in a unified coordinate system. For each inserted object, we compute the tight axis-aligned bounding box directly from its transported mask, $\hat{\mathcal{B}}_{q,j} = \mathrm{bbox}(\hat{M}_{s,j})$. We collect these boxes as the positive visual prompts $\mathcal{B}_\mathcal{S} = \{\hat{\mathcal{B}}_{q,j}\}_{j=1}^L$. Each box refers to a support object in $\hat{I}_q$. We feed the composite image and the full set of boxes jointly to frozen SAM3, following Eq. 2. The SAM3 decoder predicts the matching concept instances:

$$\hat{M}_q = \mathcal{G}_\theta(\hat{I}_q; \mathcal{B}_\mathcal{S}) = \mathcal{D}_{\mathrm{SAM3}}\big(\hat{F}_q, \Psi(\mathcal{B}_\mathcal{S})\big). \tag{14}$$

The resulting binary prediction includes matching query instances and transported support foreground. We remove the union of all transported masks to obtain the final query segmentation M_q:

$$M_\mathcal{S} = \bigcup_{j=1}^{L} \hat{M}_{s,j}, \qquad M_q = \hat{M}_q \odot (1 - M_\mathcal{S}). \tag{15}$$

A.3 Placement Constraints

Following Eq. 8 and Eq. 9, we select a location-scale pair that preserves the position-debiased positive region $\widetilde{\mathcal{R}}^+$, places the support object centroid within the negative region $\widetilde{\mathcal{R}}^-$, and matches the estimated query-target size. The transformed support foreground must also remain inside I_q.

We sample scales $\alpha \in [0.5, 1.5]$ with a step of 0.1 and denote the candidate locations by $\mathcal{Z}$. The size measure $\ell(\cdot)$ is the longer side of a region's tight bounding box. For each sampled scale $\alpha \in [0.5, 1.5]$, we resize the support mask and evaluate its long-side difference from $\widetilde{\mathcal{R}}^+$. We then search $\mathcal{Z}$ for locations satisfying the placement constraints and retain the feasible pair with the smallest difference. The selected $(z^\star, \alpha^\star)$ specifies the support object and mask transformations in Eq. 10. The pseudocode below details this search.

A.4 Efficiency Measurement

The timing statistics cover feature extraction for the semantic prior and candidate regions construction, support object extraction and placement, composite-image inference, and removal of transported regions. Estimation of the position-related subspace basis is a one-time preprocessing step and is measured separately from per-episode inference. Memory denotes peak GPU memory allocated during an episode, with the peak statistic reset before each episode.

The full-support composition setting in Tab. 4 spatially concatenates complete support and query images using its default hyperparameters. The resulting composite is not resized to a fixed resolution, so adding support images enlarges its input composite. In contrast, VCT-SAM3 only inserts compact support objects into the composite query image, while multiple shots add support objects without appending complete support scenes. This distinction in composite construction is the basis of the reported latency and memory comparison.

Algorithm: target-preserving placement

Input: Query image I_q, support mask M_s, position-debiased positive and negative regions $\widetilde{\mathcal{R}}^+, \widetilde{\mathcal{R}}^-$, scales $\alpha \in [0.5, 1.5]$ sampled with a step of 0.1, and candidate locations $\mathcal{Z}$.
Output: $(z^\star, \alpha^\star)$.

$d^\star \leftarrow +\infty$; $\ell_+ \leftarrow \ell(\widetilde{\mathcal{R}}^+)$
for each sampled scale $\alpha \in [0.5, 1.5]$ **do**
  $M_s^\alpha \leftarrow \mathrm{Resize}(M_s, \alpha)$; $d \leftarrow |\ell(M_s^\alpha) - \ell_+|$
  for each $z \in \mathcal{Z}$ **do**
    $M \leftarrow \mathrm{Translate}(M_s^\alpha, z)$
    if M lies within I_q **and** $M \cap \widetilde{\mathcal{R}}^+ = \varnothing$
      and $c(M) \in \widetilde{\mathcal{R}}^-$ **and** $d < d^\star$ **then**
      $(z^\star, \alpha^\star, d^\star) \leftarrow (z, \alpha, d)$
    end if
  end for
end for
return $(z^\star, \alpha^\star)$

B Diagnostic Analysis

Table 5: Controlled comparison of transported content (%). P/C denotes PASCAL-5i/COCO-20i. Both variants share the same support object scale, placement location, and visual prompt.

Transported content	Shot	mIoU (P/C)	FB-IoU (P/C)
Tight-box crop	1	77.3/71.2	87.5/83.9
Foreground cutout	1	**80.8/74.5**	**89.2/86.3**
Tight-box crop	5	76.4/70.8	86.9/83.3
Foreground cutout	5	**81.8/76.6**	**89.4/86.8**

B.1 Controlled Comparison of Transported Content

This comparison examines the visual evidence supplied by two forms of transported content: 1) a tight-box crop that retains the annotated foreground and background inside its box; 2) a foreground cutout that retains only the annotated object. The paired variants share the position-debiased semantic prior, support object scale, placement location, and visual prompt. The crop replaces the full rectangular footprint, whereas the cutout replaces only foreground pixels, preserving query content elsewhere inside the box. Tab. 5 shows that foreground cutouts improve mIoU by 3.5/3.3 mIoU points on PASCAL/COCO in the 1-shot setting and by 5.4/5.8 mIoU points in the 5-shot setting. FB-IoU also improves in all four comparisons. The larger 5-shot gap shows that the choice of transported content becomes more consequential as more support evidence is inserted. These results support foreground-only transportation, which supplies focused concept evidence while preserving more of the query content inside the tight bounding box of each transported support object. In contrast, the tight-box crop inevitably introduces irrelevant support context into the query image, disrupting the original contextual continuity and introducing additional ambiguity for concept propagation, which can lead to inaccurate target segmentation.

Fig. 5 further illustrates this effect qualitatively. When the support object is partially occluded, the tight-box crop inevitably includes background or occluding content within its rectangular region, which is transported together with the target object and can interfere with the concept conveyed by the visual prompt. For example, in Fig. 5, the support *hydrant* is partially occluded by a *barricade*. Consequently, its tight bounding box inevitably includes part of the *barricade*, which is transported into the query image together with the *hydrant*. This irrelevant visual evidence contaminates the concept conveyed by the visual prompt, leading to scattered decoder activations and inaccurate propagation to the target hydrants. In contrast, the foreground cutout isolates the annotated *hydrant* from the occluding *barricade*, providing cleaner support evidence and enabling SAM3 to correctly activate the target *hydrant* in the query image.

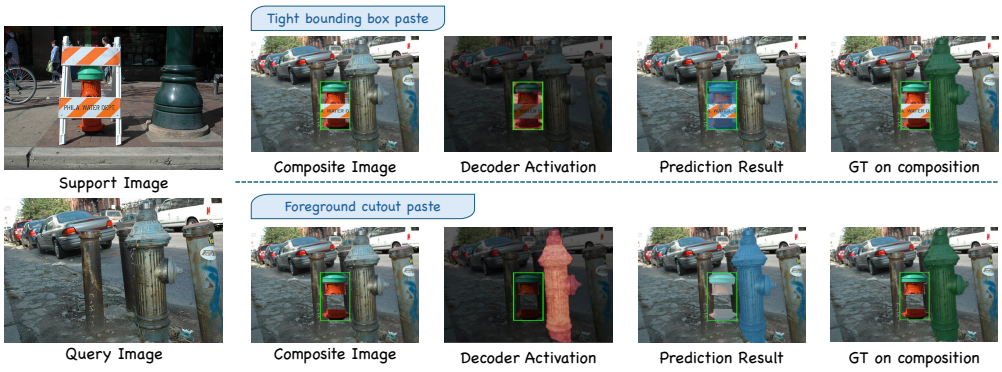


Figure 5: Visual comparison of tight box and foreground cutout for visual concept transportation.

B.2 Fold-Level Comparison

Tab. 6 compares VCT-SAM3 with Matcher (Liu et al., 2024b), GF-SAM (Zhang et al., 2024a), and FSS-SAM3 (Tsai et al., 2026) without text prompts across the four standard folds in the 1-shot and 5-shot settings. Each cell reports fold-level mIoU.

On COCO-20i, VCT-SAM3 exceeds FSS-SAM3 on every 1-shot fold, with gains ranging from 6.6 to 9.1 mIoU points. In the 5-shot setting, it improves threefold and is within 0.3 mIoU points on Fold 3. On PASCAL-5i, the comparison varies across folds: the largest gains over FSS-SAM3 occur on Fold 0, reaching 8.2 and 4.4 mIoU points for 1-shot and 5-shot, respectively. This breakdown shows that the strong COCO 1-shot improvement is distributed across all four folds, while the benefit on PASCAL is more dependent on the evaluated fold.

Table 6: Fold-level comparison on PASCAL-5i and COCO-20i. Each fold cell reports mIoU (%).

Dataset	Method	Shot	Fold 0	Fold 1	Fold 2	Fold 3
PASCAL-5i	Matcher	1	67.7	70.7	66.9	67.0
	GF-SAM	1	71.1	75.7	69.2	73.3
	FSS-SAM3	1	70.4	**84.6**	**80.9**	82.3
	VCT-SAM3 (Ours)	1	**78.6**	82.2	79.4	**83.0**
	Matcher	5	71.4	77.5	74.1	72.8
	GF-SAM	5	**81.5**	86.3	79.7	82.9
	FSS-SAM3	5	74.8	**86.8**	80.4	**82.9**
	VCT-SAM3 (Ours)	5	79.2	85.0	**81.1**	81.7
COCO-20i	Matcher	1	52.7	53.5	52.6	52.1
	GF-SAM	1	56.6	61.4	59.6	57.1
	FSS-SAM3	1	64.2	66.8	67.7	67.5
	VCT-SAM3 (Ours)	1	**73.3**	**75.4**	**74.3**	**74.8**
	Matcher	5	60.1	62.7	60.9	59.2
	GF-SAM	5	67.1	69.4	66.0	64.8
	FSS-SAM3	5	68.6	72.5	72.5	**73.6**
	VCT-SAM3 (Ours)	5	**72.4**	**83.1**	**77.6**	73.3

B.3 Case Study

When the target occupies most of the query image, the remaining space may not admit a support-object placement that satisfies the constraints at any candidate scale. We examine all 4,000 evaluation episodes on each benchmark, where query images may be repeatedly sampled across episodes. As reported in Tab. 7, only 18 episodes on PASCAL-5i and 4 episodes on COCO-20i require composite extension, accounting for 0.45% and 0.10% of the evaluated episodes, respectively. This indicates that valid placement can be found within the original query extent for nearly all episodes, while such extreme cases occur only rarely.

For these rare episodes, we extend the composite query with a blank strip along its shorter dimension, adding only the space required to accommodate the support object at its original scale ($\alpha = 1$). As illustrated in Fig. 6, the support object is placed entirely within the extended region, leaving the original query content intact. We then prompt SAM3 with its tight bounding box as usual, remove the transported support mask, and crop the prediction back to the original query extent. This fallback handles the few infeasible placements without altering the original query content.

Table 7: statistic of extreme cases on PASCAL and COCO.

Dataset	Inspected query episode	Affected query episode	Fraction (%)
PASCAL-5i	4000	18	0.45
COCO-20i	4000	4	0.10

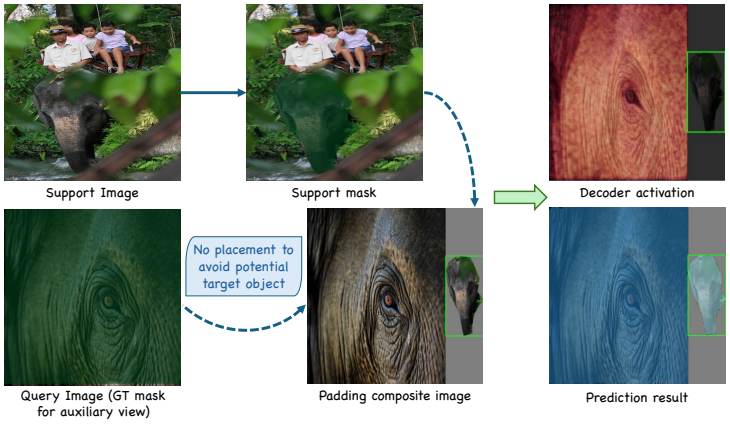


Figure 6: Visual illustration of the padding operation for these extreme cases.

C Extra Ablation Studies

C.1 Position-Related Subspace Controls

We further examine whether the improvement of PDSM comes from removing the estimated position-related subspace, rather than simply suppressing arbitrary feature directions. To this end, we compare raw SAM3 features, random rank-64 subspace removal, and PDSM bases estimated from two different content-free inputs.

Table 8: Subspace controls on 1-shot FSS. P/C denotes PASCAL/COCO benchmarks. All variants evaluate the same 4,000 episodes per benchmark.

Representation	mIoU (P/C)	$\text{IoU}^{+}_{\text{fg}}$ (P/C)	$\text{IoU}^{+}_{\text{box}}$ (P/C)
Raw SAM3 features	78.1/72.2	67.4/62.3	70.5/61.8
Random rank-64 removal	78.3/72.7	68.3/59.3	72.4/63.0
Black-input PDSM	80.8/74.5	72.1/65.8	73.9/66.4
Mean-input PDSM	80.6/74.3	72.2/65.9	73.8/66.4

Tab. 8 shows that black-input PDSM improves mIoU over raw SAM3 features by 2.7% mIoU on PASCAL and 2.3% mIoU on COCO. Random rank-64 removal yields smaller gains of 0.2 and 0.5 points, respectively, and reduces COCO foreground-prior IoU from 62.3% to 59.3%. In contrast, both content-free PDSM variants improve foreground and box IoU on both benchmarks. Thus, removing directions estimated from the positional response produces more consistent benefits than removing an arbitrary subspace of the same rank. The black-input and mean-input PDSM differ by only 0.2% mIoU on each benchmark, with prior-quality differences of at most 0.1% mIoU. We use the black input as the default.

C.2 Sensitivity to the PDSM Rank

We vary the removed rank over $r \in \{16, 32, 64, 128\}$ on COCO-20^i 1-shot. We report segmentation accuracy, IoU^+_{fg} and $\text{IoU}^+_{\text{box}}$ used in the main paper.

Table 9: Rank sensitivity on COCO-20^i in the 1-shot setting.

Rank	mIoU	FB-IoU	IoU^+_{fg}	$\text{IoU}^+_{\text{box}}$
16	73.8	85.9	65.2	65.6
32	74.4	86.2	65.0	66.0
64	74.5	86.3	65.8	66.4
128	73.3	84.8	59.0	60.6

Tab. 9 shows that mIoU remains within 0.7% mIoU for $r = 16$–64. Rank 64 achieves the highest segmentation and prior-quality scores among the tested settings. Increasing the rank to 128 reduces 1.2% mIoU and foreground/box prior IoU by 6.8%/5.8% relative to rank 64. This pattern suggests that removing too many directions can also discard information useful for semantic matching. We therefore adopt $r = 64$ as the default.

C.3 Feature Sources for Semantic Prior

We further investigate the choice of feature source for constructing the semantic prior. Specifically, we replace the default SAM3 features with DINOv2, DINOv3, and CLIP features, while applying the same PDSM procedure to each feature extractor with $r = 64$. We evaluate both the final segmentation performance and the quality of the semantic prior map using IoU^+_{fg} and $\text{IoU}^+_{\text{box}}$.

Table 10: Placement feature-source comparison on 1-shot FSS. P/C denotes PASCAL/COCO. External features introduce an additional frozen VFM.

Feature source	Rank	Extra VFM	mIoU (P/C)	IoU^+_{fg} (P/C)	$\text{IoU}^+_{\text{box}}$ (P/C)
SAM3	64	–	80.8/74.5	72.1/65.8	73.9/66.4
DINOv2	64	✓	81.3/74.2	74.0/66.2	75.1/66.8
DINOv3	64	✓	81.8/74.9	74.3/66.7	74.4/67.2
CLIP	64	✓	70.5/71.2	68.9/62.4	70.7/62.0

Tab. 10 shows that DINOv3 improves mIoU over SAM3 features by 1.0% mIoU on PASCAL and 0.4% on COCO, while improving both prior-quality metrics. DINOv2 also improves the prior-quality scores, but its segmentation gains are mixed: PASCAL mIoU increases by 0.5% mIoU, and COCO mIoU decreases by 0.3% mIoU. CLIP yields lower segmentation and prior-quality scores on both benchmarks under the same matching procedure. SAM3 achieves competitive results with the backbone already available in our pipeline, whereas external alternatives require an additional frozen feature extractor. To summarize, DINO features can provide stronger semantic priors, but require an additional VFM and corresponding computation. SAM3 provides a better trade-off: its internal features already produce competitive placement quality while requiring no external feature extractor. We therefore use SAM3 as the default feature source rather than introducing an additional VFM.

C.4 Sensitivity to Positive and Negative Region Thresholds

We examine the sensitivity of positive and negative threshold on COCO-20^i 1-shot by varying one threshold at a time: $\tau^+ \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ with $\tau^- = 0.3$, and $\tau^- \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ with $\tau^+ = 0.7$. The semantic regions and placements are recomputed for each setting using Eq. 5.

In addition to segmentation accuracy, we report the number of episodes for which no feasible placement can be found within the original query extent. This statistic excludes the extreme episodes in Sec. B.3, where the query target itself occupies most of the image. For any remaining episode without a feasible placement, we apply the same boundary-case procedure by extending the composite with a blank region to accommodate the support object.

Table 11: Region-threshold sensitivity on COCO-20i 1-shot. Each sweep fixes the other threshold at its default.

Sweep	τ^+	τ^-	mIoU	FB-IoU	No placement
Positive threshold	0.5	0.3	73.6	85.7	11
	0.6	0.3	74.1	86.0	2
	0.7	0.3	**74.5**	**86.3**	0
	0.8	0.3	72.2	81.1	0
	0.9	0.3	67.8	75.8	0
Negative threshold	0.7	0.1	73.7	85.8	0
	0.7	0.2	74.1	86.0	0
	0.7	0.3	**74.5**	**86.3**	0
	0.7	0.4	74.2	86.1	0
	0.7	0.5	73.9	85.9	0

As shown in Tab. 11, the default thresholds $(\tau^+, \tau^-) = (0.7, 0.3)$ achieve the best performance, while moderate variations lead to only small degradation. A low τ^+ enlarges the protected positive region and consequently reduces the space available for support object placement, resulting in several infeasible episodes. To avoid the *no placement* issue, we prefer the default parameter setting. Moreover, an excessively high τ^+ provides insufficient protection for potential target regions, leading to occlusion and severe performance degradation.

C.5 Sensitivity to the Support Object Scale Range

We further study how the candidate scale range affects support-object transportation on COCO-20i 1-shot. Similar to Sec. C.4, we also report the number of *no-placement* episodes.

Table 12: Support-object scale-range sensitivity on COCO-20i 1-shot. All intervals use a step of 0.1, including both endpoints.

Scale range	# Scales	mIoU	FB-IoU	No placement
$[1.0, 1.0]$ (fixed)	1	73.5	85.6	14
$[0.8, 1.0]$	3	73.9	85.9	6
$[0.6, 1.0]$	5	74.2	86.1	2
$[0.5, 1.5]$ (default)	11	**74.5**	**86.3**	0
$[0.3, 1.7]$	15	73.9	85.9	0

As shown in Tab. 12, increasing scale flexibility progressively improves placement feasibility. When the minimum scale is relatively large, a large support object cannot be sufficiently reduced to fit the available query region, resulting in *no-placement* episodes. Lowering the minimum scale to 0.5 eliminates these cases and achieves the best segmentation performance. Further extending the range to $[0.3, 1.7]$ brings no additional benefit and slightly degrades accuracy. We therefore adopt $[0.5, 1.5]$ as the default range, which provides sufficient flexibility for placement while avoiding unnecessary scale variation.