PROTOTWINMAMBA: PROTOTYPE-GUIDED SPATIAL ORDERING AND TEMPORAL MODELING FOR VIDEO POLYP SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video polyp segmentation remains challenging due to foreground–background ambiguity and inter-frame appearance variations. In recurrent spatial modeling, token order shapes how contextual information propagates, yet fixed grid traversals do not explicitly account for foreground and background relevance. Motivated by this observation, we propose ProtoTwinMamba, which uses region guidance not only to refine features but also to organize their spatial processing order. Specifically, a coarse prediction provides soft pooling weights for constructing frame-wise foreground and background prototypes. Prototype similarities are combined with the coarse prior to produce region-specific guidance maps, shared by masked linear attention for feature refinement and two independently parameterized Mamba branches for spatial scan ordering. Sorting tokens by their respective guidance scores groups high-relevance tokens together, forming foreground- and background-conditioned sequences without hard token partitioning. Each branch then restores the original feature grid before bidirectional temporal aggregation at fixed spatial locations. Moreover, a shallow detail-enhancement pathway preserves local detail, while a convolutional decoder fuses shallow and deep features to produce the final segmentation masks. Extensive experiments on SUN-SEG and CVC-612 demonstrate that ProtoTwinMamba achieves state-of-the-art performance on both benchmarks.

1 INTRODUCTION

Video Polyp Segmentation (VPS) (Qayoom et al., 2025; Mei et al., 2025; Jha et al., 2021) aims to delineate polyp regions in colonoscopy videos by combining spatial information with neighboring-frame context (Ji et al., 2022). Accurate delineation can support clinicians in locating and assessing lesions (Shao et al., 2024; Wu & Zhao, 2025). However, polyps often resemble the surrounding mucosa in color and texture, making their boundaries difficult to identify. Camera motion, viewpoint changes, and illumination variations further alter their appearance across frames (Fang et al., 2024a; Hu et al., 2024; Zhang et al., 2026). As illustrated in Figure 1(a), these challenges require spatial representations that distinguish foreground from background, together with temporal context that supplements individual-frame observations.

Existing methods exploit spatial and temporal information through attention (Ji et al., 2021; 2022; Zhao et al., 2022; Chen et al., 2024), alignment and memory (Hu et al., 2024; Zhao et al., 2026), and region linking with scale alignment (Fang et al., 2024a). Other approaches investigate diffusion-based temporal reasoning (Lu et al., 2024), adapt the Segment Anything Model (SAM) (Kirillov et al., 2023; Fang et al., 2024b; Wul & Qin, 2026; Li & Masood, 2026; Guo et al., 2026), or distill foundation-model representations (Leng et al., 2026). Frequency–spatial interactions and wavelet-based aggregation provide additional representation cues (He et al., 2026; Zhang et al., 2026). Complementing these directions, we investigate how foreground–background relevance can organize spatial processing before temporal context is incorporated.

Selective State Space Models (SSMs) offer a sequence-based mechanism for contextual modeling. Mamba introduces input-dependent state updates (Gu & Dao, 2023), while VMamba extends selective scanning to visual representations (Liu et al., 2024; 2026). For VPS, STDDNet combines

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

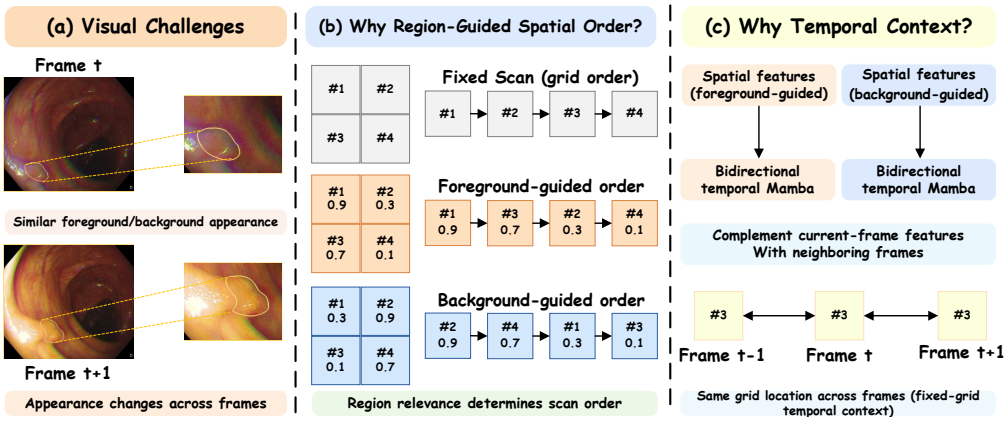


Figure 1: Motivation. (a) Foreground–background ambiguity and inter-frame appearance changes. (b) Region guidance organizes spatial tokens by their foreground or background relevance instead of grid order. (c) Temporal modeling supplements the resulting spatial features with neighboring-frame context. Token IDs denote original grid locations; guidance scores are illustrative.

Mamba-based temporal modeling with spatial processing and dynamic feature extraction (Chen et al., 2025). In recurrent spatial modeling, token order shapes how contextual information propagates. Although selective state updates depend on the input, a fixed grid traversal organizes tokens by spatial position rather than foreground–background relevance. This motivates the question: *can region guidance determine not only how features are weighted, but also the order in which spatial context is accumulated?*

Our starting point is to construct foreground- and background-conditioned sequences from the same feature grid. As illustrated in Figure 1(b), sorting tokens by their relevance to a region places high-relevance tokens together near the beginning of the sequence. We hypothesize that this arrangement provides a useful order for accumulating regional context, allowing evidence from these tokens to contribute to subsequent state updates. Applying the ordering separately to foreground and background guidance produces two region-conditioned processing orders. Importantly, both branches retain all spatial tokens: the guidance changes their processing order rather than partitioning them into disjoint foreground and background sets.

Spatial guidance estimated from a single frame can nevertheless be uncertain when boundaries are ambiguous or appearance is degraded. Neighboring frames provide additional observations that can complement the resulting spatial features. We therefore follow region-guided spatial processing with temporal aggregation within each branch, as illustrated in Figure 1(c). Since the spatial permutation can vary across frames, we restore the original grid positions before temporal modeling. This preserves consistent spatial indexing rather than treating frame-specific sorting ranks as corresponding locations; it does not perform explicit motion alignment.

Building on this design, we propose *ProtoTwinMamba*, a framework that couples region-guided feature refinement and spatial sequence construction with temporal context aggregation. At each deep feature level, a coarse prediction provides soft pooling weights for constructing frame-wise foreground and background prototypes. Prototype similarities are combined with the coarse prior to produce two region-specific guidance maps. These maps jointly guide Masked Linear Attention (MLA) for feature refinement and Prototype-Guided Semantic Scan (PGSS) for spatial ordering in independently parameterized Mamba branches. Whereas prototypes have been used for long-term memory or reference selection (Chen et al., 2026; Wang et al., 2026), we use them as shared guidance for feature refinement and spatial sequence construction.

After spatial processing and grid restoration, Location-preserved Bidirectional Temporal Scan (LBTS) aggregates neighboring-frame context at fixed grid locations. Moreover, a shallow detail-enhancement pathway retains local information without coarse-mask gating, while a convolutional decoder fuses shallow and deep representations to predict segmentation masks.

We evaluate ProtoTwinMamba on SUN-SEG (Ji et al., 2022) and CVC-612 (Bernal et al., 2015), comparing segmentation performance and examining the contributions of its components. Our main contributions are summarized as follows:

- We introduce shared foreground and background prototype guidance that couples contextual feature refinement with spatial sequence construction, using region relevance to guide both feature weighting and scan ordering.
- We develop twin spatial–temporal Mamba branches that process region-conditioned spatial sequences independently and incorporate neighboring-frame context after restoring the original feature grid.
- Extensive experiments on SUN-SEG and CVC-612 demonstrate that ProtoTwinMamba achieves state-of-the-art performance on both benchmarks.

2 Related Work

2.1 Video Polyp Segmentation

VPS combines spatial discrimination with temporal context. PNS-Net (Ji et al., 2021), PNS+ (Ji et al., 2022), and MAST (Chen et al., 2024) explore spatiotemporal dependencies through attention, while EUVPS (Fang et al., 2024a) strengthens spatial representations through region linking and scale alignment. HFSTI-Net (He et al., 2026) and WavePolyp (Zhang et al., 2026) further incorporate frequency information into spatiotemporal modeling. To exploit neighboring and historical observations, SALI (Hu et al., 2024) combines short-term alignment with long-term memory, and DALA (Zhao et al., 2026) uses deformable alignment and local attention. Diff-VPS (Lu et al., 2024) instead explores diffusion-based segmentation with adversarial temporal reasoning. Foundation-model-based methods, including VP-SAM (Fang et al., 2024b) and FSD-Net (Leng et al., 2026), introduce additional priors through adaptation and spatiotemporal distillation. Complementing these approaches, we organize within-frame spatial processing by foreground/background relevance, then use temporal context to supplement region features.

2.2 Prototype Guidance and State Space Modeling

Prototypes summarize region features for video modeling. VPSentry (Chen et al., 2026) employs long-term prototype fusion for semi-supervised VPS, while CMSA-Net (Wang et al., 2026) uses prototype separability and temporal consistency to guide reference selection. Meanwhile, Mamba (Gu & Dao, 2023) introduces selective state updates, and VMamba (Liu et al., 2024) extends selective scanning to visual representations. STDDNet (Chen et al., 2025) applies Mamba-based temporal propagation alongside spatial modeling and dynamic feature extraction for VPS. Whereas CP-Mamba (Zhang et al., 2025) orders tokens by category probabilities for nuclei analysis, our frame-wise foreground/background prototypes and coarse prior jointly guide feature refinement and spatial ordering, followed by temporal aggregation on restored feature grids.

3 Method

3.1 Overview

ProtoTwinMamba organizes within-frame spatial information according to foreground and background relevance, then incorporates cross-frame context to complement the resulting representations. As shown in Figure 2, the same prototype-based guidance is used for contextual feature refinement and spatial scan ordering. Two independently parameterized branches process the foreground-guided and background-guided sequences, respectively, before aggregating temporal information on the restored feature grid.

Specifically, given a clip $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{B\times T\times 3\times H\times W}$, a shared frame-wise encoder extracts four feature levels. Independent 1×1 convolutions then map them to $\boldsymbol{F}_i \in \mathbb{R}^{B\times T\times C\times H_i\times W_i}$, $i \in \{1, 2, 3, 4\}$, where H_i and W_i denote the spatial dimensions at level i. We apply prototype-guided refinement and twin Mamba modeling to levels 3 and 4. In parallel, levels 1 and 2 undergo detail enhancement

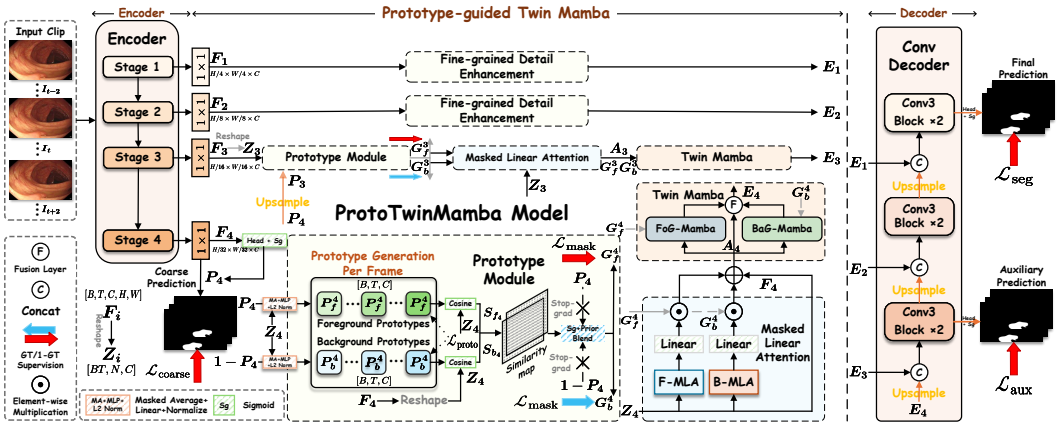


Figure 2: Overview of ProtoTwinMamba. Frame-wise prototype guidance refines deep features and determines foreground/background spatial scan orders. The two branches subsequently aggregate temporal context on restored feature grids. Shallow detail enhancement and a convolutional decoder complement and integrate the resulting representations.

without coarse-mask gating. A convolutional decoder integrates the four enhanced features $\{\boldsymbol{E}_i\}_{i=1}^{4}$ to predict segmentation masks. For operations performed independently on each frame, we omit the batch and frame indices. The operator $\mathcal{R}$ flattens a feature grid into spatial tokens, and $\mathcal{R}^{-1}$ restores the grid. We use $[\cdot,\cdot]$ for channel concatenation, $\odot$ for broadcast element-wise multiplication, σ for sigmoid, and n_2 for ℓ_2 normalization. Unless otherwise specified, $\mathcal{C}_k$ denotes a $k \times k$ convolution followed by group normalization and GELU.

3.2 Prototype-Guided Feature Refinement

We first estimate the relevance of each spatial token to the foreground and background. To this end, a coarse prediction provides pooling weights for constructing frame-wise prototypes. The resulting prototype guidance is used both to refine features in this subsection and to organize spatial sequences in Section 3.3.

Coarse Prior and Frame-Wise Prototype Generation. As shown in Figure 2, a 1×1 prediction head h_{c} maps the deepest feature $\boldsymbol{F}_4$ to coarse logits. We apply sigmoid to obtain the foreground prior and resize it for level 3: $\boldsymbol{L}^{\mathrm{c}} = h_{\mathrm{c}}(\boldsymbol{F}_4)$, $\boldsymbol{P}_4 = \sigma(\boldsymbol{L}^{\mathrm{c}})$, and $\boldsymbol{P}_3 = \mathrm{Up}_3(\boldsymbol{P}_4)$, where Up_3 denotes bilinear resizing to (H_3, W_3). For each frame at level $i \in \{3, 4\}$, we flatten $\boldsymbol{F}_i$ into $\boldsymbol{Z}_i = \mathcal{R}(\boldsymbol{F}_i) \in \mathbb{R}^{N_i \times C}$, where $N_i = H_i W_i$. Let $\boldsymbol{z}_{i,n} \in \mathbb{R}^C$ denote the column-vector representation of token n, and let $p_{i,n}$ be its foreground prior. The pooling weights are $w_{\mathrm{f},i,n} = p_{i,n}$ and $w_{\mathrm{b},i,n} = 1 - p_{i,n}$. We compute a weighted average of the tokens for each region, followed by a region-specific affine projection and ℓ_2 normalization:

$$\boldsymbol{\mu}_r^i = \frac{\sum_{n=1}^{N_i} w_{r,i,n} \boldsymbol{z}_{i,n}}{\max\left(\sum_{n=1}^{N_i} w_{r,i,n}, \epsilon\right)}, \qquad \boldsymbol{p}_r^i = \mathrm{n}_2\left(\ell_{r,i}(\boldsymbol{\mu}_r^i)\right), \quad r \in \{\mathrm{f}, \mathrm{b}\}. \tag{1}$$

Here, ϵ prevents division by zero, and $\ell_{r,i}$ denotes the learned projection. Each frame has one foreground prototype and one background prototype at each deep level. Each prototype has C dimensions; collecting it across a batch of clips gives a tensor of shape $B \times T \times C$.

Soft Prototype Guidance. We next measure the relevance of each token to each region using its cosine similarity to the corresponding prototype: $s_{r,i,n} = \mathrm{n}_2(\boldsymbol{z}_{i,n})^\top \boldsymbol{p}_r^i$. Since the prototypes are already normalized, only the tokens require normalization at this step. We map the similarity to a soft score and combine it with the coarse prior:

$$g_{r,i,n} = \frac{1}{2}\sigma(\tau_i s_{r,i,n}) + \frac{1}{2}\,\mathrm{sg}(w_{r,i,n}), \tag{2}$$

where τ_i is a learnable scale initialized to 5 and clamped to $[1, 20]$, and sg denotes stop-gradient. This stop-gradient applies only to the prior term in Eq. equation 2; the prior remains differentiable

through the pooling operation in Eq. equation 1. The scores form $\boldsymbol{g}_r^i \in \mathbb{R}^{N_i}$, whose spatial form is denoted by $\boldsymbol{G}_r^i$. Similarly, $\boldsymbol{S}_r^i$ denotes the spatial similarity map. Across a batch of clips, $\boldsymbol{G}_r^i \in \mathbb{R}^{B\times T\times 1\times H_i\times W_i}$. Although the pooling weights are complementary, the two prototype similarities are computed separately. Thus, the foreground and background guidance maps are not constrained by construction to sum to one.

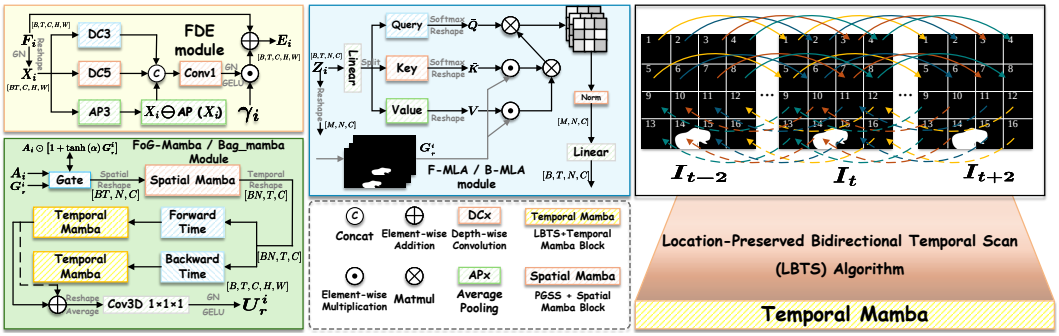


Figure 3: Components of ProtoTwinMamba. Shallow Fine-grained Detail Enhancement (FDE) preserves local information. F-MLA and B-MLA use region guidance to weight keys and values. Each guided Mamba branch models spatial context before aggregating information across frames.

Masked Linear Attention. Using these guidance maps, F-MLA and B-MLA refine the features with region-weighted context, as illustrated in Figure 3. The two modules have independent parameters. For one of J heads (Vaswani et al., 2017), learned projections of $\boldsymbol{Z}_i$ produce $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{N_i\times d}$, where $d = C/J$. We omit branch, level, and head indices below for readability. Specifically, queries are normalized over channels and keys over spatial positions: $\overline{\boldsymbol{Q}} = \mathrm{Softmax}_d(\boldsymbol{Q})$ and $\overline{\boldsymbol{K}} = \mathrm{Softmax}_{N_i}(\boldsymbol{K})$. The values receive no softmax normalization. We then weight both keys and values using the region guidance: $\widetilde{\boldsymbol{K}} = \overline{\boldsymbol{K}} \odot \boldsymbol{g}_r^i$ and $\widetilde{\boldsymbol{V}} = \boldsymbol{V} \odot \boldsymbol{g}_r^i$, with guidance broadcast over channels. These weighted features form a context matrix $\widetilde{\boldsymbol{K}}^\top \widetilde{\boldsymbol{V}} \in \mathbb{R}^{d\times d}$. For query position n, the output is

$$\boldsymbol{o}_{r,i,n}^\top = \frac{\overline{\boldsymbol{q}}_n^\top \left(\widetilde{\boldsymbol{K}}^\top \widetilde{\boldsymbol{V}}\right)}{\max\left(\overline{\boldsymbol{q}}_n^\top \sum_{m=1}^{N_i} \widetilde{\boldsymbol{k}}_m, \epsilon\right)}. \tag{3}$$

Here, $\overline{\boldsymbol{q}}_n$ and $\widetilde{\boldsymbol{k}}_m$ are column-vector representations of the corresponding matrix rows. The head outputs are concatenated and projected to obtain $\mathcal{A}_{r,i}(\boldsymbol{Z}_i, \boldsymbol{g}_r^i)$. This computation avoids an $N_i \times N_i$ attention matrix and has a per-head aggregation cost of $O(N_i d^2)$. We use $J = 4$ and $C = 128$, giving $d = 32$.

Guided Feature Fusion. We then combine the two attention outputs with the original tokens. Each output passes through an additional branch-specific projection and is weighted by its guidance:

$$\widetilde{\boldsymbol{Z}}_i = \boldsymbol{Z}_i + \sum_{r\in\{\mathrm{f,b}\}} \ell_{r,i}^{\mathrm{out}}\left(\mathcal{A}_{r,i}(\boldsymbol{Z}_i, \boldsymbol{g}_r^i)\right) \odot \boldsymbol{g}_r^i, \tag{4}$$

where $\ell_{r,i}^{\mathrm{out}}$ denotes the additional linear projection. After restoring the refined tokens to the grid, we concatenate them with the original features and a foreground-weighted feature map:

$$\boldsymbol{A}_i = \boldsymbol{F}_i + \mathcal{C}_1\left(\left[\boldsymbol{F}_i,\ \mathcal{R}^{-1}(\widetilde{\boldsymbol{Z}}_i),\ \boldsymbol{F}_i \odot \boldsymbol{G}_{\mathrm{f}}^i\right]\right). \tag{5}$$

The residual pathway retains the original features alongside the guided context. Next, $\boldsymbol{A}_i$ enters the twin Mamba branches, where the same guidance maps determine spatial scan order.

3.3 Prototype-Guided Twin Mamba

Feature refinement uses region relevance to weight contextual information. We now use that relevance to organize the sequence in which spatial tokens are processed. The foreground and background branches receive the same refined feature $\boldsymbol{A}_i$, but use their respective guidance maps and

independent parameters. Within each branch, spatial modeling is followed by temporal aggregation, allowing cross-frame context to supplement the region-conditioned spatial features.

For both operations, we use a residual sequence block $\mathcal{S}(\boldsymbol{X}) = \boldsymbol{X} + \ell\big(\mathrm{Mamba}(\mathrm{LN}(\boldsymbol{X}))\big)$, where LN denotes layer normalization and ℓ is a learned linear projection. The spatial block and the two temporal blocks have independent parameters.

Prototype-Guided Semantic Scan. For each frame, we first modulate the refined feature with a region-specific residual gate: $\boldsymbol{\Omega}_r^i = \boldsymbol{A}_i \odot \big(1 + \tanh(\alpha_r^i)\boldsymbol{G}_r^i\big)$, where α_r^i is a learnable scalar initialized to 1. We then construct a spatial permutation by sorting the guidance scores in descending order: $\boldsymbol{\pi}_r^i = \mathrm{Argsort}_{\downarrow}(\boldsymbol{g}_r^i)$. Thus, guidance determines both feature modulation and the order of spatial state updates. As illustrated in Figure 4, we gather the flattened tokens according to $\boldsymbol{\pi}_r^i$ and process them with a spatial Mamba block (Gu & Dao, 2023). The output is then returned to the original grid:

$$\boldsymbol{\Theta}_r^i = \mathcal{R}^{-1}\left[\mathcal{P}_{(\boldsymbol{\pi}_r^i)^{-1}}\left(\mathcal{S}_{r,i}^{\mathrm{s}}\left(\mathcal{P}_{\boldsymbol{\pi}_r^i}\left(\mathcal{R}(\boldsymbol{\Omega}_r^i)\right)\right)\right)\right], \tag{6}$$

where $\mathcal{P}_{\boldsymbol{\pi}}$ gathers tokens using permutation $\boldsymbol{\pi}$, and $\mathcal{S}_{r,i}^{\mathrm{s}}$ denotes the spatial block.

Tokens with higher region guidance appear earlier in the sequence, so the two branches can follow different spatial orders. All tokens are retained, and feature channels are not reordered. Before temporal modeling, the inverse permutation restores grid indexing so that features are grouped by location, rather than by frame-specific sorting rank. The discrete sorting indices are not differentiable; gradients still pass through the gathered features, guidance-dependent gating, and MLA. Sorting costs $O(N_i \log N_i)$ per frame and branch.

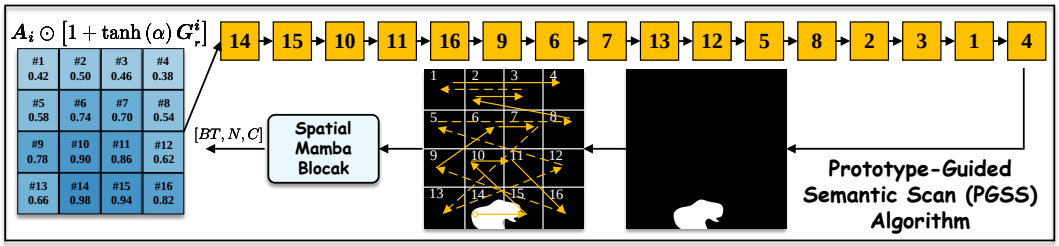


Figure 4: PGSS. Region guidance determines the spatial token order. Gated tokens are gathered, processed by spatial Mamba, and restored to their original grid positions before LBTS.

Location-preserved Bidirectional Temporal Scan. PGSS constructs spatial representations separately for each frame. To complement these representations with neighboring-frame information, we next collect features at each restored grid location over the T frames. This rearranges $\boldsymbol{\Theta}_r^i$ into $\boldsymbol{\mathcal{T}}_r^i \in \mathbb{R}^{(BN_i)\times T\times C}$, where each sequence corresponds to one grid location within a clip.

We process each sequence in chronological and reverse order using independent temporal blocks. The backward output is reversed again before the two directions are combined:

$$\widehat{\boldsymbol{\mathcal{T}}}_r^i = \frac{1}{2}\left[\mathcal{S}_{r,i}^{\rightarrow}(\boldsymbol{\mathcal{T}}_r^i) + \mathcal{J}\left(\mathcal{S}_{r,i}^{\leftarrow}\left(\mathcal{J}(\boldsymbol{\mathcal{T}}_r^i)\right)\right)\right], \tag{7}$$

where $\mathcal{J}$ reverses only the temporal dimension. As shown in Figure 3, this operation incorporates both earlier and later frames within the clip. Finally, we reshape the output to $B \times C \times T \times H_i \times W_i$ and apply a $1 \times 1 \times 1$ convolution, group normalization, and GELU. Rearranging the dimensions gives $\boldsymbol{U}_r^i \in \mathbb{R}^{B\times T\times C\times H_i\times W_i}$. LBTS operates on spatially contextualized features at fixed grid locations; it does not explicitly align moving objects.

Twin-Branch Fusion. We fuse the foreground and background outputs with the refined input to obtain deep features for decoding: $\boldsymbol{E}_i = \mathcal{C}_1\left([\boldsymbol{A}_i, \boldsymbol{U}_{\mathrm{f}}^i, \boldsymbol{U}_{\mathrm{b}}^i]\right), i \in \{3, 4\}$.

3.4 Detail Enhancement and Decoding

The deep branches provide region-conditioned spatial and temporal context. To retain complementary local information, we enhance $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ through a shallow pathway without coarse-mask

Table 1: Quantitative comparison on Easy.

SUN-SEG-Easy-Seen								
Method	Publication	Backbone	$S_\alpha \uparrow$	$E_\phi^m \uparrow$	$F_\beta^w \uparrow$	Dice↑	IoU↑	MAE↓
COSNet	TPAMI 2019	–	0.845	0.836	0.727	0.730	0.648	0.034
PraNet	MICCAI 2020	Res2Net-50	0.918	0.942	0.877	0.883	0.825	0.020
PNS	MICCAI 2021	Res2Net-50	0.906	0.910	0.836	0.841	0.783	0.020
PNS+	MIR 2022	Res2Net-50	0.917	0.925	0.848	0.855	0.787	0.021
SALI	MICCAI 2024	PVTv2-B2	0.902	0.932	0.849	0.858	0.789	0.024
SALI	MICCAI 2024	PVTv2-B5	0.907	0.937	0.851	0.862	0.796	0.022
STDDNet	ICCV 2025	Res2Net-50	0.935	0.960	0.897	0.905	0.850	0.015
Ours	–	Res2Net-50	0.933	0.960	0.899	0.905	0.851	0.016
Ours	–	PVTv2-B2	**0.945**	**0.972**	**0.917**	**0.923**	**0.871**	**0.013**
SUN-SEG-Easy-Unseen								
Method	Publication	Backbone	$S_\alpha \uparrow$	$E_\phi^m \uparrow$	$F_\beta^w \uparrow$	Dice↑	IoU↑	MAE↓
COSNet	TPAMI 2019	–	0.654	0.600	0.431	0.423	0.342	0.073
PraNet	MICCAI 2020	Res2Net-50	0.781	0.788	0.663	0.665	0.582	0.052
PNS	MICCAI 2021	Res2Net-50	0.767	0.744	0.616	0.618	0.545	0.048
PNS+	MIR 2022	Res2Net-50	0.806	0.798	0.676	0.678	0.591	0.044
SALI	MICCAI 2024	PVTv2-B2	0.731	0.752	0.587	0.592	0.502	0.063
SALI	MICCAI 2024	PVTv2-B5	0.771	0.821	0.646	0.656	0.568	0.055
STDDNet	ICCV 2025	Res2Net-50	0.817	0.830	0.721	0.724	0.643	0.037
Ours	–	Res2Net-50	0.834	0.846	0.743	0.749	0.673	0.038
Ours	–	PVTv2-B2	**0.855**	**0.889**	**0.791**	**0.796**	**0.715**	**0.032**

Table 2: Quantitative comparison on Hard.

SUN-SEG-Hard-Seen								
Method	Publication	Backbone	$S_\alpha \uparrow$	$E_\phi^m \uparrow$	$F_\beta^w \uparrow$	Dice↑	IoU↑	MAE↓
COSNet	TPAMI 2019	–	0.785	0.772	0.626	0.633	0.541	0.046
PraNet	MICCAI 2020	Res2Net-50	0.884	0.919	0.831	0.839	0.766	0.031
PNS	MICCAI 2021	Res2Net-50	0.870	0.892	0.787	0.796	0.721	0.033
PNS+	MIR 2022	Res2Net-50	0.887	0.902	0.806	0.813	0.728	0.030
SALI	MICCAI 2024	PVTv2-B2	0.868	0.909	0.799	0.810	0.726	0.034
SALI	MICCAI 2024	PVTv2-B5	0.866	0.910	0.797	0.810	0.729	0.038
STDDNet	ICCV 2025	Res2Net-50	0.913	0.952	0.869	0.881	0.810	**0.023**
Ours	–	Res2Net-50	0.907	0.943	0.862	0.873	0.803	0.026
Ours	–	PVTv2-B2	**0.919**	**0.953**	**0.879**	**0.889**	**0.820**	**0.023**
SUN-SEG-Hard-Unseen								
Method	Publication	Backbone	$S_\alpha \uparrow$	$E_\phi^m \uparrow$	$F_\beta^w \uparrow$	Dice↑	IoU↑	MAE↓
COSNet	TPAMI 2019	–	0.670	0.627	0.443	0.438	0.353	0.070
PraNet	MICCAI 2020	Res2Net-50	0.787	0.802	0.667	0.675	0.587	0.053
PNS	MICCAI 2021	Res2Net-50	0.767	0.755	0.609	0.615	0.539	0.050
PNS+	MIR 2022	Res2Net-50	0.798	0.793	0.654	0.661	0.571	0.050
SALI	MICCAI 2024	PVTv2-B2	0.728	0.759	0.569	0.579	0.487	0.068
SALI	MICCAI 2024	PVTv2-B5	0.765	0.813	0.620	0.636	0.547	0.057
STDDNet	ICCV 2025	Res2Net-50	0.834	0.856	0.741	0.750	0.673	0.037
Ours	–	Res2Net-50	0.841	0.863	0.750	0.762	0.682	0.038
Ours	–	PVTv2-B2	**0.859**	**0.897**	**0.792**	**0.803**	**0.722**	**0.032**

Best values among the listed methods are bold, including ties at three decimals. Shaded rows denote our models.

gating. As shown in Figure 3, each module combines 3×3 and 5×5 depthwise convolutions with a local high-pass residual. Feature fusion and a learnable channel-wise residual scale then produce $\boldsymbol{E}_1$ and $\boldsymbol{E}_2$. A top-down convolutional decoder progressively upsamples and fuses all four enhanced levels, as illustrated in Figure 2. The finest decoder feature produces the final logits, while an auxiliary head at level 3 provides additional supervision. We bilinearly resize the final, auxiliary, and coarse logits to $H \times W$ before applying sigmoid, obtaining $\widehat{\boldsymbol{Y}}$, $\widehat{\boldsymbol{Y}}^{\mathrm{aux}}$, and $\widehat{\boldsymbol{Y}}^{\mathrm{c}}$, respectively. Here, $\boldsymbol{P}_4$ is the low-resolution prior used for prototype generation, whereas $\widehat{\boldsymbol{Y}}^{\mathrm{c}}$ is the input-resolution coarse prediction used for supervision. Detailed module definitions are provided in Appendix A.

3.5 Training Objectives

We supervise the final, auxiliary, and coarse predictions using binary cross-entropy and soft Dice loss (Milletari et al., 2016). The guidance loss $\mathcal{L}_{\mathrm{mask}}$ supervises foreground/background maps and penalizes overlap and non-complementarity. The prototype loss $\mathcal{L}_{\mathrm{proto}}$ promotes consistency across adjacent frames and feature levels while discouraging excessive foreground–background similarity. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{seg}} + \lambda_{\mathrm{aux}} \left(\tfrac{1}{2}\mathcal{L}_{\mathrm{aux}} + \tfrac{1}{4}\mathcal{L}_{\mathrm{coarse}} \right) + \lambda_{\mathrm{proto}}\mathcal{L}_{\mathrm{proto}} + \lambda_{\mathrm{mask}}\mathcal{L}_{\mathrm{mask}}, \tag{8}$$

where $\lambda_{\mathrm{aux}} = 0.4$, $\lambda_{\mathrm{proto}} = 0.1$, and $\lambda_{\mathrm{mask}} = 0.2$. Only valid frames and valid adjacent-frame pairs contribute to the corresponding losses. Ground truth provides supervision but is not used for prototype generation or scan ordering. Details are provided in Appendix B.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate on SUN-SEG (Ji et al., 2022) and CVC-612 (Bernal et al., 2015). For SUN-SEG, we use all 112 training sequences containing 19,544 frames. Results are reported separately on Easy-Seen, Easy-Unseen, Hard-Seen, and Hard-Unseen. For CVC-612, we use a fixed sequence-disjoint split with 363 training frames from 17 sequences, 130 validation frames from 6 sequences, and 119 test frames from 6 sequences. This split is specified by a shared manifest rather than assumed to match the splits used in previous publications.

Evaluation Metrics and Implementation Details. We use the structure measure S_α (Fan et al., 2017), mean enhanced-alignment measure E_ϕ^{mean} (Fan et al., 2021), weighted F-measure F_β^w (Margolin et al., 2014), meanDice, meanIoU, and MAE for evaluation. We instantiate ProtoTwinMamba with ImageNet-pretrained Res2Net-50 (Gao et al., 2019) and PVTv2-B2 (Wang et al., 2022). For SUN-SEG, inputs contain 5 frames resized to 352×352. We train for 30 epochs on two RTX 3090

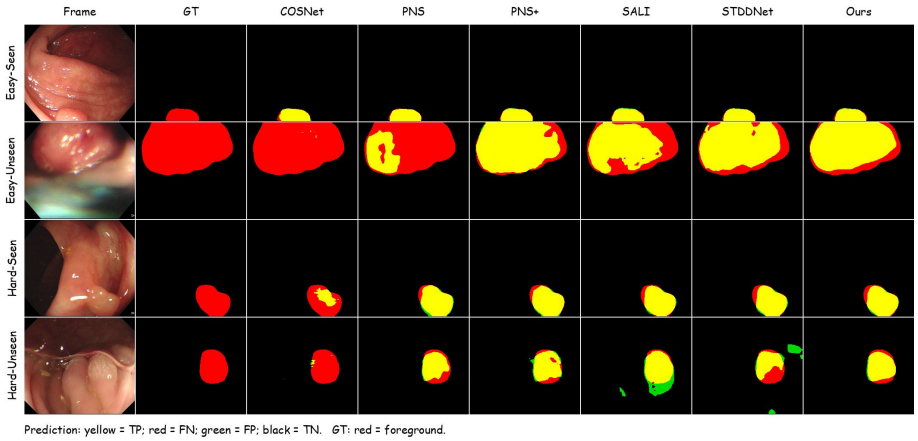


Figure 5: Qualitative comparison on four SUN-SEG subsets. Yellow, red, and green indicate true positives, false negatives, and false positives, respectively. GT foreground regions are shown in red.

Table 3: Performance comparison on CVC-612.

Method	$S_\alpha \uparrow$	$E_\phi^m \uparrow$	$F_\beta^w \uparrow$	Dice↑	IoU↑	MAE↓
STDDNet	0.737	0.833	0.407	0.716	0.605	0.081
Ours	**0.787**	**0.855**	**0.540**	**0.726**	**0.628**	**0.047**

Table 4: Model complexity comparison.

Method	Backbone	Params (M)↓	GFLOPs↓
SALI	PVTv2-B5	82.730	86.580
STDDNet	Res2Net-50	38.160	75.170
Ours	Res2Net-50	**27.930**	35.725
Ours	PVTv2-B2	28.750	**33.748**

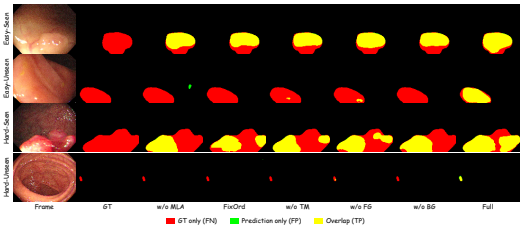


Figure 6: Some module-ablation examples on SUN-SEG.

GPUs using AdamW, with an initial learning rate of 10^{-4} for both the backbone and other parameters and weight decay of 10^{-4}. A three-epoch warmup is followed by cosine decay. The per-GPU batch sizes are 10 and 24 for Res2Net-50 and PVTv2-B2, respectively.

4.2 Comparison with Existing Methods

We compare against COSNet (Lu et al., 2019), PraNet (Fan et al., 2020), PNS (Ji et al., 2021), PNS+ (Ji et al., 2022), SALI (Hu et al., 2024), and STDDNet (Chen et al., 2025) for SUN-SEG. For CVC-612, only methods trained under the shared split are included in the direct comparison. Baseline results in Tables 1 and 2 are taken from existing publications.

Quantitative Comparisons. Tables 1 and 2 summarize results on SUN-SEG. With PVTv2-B2, ProtoTwinMamba achieves the best scores among the listed methods across all four subsets and six metrics, including a tied MAE on Hard-Seen. Compared with the reported SALI results using the same backbone, Dice improves by 20.4 and 22.4 percentage points on Easy-Unseen and Hard-Unseen, respectively. With Res2Net-50, the corresponding gains over STDDNet are 2.5 and 1.2 points, showing consistent improvements on unseen sequences across both backbones. On CVC-612 (Table 3), our method outperforms STDDNet in all metrics.

Qualitative Comparisons. Figure 5 shows visual comparisons across the four SUN-SEG subsets. In the Easy-Unseen example, ProtoTwinMamba produces a largely continuous foreground mask, recovering regions missed by several baselines and avoiding interior gaps. In the Hard-Unseen example, our method suppresses the background false positives observed in SALI and STDDNet while preserving most of the polyp region. These examples illustrate its ability to recover polyp regions and distinguish them from surrounding tissue under challenging appearances.

Complexity. Table 4 compares model complexity. Our Res2Net-50 and PVTv2-B2 variants contain 27.930M and 28.750M parameters, respectively, fewer than the reported SALI and STDDNet

Table 5: Component ablations on SUN-SEG with Res2Net-50.

Variant	Easy-Seen		Easy-Unseen		Hard-Seen		Hard-Unseen	
	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑
Full	**0.905**	**0.851**	**0.749**	**0.673**	**0.873**	**0.803**	**0.762**	0.682
FixOrd	0.902	0.847	0.723	0.647	0.865	0.794	0.740	0.660
w/o TM	0.903	0.848	0.729	0.651	0.866	0.797	0.732	0.652
w/o MLA	0.904	**0.851**	0.717	0.643	0.863	0.792	0.744	0.665
w/o FG	0.901	0.850	0.731	0.654	0.870	0.801	0.761	**0.685**
w/o BG	0.900	0.844	0.732	0.656	0.868	0.798	0.734	0.655

Table 6: Temporal design ablations on SUN-SEG with Res2Net-50.

Variant	Easy-Seen		Easy-Unseen		Hard-Seen		Hard-Unseen	
	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑
Full	**0.905**	**0.851**	**0.749**	**0.673**	**0.873**	**0.803**	**0.762**	**0.682**
Uni-TM	0.903	0.847	0.733	0.655	0.863	0.791	0.750	0.670
No-L3	0.901	0.847	0.705	0.630	0.866	0.796	0.713	0.637

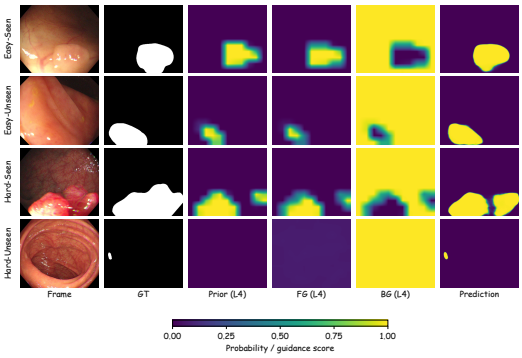


Figure 7: Prototype guidance at level 4. Columns show frames, GT, coarse priors, foreground guidance, background guidance, and final predictions. Heatmaps share a $[0, 1]$ scale without per-image normalization.

models. Their estimated amortized costs are 35.725 and 33.748 GFLOPs per frame. These results suggest a favorable balance between segmentation performance and model complexity.

4.3 Ablation Studies

Effectiveness of Components. Table 5 reports ablations with Res2Net-50. FixOrd denotes fixed spatial ordering; TM, MLA, FG, and BG denote temporal modeling, masked linear attention, foreground Mamba, and background Mamba, respectively. Replacing guided ordering with FixOrd reduces Dice by 2.6 and 2.2 percentage points on Easy-Unseen and Hard-Unseen. Removing TM causes corresponding drops of 2.0 and 3.0 points, supporting the combination of region-guided spatial processing and temporal context. Removing MLA decreases Dice by 3.2 and 1.8 points, demonstrating the benefit of guided feature refinement. Removing BG yields drops of 1.7 and 2.8 points, whereas removing FG produces smaller drops of 1.8 and 0.1 points, with a slight improvement in Hard-Unseen IoU. Figure 6 presents some examples where the full model recovers polyp regions missed by ablated variants and produces more complete foreground masks, illustrating the benefits of combining region-guided spatial processing with temporal context. Actual scan orders are visualized in Appendix C.4. Failure cases and limitations are discussed in Appendix C.5.

Effectiveness of Temporal Design. Table 6 compares the full model with unidirectional temporal modeling (Uni-TM) and the No-L3 variant. Uni-TM reduces Dice by 1.6 and 1.2 percentage points on Easy-Unseen and Hard-Unseen, respectively, supporting bidirectional temporal aggregation. No-L3 yields larger drops of 4.4 and 4.9 points, highlighting the contribution of the level-3 processing pathway. Further ablations are provided in Appendix C.

Visualization of Prototype Guidance. Figure 7 visualizes the guidance at level 4. Foreground and background maps exhibit distinct regional responses, while prototype guidance adjusts the coarse priors in strength and spatial distribution. These maps provide soft cues rather than complete masks.

5 Conclusion

We presented ProtoTwinMamba, a video polyp segmentation framework that uses prototype guidance to couple feature refinement with spatial scan ordering. Its foreground and background branches construct region-conditioned spatial representations, restore their original grid positions, and aggregate neighboring-frame context through bidirectional temporal modeling. A shallow detail-enhancement pathway and a convolutional decoder further integrate local detail with the deep spatial–temporal representations. Experiments on SUN-SEG and CVC-612 demonstrate the effectiveness of this design. Future work will explore motion-aware temporal modeling and more robust segmentation of small polyps under challenging appearance changes.

AI USE STATEMENT

In this work, we used generative AI tools for language translation and polishing. We did not use generative AI tools for research idea generation, methodology design, experiment implementation, data analysis, or result interpretation. All scientific content was produced by the authors. We have reviewed all AI-assisted text and take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies VPS using existing public research datasets, SUN-SEG and CVC-612, without recruiting participants or collecting new patient data. Use of these datasets should follow their applicable access conditions and privacy requirements. The proposed method is intended for research and has not been validated for clinical deployment. Segmentation errors, including missed polyps and false positives, may affect downstream interpretation; model outputs should therefore not be treated as clinical diagnoses. Performance on the evaluated datasets does not establish reliability across patient populations, institutions, or imaging devices. Further external validation and appropriate clinical oversight would be required before clinical use.

REPRODUCIBILITY STATEMENT

The Method section describes the model architecture and training objective, with detailed loss definitions provided in Appendix B. The experimental description and supplementary materials should be consulted for dataset splits, preprocessing, training settings, checkpoint selection, and evaluation procedures. In particular, reproducing the reported results requires matching the evaluated frames, prediction resizing, metric definitions, and aggregation rules.

REFERENCES

Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Debora Gil, Cristina Rodríguez, and Fernando Vilariño. Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized medical imaging and graphics*, 43:99–111, 2015.

Geng Chen, Junqing Yang, Xiaozhou Pu, Ge-Peng Ji, Huan Xiong, Yongsheng Pan, Hengfei Cui, and Yong Xia. Mast: Video polyp segmentation with a mixture-attention siamese transformer. *arXiv preprint arXiv:2401.12439*, 2024.

Guilian Chen, Huisi Wu, and Jing Qin. Stddnet: Harnessing mamba for video polyp segmentation via spatial-aligned temporal modeling and discriminative dynamic representation learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21364–21373. IEEE, 2025.

Guilian Chen, Xiaoling Luo, Huisi Wu, and Jing Qin. Vpsentry: Semi-supervised video polyp segmentation via sentry-guided long-term prototype fusion with correlation dynamic propagation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2850–2858, 2026.

Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4558–4567. IEEE, 2017.

Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 263–273. Springer, 2020.

Deng-Ping Fan, Ge-Peng Ji, Xuebin Qin, and Ming-Ming Cheng. Cognitive vision inspired object segmentation metric and loss function. *Scientia Sinica Informationis*, 6(6):5, 2021.

Zhixue Fang, Xinrong Guo, Jingyin Lin, Huisi Wu, and Jing Qin. An embedding-unleashing video polyp segmentation framework via region linking and scale alignment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 1744–1752, 2024a.

Zhixue Fang, Yuzhi Liu, Huisi Wu, and Jin Qin. Vp-sam: Taming segment anything model for video polyp segmentation via disentanglement and spatio-temporal side network. In *European conference on computer vision*, pp. 367–383. Springer, 2024b.

Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):652–662, 2019.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Guangqian Guo, Pengfei Chen, Yong Guo, Huafeng Chen, Boqiang Zhang, and Shan Gao. Boosting segment anything model to generalize visually non-salient scenarios. *IEEE Transactions on Image Processing*, 35:1143–1157, 2026.

Yuanqin He, Guilian Chen, Yuhua Zhang, Huisi Wu, and Jing Qin. Hfsti-net: Hierarchical frequency-spatial-temporal interactions for video polyp segmentation. In *The Fourteenth International Conference on Learning Representations*, 2026.

Qiang Hu, Zhenyu Yi, Ying Zhou, Fang Peng, Mei Liu, Qiang Li, and Zhiwei Wang. Sali: Short-term alignment and long-term interaction network for colonoscopy video polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 531–541. Springer, 2024.

Debesh Jha, Nikhil Kumar Tomar, Sharib Ali, Michael A Riegler, Håvard D Johansen, Dag Johansen, Thomas De Lange, and Pål Halvorsen. Nanonet: Real-time polyp segmentation in video capsule endoscopy and colonoscopy. In *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*, pp. 37–43. IEEE, 2021.

Ge-Peng Ji, Yu-Cheng Chou, Deng-Ping Fan, Geng Chen, Huazhu Fu, Debesh Jha, and Ling Shao. Progressively normalized self-attention network for video polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 142–152. Springer, 2021.

Ge-Peng Ji, Guobao Xiao, Yu-Cheng Chou, Deng-Ping Fan, Kai Zhao, Geng Chen, and Luc Van Gool. Video polyp segmentation: A deep learning perspective. *Machine Intelligence Research*, 19(6):531–549, 2022.

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.

Yahang Leng, Shiquan Min, and Chengzhou Li. Fsd-net: Foundation-guided spatiotemporal distillation for video polyp segmentation. In *European Conference on Computer Vision*, pp. 562–579. Springer, 2026.

Hao Li and Anum Masood. Hat-sam3: endoscopy-aware adaptation of a foundation segmentation model for generalizable polyp segmentation. *The Visual Computer*, 42(9):371, 2026.

Xiao Liu, Chenxu Zhang, Fuxiang Huang, Shuyin Xia, Guoyin Wang, and Lei Zhang. Vision mamba: A comprehensive survey and taxonomy. *IEEE Transactions on Neural Networks and Learning Systems*, 37(2):505–525, 2026.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2024.

Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3618–3627. IEEE, 2019.

Yingling Lu, Yijun Yang, Zhaohu Xing, Qiong Wang, and Lei Zhu. Diff-vps: Video polyp segmentation via a multi-task diffusion network with adversarial temporal reasoning. In *International conference on medical image computing and computer-assisted intervention*, pp. 165–175. Springer, 2024.

Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 248–255, 2014.

Jiaxin Mei, Tao Zhou, Kaiwen Huang, Yizhe Zhang, Yi Zhou, Ye Wu, and Huazhu Fu. A survey on deep learning for polyp segmentation: Techniques, challenges and future trends. *Visual Intelligence*, 3(1):1, 2025.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pp. 565–571. Ieee, 2016.

Abdul Qayoom, Juanying Xie, and Haider Ali. Polyp segmentation in medical imaging: challenges, approaches and future directions. *Artificial Intelligence Review*, 58(6):169, 2025.

Hao Shao, Yang Zhang, and Qibin Hou. Polyper: Boundary sensitive polyp segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 4731–4739, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Tong Wang, Yaolei Qi, Siwen Wang, Imran Razzak, Guanyu Yang, and Yutong Xie. Cmsa-net: Causal multi-scale aggregation with adaptive multi-source reference for video polyp segmentation. *arXiv preprint arXiv:2602.22821*, 2026.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational visual media*, 8(3):415–424, 2022.

Huisi Wu and Zebin Zhao. Epsegnet: Lightweight semantic recalibration and assembly for efficient polyp segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 36(8):13805–13817, 2025.

Huisi Wul and Jin Qin. Fst-sam3: Taming sam 3 with frequency-spatio-temporal refinement for video polyp segmentation. In *Computer Vision–ECCV 2026: 19th European Conference, Malmö, Sweden, September 8–12, 2026, Proceedings, Part LXXV*, pp. 231. Springer Nature, 2026.

Ye Zhang, Zijie Fang, Yifeng Wang, Lingbo Zhang, Xianchao Guan, and Yongbing Zhang. Category prompt mamba network for nuclei segmentation and classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10284–10292, 2025.

Yuhua Zhang, Guilian Chen, Yuanqin He, Huisi Wu, and Jing Qin. Wavepolyp: Video polyp segmentation via hierarchical wavelet-based feature aggregation and inter-frame divergence perception. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xinkai Zhao, Zhenhua Wu, Shuangyi Tan, De-Jun Fan, Zhen Li, Xiang Wan, and Guanbin Li. Semi-supervised spatial temporal attention network for video polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 456–466. Springer, 2022.

Yifei Zhao, Xiaoying Wang, and Junping Yin. Efficient video polyp segmentation by deformable alignment and local attention. *IEEE Journal of Biomedical and Health Informatics*, 30(2):1534–1543, 2026.

A Detail Enhancement and Decoding

The deep branches organize spatial features by region relevance and supplement them with temporal context. In parallel, the shallow pathway preserves local details without coarse-mask gating. This appendix describes the detail-enhancement modules and the convolutional decoder that integrates both pathways.

A.1 Fine-Grained Detail Enhancement

As illustrated in Figure 3, we enhance the shallow features at levels $i \in \{1, 2\}$ independently of the coarse prior. Specifically, we first apply group normalization: $\boldsymbol{X}_i = \mathrm{GN}(\boldsymbol{F}_i)$. Two depthwise convolutions with 3×3 and 5×5 kernels then extract local features at different spatial scales. In parallel, we subtract a local average to obtain a high-pass residual:

$$\boldsymbol{H}_i^{\mathrm{detail}} = \boldsymbol{X}_i - \mathrm{AvgPool}_3(\boldsymbol{X}_i), \tag{9}$$

where $\mathrm{AvgPool}_3$ uses a 3×3 window with unit stride and excludes padded entries from the average.

We concatenate the two convolution outputs and the high-pass residual, fuse them through $\mathcal{C}_1$, and add the scaled result to the original feature:

$$\boldsymbol{E}_i = \boldsymbol{F}_i + \boldsymbol{\gamma}_i \odot \mathcal{C}_1\left(\left[\mathrm{DWConv}_3(\boldsymbol{X}_i),\, \mathrm{DWConv}_5(\boldsymbol{X}_i),\, \boldsymbol{H}_i^{\mathrm{detail}}\right]\right). \tag{10}$$

The learnable channel-wise scale $\boldsymbol{\gamma}_i \in \mathbb{R}^C$ is initialized to 0.1 and broadcast over the remaining dimensions. It controls the contribution of the added detail features. The resulting $\boldsymbol{E}_1$ and $\boldsymbol{E}_2$ complement the region-conditioned deep representations.

A.2 Top-Down Decoding

The decoder integrates $\{\boldsymbol{E}_1, \boldsymbol{E}_2, \boldsymbol{E}_3, \boldsymbol{E}_4\}$, as shown in Figure 2. Starting from $\boldsymbol{D}_4 = \boldsymbol{E}_4$, we progressively resize the deeper feature, concatenate it with the feature at the next level, and apply two convolutional blocks:

$$\boldsymbol{D}_i = \mathcal{C}_{3,i}^{(2)}\left(\mathcal{C}_{3,i}^{(1)}\left([\mathrm{Up}_i(\boldsymbol{D}_{i+1}),\, \boldsymbol{E}_i]\right)\right), \qquad i = 3, 2, 1, \tag{11}$$

where Up_i denotes bilinear resizing to (H_i, W_i). Each $\mathcal{C}_{3,i}^{(k)}$ consists of a 3×3 convolution, group normalization, and GELU.

After the final fusion stage, another 3×3 convolutional block reduces the channel dimension of $\boldsymbol{D}_1$ from C to $C/2$. A 1×1 head produces the final logits. An auxiliary 1×1 head attached to $\boldsymbol{D}_3$ provides additional supervision during training.

For the final, auxiliary, and coarse predictions, we bilinearly resize the logits to $H \times W$ before applying sigmoid. The resulting probability maps are denoted by $\widehat{\boldsymbol{Y}}$, $\widehat{\boldsymbol{Y}}^{\mathrm{aux}}$, and $\widehat{\boldsymbol{Y}}^{\mathrm{c}}$, respectively. Both $\boldsymbol{P}_4$ and $\widehat{\boldsymbol{Y}}^{\mathrm{c}}$ are derived from the coarse logits $\boldsymbol{L}^{\mathrm{c}}$. The former applies sigmoid at the native feature resolution for prototype generation, whereas the latter applies sigmoid after resizing the logits to the input resolution for supervision.

B Detailed Training Objectives

This appendix details the objectives in Eq. equation 8. Segmentation losses supervise the predictions, guidance supervision associates the two branches with their respective regions, and prototype regularization encourages consistent yet distinct region representations.

Let $\boldsymbol{Y}$ denote the binary ground-truth masks. Unless otherwise specified, averages use valid frames, and temporal constraints use only valid adjacent-frame pairs. We denote averaging over the indicated elements by $\langle\cdot\rangle$ and use $\epsilon = 10^{-6}$ for numerical stability.

B.1 Segmentation Supervision

We supervise the final, auxiliary, and coarse predictions using binary cross-entropy (BCE) and soft Dice loss. For a predicted probability map $\widehat{\boldsymbol{Y}}$, the segmentation loss is

$$\begin{aligned}\mathcal{B}(\widehat{\boldsymbol{Y}}, \boldsymbol{Y}) = \ & \mathrm{BCE}(\widehat{\boldsymbol{Y}}, \boldsymbol{Y}) \\ & + \left\langle 1 - \frac{2\sum_n \widehat{y}_n y_n + \epsilon}{\sum_n \widehat{y}_n + \sum_n y_n + \epsilon} \right\rangle_{\text{frames}},\end{aligned} \tag{12}$$

where n indexes spatial positions within a frame. BCE is averaged over pixels in valid frames, while the Dice term is computed for each valid frame and then averaged. For prediction heads, BCE is evaluated directly from logits for numerical stability.

Applying this objective to the three predictions gives

$$\begin{aligned}\mathcal{L}_{\text{seg}} &= \mathcal{B}(\widehat{\boldsymbol{Y}}, \boldsymbol{Y}), \\ \mathcal{L}_{\text{aux}} &= \mathcal{B}(\widehat{\boldsymbol{Y}}^{\text{aux}}, \boldsymbol{Y}), \\ \mathcal{L}_{\text{coarse}} &= \mathcal{B}(\widehat{\boldsymbol{Y}}^{\text{c}}, \boldsymbol{Y}).\end{aligned} \tag{13}$$

For each head, logits are bilinearly resized to the input resolution before computing the loss. The probability maps in Eq. equation 13 are obtained by applying sigmoid to these resized logits.

B.2 Guidance Supervision

The guidance maps are shared by MLA, residual gating, and PGSS. We supervise these maps to associate their scores with foreground and background regions. At levels $i \in \{3, 4\}$, the respective targets are $\boldsymbol{Y}_i$ and $1 - \boldsymbol{Y}_i$, where $\boldsymbol{Y}_i$ is obtained by nearest-neighbor resizing of $\boldsymbol{Y}$ to (H_i, W_i).

Before computing the loss, we clip the guidance values for numerical stability:

$$\overline{\boldsymbol{G}}_r^{\,i} = \mathrm{Clip}\left(\boldsymbol{G}_r^i, \delta, 1-\delta\right), \qquad \delta = 10^{-4}, \quad r \in \{\mathrm{f}, \mathrm{b}\}. \tag{14}$$

In addition to mask supervision, we penalize overlap between the two guidance maps and deviations of their sum from one:

$$\begin{aligned}\mathcal{O}_i &= \left\langle \overline{\boldsymbol{G}}_{\mathrm{f}}^{\,i} \odot \overline{\boldsymbol{G}}_{\mathrm{b}}^{\,i} \right\rangle_{\text{pixels}}, \\ \mathcal{Q}_i &= \left\langle \left| \overline{\boldsymbol{G}}_{\mathrm{f}}^{\,i} + \overline{\boldsymbol{G}}_{\mathrm{b}}^{\,i} - 1 \right| \right\rangle_{\text{pixels}}.\end{aligned} \tag{15}$$

Here, pixel averages include only valid frames. These penalties encourage complementarity during training without imposing it exactly in the guidance construction.

The complete guidance loss is

$$\begin{aligned}\mathcal{L}_{\text{mask}} = \frac{1}{2} \sum_{i \in \{3,4\}} \Bigg\{ & \frac{1}{4}\Big[\mathcal{B}\left(\overline{\boldsymbol{G}}_{\mathrm{f}}^{\,i}, \boldsymbol{Y}_i\right) + \mathcal{B}\left(\overline{\boldsymbol{G}}_{\mathrm{b}}^{\,i}, 1 - \boldsymbol{Y}_i\right)\Big] \\ & + \frac{\lambda_{\text{part}}}{2}\left(\mathcal{O}_i + \mathcal{Q}_i\right) \Bigg\},\end{aligned} \tag{16}$$

where $\lambda_{\text{part}} = 0.25$. The outer factor averages the losses from the two feature levels. This objective supervises the guidance values, not the discrete sorting indices used in PGSS.

B.3 Prototype Regularization

We regularize the prototypes through temporal consistency, foreground–background separation, and cross-level consistency. These constraints encourage each region's representation to remain consistent across observations while discouraging excessive similarity between the two regions.

Let $\boldsymbol{p}_{r,t}^{\,i}$ denote the normalized prototype for region r, frame t, and level i. For each clip, we compute a normalized mean prototype:

$$\overline{\boldsymbol{p}}_r^{\,i} = \mathrm{n}_2\left(\left\langle \boldsymbol{p}_{r,t}^{\,i} \right\rangle_{\text{valid } t}\right). \tag{17}$$

Batch indices are omitted for readability.

Temporal Consistency. We encourage prototypes from adjacent valid frames to remain consistent by minimizing their cosine distance:

$$a_{r,i} = \left\langle 1 - (\boldsymbol{p}^i_{r,t})^\top \boldsymbol{p}^i_{r,t-1} \right\rangle_{\text{valid pairs}}. \tag{18}$$

The average includes adjacent pairs for which both frames are valid, across the batch.

Foreground–Background Separation. To discourage similar foreground and background representations, we penalize clip-level cosine similarity above a margin:

$$b_i = \left\langle \left[(\overline{\boldsymbol{p}}^i_{\text{f}})^\top \overline{\boldsymbol{p}}^i_{\text{b}} - m \right]_+ \right\rangle_{\text{clips}}, \tag{19}$$

where $[x]_+ = \max(x, 0)$ and $m = 0.2$. This term discourages excessive similarity without requiring the two prototypes to be orthogonal.

Cross-Level Consistency. We also encourage prototypes for the same region to agree across the two deep feature levels:

$$c_r = \left\langle 1 - (\overline{\boldsymbol{p}}^3_r)^\top \overline{\boldsymbol{p}}^4_r \right\rangle_{\text{clips}}. \tag{20}$$

Combined Prototype Loss. When valid adjacent-frame pairs are available, we average the four temporal terms, two separation terms, and two cross-level terms:

$$\mathcal{L}_{\text{proto}} = \frac{1}{8} \left[\sum_{i \in \{3,4\}} (a_{\text{f},i} + a_{\text{b},i} + b_i) + c_{\text{f}} + c_{\text{b}} \right]. \tag{21}$$

If no valid adjacent-frame pair exists in the batch, we omit the temporal terms and average the four remaining terms:

$$\mathcal{L}_{\text{proto}} = \frac{1}{4} \left(b_3 + b_4 + c_{\text{f}} + c_{\text{b}} \right). \tag{22}$$

Prototype regularization constrains relationships among predicted prototypes rather than matching them to ground-truth-derived prototype targets. Ground truth supervises the segmentation outputs and guidance maps; it is not an input to prototype generation or scan ordering. All losses are combined using the weights specified in Eq. equation 8.

C Additional Experiments and Analyses

C.1 Additional Component Ablations

Table 7 evaluates fine-grained detail enhancement (FDE), the complete prototype-guided module, and the joint foreground/background Mamba branches.

Effectiveness of Detail Enhancement. Removing FDE reduces Dice across all four subsets, with the largest drop of 0.6 percentage points on Hard-Seen. The corresponding IoU decreases from 0.803 to 0.796, supporting the contribution of shallow detail features.

Effectiveness of the Prototype-Guided Module. As shown in Table 7, removing the prototype module reduces Dice by 0.6, 0.9, 0.9, and 0.3 percentage points on Easy-Seen, Easy-Unseen, Hard-Seen, and Hard-Unseen, respectively. IoU also decreases across all four subsets, supporting the contribution of prototype guidance to segmentation on both seen and unseen sequences.

Effectiveness of Joint Foreground–Background Modeling. Removing both Mamba branches reduces Dice by 2.3 and 1.3 percentage points on Easy-Unseen and Hard-Unseen, respectively, with corresponding IoU drops of 2.3 and 1.1 points. These results support the joint contribution of the region-conditioned branches, particularly on unseen sequences.

Table 7: Additional component ablations on SUN-SEG with Res2Net-50.

Variant	Easy-Seen		Easy-Unseen		Hard-Seen		Hard-Unseen	
	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑
Full	**0.905**	**0.851**	**0.749**	**0.673**	**0.873**	**0.803**	**0.762**	**0.682**
w/o FDE	0.901	0.850	0.744	0.665	0.867	0.796	0.761	0.681
w/o Proto Module	0.899	0.845	0.740	0.664	0.864	0.789	0.759	0.681
w/o FG+BG	0.902	0.849	0.726	0.650	0.869	0.799	0.749	0.671

Table 8: Supervision ablations on SUN-SEG with Res2Net-50.

Variant	Easy-Seen		Easy-Unseen		Hard-Seen		Hard-Unseen	
	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑	Dice↑	IoU↑
Full	**0.905**	**0.851**	**0.749**	**0.673**	**0.873**	**0.803**	**0.762**	**0.682**
w/o Proto Loss	0.903	0.847	0.735	0.656	0.858	0.786	0.760	0.679
w/o Guidance Loss	0.901	**0.851**	0.745	0.671	0.872	0.801	0.758	0.674
FG-only Guidance Loss	0.898	0.841	0.710	0.632	0.859	0.787	0.727	0.645

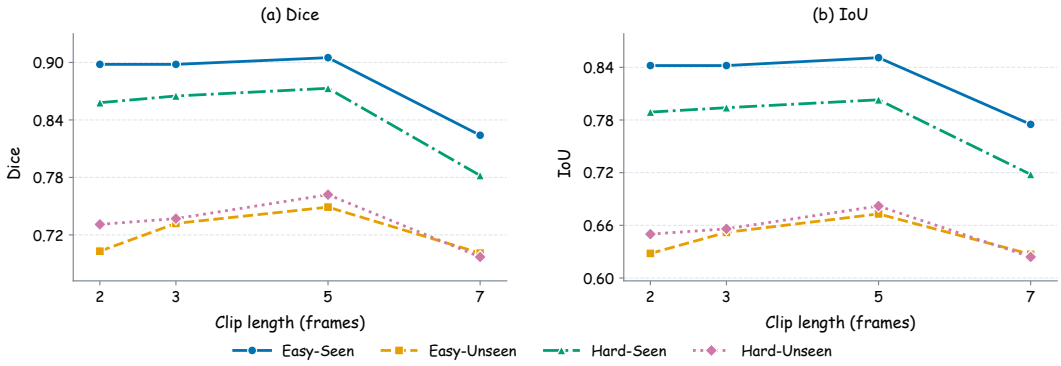


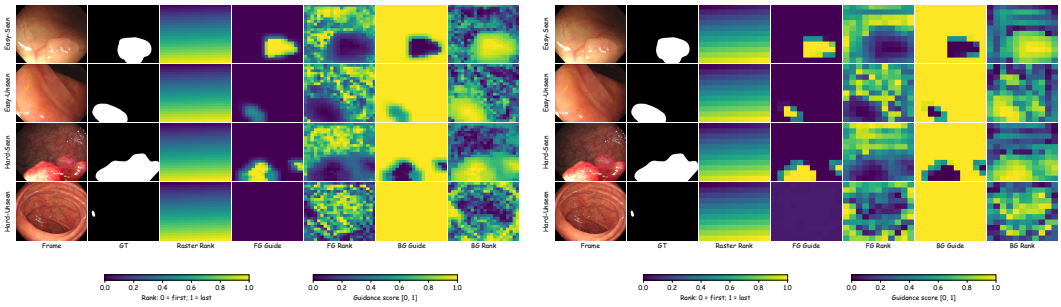
Figure 8: Effect of clip length on SUN-SEG with Res2Net-50. Dice and IoU are reported on all four test subsets. Five-frame clips achieve the best performance among the evaluated settings.

C.2 Ablations of Supervision Terms

Table 8 evaluates prototype regularization and guidance supervision. Removing the prototype loss reduces Dice by 1.4 and 1.5 percentage points on Easy-Unseen and Hard-Seen, respectively, supporting its contribution to segmentation. Removing the guidance loss produces smaller but consistent Dice drops across all four subsets. Replacing joint guidance supervision with FG-only guidance reduces Dice by 3.9 and 3.5 points on the two unseen subsets. This variant also removes the overlap and complementarity penalties and changes the relative foreground weighting; thus, the results support the joint supervision design rather than isolating the effect of background supervision.

C.3 Effectiveness of Clip Length.

Figure 8 compares clip lengths of $T \in \{2, 3, 5, 7\}$. Performance generally improves from two to five frames, with $T = 5$ achieving the highest Dice and IoU across all four subsets. Extending the clip to seven frames reduces performance, showing that longer temporal context does not necessarily yield better segmentation. These results support our default choice of $T = 5$.



(a) Level 3. (b) Level 4.

Figure 9: Guided spatial ordering on selected SUN-SEG examples. Raster ranks provide a fixed-order reference. FG/BG ranks show the actual scan positions mapped back to the spatial grid, normalized from 0 (first) to 1 (last). Higher guidance scores correspond to earlier scan positions.

C.4 Visualization of Guided Spatial Ordering

Figure 9 visualizes the actual spatial scan ranks at levels 3 and 4. Unlike raster traversal, guided ordering places tokens with higher foreground or background relevance earlier in the corresponding branch. In the first three examples, polyp-associated regions generally appear earlier in the foreground scan and later in the background scan. This illustrates how the two branches organize spatial information differently before temporal aggregation. For the small target in the last row, guidance is nearly uniform, so rank variations should not be interpreted as strong semantic distinctions. The visualizations describe the ordering mechanism; its effectiveness is evaluated by the FixOrd ablation in Table 5.

C.5 Failure Cases and Limitations

Failure Cases. Figure 10 presents low-Dice examples selected from the four SUN-SEG subsets. In the first, second, and fourth rows, the model misses the annotated polyp while predicting a different region. The third example shows an almost complete miss despite the large target size. These cases reveal limitations in target localization rather than only boundary delineation. Uneven illumination, specular highlights, and ambiguous appearance may contribute to these errors, although individual images cannot establish their causes. Since the examples are selected for low performance, they illustrate failure modes rather than average behavior.

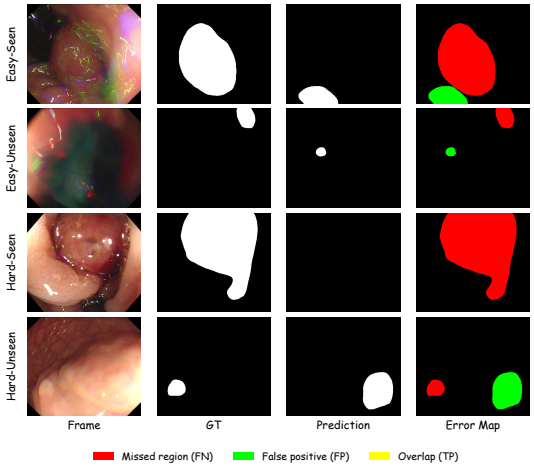


Figure 10: Several failure cases on SUN-SEG. Rows correspond to Easy-Seen, Easy-Unseen, Hard-Seen, and Hard-Unseen. Predictions are binarized at a threshold of 0.5. Red, green, and yellow indicate false negatives, false positives, and true positives, respectively.

Limitations. Our region guidance depends partly on a predicted coarse prior; an inaccurate prior may provide misleading cues for prototype construction and spatial ordering. Furthermore, restoring grid indices does not explicitly align moving anatomical regions across frames. Large motion may therefore limit temporal aggregation. The centered bidirectional configuration also uses future frames and is not strictly causal. Future work will investigate more robust regional guidance and motion-aware, causal temporal modeling.