# Unlocking Generalization in Polyp Segmentation with DINO Self-Attention "keys"

**Carla Monteiro**[*1,2,3,5] CARLA.S.MONTEIRO@INESCTEC.PT
**Valentina Corbetta**[*2,3,4] V.CORBETTA@NKI.NL
**Regina Beets-Tan**[3,4] R.BEETSTAN@NKI.NL
**Luís F. Teixeira**[1,5] LUISFT@FE.UP.PT
**Wilson Silva**[2,3] W.J.DOSSANTOSSILVA@UU.NL

[1] *Faculty of Engineering, University of Porto*
[2] *AI Technology for Life, Utrecht University*
[3] *Department of Radiology, The Netherlands Cancer Institute*
[4] *GROW Research Institute for Oncology and Reproduction, Maastricht University*
[5] *INESC TEC*

**Editors:** Under Review for MIDL 2026

## Abstract

Automatic polyp segmentation is crucial for improving the clinical identification of colorectal cancer (CRC). While Deep Learning (DL) techniques have been extensively researched for this problem, current methods frequently struggle with generalization, particularly in data-constrained or challenging settings. Moreover, many existing polyp segmentation methods rely on complex, task-specific architectures. To address these limitations, we present a framework that leverages the intrinsic robustness of DINO self-attention "key" features for robust segmentation. Unlike traditional methods that extract tokens from the deepest layers of the Vision Transformer (ViT), our approach leverages the key features of the self-attention module with a simple convolutional decoder to predict polyp masks, resulting in enhanced performance and better generalizability. We validate our approach using a multi-center dataset under two rigorous protocols: Domain Generalization (DG) and Extreme Single Domain Generalization (ESDG). Our results, supported by a comprehensive statistical analysis, demonstrate that this pipeline achieves state-of-the-art (SOTA) performance, significantly enhancing generalization, particularly in data-scarce and challenging scenarios. While avoiding a polyp-specific architecture, we surpass well-established models like nnU-Net and UM-Net. Additionally, we provide a systematic benchmark of the DINO framework's evolution, quantifying the specific impact of architectural advancements on downstream polyp segmentation performance.

**Keywords:** representation learning, polyp segmentation, self supervised learning, generalizability, vision transformers

## 1. Introduction

Colorectal Cancer (CRC) is the third most common cancer type and the second leading cause of cancer-related deaths. Around 95% of cases arise from the development of polyps in the colon or rectum, which, if undetected, can lead to cancer (Gupta and Mishra, 2024).

[*] Equal contribution
Code is available at: https://github.com/Trustworthy-AI-UU-NKI/Unlocking-Generalization-in-Polyp-Segmentation-with-DINO-Self-Attention-keys-.git

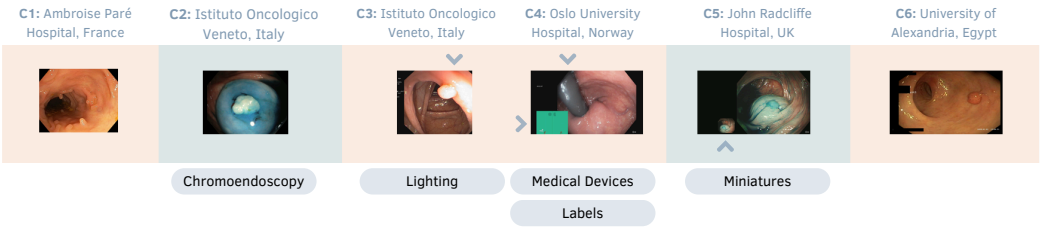


Figure 1: Illustration of the variability of PolypGen (Ali et al., 2023) samples across different centers. Blue highlights chromoendoscopy samples, with arrows indicating other segmentation challenges, such as varying lighting and device-related labels.

Colonoscopy remains the gold standard for polyp screening. However, polyps often resemble normal tissue, making detection particularly difficult, especially for less experienced operators. Missed or inconsistent detections can also result from operator-dependent factors (e.g. colonoscope speed) (Wu et al., 2024). Several studies have focused on automatic polyp segmentation to improve detection and patient outcomes (Illimoottil and Ginat, 2023).

The introduction of U-Net (Ronneberger et al., 2015) marked a significant advancement in semantic segmentation and inspired a variety of general (Zhou et al., 2018; Chen et al., 2017) and polyp segmentation specific (Safarov and Whangbo, 2021; Sun et al., 2019; Nguyen-Mau et al., 2023; Murugesan et al., 2019; Patel et al., 2021; Yeung et al., 2021) Convolutional Neural Network (CNN)-based methods. Criticizing the growing number of task-specific U-Net-based architectures, Isensee et al. (2021) introduced nnU-Net, a self-configuring U-Net model designed to be applicable to any medical segmentation task, including polyp segmentation (Huang et al., 2024). It automatically configures the most appropriate U-Net architecture for the dataset at hand, thus mitigating the need for laborious adaptation. Nevertheless, promising U-Net-based architectures continue to emerge. For example, Du et al. (2025) introduced a novel polyp segmentation method, UM-Net, based on the hypothesis that incorporating reverse-contour information, adaptive learning of local and global contexts, and multi-level feature fusion can enhance polyp segmentation performance. UM-Net effectively outperformed several advanced polyp segmentation methods such as PraNet (Fan et al., 2020), SFANet (Fang et al., 2019), ACSNet (Zhang et al., 2020b), Polyp-PVT (Dong et al., 2021) and FANet (Tomar et al., 2022), becoming state-of-the-art (SOTA). In a broader context, transformer-based models, originally developed for Natural Language Processing (NLP), have become increasingly popular in vision tasks, as their attention mechanisms capture rich contextual relationships (Wang et al., 2023; Ling et al., 2023; Xiao et al., 2024).

Despite the significant progress made by these methods, challenges still persist in their integration into the clinical workflow. As shown in Figure 1, colonoscopy data is highly heterogeneous, influenced by factors such as the acquisition device, patient demographics, technique, and lighting conditions (Wei et al., 2021). These variations introduce specific characteristics in the data that do not always align with clinically relevant information. When DL models are trained on such data, they tend to learn these non-relevant features and mistakenly associate them with clinical meaning. Consequently, models base their

predictions on spurious correlations rather than robust clinical features. This bias leads to significant performance degradation when applied to unseen domains where such misleading cues are absent (Zhang et al., 2018; Mehrabi et al., 2021; Geirhos et al., 2020). Therefore, for Artificial Intelligence (AI) models to be efficiently integrated into clinical workflows, they must be robust to these shifts in data distribution. While recent advancements in polyp segmentation are promising, their performance tends to degrade on unseen data distributions (Mei et al., 2025). Transfer learning and domain adaptation methods can address these challenges by adapting models to new data, but they often require large, annotated datasets, which are expensive and time-consuming to collect (Zhang et al., 2020a).

In contrast, Self-Supervised Learning (SSL) has gained significant attention for its ability to train models without labeled data while producing more robust representations, making it a promising way to overcome the limitations of large annotated datasets (Jing and Tian, 2020; Tendle and Hasan, 2021). Specifically, Distillation with No Labels (DINO) (Caron et al., 2021) has emerged as one of the most effective approaches, particularly when combined with Vision Transformers (ViTs) (Dosovitskiy, 2020). In ViTs, the decoder typically receives learned tokens for prediction, while the self-attention mechanism generates additional data representations used for attention scoring. Notably, the auxiliary "key" representations obtained through DINO pretraining offer greater robustness and interpretability than the tokens, potentially enhancing the model's generalization capabilities (Amir et al., 2021).

Validating this hypothesis, we demonstrate that the intrinsic robustness of DINO "key" features directly translates to superior generalization in polyp segmentation, particularly in data-constrained scenarios. Furthermore, we systematically evaluate the evolution of the DINO framework, quantifying how architectural advancements across versions impact segmentation performance (Caron et al., 2021; Oquab et al., 2023; Siméoni et al., 2025).

The main contributions of this study can be summarized as follows: **(1)** We present a straightforward supervised approach leveraging DINO ViT "key" features that achieves SOTA results without relying on complex, task-specific architectures. **(2)** We validate the model's robustness through stringent Domain Generalization (DG) and Extreme Single Domain Generalization (ESDG) protocols, employing a comprehensive statistical analysis to confirm its superiority in data-scarce settings. **(3)** We provide a comparative benchmark of the DINO framework's progression, quantifying the specific impact of architectural advancements on segmentation performance. **(4)** Through targeted ablation studies, we empirically demonstrate that pre-trained DINO features are sufficiently robust to render task-specific fine-tuning unnecessary. **(5)** We demonstrate that our approach achieves the best performance and lower computation cost than most baselines, facilitating clinical integration.

## 2. Materials and Methods

### 2.1. Data

PolypGen (Ali et al., 2023) is a comprehensive colonoscopy dataset collected from six hospitals and comprises 3,762 annotated polyp pictures. These include sequence frames and single frames, all validated by six senior gastroenterologists. Our analysis focuses on the single-frame subset, which contains 1,537 samples. Their distribution across institutions is as follows: **C1**: 256, **C2**: 301, **C3**: 457, **C4**: 227, **C5**: 208, and **C6**: 88.

This dataset not only enables a comprehensive evaluation of model segmentation generalization, as it includes data from multiple hospitals, but it is also highly heterogeneous. As seen in Figure 1, images might contain device-related annotations, such as dates and other non-anatomical elements, and vary significantly in terms of light conditions and blurring. Additionally, the dataset includes samples from chromoendoscopy (Song et al., 2007), a contrast-enhanced colonoscopy technique using methylene blue, which also introduces significant variability. We consider this to be the most challenging publicly available dataset for the task, likely better representing real deployment scenarios.

### 2.2. Distillation with no labels (DINO)

**DINOv1** DINO (Caron et al., 2021) is a SSL method trained on the ImageNet dataset (Russakovsky et al., 2015), designed to improve feature learning through knowledge distillation (Ohri and Kumar, 2021). It trains two identical networks—a student and a teacher—each receiving different views of the same image. Each model consists of a ViT and a DINO head with three Multilayer Perceptron (MLP) layers. A ViT model encodes an image by embedding each patch as a separate token. To this sequence, it adds a learnable [CLS] token whose role is to absorb global contextual information through the self-attention mechanism. The final [CLS] embedding is then used as a compact, image-level representation. This [CLS] token from the ViT is passed to the DINO head, generating a representation of the input. While the teacher generates this representation from a global view of the input, the student receives either a global or local crop. The student learns by attempting to imitate the teacher's output, with the teacher's parameters updated gradually using an Exponential Moving Average (EMA) of the student's past iterates. Furthermore, to prevent collapse of these representations during training, the teacher's output undergoes centering and sharpening. Centering prevents any one dimension from dominating, but may promote a uniform distribution, while sharpening has the opposite effect.

**DINOv2** (Oquab et al., 2023) expands the pretraining dataset size through an automatic curation pipeline, creating the LVD-142M dataset. It combines the learning objectives of the original DINO and the iBOT SSL framework (Zhou et al., 2021), resulting in an image-level objective that compares the teacher's and student's predictions on the teacher's global views, and a patch-level objective where the student predicts masked patches to match the iBOT teacher, which observes the corresponding visible patches. Instead of the original softmax centering for the teacher's predictions, DINOv2 employs Sinkhorn-Knopp centering (Caron et al., 2020).

Overall, the key technical improvements introduced in DINOV2 primarily focus on enabling larger models to better handle and learn from vast amounts of data.

**Registers** Darcet et al. (2023) observed the emergence of high-norm tokens in the background areas of the images, a new behavior in DINOv2 that was not present in the original version of the framework. These artifacts primarily appear in tokens containing redundant information. Based on this, the authors hypothesize that the model may be identifying irrelevant information and reusing these patches to aggregate global features, a theory further supported by the fact that these tokens appear only in the later layers of the ViT. To address this issue, the authors propose the addition of four "register tokens" at the end of

the token sequence in the ViT. These tokens seem to mitigate the observed artifacts by isolating and redirecting this ViT tendency to encode global computations in arbitrary patch tokens, thereby preventing this behaviour from interfering with patch-level representations.

**DINOv3** Siméoni et al. (2025) built upon DINOv2 with register tokens, introducing DINOv3. Based on the advances in automatic curation of large-scale, uncurated data explored in DINOv2 with the LVD-142M dataset, the authors constructed the LVD-1689M dataset, sourced from approximately 17 billion publicly available Instagram images. The learning objective also remains similar to DINOv2, incorporating the image-level objective (Caron et al., 2021) and the patch-level objective from iBOT (Zhou et al., 2021).

In DINOv2, the backbone used for self-supervised training and later distilled into smaller variants is the ViT-g architecture, which has 1.1 billion parameters. In DINOv3, the authors introduce a new variant, ViT-7B, with 6.7 billion parameters. Compared to ViT-g, ViT-7B retains the same depth of 40 blocks but increases the patch size from 14 to 16 and the embedding dimensions from 1536 to 4096. Additionally, the authors modify the traditional position embeddings to a custom version of Relative Position Embeddings (RoPE) (Su et al., 2024). In RoPE, the position of patches is represented within a coordinate box in the range [-1, 1]. The authors modify this by introducing "RoPE-box jittering", that randomly scales the coordinates of each patch within the interval [-$s$,$s$], where $s$ ranges between 0.5 and 2.

One of the most significant advancements in DINOv3 is the introduction of Gram Anchoring. The authors observe that the cosine similarity between the patch outputs and the [CLS] token increases during longer training periods, leading to the loss of local-specific information. To address this issue, the authors propose a new loss function based on the Gram matrices (Gatys et al., 2016) of the student and teacher models.

The Gram matrix is a matrix of pairwise dot products between patch features in an image that captures the relationships between these features. With this new loss function, the Gram matrix of the student model is pushed to align with the Gram matrix of an earlier iteration of the teacher model, as the teacher's early iteration exhibits superior dense properties. By minimizing this loss, the model is encouraged to preserve patch-level consistency without directly modifying the features themselves.

In addition to Gram Anchoring, the authors introduce resolution scaling, which helps the model better adapt to higher image resolutions. This involves training the model for an additional 10k iterations using crops with varying resolutions up to 768 pixels.

### 2.3. Model Architecture

Figure 2 illustrates the proposed architecture. It consists of a ViT small (ViT-s) feature extractor, pretrained with DINO, and paired with a CNN-based decoder. Amir et al. (2021) studied DINO-pretrained ViT features, focusing on the interpretability of the ViT's self-attention mechanism and the robustness of DINO representations. Amir et al. (2021) found that "key" features offer a more object-centered representation, preserving positional information better than other computed features. Moreover, compared to supervised ViTs, DINO's self-supervised feature space organizes descriptors based on semantic parts, independent of the original class, showing increased robustness and potential for better generalization. Following these observations, we extract the "key" features from the self-attention module at the last layer of the ViT-s for further processing. We then add a projection

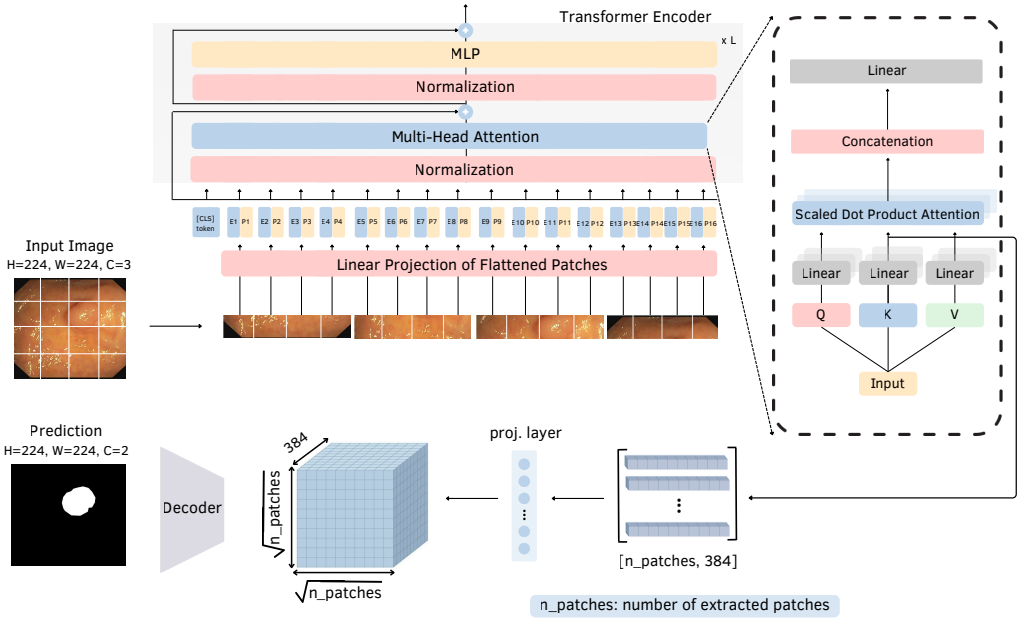


Figure 2: Model architecture for polyp segmentation using DINO ViT "key" features. H, W and C represent the height, width and number of channels, respectively. The input image is passed through a ViT encoder, and "key" features are extracted from the multi-head attention block in the last layer. These features are reshaped and projected to match the spatial dimensions required by the decoder, which generates the final segmentation mask.

layer as a bridge to the decoder, mapping the 384 elements of the patch embeddings to 1024 channels. Our decoder is composed of five unit blocks. The first four blocks include a bilinear upsampling operation followed by double convolution, which progressively refines the feature maps. Then, the final block performs a pointwise convolution operation that maps the features to the end dense prediction mask.

We intentionally keep the architecture simple, using the smallest ViT version and a low-depth, straightforward CNN-based decoder. Without adding unnecessary complexities, the strength of this approach lies in the robustness of the extracted "key" features.

### 2.4. Baselines

To validate the effectiveness of our approach, we compare it with **U-Net** with 4 stages of depth and bilinear upsampling instead of transposed convolutions (similarly to our decoder); nnU-Net (Isensee et al., 2021); and different variations of UM-Net (Du et al., 2025).

**nnU-Net.** Isensee et al. (2021) introduced an adaptive U-Net approach, which has gained widespread recognition for delivering SOTA results. In the same way, the pre-processing of input images and the post-processing of predictions are selected through automatic design choices. nnU-Net groups these decisions into three categories: fixed, heuristic (rule-

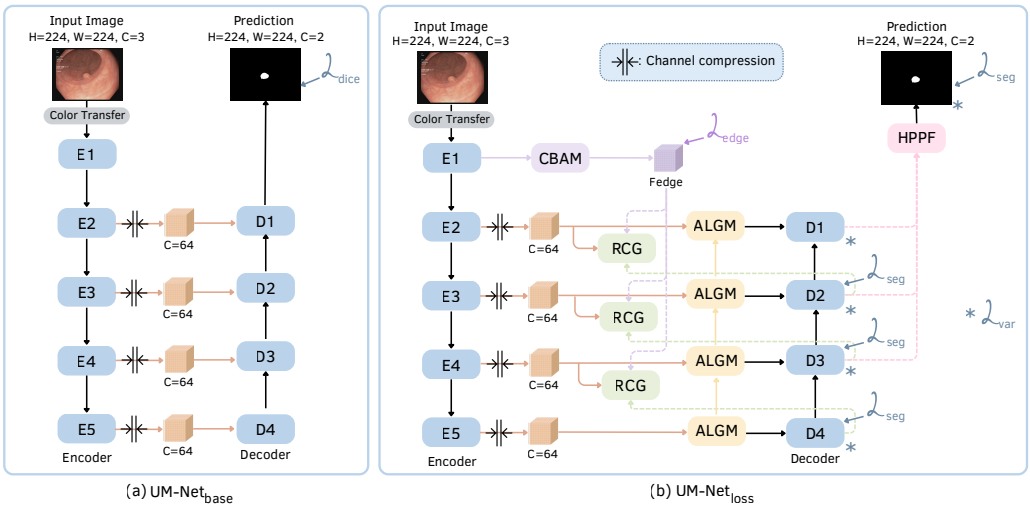


Figure 3: UM-Net variants. (a) UM-Net$_{\text{base}}$, the U-Net baseline of UM-Net. (b) UM-Net$_{\text{loss}}$, the full implementation of UM-Net, incorporating the considered losses. The segmentation loss, $L_{\text{seg}}$, is composed of the Dice loss ($L_{\text{dice}}$) and the Binary Cross-Entropy (BCE) loss. $L_{\text{edge}}$ represents the focal loss, while $L_{\text{var}}$ denotes the variance loss, applied to the variance between deep predictions. H, W and C represent the height, width and number of channels, respectively. $E_{\text{n}}$ and $D_{\text{n}}$ refer to the $n^{\text{th}}$ encoder and decoder blocks, respectively. CBAM represents the Convolutional Block Attention Module, RCG the Reverse-Contour Guidance Module, ALGM the Adaptive Local-Global Context Module and, finally, the HPPF the Hybrid Pyramid Pooling Fusion Module. This figure is adapted from Du et al. (2025).

based), and empirical parameters. Fixed parameters remain constant across datasets and correspond to configurations that have been shown to work reliably in a wide range of tasks. Heuristic parameters depend on the extracted dataset fingerprint and are determined through explicit logical rules rather than learned from data. Finally, empirical parameters are selected using the training data via cross-validation, by choosing the configuration that performs best. Overall, nnU-Net has demonstrated excellent results across diverse segmentation tasks (Isensee et al., 2020; Ferrante et al., 2022) and effectively addresses the problem of laborious, dataset-specific adaptation. As a result, it has become one of the standard segmentation baselines that new methods aim to outperform.

**UM-Net.** In UM-Net (Du et al., 2025), the input data is first processed by a 5-block ResNet34 encoder. The output from the first and shallowest encoder block ($E_1$) is passed through a Convolutional Block Attention Module (CBAM), which assigns attention weights across both the spatial and channel dimensions. The processed features undergo 2D convolution with a single output channel to emphasize the edges ($F_{\text{edge}}$). The outputs from $E_2$-$E_5$ are compressed to 64 channels, while preserving the spatial dimensions of the features. The compressed output of $E_5$ is processed by an Adaptive Local-Global Context Module (ALGM) before being fed to the decoder. This module is designed to capture multi-scale

information through two submodules: a global module and a local module. The global module captures long-range dependencies between pixels in the feature map, while the local module applies cascaded dilated convolutions to the features, integrating information across multiple scales. The outputs from both modules are then combined and processed through a squeeze-and-excitation (SE) block, which re-weights the channels to emphasize the most relevant features.

The Reverse-Contour Guidance Module (RCG) combines $F_{\text{edge}}$ with reverse features, obtained by generating a reverse attention map from the output of the previous decoder block. In addition, it incorporates compressed encoder features, containing high-level information. By combining enhanced edge information with reverse semantic context, the RCG module generates refined features, which are then fed to the decoder blocks.

With the exception of the deepest decoder block, each decoder block receives a 192-channel input and delivers a 64-channel output. The inputs consist of the RCG-processed features, ALGM-processed features, and the features from the previous decoder block. The final prediction is determined by a fusion between the outputs of $D_1$, $D_2$ and $D_3$.

Each decoder block consists of two convolutional layers followed by an upsampling layer, doubling the spatial dimensions of the input. The first convolutional layer reduces the number of channels from the input to one-quarter of the input channels, using a 3x3 kernel with a stride of 1. After the convolution, batch normalization is applied to stabilize the activations, followed by a ReLU activation. The second convolutional layer further processes these features, mapping the output from the first layer to the final 64 output channels, using a 3x3 kernel and a stride of 1. Finally, the upsampling layer uses bilinear interpolation to scale the spatial dimensions by a factor of 2.

To properly deal with inconsistencies in color in colonoscopy images, the authors also propose a color transfer operation as a new type of color invariance method. For a certain input image, a random reference image is selected. Both are converted to the LAB color space, where the target colors are mapped to the distribution of its selected reference.

The authors train their architecture with the Dice loss, Binary Cross-Entropy (BCE) loss, a focal loss (focused on the boundaries of the segmentation), and an introduced variance loss, using deep supervision in the decoder.

We consider the following UM-Net variations, depicted in Figure 3: **(1)** UM-Net$_{\text{base}}$: the U-Net baseline on which UM-Net is built. **(2)** UM-Net: trained using the Dice loss. **(3)** UM-Net$_{\text{loss}}$: trained with the complete set of losses described in Du et al. (2025).

### 2.5. Implementation details

We summarize the key differences in implementation across DINO versions in Table 4 in Appendix A.

For the experiments conducted using DINOV1, the patch size used was 16 pixels, with a stride set to 50% of the patch size (i.e., 8 pixels). With DINOV2, the model was pretrained on the LVD-142M dataset. For this version, the available checkpoints were pretrained with a patch size of 14. Similarly to the original version, we consider a stride of 50% of the patch size. Finally, DINOV3 (Siméoni et al., 2025) uses a patch size of 16, and was pretrained on the LVD-1689M dataset, built on an automatic curation of Instagram public images. The stride used was the same as in previous versions.

The ViT extractor is kept frozen for all experiments. Consistent with the DINO authors' findings (Oquab et al., 2023; Siméoni et al., 2025), our observations confirm that these features are already sufficiently robust for the downstream task. Section 2.8 describes the experiments that lead to this observation, and Appendix D has the corresponding results.

The images are reshaped to 224x224. During training, the image intensities are normalized and randomly receive transformations such as horizontal and vertical flipping, rotation (limited to 45° and with a probability of 50%) and color jittering (with a probability of 30%). UM-Net variants also include color transfer, as introduced by the authors. nnU-Net does not follow our determined augmentations as it has its own preprocessing configurations. The models are trained on an NVIDIA GPU RTX8000 with a batch size of 4 for 300 epochs using the Adam optimizer (Adam et al., 2014), and a learning rate of $1 \times 10^{-5}$. We select the evaluation checkpoint using the maximum validation Dice Score Coefficient (DSC). All models are trained with the Dice loss, except for UM-Net$_{\text{loss}}$, where we replicate the loss functions described in Du et al. (2025).

### 2.6. Generalization assessment

We evaluated our model against the baselines in two generalization tests: the Domain Generalization (DG) and the Extreme Single Domain Generalization (ESDG) tests (Che et al., 2023). These tests are designed to simulate real-world deployment scenarios, where models are required to perform well on unseen data from different domains or environments.

**DG Test.** This test follows the leave-one-domain-out protocol. In this setup, the model is trained using data from multiple centers, but the performance is assessed by excluding one center at a time during the training phase, testing the model's ability to generalize to that unseen domain. This is a challenging test since it evaluates how well the model can handle data from entirely new domains that were not part of its training process.

**ESDG Test.** The ESDG test is an even more stringent evaluation, simulating a real-world scenario where the model is trained on a single domain (in this case, one center's data) and then tested on the remaining centers. This test assesses the ability of the model to generalize to other domains without having seen them during training, in a much more data-limited setting.

### 2.7. Statistical analysis

It is currently common practice to report performance improvements by directly comparing results with baseline models. However, some observed differences may lack statistical significance and therefore do not provide sufficient evidence that a given model genuinely outperforms its baselines. To ensure a rigorous assessment of our results—both in the main experiments and in the ablation studies—we evaluate the statistical significance of the differences observed in DSC performance. For detailed explanation of the considered tests, please refer to Appendix B.

Statistical testing is performed independently for each generalization setting. For each fold (for example, fold 1 under the DG evaluation), we compute a single average DSC value per model. This produces a paired set of observations across folds for any two models under comparison. The Wilcoxon signed-rank test is applied to these paired differences,

evaluating whether the median difference deviates significantly from zero. In doing so, it assesses whether one model consistently achieves higher segmentation performance than another in a statistically meaningful way.

### 2.8. Complementary experiments

We conduct two ablation studies—one examining the impact of our decoder architecture and another assessing the impact of fine-tuning the DINO ViT features. Additionally, we examine the computational load of each model.

**Ablation: Decoder Architecture.** The original UM-Net decoder could not be directly integrated into our pipeline, primarily because it relies on skip connections. This requirement makes the number of input and output channels incompatible with the unidirectional flow of features in our approach. Because of this, we replaced it with a slightly different, simple CNN-based decoder. However, to ensure that this architectural change does not introduce a confounding factor, we perform a dedicated ablation using UM-Net$_{\text{base}}$. Specifically, we take the encoder from UM-Net$_{\text{base}}$ and append a projection layer that maps its 512 channel output to the 1024 channels required by our decoder. We then attach our decoder and evaluate the performance of this changed configuration. By comparing the results with those obtained using the original UM-Net$_{\text{base}}$ decoder, we verify whether our decoder introduces any significant difference. Demonstrating that this change does not notably affect performance, we confirm that these decoder differences do not compromise the validity of our study. Results of this ablation are provided in Appendix C.

**Ablation: Supervised Fine-tuning of DINO Features.** To determine whether polyp segmentation benefits from supervised fine-tuning of the DINO ViT extractor, we conduct an ablation study comparing the performance of each version (V1, V2, V2R, and V3) when using the pretrained features directly, i.e., without any task-specific tuning, against the performance obtained after supervised fine-tuning for polyp segmentation. We then assess the statistical significance of the differences between these two settings to determine whether these representations gain meaningful improvements from task-specific supervision or whether their strength derives primarily from self-supervised pretraining, thus making further fine-tuning unnecessary. Results of this ablation are provided in Appendix D.

**Computational Load.** For each model and generalization test, we calculate the average training time per epoch over the first 10 epochs. From this, we can derive the average running time per epoch for each fold and model. These folds correspond to those used in the generalization tests, with 6 folds for the DG test (e.g., DG1 uses data from centers 2-6 for training and validation, and data from center 1 for testing) and another 6 folds for the ESDG test (e.g., ESDG1 uses data from center 1 for training and validation, and data from centers 2-6 for testing). With these findings, we can objectively assess the relative computing loads of these models. Furthermore, we aim to compare inference times by evaluating the elapsed time for making a prediction on a random input with dimensions (H, W, C) = (224, 224, 3). Due to its need for input sizes divisible by 256 and greater than 256, nnU-Net is an exception, accepting an input with dimensions (H, W, C) = (512, 512, 3). The Least Common Multiple (LCM) 1792, which takes into account 14, 16, and

Table 1: Domain Generalization (DG) and Extreme-Single Domain Generalization (ESDG) test results represented as $(100 \times \text{avg})_{100\times\text{std}}$. Best performance per column is depicted in **bold**.

| Model | DG | | | ESDG | | |
|---|---|---|---|---|---|---|
| | DSC | Recall | Acc | DSC | Recall | Acc |
| U-Net | $60.75_{12.78}$ | $59.98_{15.73}$ | $95.84_{0.80}$ | $38.15_{6.78}$ | $48.81_{11.01}$ | $89.48_{5.80}$ |
| nnU-Net | $65.09_{14.75}$ | $65.68_{15.15}$ | $95.50_{1.64}$ | $52.36_{7.97}$ | $57.40_{10.63}$ | $93.21_{1.65}$ |
| UM-Net$_{\text{base}}$ | $69.69_{14.16}$ | $69.41_{15.92}$ | $96.82_{1.33}$ | $61.04_{3.38}$ | $66.07_{2.43}$ | $95.19_{0.43}$ |
| UM-Net | $71.44_{12.81}$ | $68.46_{17.39}$ | $97.13_{0.78}$ | $60.77_{4.05}$ | $63.73_{5.10}$ | $95.07_{0.87}$ |
| UM-Net$_{\text{loss}}$ | $73.26_{13.42}$ | $70.88_{16.02}$ | $97.42_{0.88}$ | $63.22_{3.23}$ | $65.03_{4.05}$ | $95.60_{0.31}$ |
| Proposed$_{\text{V1}}$ | $71.16_{14.86}$ | $69.93_{16.64}$ | $97.00_{1.37}$ | $64.03_{4.10}$ | $66.53_{3.19}$ | $95.72_{0.60}$ |
| Proposed$_{\text{V2}}$ | $72.82_{12.57}$ | $69.89_{16.25}$ | $97.37_{0.84}$ | $65.31_{4.58}$ | $71.54_{3.13}$ | $95.54_{1.03}$ |
| Proposed$_{\text{V2R}}$ | $77.00_{11.43}$ | $72.35_{16.43}$ | $97.72_{0.58}$ | $71.03_{2.75}$ | $72.04_{2.82}$ | $96.47_{0.49}$ |
| Proposed$_{\text{V3}}$ | $\mathbf{80.00}_{8.31}$ | $\mathbf{75.21}_{14.94}$ | $\mathbf{98.07}_{0.36}$ | $\mathbf{73.64}_{3.33}$ | $\mathbf{75.51}_{2.21}$ | $\mathbf{96.97}_{0.40}$ |

256, could not be used because its size and memory requirements exceed the limits of the current hardware. The detailed results of these experiments are provided in Appendix E.

## 3. Results

We present the results for the generalization tests in Table 1, using the evaluation metrics described in Appendix F: **Accuracy**, **Recall** and **DSC**. In the notation for the proposed variants, we indicate the DINO version (V1, V2, or V3), along with the presence of register tokens (denoted as R). The R notation for V3 is omitted, as the registers are already included in the base for this DINO version.

The proposed model achieves the best performance over both tests. Notably, the performance improvement brought by the proposed configuration is particularly pronounced in the most challenging evaluation setting, where the model is trained on single-domain data and evaluated on the remaining five domains. This finding is crucial for the model's clinical applicability, demonstrating that our approach can effectively generalize to diverse data distributions, even with as few as 88 training and validation samples (center 6). Overall, we observe a clear progression in performance across the different DINO versions, with the most pronounced improvement corresponding to the introduction of registers in DINOV2.

U-Net exhibits the second largest performance drop from the DG to the ESDG test (around 37%). This model, while effective in standard conditions, struggled significantly when trained on a single domain, which highlights its limited generalization capacity and high sensitivity to domain shifts. Contrarily to expectations, nnU-Net only slightly outperformed U-Net in most scenarios. Additionally, it experienced the largest drop in DSC, with a 38% decrease from DG to ESDG testing. On the other hand, UM-Net exhibited the best generalization among the evaluated baselines, with its performance further improved by replicating the original training conditions (UM-Net$_{\text{loss}}$). For the tested variations of this model, the average relative performance drop between tests was between 12 and 20%. How-

Table 2: Pairwise p-values from Wilcoxon Signed-Rank Test on DG DSC results. A "-" indicates comparison of the same model. P-values greater than 0.05 (indicating a non-significant difference) are shaded grey.

| | U-Net | nnU-Net | UM-Net$_{base}$ | UM-Net | UM-Net$_{loss}$ | V1 | V2 | V2R | V3 |
|---|---|---|---|---|---|---|---|---|---|
| U-Net | - | 0.0625 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| nnU-Net | 0.0625 | - | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| UM-Net$_{base}$ | 0.0313 | 0.0313 | - | 0.2188 | 0.0313 | 0.1563 | 0.0625 | 0.0313 | 0.0313 |
| UM-Net | 0.0313 | 0.0313 | 0.2188 | - | 0.2188 | 0.6875 | 0.4375 | 0.0313 | 0.0313 |
| UM-Net$_{loss}$ | 0.0313 | 0.0313 | 0.0313 | 0.2188 | - | 0.0625 | 1.0 | 0.0313 | 0.0313 |
| V1 | 0.0313 | 0.0313 | 0.1563 | 0.6875 | 0.0625 | - | 0.3125 | 0.0313 | 0.0313 |
| V2 | 0.0313 | 0.0313 | 0.0625 | 0.4375 | 1.0 | 0.3125 | - | 0.0313 | 0.0313 |
| V2R | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | - | 0.0679 |
| V3 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0679 | - |

ever, it still trailed behind the proposed model, especially in the ESDG tests. Our model, without any polyp-specific architectural components, demonstrated the best performance under the most challenging conditions, with DG to ESDG performance drops of 8-10%, notably lower than those observed in the baselines. While UM-Net remains competitive, it does not match the robustness and cross-domain adaptability of the proposed framework.

Some of the performance ranges overlap or are quite close, stressing the importance of statistical analysis to determine whether the observed differences are significant.

Applying the Friedman test to the average DSC results for the DG and ESDG tests, we obtain p-values of $9.2748 \times 10^{-7}$ and $3.2439 \times 10^{-7}$, respectively. Given that both p-values are significantly lower than the chosen significance level of $\alpha = 0.05$, we can reject the null hypothesis and conclude that there are statistically significant differences between the performances of at least one of these models.

To further assess which pairs of models exhibit significant performance differences, we apply the Wilcoxon test with a significance level of $\alpha = 0.05$. The results of this pairwise analysis are presented in Tables 2 and 3. Highlighted in grey are the model pairs that do not show a significant difference in average DSC results across both tests.

In both cases, nnU-Net does not show a significant improvement over U-Net. Examining the variations of UM-Net, we find no substantial differences between UM-Net$_{base}$ and UM-Net, but performance improves notably with UM-Net$_{loss}$, suggesting that these additional losses are effectively guiding the model's learning. For the proposed variants, V2R and V3 outperform all baselines with significant improvements. While V1 and V2 show gains over U-Net and nnU-Net in DG tests, they do not surpass the performance of the UM-Net variants. In the ESDG tests, V1 outperforms UM-Net but does not surpass UM-Net$_{base}$ or UM-Net$_{loss}$, while V2 improves over both UM-Net and UM-Net$_{base}$, but not over UM-Net$_{loss}$.

When comparing the proposed variants, V2 does not appear to provide significant gains over V1; nonetheless, the addition of register tokens appears to play an important role in performance enhancement. V3 outperforms V1 and V2, but not V2R. Based on these findings, it appears that the enhancements offered in V2, particularly when combined with the usage of register tokens to smoothen patch features, are the most significant modifications introduced by DINO for this specific application.

Table 3: Pairwise p-values from Wilcoxon Signed-Rank Test on ESDG DSC results. A "-" indicates comparison of the same model. P-values greater than 0.05 (indicating a non-significant difference) are shaded grey.

| | U-Net | nnU-Net | UM-Net$_{base}$ | UM-Net | UM-Net$_{loss}$ | V1 | V2 | V2R | V3 |
|---|---|---|---|---|---|---|---|---|---|
| U-Net | - | 0.0625 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| nnU-Net | 0.0625 | - | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| UM-Net$_{base}$ | 0.0313 | 0.0313 | - | 1.0 | 0.0625 | 0.0625 | 0.0313 | 0.0313 | 0.0313 |
| UM-Net | 0.0313 | 0.0313 | 1.0 | - | 0.0625 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| UM-Net$_{loss}$ | 0.0313 | 0.0313 | 0.0625 | 0.0625 | - | 0.2188 | 0.0938 | 0.0313 | 0.0313 |
| V1 | 0.0313 | 0.0313 | 0.0625 | 0.0313 | 0.2188 | - | 0.0625 | 0.0313 | 0.0313 |
| V2 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0938 | 0.0625 | - | 0.0313 | 0.0313 |
| V2R | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | - | 0.0679 |
| V3 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0679 | - |

We present qualitative examples in Figure 4 from the ESDG setting, where model differences were strongest and segmentation was most difficult. They include device-related artefacts (1st, 2nd, 4th), chromoendoscopy frames (2nd, 4th), miniatures (3rd), and medical devices (3rd, 4th). U-Net performed the worst, failing on all examples. nnU-Net showed clear improvements, though it still struggled particularly with miniatures and chromoendoscopy. UM-Net variants were less sensitive to these challenges but produced less precise predictions than our variants, struggling mostly with miniatures and chromoendoscopy. In contrast, our variants delivered the most accurate results overall: V3 was particularly robust to miniatures, isolating only the polyp, and only V2 failed to detect the larger polyp in the 2nd example. We also observe consistent improvements from V1 to V3—especially in V2R and V3——matching the trends seen in the quantitative analysis.

These results confirm that our model meaningfully outperforms all baselines, especially with the most recent DINO variants, and offers the largest gains in data-scarce settings. In clinical practice, segmentation models must remain robust across variable data distributions and prioritize medically relevant features over spurious cues. Furthermore, highly specialized architectures often fail to generalize beyond their target task; in contrast, our approach achieves strong performance without task-specific components, using a simple design that could be adapted to other segmentation problems. This configuration also shows a reduced computational cost compared to most baselines, as observed in Table 7 of Appendix E.

We depict failure results in Figures 9 and 10 of Appendix G. Interestingly, we observe that when the test set includes samples from centers 4 and 5, failure rates markedly increase. These centers contain the highest density of device-related labels and the largest proportion of negative samples, underrepresented in the single-frame data. Although reaching meaningful contributions, we identify some limitations. Ideally, we would use DINO versions pretrained under identical conditions (patch size, dataset), which was not possible due to reliance on publicly available weights. Due to computational constraints, we used the ViT-s architecture for the feature extractor, which is distilled from larger models in V2 and V3. However, using larger variants would likely further increase the performance gap between our model and the baselines.

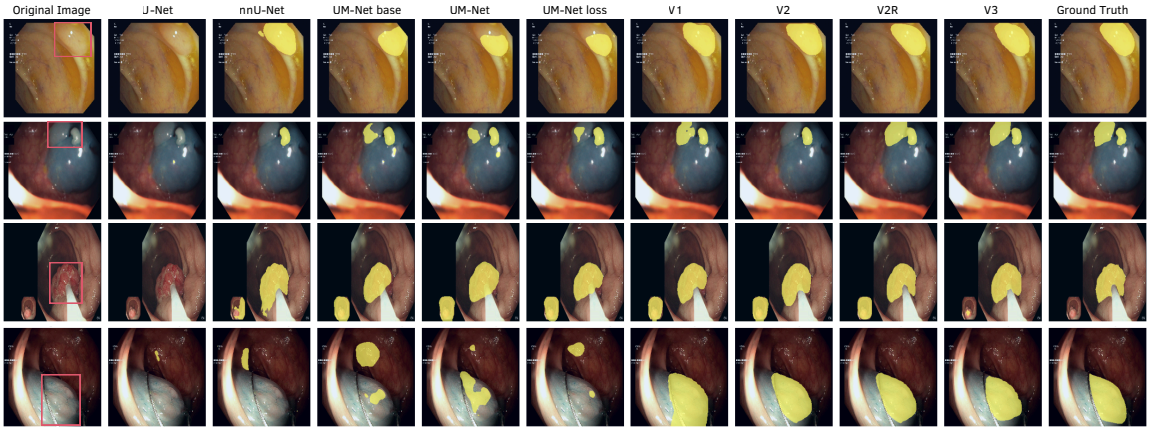


Figure 4: Qualitative ESDG results of the baselines and variations of the proposed pipeline. Predictions and Ground Truth (GT) are overlayed in yellow.

In nnU-Net, self-optimized U-Net consists of a 9-stage encoder-decoder with transposed convolutions. Given that the nnU-Net has a decoder depth twice that of our model, it is plausible to expect that a deeper decoder would result in better segmentation performance. Since this might put our proposed pipeline at a disadvantage rather than nnU-Net, we argue that it does not invalidate the comparison.

Lastly, UM-Net's authors released the architecture code but not the training pipeline. Therefore, UM-Net$_{\text{loss}}$ is a result of our implementation based on the paper's description, and thus may deviate slightly from the author's original approach.

## 4. Conclusions

In this study, we explore DINO ViT "key" features to enhance generalization efficiency in polyp segmentation, without requiring a tailored architecture. Our experiments show that, particularly with DINOv2 (using register tokens) and the latest DINOv3, we significantly outperform the baselines, especially in challenging single-center training. Statistical analysis confirmed that the observed performance differences were meaningful, validating our results. This finding not only paves the way for improved generalization in polyp segmentation but also opens opportunities for applying this method to other segmentation tasks. Notably, with increased model complexity, larger ViT versions could further enhance this performance.

Future work will focus on applying this approach to polyp video segmentation and investigating whether this improved generalization is transferable to video data. Colonoscopy videos typically run at 30 FPS (Kim et al., 2024; Cho et al., 2019), requiring model inference times below 33 ms (1 s / 30). As shown in Table 8 in Appendix E, all proposed variants meet this requirement, achieving inference times suitable for real-time video integration. Furthermore, we are also interested in studying the scalability of this approach to the segmentation of other lesions, such as tumors in Computed Tomography (CT) or Magnetic Resonance Imaging (MRI).

## Acknowledgments

Research at the Netherlands Cancer Institute is supported by grants from the Dutch Cancer Society and the Dutch Ministry of Health, Welfare and Sport. The authors would like to acknowledge the Research High Performance Computing (RHPC) facility of the Netherlands Cancer Institute (NKI).

## References

Kingma DP Ba J Adam et al. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6), 2014.

Sharib Ali, Debesh Jha, Noha Ghatwary, Stefano Realdon, Renato Cannizzaro, Osama E Salem, Dominique Lamarque, Christian Daul, Michael A Riegler, Kim V Anonsen, et al. A multi-centre polyp detection and segmentation dataset for generalisability assessment. *Scientific Data*, 10(1):75, 2023.

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

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

Haoxuan Che, Yuhan Cheng, Haibo Jin, and Hao Chen. Towards generalizable diabetic retinopathy grading in unseen domains. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 430–440. Springer, 2023.

Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

Minwoo Cho, Jee Hyun Kim, Kyoung Sup Hong, Joo Sung Kim, Hyoun-Joong Kong, and Sungwan Kim. Identification of cecum time-location in a colonoscopy video by deep learning analysis of colonoscope movement. *PeerJ*, 7:e7256, 2019.

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

Bo Dong, Wenhai Wang, Deng-Ping Fan, Jinpeng Li, Huazhu Fu, and Ling Shao. Polyp-pvt: Polyp segmentation with pyramid vision transformers. *arXiv preprint arXiv:2108.06932*, 2021.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Xiuquan Du, Xuebin Xu, Jiajia Chen, Xuejun Zhang, Lei Li, Heng Liu, and Shuo Li. Umnet: Rethinking icgnet for polyp segmentation with uncertainty modeling. *Medical Image Analysis*, 99:103347, 2025.

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*, pages 263–273. Springer, 2020.

Yuqi Fang, Cheng Chen, Yixuan Yuan, and Kai-yu Tong. Selective feature aggregation network with area-boundary constraints for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 302–310. Springer, 2019.

Matteo Ferrante, Lisa Rinaldi, Francesca Botta, Xiaobin Hu, Andreas Dolp, Marta Minotti, Francesca De Piano, Gianluigi Funicelli, Stefania Volpe, Federica Bellerba, et al. Application of nnu-net for automatic segmentation of lung lesions on ct images and its implication for radiomic models. *Journal of clinical medicine*, 11(24):7334, 2022.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Mayuri Gupta and Ashish Mishra. A systematic review of deep learning based image segmentation to detect polyp. *Artificial Intelligence Review*, 57(1):7, 2024.

Lina Huang, Alina Miron, Kate Hone, and Yongmin Li. Segmenting medical images: from unet to res-unet and nnunet. In *2024 IEEE 37th International Symposium on Computer-Based Medical Systems (CBMS)*, pages 483–489. IEEE, 2024.

Mathew Illimoottil and Daniel Ginat. Recent advances in deep learning and medical imaging for head and neck cancer treatment: Mri, ct, and pet scans. *Cancers*, 15(13):3267, 2023.

Ronald L Iman and James M Davenport. Approximations of the critical region of the fbietkan statistic. *Communications in Statistics-Theory and Methods*, 9(6):571–595, 1980.

Fabian Isensee, Paul F Jäger, Peter M Full, Philipp Vollmuth, and Klaus H Maier-Hein. nnu-net for brain tumor segmentation. In *International MICCAI Brainlesion Workshop*, pages 118–132. Springer, 2020.

Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.

Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43 (11):4037–4058, 2020.

Byeong Soo Kim, Minwoo Cho, Goh Eun Chung, Jooyoung Lee, Hae Yeon Kang, Dan Yoon, Woo Sang Cho, Jung Chan Lee, Jung Ho Bae, Hyoun-Joong Kong, et al. Density clustering-based automatic anatomical section recognition in colonoscopy video using deep learning. *Scientific Reports*, 14(1):872, 2024.

Tianyi Ling, Chengyi Wu, Huan Yu, Tian Cai, Da Wang, Yincong Zhou, Ming Chen, and Kefeng Ding. Probabilistic modeling ensemble vision transformer improves complex polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 572–581. Springer, 2023.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54 (6):1–35, 2021.

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.

Balamurali Murugesan, Kaushik Sarveswaran, Sharath M Shankaranarayana, Keerthi Ram, Jayaraj Joseph, and Mohanasankar Sivaprakasam. Psi-net: Shape and boundary aware joint multi-task deep network for medical image segmentation. In *2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 7223–7226. IEEE, 2019.

Trong-Hieu Nguyen-Mau, Quoc-Huy Trinh, Nhat-Tan Bui, Phuoc-Thao Vo Thi, Minh-Van Nguyen, Xuan-Nam Cao, Minh-Triet Tran, and Hai-Dang Nguyen. Pefnet: Positional embedding feature for polyp segmentation. In *International Conference on Multimedia Modeling*, pages 240–251. Springer, 2023.

Kriti Ohri and Mukesh Kumar. Review on self-supervised image recognition using deep neural networks. *Knowledge-Based Systems*, 224:107090, 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.

Krushi Patel, Andrés M Bur, and Guanghui Wang. Enhanced u-net: A feature enhancement network for polyp segmentation. In *2021 18th conference on robots and vision (CRV)*, pages 181–188. IEEE, 2021.

Oona Rainio, Jarmo Teuho, and Riku Klén. Evaluation metrics and statistical tests for machine learning. *Scientific Reports*, 14(1):6086, 2024.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Sirojbek Safarov and Taeg Keun Whangbo. A-denseunet: Adaptive densely connected unet for polyp segmentation in colonoscopy images with atrous convolution. *Sensors*, 21(4): 1441, 2021.

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

Louis Michel Wong Kee Song, Douglas G Adler, Bipan Chand, Jason D Conway, Joseph MB Croffie, James A DiSario, Daniel S Mishkin, Raj J Shah, Lehel Somogyi, William M Tierney, et al. Chromoendoscopy. *Gastrointestinal endoscopy*, 66(4):639–649, 2007.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Xinzi Sun, Pengfei Zhang, Dechun Wang, Yu Cao, and Benyuan Liu. Colorectal polyp segmentation by u-net with dilation convolution. In *2019 18th IEEE international conference on machine learning and applications (ICMLA)*, pages 851–858. IEEE, 2019.

Atharva Tendle and Mohammad Rashedul Hasan. A study of the generalizability of self-supervised representations. *Machine Learning with Applications*, 6:100124, 2021.

Nikhil Kumar Tomar, Debesh Jha, Michael A Riegler, Håvard D Johansen, Dag Johansen, Jens Rittscher, Pål Halvorsen, and Sharib Ali. Fanet: A feedback attention network for improved biomedical image segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):9375–9388, 2022.

Ao Wang, Ming Wu, Hao Qi, Hong Shi, Jianhua Chen, Yinran Chen, and Xiongbiao Luo. Pyramid transformer driven multibranch fusion for polyp segmentation in colonoscopic video images. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 2350–2354. IEEE, 2023.

Jun Wei, Yiwen Hu, Ruimao Zhang, Zhen Li, S Kevin Zhou, and Shuguang Cui. Shallow attention network for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 699–708. Springer, 2021.

Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics: Methodology and distribution*, pages 196–202. Springer, 1992.

Zhenyu Wu, Fengmao Lv, Chenglizhao Chen, Aimin Hao, and Shuo Li. Colorectal polyp segmentation in the deep learning era: A comprehensive survey. *arXiv preprint arXiv:2401.11734*, 2024.

Bin Xiao, Jinwu Hu, Weisheng Li, Chi-Man Pun, and Xiuli Bi. Ctnet: Contrastive transformer network for polyp segmentation. *IEEE Transactions on Cybernetics*, 54(9):5040–5053, 2024.

Michael Yeung, Evis Sala, Carola-Bibiane Schönlieb, and Leonardo Rundo. Focus u-net: A novel dual attention-gated cnn for polyp segmentation during colonoscopy. *Computers in biology and medicine*, 137:104815, 2021.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

Ling Zhang, Xiaosong Wang, Dong Yang, Thomas Sanford, Stephanie Harmon, Baris Turkbey, Bradford J Wood, Holger Roth, Andriy Myronenko, Daguang Xu, et al. Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation. *IEEE transactions on medical imaging*, 39(7):2531–2540, 2020a.

Ruifei Zhang, Guanbin Li, Zhen Li, Shuguang Cui, Dahong Qian, and Yizhou Yu. Adaptive context selection for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 253–262. Springer, 2020b.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *International workshop on deep learning in medical image analysis*, pages 3–11. Springer, 2018.

## Appendix A. DINO Implementation Details

Table 4: Implementation details across versions of DINO.

| DINO Version | Patch Size | Stride | Pretraining Dataset |
|---|---|---|---|
| V1 | 16 pixels | 8 pixels (50% of patch size) | ImageNet |
| V2 | 14 pixels | 7 pixels (50% of patch size) | LVD-142M dataset |
| V3 | 16 pixels | 8 pixels (50% of patch size) | LVD-1689M dataset |

## Appendix B. Statistical Analysis

When performing statistical analyses, t-tests are often the default choice, but they are not ideal in segmentation settings because their underlying assumptions—such as normality and equal variances—are typically violated. For comparisons involving more than two models, the most appropriate approach is the Friedman test, specifically with the Iman–Davenport correction present in equation 1 (Iman and Davenport, 1980), which offers greater reliability and statistical power.

$$F_{ID} = \frac{(J-1)\chi_F^2}{J(K-1)-\chi_F^2}, \tag{1}$$

where we have K models evaluated in J folds (repeated measurements).

For pairwise model comparisons, we use the Wilcoxon signed-rank test (Wilcoxon, 1992). This test is non-parametric—meaning it does not assume any particular distribution for the underlying data—and is therefore robust to outliers and well suited for repeated cross-validation scenarios (Rainio et al., 2024). The test is formulated as

$$T = \min(R^+, R^-), \tag{2}$$

where $R^+$ and $R^-$ denoting the sum of ranks associated with positive paired differences and the sum of ranks corresponding to negative differences, respectively.

## Appendix C. Decoder Ablation

To determine whether the differences between our CNN decoder and UM-Net's decoder could be considered a confounding factor, we conduct an ablation study of UM-Net's baseline, by comparing both their base decoder and our decoder. We then apply the Wilcoxon Signed-Rank test to the DSC results to evaluate the significance of these differences.

The distribution of average DSC between both generalization tests is represented in Figure 5.

The performance difference appears to be more pronounced in ESDG instances, where UM-Net's base decoder has a greater range of values than ours. In the DG cases, the median values are nearly identical, while in the ESDG situations, our decoder's median is somewhat lower. Overall, the difference appears to be negligible, as verified by the Wilcoxon test results, which show a p-value of 0.6875 for the DG comparison and 0.0938

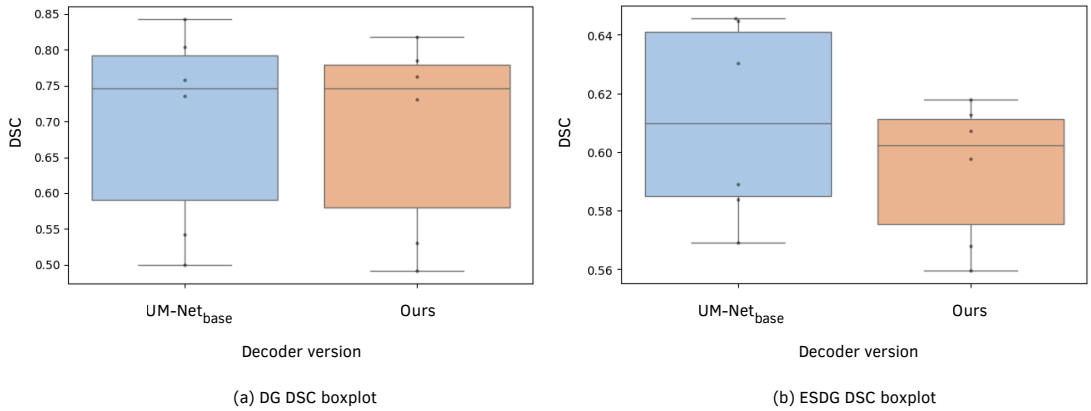


Figure 5: Boxplots of the DSC results with the two decoder versions - UM-Net's base and ours, for (a) DG tests, and (b) ESDG tests.

for the ESDG comparison. Both p-values exceed the $\alpha$ threshold of 0.05, indicating that the observed performance differences are not statistically significant.

## Appendix D. Ablation on the Supervised fine-tuning of DINO Features

We aim to determine whether training the ViT extractor for supervised polyp segmentation alongside the decoder leads to improved results, or if it is unnecessary, given that the features provided by DINO already lead to improved SOTA results. For that, we repeat the same experiments seen in the main body but without freezing the feature extractor.

The notation used follows:

- A for experiments using DINO features directly, i.e., where the feature extractor is kept frozen during decoder training.
- B for experiments where the feature extractor is not frozen during supervised training, i.e., where it is fine-tuned for polyp segmentation.

We present quantitative results for DG and ESDG tests across variations of the proposed pipeline in Table 5.

V3 appears to be the variation with the highest average performance. However, when comparing cases A and B, no consistent trend emerges regarding which one is more advantageous: in some metrics, version B achieves a higher average, while in others it does not. Therefore, looking at these tables alone does not lead to a definite conclusion on whether fine-tuning the DINO features with supervision leads to improved performance.

Using the Friedman test on the average DSC values, we get p-values of $6 \times 10^{-6}$ and $3 \times 10^{-6}$ for the DG and ESDG tests, respectively. Given a $\alpha$ value of 0.05, at least one of these models has significantly different DSC performance than the others.

Table 5: Domain Generalization (DG) and Extreme Single-Domain Generalization (ESDG) ablation test results represented as $(100 \times \text{avg})_{100\times\text{std}}$. Best performance per column is depicted in **bold**.

| Model | DG | | | ESDG | | |
|---|---|---|---|---|---|---|
| | DSC | Recall | Acc | DSC | Recall | Acc |
| $\text{Proposed}_{\text{V1A}}$ | $71.16_{14.86}$ | $69.93_{16.64}$ | $97.00_{1.37}$ | $64.03_{4.10}$ | $66.53_{3.19}$ | $95.72_{0.60}$ |
| $\text{Proposed}_{\text{V1B}}$ | $71.85_{13.04}$ | $69.32_{17.28}$ | $97.10_{0.86}$ | $64.18_{4.19}$ | $68.45_{2.32}$ | $95.66_{0.80}$ |
| $\text{Proposed}_{\text{V2A}}$ | $72.82_{12.57}$ | $69.89_{16.25}$ | $97.37_{0.84}$ | $65.31_{4.58}$ | $71.54_{3.13}$ | $95.54_{1.03}$ |
| $\text{Proposed}_{\text{V2B}}$ | $71.88_{13.04}$ | $71.38_{16.94}$ | $97.07_{1.26}$ | $66.19_{5.16}$ | $70.61_{1.92}$ | $95.79_{0.88}$ |
| $\text{Proposed}_{\text{V2RA}}$ | $77.00_{11.43}$ | $72.35_{16.43}$ | $97.72_{0.58}$ | $71.03_{2.75}$ | $72.04_{2.82}$ | $96.47_{0.49}$ |
| $\text{Proposed}_{\text{V2RB}}$ | $76.34_{12.13}$ | $72.06_{17.13}$ | $97.71_{0.79}$ | $70.92_{2.83}$ | $73.17_{2.93}$ | $96.58_{0.49}$ |
| $\text{Proposed}_{\text{V3A}}$ | $\mathbf{80.00}_{8.31}$ | $75.21_{14.94}$ | $\mathbf{98.07}_{0.36}$ | $\mathbf{73.64}_{3.33}$ | $75.51_{2.21}$ | $\mathbf{96.97}_{0.40}$ |
| $\text{Proposed}_{\text{V3B}}$ | $79.88_{8.98}$ | $\mathbf{75.71}_{14.88}$ | $97.91_{0.78}$ | $73.47_{3.27}$ | $\mathbf{76.51}_{3.93}$ | $96.90_{0.37}$ |

Table 6: Pairwise P-values from the Wilcoxon Signed-Rank Test comparing non-fine-tuned (A) and fine-tuned (B) variants for the Domain Generalization (DG) and Extreme Single-Domain Generalization (ESDG). P-values greater than 0.05 are shaded grey.

| Model 1 | Model 2 | DG p-value | ESDG p-value |
|---|---|---|---|
| V1A | V1B | 0.8438 | 0.8927 |
| V2A | V2B | 0.2188 | 0.5625 |
| V2RA | V2RB | 0.2188 | 0.6875 |
| V3A | V3B | 0.6875 | 0.6875 |

In Table 6, we report the p-values for the A–B version pairs of the proposed pipeline on the DG and ESDG test sets. Across all comparisons, fine-tuning the extractor features for supervised polyp segmentation does not yield significant gains. This indicates that the pretrained features are already well aligned with the requirements of the task and do not benefit meaningfully from additional supervised refinement.

## Appendix E. Computational Load

In this section, we assess the computational load between our model and the baselines. We include results for the average training time per epoch for each model and fold (Table 7), the inference time for a randomized input of dimensions (H, W, C) = (224, 224, 3) and number of parameters (Table 8).

U-Net consistently had the quickest training time per epoch in both types of folds, followed by $\text{UM-Net}_{\text{base}}$. This is the simplest architecture amongst comparisons; thus, these results are as expected. On the other hand, nnU-Net had the longest training time, followed by $\text{UM-Net}_{\text{loss}}$ and UM-Net. Overall, the proposed variants exhibited training times

Table 7: Average running time per epoch, over the first 10 epochs of training, for every fold and model. Lowest results are depicted in **bold**.

| Model | Avg $t$ p/epoch DG folds (s) | | | | | | Avg $t$ p/epoch ESDG folds (s) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 |
| U-Net | **23.3** | **23.2** | **19.0** | **24.9** | **25.1** | **28.0** | **5.9** | 7.5 | **9.4** | 8.9 | 9.0 | 5.5 |
| nnU-Net | 60.3 | 54.3 | 62.8 | 53.7 | 63.7 | 65.1 | 59.6 | 59.4 | 59.4 | 64.1 | 63.8 | 65.6 |
| UM-Net$_{\text{base}}$ | 27.8 | 29.2 | 25.6 | 27.8 | 28.8 | 33.4 | 6.0 | **7.3** | 10.7 | **5.8** | **5.3** | 3.8 |
| UM-Net | 45.8 | 45.8 | 34.3 | 45.7 | 45.9 | 50.4 | 9.3 | 10.9 | 16.2 | 8.2 | 8.1 | 4.2 |
| UM-Net$_{\text{loss}}$ | 48.6 | 49.7 | 41.5 | 52.4 | 52.7 | 61.5 | 10.1 | 14.5 | 20.3 | 10.3 | 9.2 | 5.1 |
| Proposed$_{\text{V1}}$ | 32.8 | 32.5 | 27.1 | 33.7 | 34.6 | 41.1 | 7.0 | 8.2 | 12.4 | 6.8 | 6.1 | **3.2** |
| Proposed$_{\text{V2}}$ | 31.4 | 30.9 | 26.4 | 32.4 | 33.4 | 38.4 | 6.7 | 7.8 | 12.0 | 6.4 | 5.8 | 3.3 |
| Proposed$_{\text{V2R}}$ | 30.3 | 31.5 | 26.1 | 32.3 | 34.0 | 38.8 | 6.2 | 8.1 | 13.6 | 6.6 | 6.1 | **3.2** |
| Proposed$_{\text{V3}}$ | 38.9 | 38.6 | 31.9 | 39.3 | 41.4 | 47.3 | 8.1 | 9.8 | 14.9 | 8.0 | 6.9 | 3.7 |

inferior to most baselines, offering a balanced trade-off between computational load and performance, important for clinical integration. Focusing solely on the proposed variants, we observe that the running times for V1 through V2R are similar, with a noticeable increase in time when transitioning to V3.

In general, we consider our tradeoff between performance and computational load to be efficient, as we demonstrate improved performance over SOTA models with quicker training.

In Table 8, we can observe that, while proposed variants have the highest number of parameters, inference time is lower than UM-Net and UM-Net$_{\text{loss}}$, demonstrating once again improved performance over increased computation efficiency.

Table 8: Evaluation of model size and inference time. The best result is depicted in **Bold**.

| **Model** | **Params (M)** | **Inference Time (ms)** |
|---|---|---|
| U-Net | **17.27** | **5.61** |
| nnU-Net | 59.18 | 14.57 |
| UM-Net$_{\text{base}}$ | 21.55 | 8.96 |
| UM-Net | 22.79 | 27.22 |
| UM-Net$_{\text{loss}}$ | 22.79 | 27.00 |
| Proposed$_{\text{V1}}$ | 31.47 | 11.07 |
| Proposed$_{\text{V2}}$ | 31.86 | 15.25 |
| Proposed$_{\text{V2R}}$ | 31.86 | 15.31 |
| Proposed$_{\text{V3}}$ | 31.40 | 20.94 |

## Appendix F. Evaluation Metrics

Accuracy defines the percentage of pixels of the image that were correctly classified. It ranges from [0, 1], 1 being perfect pixel-wise classification, and 0 translating that none of the pixel-wise predictions made were correct.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

where TP, FP, TN, FN represent the number of true positives, false positives, true negatives and false negatives identified.

Recall represents the proportion of TP correctly identified by the segmentation algorithm.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

It ranges from [0, 1], 1 representing that all polyp pixels in the image were correctly identified, and 0 being none.

And, finally, the DSC is given by the formula:

$$\text{DSC} = \frac{2 \cdot |\text{A} \cap \text{B}|}{|\text{A}| + |\text{B}|} = \frac{2 \cdot \text{TP}}{(2 \cdot \text{TP}) + \text{FP} + \text{FN}} \tag{5}$$

where $A$ and $B$ denote the sets of pixels corresponding to the predicted segmentation and the ground truth, respectively. $|A \cap B|$ represents the number of common pixels of both sets, while $|A|$ and $|B|$ are the sizes of the respective sets. It is a measure of the overlap of the true class in the prediction and the ground truth mask, ranging [0, 1], where 1 represents a perfect overlap.

## Appendix G. Failure cases

This section examines failure trends for both the best baseline and the best-performing proposed version. Our goal is to determine whether these failure examples exhibit any intriguing patterns that may be useful for further research. Initially, we group samples by origin center and examine the DSC distribution across DG and ESDG tests. Figures 6, 7 and 8 show these results.

It is noteworthy that samples from centers 4 and 5 achieved substantially lower average performance than those from the other centers. In particular, these centers show a much higher number of segmentation failures, suggesting that their images contain domain characteristics that remain challenging for all models and demand further investigation.

Figures 9 and 10 illustrate qualitative failure cases from centers 4 and 5. Across models, a common pattern emerges: colon folds are often misinterpreted as elongated polyps, and the reverse also occurs, with true polyps mistakenly treated as folds and left unsegmented. Another recurrent issue is that models frequently produce positive predictions even when the ground truth is empty. This is particularly pronounced for center 4, where roughly 35% of samples contain no polyps. Since negative samples are underrepresented across the dataset, the models learn a bias toward predicting non-empty masks. Additionally, these two centers contain a higher concentration of large device-related annotations, which may further degrade performance. Finally, consistent with known challenges in polyp segmentation, small and flat polyps remain among the most common failure types.

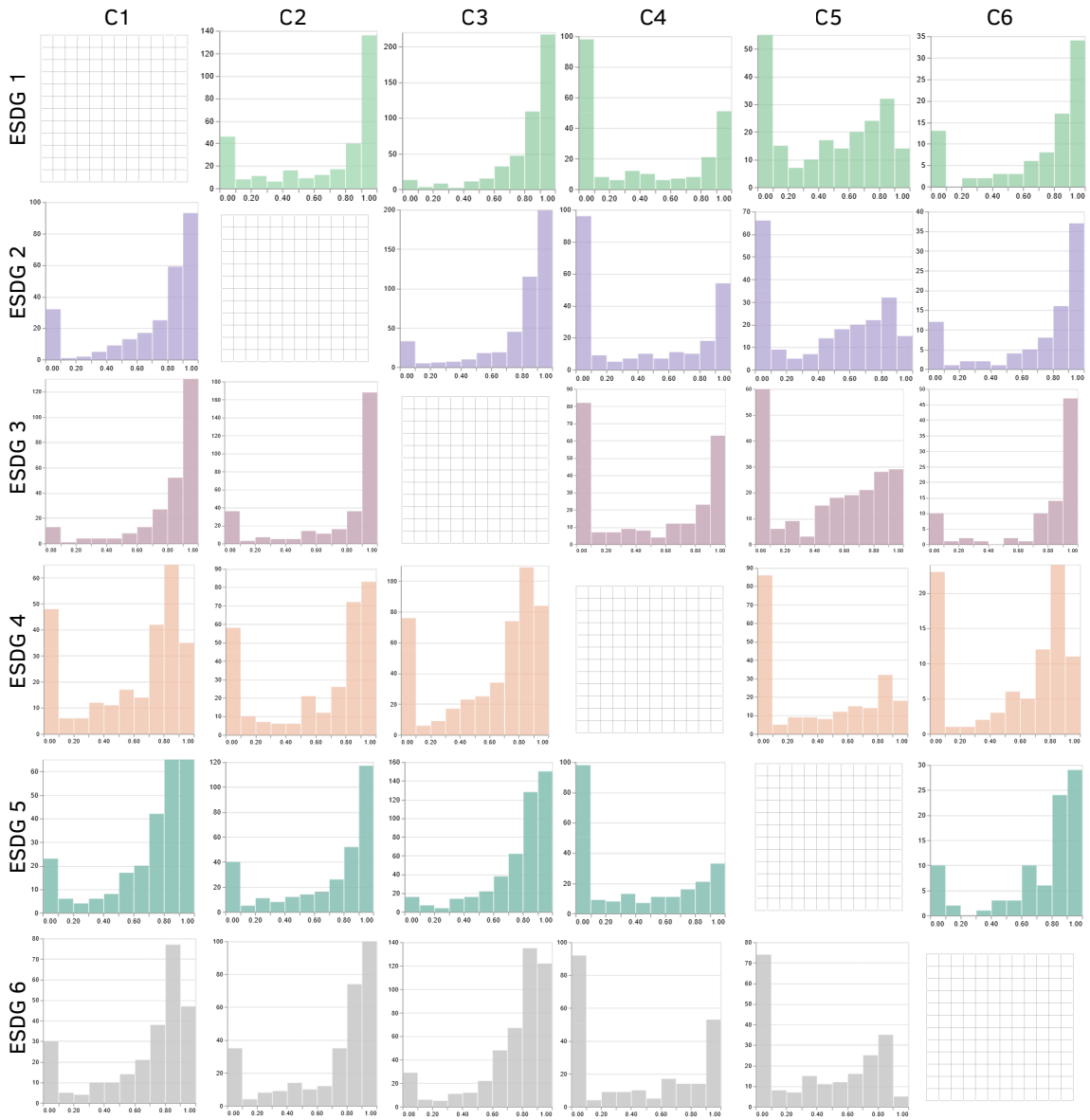


Figure 6: DSC distribution of UM-Net$_{\text{loss}}$ for each ESDG test and each center's samples.

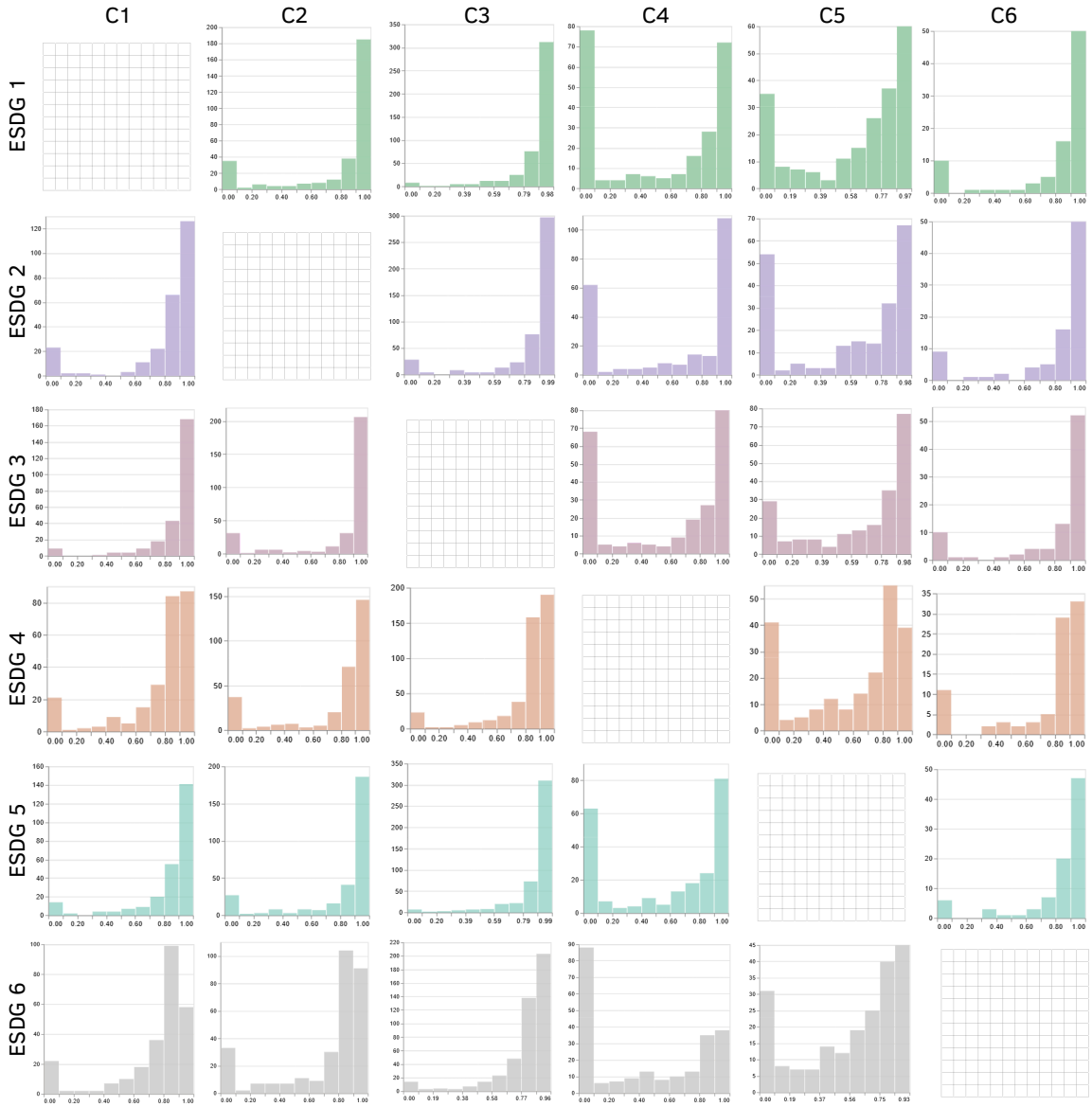


Figure 7: DSC distribution of Proposed$_{V3}$ for each ESDG test and each center's samples.

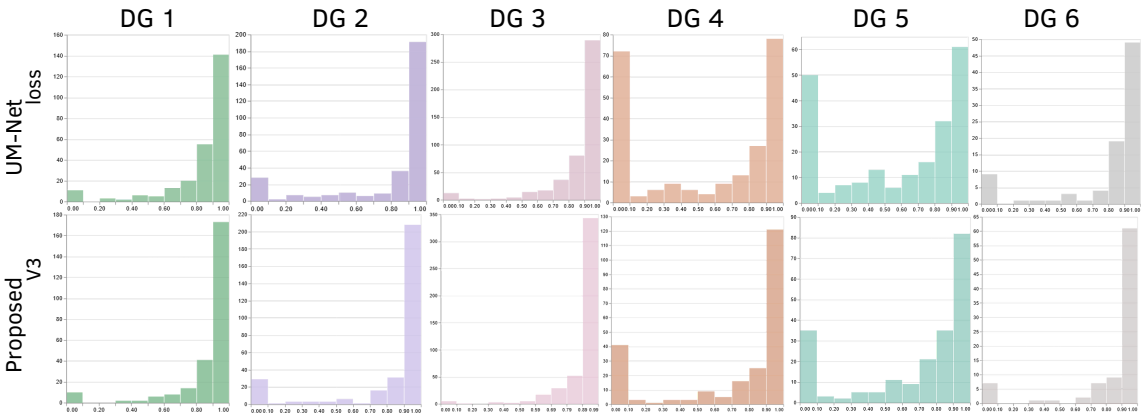


Figure 8: DSC distribution of UM-Net$_{\text{loss}}$ and Proposed$_{\text{V3}}$ for each DG test.

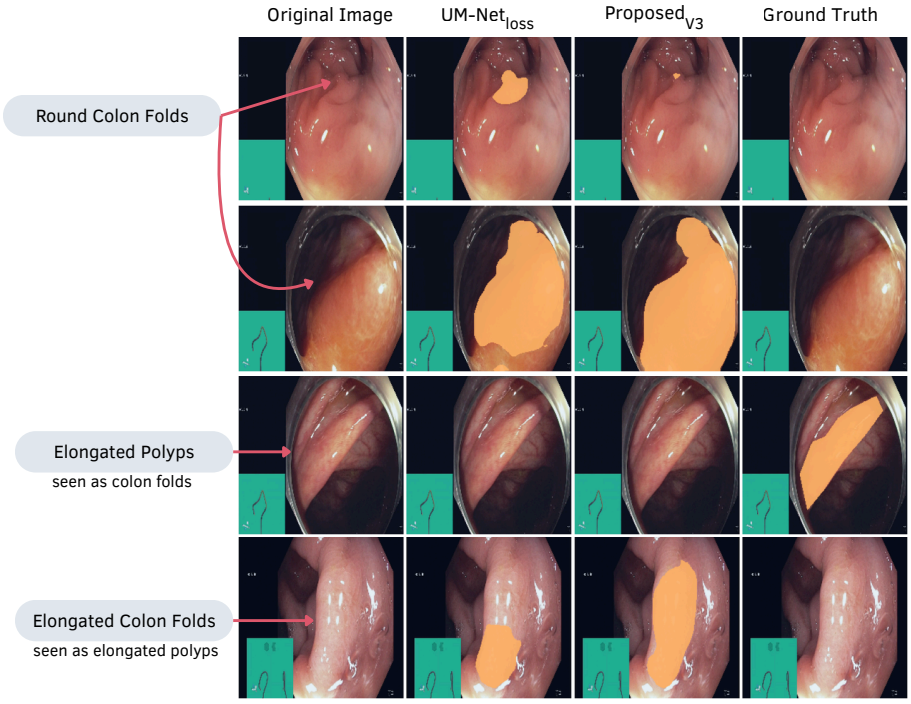


Figure 9: Examples of failure cases for UM-Net$_{\text{loss}}$ and Proposed$_{\text{V3}}$ in samples from C4.

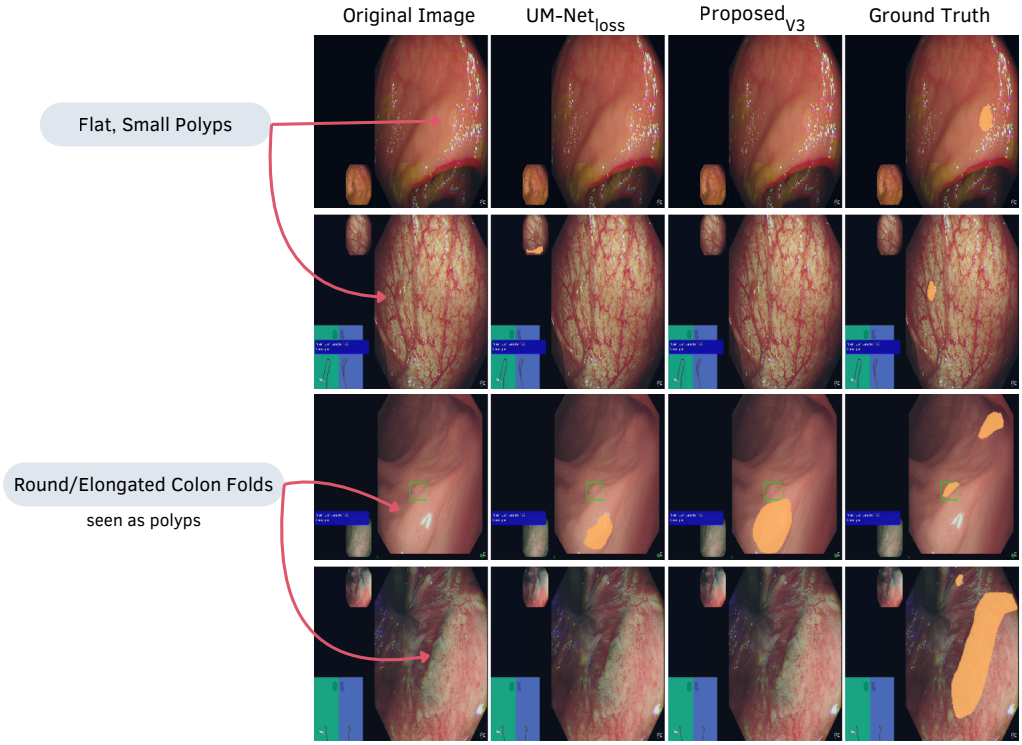


Figure 10: Examples of failure cases for UM-Net$_{\text{loss}}$ and Proposed$_{\text{V3}}$ in samples from C5.