



Task-Conditioned 3D U-Nets via Hypernetworks for Data-Scarce Medical Segmentation

Luca Hagen¹ 

Johanna P. Müller¹ 

Moritz Gmeiner¹ 

Bernhard Kainz^{1,2} 

LUCA.HAGEN@FAU.DE

JOHANNA.PAULA.MUELLER@FAU.DE

MORITZ.GMEINER@FAU.DE

BERNHARD.KAINZ@FAU.DE

¹ *Friedrich-Alexander University Erlangen-Nuremberg, GER*

² *Imperial College London, UK*

Editors: Under Review for MIDL 2026

Abstract

Training 3D segmentation models typically requires extensive expert annotation, which is costly and often unavailable for rare or low-prevalence pathologies. We propose a hypernetwork-based framework that amortises the prediction of parameters for compact 3D U-Nets, enabling task-specific specialisation from as little as a single annotated volume. By learning shared anatomical structure, such as coarse shape, scale, and spatial organisation, across organs and imaging modalities, the hypernetwork generates task-conditioned network parameters, allowing controlled adaptation to previously unseen but anatomically related targets without full retraining. We evaluate the proposed approach on the CT TotalSegmentator and Medical Segmentation Decathlon benchmarks. The method achieves strong one-shot performance for anatomically homogeneous structures (*e.g.*, liver, spleen, atrium) and demonstrates stable few-shot adaptation for more heterogeneous or low-contrast targets (*e.g.*, tumours, prostate). In regimes with two to four annotated volumes, hypernetwork-generated U-Nets consistently outperform pretrained baselines and substantially reduce the performance gap to fully supervised models while using minimal annotation. These results indicate that weight prediction serves as an effective task-informed prior for data-scarce 3D medical image segmentation.

Keywords: Few-shot learning, Hypernetworks, 3D medical image segmentation, Data-Scarcity.

1. Introduction

Automated segmentation is a core component of many radiological workflows, with neural networks enabling accurate delineation of organs, tumours, and other anatomical structures. Architectures such as the U-Net (Ronneberger et al., 2015) and, more recently, Vision Transformer variants (Dosovitskiy et al., 2020) remain widely used for 3D medical image segmentation. Despite their success, these models typically rely on large amounts of expert-annotated data, which limits their applicability for rare pathologies or uncommon anatomical targets. The standard supervised learning pipeline requires extensive data collection, careful curation, and time-consuming expert annotation (Galbusera and Cina, 2024), followed by task-specific model training. While feasible for common and routinely imaged conditions, this paradigm breaks down when target cases are scarce. In such settings, the annotation burden represents a substantial entry barrier, despite the potential

clinical value of automation. One-shot and few-shot segmentation methods aim to address this challenge by conditioning a generic model on a small number of annotated reference examples. These approaches enable generalisation to unseen anatomical targets without retraining. Indeed, [meta-learning-based](#) segmentation has already been explored in medical imaging (Khadka et al., 2022; Farshad et al., 2022; Leng et al., 2024; Alsaleh et al., 2024; Tirpude et al., 2025). More broadly, few-shot and low-shot learning in medical imaging has become an active research area, as reviewed in recent surveys (Pachetti and Colantonio, 2024; Dissanayake et al., 2025). However, despite their flexibility, these methods exhibit two fundamental limitations.

First, their performance typically remains below that of models trained specifically for a given task, even when additional annotated examples become available. Their ability to generalise across tasks often comes at the cost of limited capacity for task-specific optimisation. Second, fine-tuning is non-trivial, as task conditioning is often realised through shared internal representations or logit modulation rather than explicit parameter adaptation. As a result, updating model weights for a single task can degrade performance on others, limiting their usefulness when more data is acquired. Other lines of work attempt metric- or embedding-based segmentation under extreme scarcity, showing promising results even in one-shot scenarios (Cui et al., 2021). Approaches that rely on self-supervision and anomaly detection have also been proposed to mitigate the problem of limited foreground/background discrimination in few-shot settings (Hansen et al., 2022). These efforts demonstrate the growing recognition of data scarcity in medical segmentation and the need for more robust, adaptable methods. We address these limitations by formulating task conditioning as an explicit parameter-prediction problem. We propose a hypernetwork that amortises the generation of parameters for compact 3D U-Nets, conditioned on limited annotated data. Trained across a diverse set of segmentation tasks, the hypernetwork learns shared anatomical structure such as coarse shape, scale, and spatial organisation. By predicting weights rather than modulating logits or embeddings, the approach produces fully instantiated, task-specific U-Nets that are decoupled from the hypernetwork after generation. This separation allows the generated U-Nets to be fine-tuned using standard optimisation as additional data becomes available, without interfering with the learned task prior or other tasks. As a result, the proposed method combines the data efficiency of few-shot learning with the robustness and interpretability of conventional segmentation models, enabling controlled adaptation to anatomically related targets under limited supervision.

Contributions. This work makes the following contributions: (1) A hypernetwork framework that generates task-specific 3D U-Net parameters from minimal annotated data for one- and few-shot segmentation of unseen targets. (2) Task conditioning via explicit weight prediction, producing fully instantiated U-Nets that can be fine-tuned independently of the hypernetwork. (3) Demonstration that the hypernetwork captures shared anatomical structure across organs and modalities, providing task-informed priors for data-efficient segmentation. (4) Extensive evaluation on CT TotalSegmentator and [the Medical Segmentation Decathlon](#) [shows that HYPERUNET achieves](#) strong one- and few-shot performance [across diverse anatomical targets with substantially reduced](#) annotation requirements.

[Unlike SAM-style prompting and in-context learning methods that perform reference-conditioned inference at test time, HYPERUNET synthesises a standalone, task-specific 3D U-Net,](#)

enabling efficient deployment and incremental fine-tuning without increasing inference-time memory or computation.

2. Background

Few-shot models have become a central strategy for medical image segmentation under data scarcity. Given only a few annotated reference volumes, these models adapt to new tasks by exploiting information extracted from the support set. Existing approaches can be broadly grouped into three paradigms. Similarity- and prototype-based methods classify query voxels by matching pixel-wise embeddings to prototypes constructed from the support set (Cui et al., 2021). While effective for anatomically homogeneous structures, their performance depends heavily on the expressiveness of the embedding space. As a result, they struggle with complex 3D anatomy, high intra-class variability, and the inherently difficult foreground-background separation in low-shot regimes (Cui et al., 2021; Hansen et al., 2022). Attention-based models use cross-attention to align query features to support features (Galbusera and Cina, 2024; Hu et al., 2025). This enables strong appearance transfer but tightly couples the inference procedure to the support examples. Since these models do not instantiate standalone segmentation models, they cannot be easily fine-tuned or deployed independently of the conditioning samples. Parameter-based meta-learning focuses on learning how to adapt a segmentation model from few reference examples. MAML (Finn et al., 2017) and its variants (Khadka et al., 2022; Leng et al., 2024; Alsaleh et al., 2024) learn shared initialisations that are rapidly fine-tuned to new tasks. Volumetric extensions show generalisation across heterogeneous targets (Farshad et al., 2022; Tirpude et al., 2025). However, because task adaptation is restricted to a few gradient steps, the resulting models remain bound to a local neighbourhood of the meta-initialisation, ultimately limiting their flexibility. First introduced in (Ha et al., 2016), hypernetworks generate the parameters of another model directly in a forward pass. They have been used for fast task-specific updates (Przewikezlikowski et al., 2024), full model parameterisations (Zhmoginov et al., 2022), and sample-conditioned filter generation (Nirkin et al., 2021). Unlike MAML-based methods, which rely on solutions near a shared initialisation, hypernetworks can represent a broader family of task-specific models. This distinction matters in few-shot 3D segmentation. MAML requires multiple fine-tuning steps, substantial per-task supervision, and operates on closely related tasks and organs. Our approach instead generates a complete task-specific U-Net in a single forward pass and applies only one light update on the generated model, leaving the hypernetwork unchanged. This enables stable, modular adaptation and effective generalisation to genuinely unseen anatomical targets from as little as one annotated volume. Motivated by these advantages, we adopt a hypernetwork to predict autonomous, fine-tunable segmentation models. This generate-once paradigm also differentiates our method from prominent in-context learning segmentation frameworks, such as UniverSeg (Butoi et al., 2023) and Iris (Gao et al., 2025), as well as promptable segmentation approaches such as MedSAM (Cheng et al., 2023; Wang et al., 2025). Similar to attention-based retrieval mechanisms, these methods maintain a tight coupling to a reference set at inference time: predictions are obtained by explicitly conditioning on, or comparing against, the provided context examples. As a consequence, incorporating

additional supervision is typically non-trivial and often requires re-running the full inference procedure with an expanded context.

In contrast, our approach compiles decision rules from a reference sample *once* in the form of a full U-Net decoder, yielding a standalone segmentation model that can be deployed independently of the original reference set. Crucially, when further annotations become available, the resulting model can be adapted using standard backpropagation, without modifying the underlying hypernetwork pipeline.

3. Method

To tackle data-scarce 3D medical image segmentation, we propose a hypernetwork framework that generates task-specific parameters for compact 3D U-Nets. Instead of feature- or logit-based conditioning, our method treats task adaptation as explicit weight prediction, producing fully instantiated U-Nets that can be fine-tuned independently. By training across diverse segmentation tasks, the hypernetwork captures shared anatomical structures, such as shape, scale, and spatial organisation, enabling adaptation to unseen but anatomically related targets with minimal annotated data. We first formalise the problem and then detail the architecture, training strategy, and task-specific parameter generation.

Problem Formulation. We focus on binary 3D medical image segmentation tasks. Each task \mathcal{T} is defined by a dataset $D = \{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R}^{H \times W \times D}$ is a 3D image volume and $y_i \in \{0, 1\}^{H \times W \times D}$ is its corresponding binary mask. For multi-class datasets with labels $y_i \in \{0, \dots, K\}^{H \times W \times D}$, we decompose the problem into K binary segmentation tasks $\{\mathcal{T}_k\}_{k=1}^K$, one per foreground class. When volumes have multiple channels (e.g., different MRI sequences), each channel-class combination is treated as a separate binary task. Given a small support set of one or a few reference volume-mask pairs for a task \mathcal{T} , our goal is to predict the weights of a compact 3D U-Net that can be deployed as an autonomous segmentation model for that task. By formulating task adaptation as explicit parameter prediction, rather than feature- or logit-based conditioning, we aim to produce models that are both task-specific and fine-tunable without affecting other tasks.

Target Architecture. Our target network is a compact 3D U-Net composed of four encoder and four decoder stages, using a residual double-convolution block (ResDoubleConv) as the primary building unit. Each block contains two $3 \times 3 \times 3$ convolutions with group normalisation and LeakyReLU activations, alongside a $1 \times 1 \times 1$ residual branch for stable gradient flow. Downsampling in the encoder is implemented via strided $3 \times 3 \times 3$ convolutions, while upsampling in the decoder uses transposed convolutions. To reduce the number of parameters, skip connections between the encoder and decoder stages are additive rather than concatenative. A final $1 \times 1 \times 1$ convolution maps decoder features to segmentation logits. Across tasks, the encoder, normalisation layers, and upsampling layers are shared and trained jointly. Task-specific adaptation is achieved by a hypernetwork that generates all convolutional weights in the decoder ResDoubleConv blocks and the final output head. This design produces compact, fully instantiated U-Nets with approximately 5.7 million trainable parameters, combining efficiency, interpretability, and the ability to fine-tune per task without affecting other tasks.

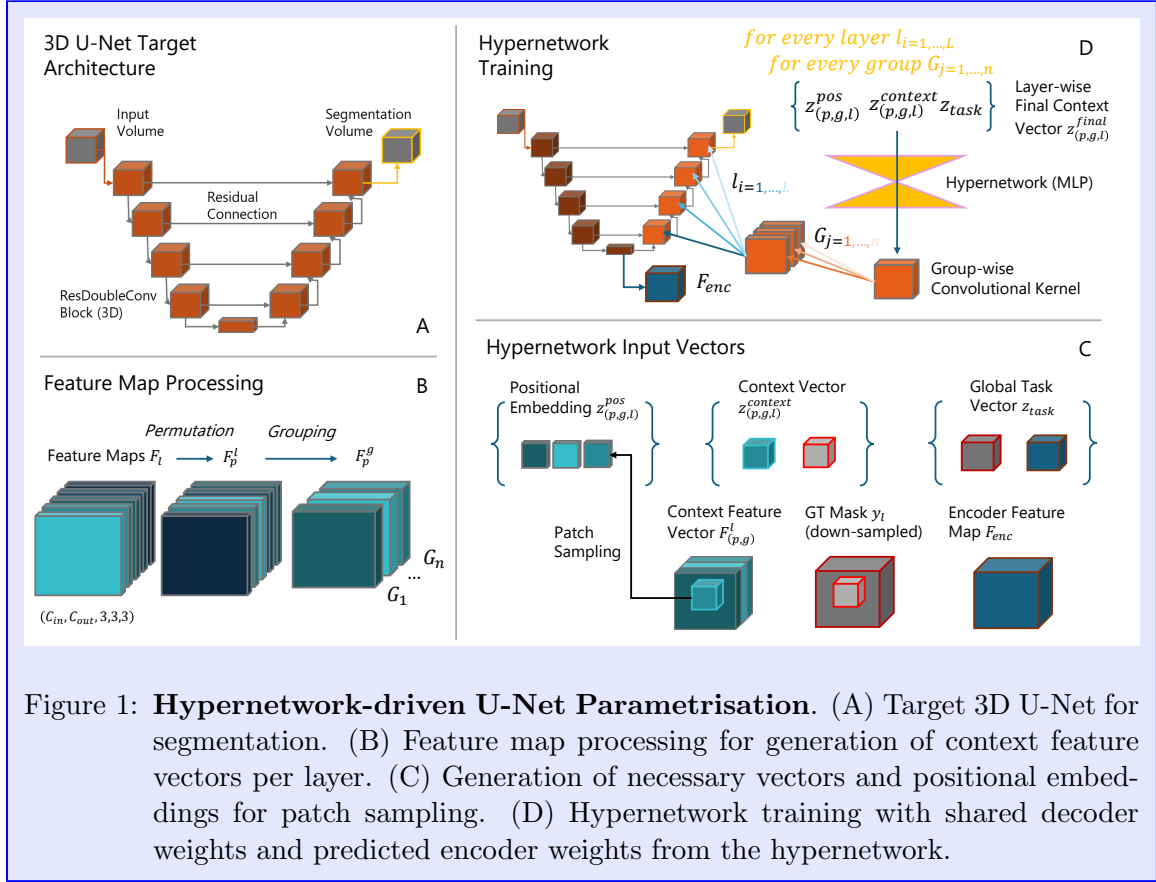


Figure 1: **Hypernetwork-driven U-Net Parametrisation.** (A) Target 3D U-Net for segmentation. (B) Feature map processing for generation of context feature vectors per layer. (C) Generation of necessary vectors and positional embeddings for patch sampling. (D) Hypernetwork training with shared decoder weights and predicted encoder weights from the hypernetwork.

Hypernetwork-based U-Net Generation. Given a reference volume-mask pair (x, y) for a task \mathcal{T} , the hypernetwork generates all decoder weights of the corresponding target U-Net in a single forward pass. This allows the resulting U-Net to be fully instantiated and fine-tuned independently of the hypernetwork. We first encode the input volume x using the shared encoder E to obtain a high-level feature map $F_{enc} = E(x) \in \mathbb{R}^{C \times H' \times W' \times D'}$. To summarise the task, we extract a global descriptor from the reference annotation by applying mask average pooling over the foreground region:

$$z_{task} = \text{AvgPool}(\text{Upsample}(F_{enc}) \odot y) \in \mathbb{R}^{1 \times C}, \quad (1)$$

where \odot denotes elementwise multiplication, and $\text{Upsample}(\cdot)$ resamples the feature map back to the original resolution. This descriptor z_{task} encodes the spatial and semantic information of the task and is subsequently used by the hypernetwork to generate decoder weights.

3.1. Context Vector Construction for Decoder Convolutions.

In our setup, the weights of each convolutional layer are generated by a forward pass of a reference pair (x, y) and on demand, i.e., exactly at the moment when the respective layer

is required during the forward pass. Consequently, the convolutional layers of the decoder are instantiated **sequentially**.

First, we compute the encoder feature map F_{enc} , it can be directly forwarded into the decoder. At this stage, the convolutional layers of the decoder are all uninitialized, while non-parametric components such as upsampling and normalisation layers can be applied as usual. Following the standard U-Net decoder structure, the first convolutional decoder layer is encountered after the initial upsampling operation and the incorporation of the corresponding encoder skip connection. We denote the resulting feature map, i.e., the input that would normally be passed through the convolutional layer, as F^l . In our notation later, feature maps F_l will be indexed by their respective convolutional decoder layer $l \in L$. In the following, we describe the context generation process for a decoder convolutional layer. In our setup, each generated convolutional layer have matching input/output channel size (in-channels = out-channels). Let C_l denote the channel size of our current convolutional layer, then we have to generate a weight tensor of shape $(C_l, C_l, 3, 3, 3)$. Instead of generating $27 \times C_l^2$ parameters directly, we generate them in chunks of size $27 \times C_{base}^2$, where C_{base} is a hyperparameter such that $C_l \bmod C_{base} = 0$ (in our setup C_{base} is chosen to be the same as the number of our U-Net base-channels; 24). Consequently, we have to generate $G = C_l / C_{base}$ different chunks of weights, where each chunk corresponds to one context vector and is obtained by passing the respective context vector through the hypernetwork.

Permutations. To construct the context vectors, we first create G copies of F_l and permute each of them individually over the channel dimension.

Channel partitioning. Each of these permuted feature maps $\{F_p^l\}_{p=1}^G$, are then further partitioned into G non-overlapping groups along the channel dimension. We denote the resulting groups as $F_p^g \in \mathbb{R}^{C_{base} \times h \times w \times d}$, with $p \in \{1, \dots, G\}$ and $g \in \{1, \dots, G\}$. Note that due to the permutation step, each of the C_{base}^2 different $F_{(p,g)}^l$'s is a unique combination of channels from the feature map F_{in} .

Mask Downsampling. As a next step, we downsample the binary support mask Y to match the spatial dimension of our $F_{(p,g)}^l$'s with $y_{down} = \text{Downsample}(y) \in \{0, 1\}^{h \times w \times d}$.

Patch Sampling. For each $F_{(p,g)}^l$ we now sample N local patches of size $3 \times 3 \times 3$

$$p_n^{(i)} \in \mathbb{R}^{C_{base} \times 3 \times 3 \times 3}, \quad n = 1, \dots, N, \quad (2)$$

from $F_{(p,g)}^l$. We sample such that $N/2$ of the resulting patches are positive and $N/2$ are negative. Additionally, our sampling is boundary-aware: For both, negative and positive patches, we ensure that at least a fixed fraction b (0.7 in our experiments) of them is a boundary patch, meaning that when looking at the respective mask-patch from y_{down} , both foreground and background are present. Positive and negative patches are labelled with respect to their centre voxel in y_{down} , meaning a negative patch has a 0 in its centre voxel when looking into the respective $3 \times 3 \times 3$ patch in y_{down} . The rationale behind using local patches is to show our hypernetwork multiple small local examples of foreground and background regions, while laying special emphasis on boundary regions, which, as we argue, hold the most information for distinction. In comparison to using voxels, patches have the advantage that they not only carry information about features but also the feature-gradient.

We stack the resulting patches in a fixed order [positive \rightarrow positive-boundary \rightarrow negative \rightarrow negative-boundary]. Flattening and concatenating the patches yields the context vector $z_{(p,g,l)}^{\text{context}} \in \mathbb{R}^{C_{\text{base}} \cdot 27 \cdot N}$ of $F_{(p,g)}^l$.

Task-aware positional embedding . While each of the $z_{(p,g,l)}^{\text{context}}$'s now carries specific information how to separate foreground from background for its group F_p^g , it does not hold any information about the position in which its resulting weights will be employed in, in the decoder architecture. To enrich the context vector z with such positional information, we assign a learnable positional embedding $z_{(p,g,l)}^{\text{pos}}$ to each group $F_{(p,g)}^l$ from decoder layer l . As we argue, that each group F_p^g can have different roles for different tasks, we enrich these $z_{(p,g,l)}^{\text{pos}}$ with our global task vector z_{task} by merging them through a small two-layer MLP:

$$z_{(p,g,l)}^* = \text{MLP}(\text{Concat}[z_{\text{task}}, z_{(p,g,l)}^{\text{pos}}]). \quad (3)$$

Final context vector. Finally, we can give the context-, position- and task-aware input vector for our hypernetwork as:

$$z_{(p,g,l)}^{\text{final}} = \text{Concat}[z_{p,g}^{\text{context}}, z_{(p,g,l)}^*] \quad (4)$$

which are then passed into the hypernetwork to generate the weights for the layer l .

Hypernetwork. The hypernetwork H is implemented as a three-layer MLP with one hidden layer of dimension 2048 and GELU activations. It maps each final context vector $z_{(p,g,l)}^{\text{final}}$ to a base convolutional filter of shape $(C_{\text{base}}, C_{\text{base}}, 3, 3, 3)$. Layer normalisation is applied after the first two linear layers to stabilise training and improve convergence. Once all base convolutional filters for layer l are generated, they are stacked along the input and output channel dimensions to produce a full weight matrix W_l for the convolutional kernel of shape $(C_{\text{out}}, C_{\text{in}}, 3, 3, 3)$, with $C_{\text{in}} = C_{\text{out}} = G \cdot C_{\text{base}}$. As soon as W_l is obtained, we can pass the current feature map F_l through the convolution parameterised by W_l . The obtained feature map F_{out} is further passed through the decoder until the next uninitialised layer is reached. The weights and the bias of the final $1 \times 1 \times 1$ segmentation head are generated by a separate two-layer MLP conditioned on the global task descriptor z_{task} . This design produces a fully instantiated decoder that is both task-specific and ready for independent fine-tuning. Once a forward pass through the entire decoder is complete, all layers are initialised, and then we can treat the model as a standard 3D U-Net.

Reference-based weight update. Once the decoder weights are generated, they are combined with the shared encoder to obtain a complete task-specific U-Net. To further leverage the information from the reference pair (x, y) , we perform a single gradient update on the U-Net parameters using a small learning rate, while keeping the hypernetwork fixed. The resulting model is a fully instantiated, task-specific U-Net that can be used directly for inference or as a task-informed initialisation for further fine-tuning when additional labelled data becomes available. This separation ensures that the hypernetwork retains its ability to generalise across tasks, while the generated U-Net can specialise to the current task.

Algorithm 1 Episodic Training for Hypernetwork-predicted U-Nets

```

1: Initialize episode counter  $episode = 1$ 
2: while  $episode \leq N_{\text{episodes}}$  do
3:   Sample a mini-batch  $\mathcal{B}$  of tasks
4:   for each task  $\mathcal{T}$  in  $\mathcal{B}$  do
5:     Randomly sample a support pair  $(x_s, y_s)$  for task  $\mathcal{T}$ 
6:     Generate task-specific U-Net:  $F_{\mathcal{T}} = \text{Hypernetwork}(x_s, y_s)$ 
7:     Sample an independent query pair  $(x_q, y_q)$  for the same task
8:     Predict outputs:  $\hat{y}_s = F_{\mathcal{T}}(x_s)$ ,  $\hat{y}_q = F_{\mathcal{T}}(x_q)$ 
9:     Compute combined loss:
10:     $\mathcal{L} = \beta L_{\text{seg}}(\hat{y}_s, y_s) + (1 - \beta) L_{\text{seg}}(\hat{y}_q, y_q)$ 
11:   end for
12:   Update parameters of  $E$ ,  $H$ , and shared decoder layers using AdamW with exponentially decaying learning rate
13:    $episode = episode + 1$ 
14: end while
15: Apply early stopping based on validation performance on unseen tasks

```

Episodic Training Objective We train the encoder E , hypernetwork H , and shared decoder components end-to-end in an episodic meta-learning setup. Each episode contains multiple tasks with support-query pairs. The training procedure is summarised in Algorithm 1.

4. Experiments

We assess our framework on tasks entirely unseen during training by generating a HYPERUNET for each new target. Performance is compared against three baselines sharing the same U-Net architecture: (i) a conventionally trained, fully supervised model for the given target task (Conv U-Net; upper bound), (ii) a model using the shared encoder, upsampling, and normalisation layers but randomised decoder weights (Rand U-Net), and (iii) a model trained jointly on all tasks including target tasks (All-classes U-Net) across both datasets, covering over 130 anatomical targets across CT and multiple MRI sequences. The All-classes U-Net baseline serves to assess whether a single multi-task model with comparable capacity can absorb knowledge across tasks under partial labels, and to contrast this with our task-conditioned decoder generation. In addition, we report three ablation studies for our method, detailed in Appendix A.

Datasets. We train and evaluate on two public benchmarks selected for anatomical diversity and suitability for episodic meta-learning. CT TotalSegmentator (Wasserthal et al., 2023) provides voxel-wise annotations for over 100 thoraco-abdominal structures and is used to define a large set of binary training tasks spanning organs, vessels, bones, and soft tissue. The Medical Segmentation Decathlon (MSD (Antonelli et al., 2022) consists of 10 heterogeneous CT and MRI datasets covering diverse anatomies and pathologies. In MSD, all evaluation targets are defined as hold-out tasks that are never used during hypernetwork

Table 1: **Organ- and Task-Shift results.** Dice (DSC) and Normalised Surface Dice (NSD) for Random U-Net, Conventional U-Net, and **1-shot HYPERUNET** (Ours) across all CT tasks. *Models trained on target tasks. **Best** and second-best per task are highlighted.

Network	Liver		Spleen		Hepatic Vessels	
	DSC	NSD	DSC	NSD	DSC	NSD
Rand. U-Net	0.176±0.050	0.110±0.025	0.091±0.052	0.045±0.017	0.019±0.010	0.032±0.012
All-classes U-Net*	0.826±0.073	0.668±0.106	0.580±0.167	0.509±0.123	0.050±0.023	0.038±0.114
HYPERUNET (Ours)	0.772±0.063	0.490±0.080	0.501±0.169	0.316±0.115	0.307±0.120	0.364±0.129
Conv. U-Net*	0.951±0.018	0.792±0.069	0.911±0.032	0.708±0.108	0.607±0.104	0.731±0.103
Network	Pancreas		Lung Tumor		Colon Tumor	
	DSC	NSD	DSC	NSD	DSC	NSD
Rand. U-Net	0.016±0.007	0.017±0.006	0.006±0.014	0.005±0.006	0.012±0.017	0.010±0.010
All-classes U-Net*	0.282±0.171	0.174±0.132	0.120±0.181	0.079±0.044	0.070±0.115	0.046±0.098
HYPERUNET (Ours)	0.153±0.138	0.102±0.078	0.025±0.067	0.019±0.058	0.050±0.126	0.046±0.109
Conv. U-Net*	0.776±0.076	0.632±0.118	0.439±0.268	0.340±0.307	0.389±0.287	0.277±0.171

training and serve exclusively for out-of-distribution evaluation across both CT and MRI domains.

Preprocessing and Training. All volumes are resampled to isotropic 1.5 mm spacing and reoriented to the RAS coordinate system. CT intensities are clipped to $[-900, 900]$ and Z-score normalised, while MRI volumes are normalised by mapping the 1st to 99th intensity percentiles and clipping outliers. A single 128^3 voxel patch covering the target structure is extracted per volume and centred when necessary. On-the-fly spatial (flips, 90° rotations, affine transforms) and intensity augmentations (Gaussian noise, smoothing, contrast adjustments) are applied. Models are trained for 20,000 episodes using a combined Dice and binary cross-entropy loss with $\beta = 0.2$, optimised with AdamW and an exponentially decaying learning rate. Early stopping is based on validation performance on unseen tasks. Training is performed on four NVIDIA A100 GPUs for approximately 12 hours.

CT Domain. First, we investigate the capabilities of our approach on unseen CT tasks. As our model was trained exclusively in the CT domain, this experiment examines the models’ abilities to generalise to adapt to unseen organs / new tasks. We evaluate our hypernetwork on several tasks from the MSD dataset and report both the Dice Score (DSC) and the normalised surface distance (NSD) with a tolerance of $\tau = 2$ mm. As shown in Fig. 1, a consistent trend emerges: HYPERUNETs substantially outperform the randomised baseline across all tasks, yet remain well below the performance of a fully supervised U-Net. Notably, HYPERUNETs outperform the jointly trained All-classes U-Net on the hepatic vessels task (DSC 0.31 vs. 0.05), highlighting limitations of a single multi-task model under partial-label supervision. In this setting, rare and fine-grained structures such as hepatic vessels receive weak and infrequent supervision and are affected by gradient interference from competing tasks, leading to unstable or degraded performance. The strongest results are observed for larger and anatomically homogeneous structures such as the liver and spleen, indicating that the hypernetwork captures coarse shape and spatial priors effectively. For more complex or fine-grained structures (e.g., pancreas, hepatic vessels), performance

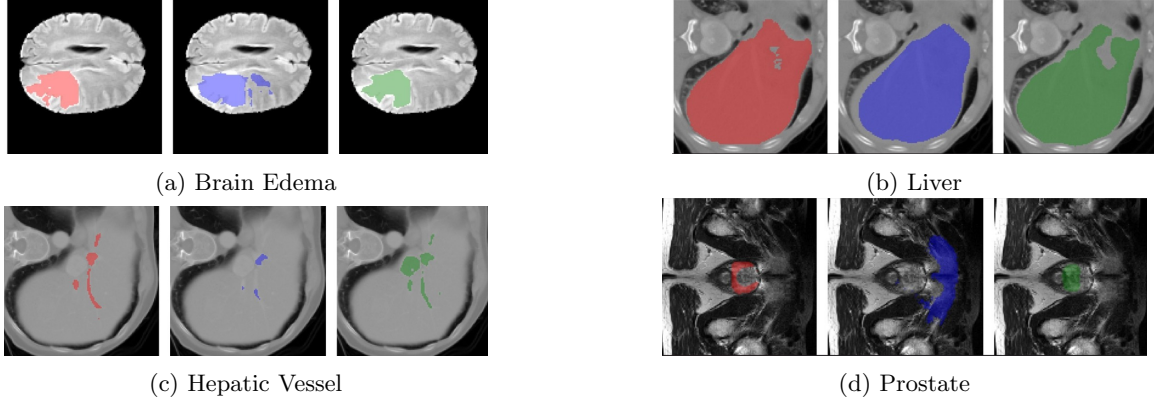


Figure 2: Qualitative evaluation on four target-tasks. Ground truth (red), and predictions of HYPERUNET (blue) and Task-specific Conv. U-Net (green).

Table 2: **Domain-, Organ-, and Task-Shift results.** Dice (DSC) and Normalised Surface Dice (NSD) for Random U-Net, Conventional U-Net, and **1-shot HYPERUNET** (Ours) across all MRI tasks. *Models trained on target tasks. **Best** and second-best per task highlighted.

Network	Brain Edema (FLAIR)		Hippocampus (T1w)	
	DSC	NSD	DSC	NSD
Rand. U-Net	0.002±0.002	0.028±0.016	0.002±0.001	0.001±0.001
All-classes U-Net*	0.508±0.195	0.436±0.167	<u>0.008±0.001</u>	<u>0.001±0.002</u>
HYPERUNET (Ours)	<u>0.326±0.208</u>	<u>0.217±0.150</u>	0.057±0.021	0.075±0.016
Conv. U-Net*	0.617±0.159	0.584±0.109	0.856±0.030	0.970±0.034
Network	Heart (bSSFP)		Prostate (T2)	
	DSC	NSD	DSC	NSD
Rand. U-Net	0.022±0.005	0.033±0.005	0.004±0.005	0.007±0.002
All-classes U-Net*	<u>0.399±0.101</u>	0.311±0.143	0.292±0.217	0.216 ±0.084
HYPERUNET (Ours)	0.500±0.119	<u>0.281±0.071</u>	<u>0.053±0.115</u>	<u>0.053±0.108</u>
Conv. U-Net*	0.859±0.034	0.619±0.111	0.468±0.089	0.364±0.056

drops relative to fully supervised training but still clearly exceeds a random baseline. For highly heterogeneous targets such as lung and colon tumours, both HYPERUNETs and the All-classes U-Net show limited performance gains, reflecting the difficulty of learning robust tumour representations under partial labels and extreme inter-case variability when only a single support example is available.

MRI Domain. In this experiment, in addition to a organ-/task-shift, we introduce a modality shift. Our hypernetwork approach was trained exclusively on CT tasks, but is now evaluated on unseen organs in the MRI domain, presenting a severe challenge. For evaluation, we again use MSD tasks and report DSC and NSD. As shown in Fig. 2, the results mirror the CT experiments: HYPERUNETs achieve strong performance on Heart and

Brain Edema, demonstrating that the hypernetwork can generate meaningful parameters for targets unseen during training and even from an unseen imaging domain. Notably, they outperform the U-Net trained on all tasks on the Heart task (0.50 vs. 0.40). The overall pattern, however, remains consistent: for large, well-defined structures, HYPERUNETs clearly exceed the random baseline, whereas for small or complex targets (e.g., hippocampus, prostate peripheral zone) they offer only marginal gains over random initialisation.

Limited Data. Further, we study how HYPERUNETs behave when more than a single reference pair is available. Instead of treating the synthesised decoder as a final segmenter, we view it as a task-informed initialisation that can be refined once additional supervision becomes available. Given N annotated samples, we form a minimal train-test split by holding out 20% of the data (at least one sample) for evaluation. A HYPERUNET is synthesised from one reference pair in the training subset and then fine-tuned on the remaining training samples using standard backpropagation with early stopping. We evaluate multiple MSD tasks for $N \in \{2, 4, 8, 16\}$, corresponding to (1/1), (3/1), (6/2), and (13/3) train/test splits.

As shown in Fig. 3, HYPERUNETs can be effectively fine-tuned with as little as one additional annotated sample, where conventional U-Nets often overfit. Performance improves consistently with more data and gradually approaches fully supervised training.

Finally, we report MAML on the spleen task: despite differing shot counts (5 and 10), it performs comparably to HYPERUNET. Appendix A (Tab. 7) further compares HYPERUNETs synthesised from five references to a 5-shot MAML baseline.

Table 3: **DSC [%] of our HyperUNet and SOTA (Gao et al., 2025) on MSD Pancreas.**

<i>Ours</i>	<i>SAM-Style/Pos. Prompts</i>		<i>In-context</i>			
HYPERUNET	SAM-Med2D	SAM-Med3D	SegGPT	UniverSeg	Tyche-IS	Iris
0.153	0.104	0.158	0.107	0.103	0.120	0.283

SOTA and similar approaches. We compare HYPERUNET to recent prompting-based and in-context learning approaches. In contrast to SAM-style prompting (Cheng et al., 2023; Wang et al., 2025) and ICL methods such as UniVerSeg (Butoi et al., 2023), SegGPT (Wang et al., 2023), Tyche-IS (Rakic et al., 2024), and Iris (Gao et al., 2025), which perform reference-conditioned inference by matching each query to a fixed support set, HYPERUNET synthesises a standalone, task-specific 3D U-Net. The generated model can be deployed directly or fine-tuned with standard supervised learning, without increasing inference-time memory or computation.

This benefit is reflected in Table 3. On MSD Pancreas, a HYPERUNET generated from a single reference pair reaches 0.153 Dice, outperforming SAM-style and most ICL baselines. While Iris achieves higher performance (0.283), it relies on extensive support, query matching and retains the full support set at inference time. By compressing task information into model weights, HYPERUNET enables efficient deployment and incremental

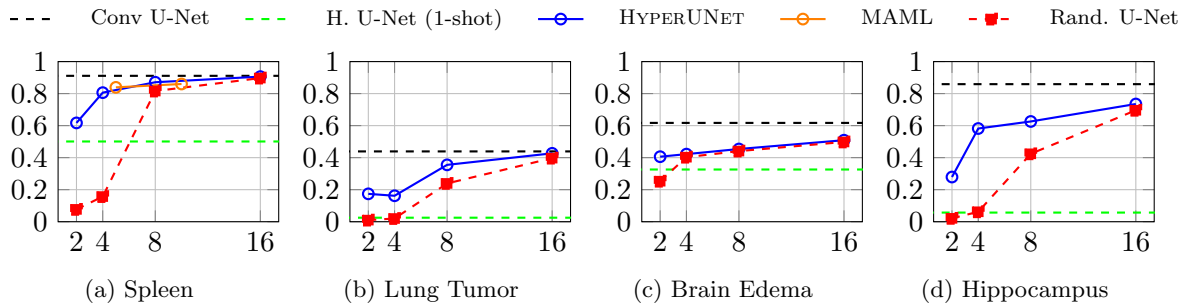


Figure 3: DSC of HyperUNets (blue) and Random U-Nets (red) trained using $N \in \{2, 4, 8, 16\}$ samples across several tasks. A U-Net trained on the entire dataset (black) and a one-shot generated HYPERUNET (green) are reported as baselines. For Spleen, MAML (orange) is reported as an additional baseline.

refinement as more annotations become available. The efficiency of the proposed generate-once paradigm is further evidenced by the inference-time analysis reported in Table A.6 (Appendix A). After an initial weight generation step (1.77 s), HYPERUNET achieves a substantially lower per-sample inference time than prompting-based and in-context learning methods, reducing inference latency from 2.0 s (previous best) to 0.06 s per sample.

5. Conclusion

In this work, we introduce HYPERUNET, a hypernetwork framework that generates compact, task-specific 3D U-Nets from as little as a single annotated volume. By predicting decoder weights directly, rather than conditioning a shared backbone at the feature or logit level, HYPERUNET produces fully instantiated segmentation models that can be deployed immediately or refined via lightweight fine-tuning, shifting task adaptation from gradient-based meta-learning to weight-level conditioning. Experiments on CT and MRI tasks from the MSD show that one-shot weight generation is effective for anatomically homogeneous structures such as liver, spleen, and cardiac chambers, and that a hypernetwork trained solely on CT generalises to MRI by capturing modality-robust anatomical priors. For more complex or fine-grained structures, including hepatic vessels and tumours, performance degrades in the strict one-shot setting, reflecting an over-reliance on generic anatomical priors (e.g. location or coarse shape) that act as distractors for heterogeneous pathologies. However, treating the generated U-Nets as task-informed initialisations and fine-tuning on as few as 2-4 labelled volumes yields substantial gains, outperforming pre-trained and randomly initialised baselines and closing much of the gap to fully supervised models. Overall, HYPERUNET offers an interesting alternative to gradient-based meta-learning by synthesising a complete task-specific segmentation model in a single forward pass, requiring only minimal additional optimisation. By explicitly separating shared priors from task-specific decoder synthesis, our method enables stable adaptation to previously unseen targets under severe annotation scarcity and occupies a distinct point in the design space. We view this work as an initial exploration of a complementary regime between in-context segmentation and fully trained task-specific models, and we hope it will motivate

further research in this direction. Future work will investigate task-dependent weighting of different priors (e.g., topology, location, texture, and intensity), richer context representations, alternative encoder architectures, and extensions to additional modalities such as PET and ultrasound.

References

- Aqilah M Alsaleh, Eid Albalawi, Abdulelah Algosaibi, Salman S Albakheet, and Surbhi Bhatta Khan. Few-shot learning for medical image segmentation using 3d u-net and model-agnostic meta-learning (maml). *Diagnostics*, 14(12):1213, 2024.
- Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. The medical segmentation decathlon. *Nature communications*, 13(1):4128, 2022.
- Victor Ion Butoi, Jose Javier Gonzalez Ortiz, Tianyu Ma, Mert R Sabuncu, John Guttag, and Adrian V Dalca. Universeg: Universal medical image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21438–21451, 2023.
- Dongjie Cheng, Ziyuan Qin, Zekun Jiang, Shaoting Zhang, Qicheng Lao, and Kang Li. Sam on medical images: A comprehensive study on three prompt modes. *arXiv preprint arXiv:2305.00035*, 2023.
- Hengji Cui, Dong Wei, Kai Ma, Shi Gu, and Yefeng Zheng. A unified framework for generalized low-shot medical image segmentation with scarce data. *IEEE Transactions on Medical Imaging*, 40(10):2656–2671, 2021. doi: 10.1109/TMI.2020.3045775.
- Theekshana Dissanayake, Yasmeen George, Dwarikanath Mahapatra, Shridha Sridharan, Clinton Fookes, and Zongyuan Ge. Few-shot learning for medical image segmentation: A review and comparative study. *ACM Computing Surveys*, 2025.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- Azade Farshad, Anastasia Makarevich, Vasileios Belagiannis, and Nassir Navab. Metamedseg: Volumetric meta-learning for few-shot organ segmentation. In *MICCAI Workshop on Domain Adaptation and Representation Transfer*, pages 45–55. Springer, 2022.
- Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.
- Fabio Galbusera and Alessandro Cina. Image annotation and curation in radiology: an overview for machine learning practitioners. *European Radiology Experimental*, 8:11, 2024. doi: 10.1186/s41747-023-00408-y. URL <https://eurradioexp.springeropen.com/articles/10.1186/s41747-023-00408-y>.
- Yunhe Gao, Di Liu, Zhuowei Li, Yunsheng Li, Dongdong Chen, Mu Zhou, and Dimitris N Metaxas. Show and segment: Universal medical image segmentation via in-context learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20830–20840, 2025.

- David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.
- Stine Hansen, Srishti Gautam, Robert Jenssen, and Michael Kampffmeyer. Anomaly detection-inspired few-shot medical image segmentation through self-supervision with supervoxels. *Medical Image Analysis*, 78:102385, 2022.
- Jiesi Hu, Jianfeng Cao, Yanwu Yang, Chenfei Ye, Yixuan Zhang, Hanyang Peng, and Ting Ma. Medverse: A universal model for full-resolution 3d medical image segmentation, transformation and enhancement. *arXiv preprint arXiv:2509.09232*, 2025.
- Rabindra Khadka, Debesh Jha, Steven Hicks, Vajira Thambawita, Michael A Riegler, Sharib Ali, and Pål Halvorsen. Meta-learning with implicit gradients in a few-shot setting for medical image segmentation. *Computers in Biology and Medicine*, 143:105227, 2022.
- Tianang Leng, Yiming Zhang, Kun Han, and Xiaohui Xie. Self-sampling meta sam: enhancing few-shot medical image segmentation with meta-learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 7925–7935, 2024.
- Yuval Nirkin, Lior Wolf, and Tal Hassner. Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4061–4070, 2021.
- Eva Pachetti and Sara Colantonio. A systematic review of few-shot learning in medical imaging. *Artificial intelligence in medicine*, 156:102949, 2024.
- Marcin Przewiezklikowski, Przemyslaw Przybysz, Jacek Tabor, Maciej Zikeba, and Przemyslaw Spurek. Hypermaml: Few-shot adaptation of deep models with hypernetworks. *Neurocomputing*, 598:128179, 2024.
- Marianne Rakic, Hallee E Wong, Jose Javier Gonzalez Ortiz, Beth A Cimini, John V Guttag, and Adrian V Dalca. Tyche: Stochastic in-context learning for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11159–11173, 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.
- Neha Tirpude, Tausif Diwan, and Meera Dhabu. Meta-transformers: A hybrid approach for medical image segmentation with u-net and meta-learning. *IEEE Access*, 2025.
- Haoyu Wang, Sizheng Guo, Jin Ye, Zhongying Deng, Junlong Cheng, Tianbin Li, Jianpin Chen, Yanzhou Su, Ziyang Huang, Yiqing Shen, et al. Sam-med3d: A vision foundation model for general-purpose segmentation on volumetric medical images. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.
- Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. Seggpt: Towards segmenting everything in context. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1130–1140, 2023.

Jakob Wasserthal, Hanns-Christian Breit, Manfred T Meyer, Maurice Pradella, Daniel Hinck, Alexander W Sauter, Tobias Heye, Daniel T Boll, Joshy Cyriac, Shan Yang, et al. Totalsegmentator: robust segmentation of 104 anatomic structures in ct images. *Radiology: Artificial Intelligence*, 5(5):e230024, 2023.

Andrey Zhmoginov, Mark Sandler, and Maksym Vladymyrov. Hypertransformer: Model generation for supervised and semi-supervised few-shot learning. In *International Conference on Machine Learning*, pages 27075–27098. PMLR, 2022.

Appendix A. Ablations

To understand the contribution of individual components in our [HYPERUNET](#) workflow, we conduct a series of ablation experiments. Each subsection isolates one specific design choice and reports its effect on segmentation performance across multiple tasks of the MSD dataset, where our [HYPERUNET](#) previously achieved strong results.

A.1. Effect of Gradient Step on the Reference Example

In our workflow, after generating a [HYPERUNET](#) from the reference [volume-mask](#) pair (x, y) , we additionally perform a single gradient update with learning rate η on this same reference pair. This step further [optimises](#) the generated model towards the reference example. To assess the impact of this design choice, we compare the performance of [HYPERUNETs](#) evaluated *before* and *after* applying the gradient step.

Table 4: **Ablation with and without gradient step.** DSC \uparrow and NSD \uparrow across all CT and MRI tasks. Best configuration per task in **bold**.

Task	Model	DSC \uparrow	NSD \uparrow
Liver	w/o gradient step	0.673 \pm 0.069	0.171 \pm 0.063
	w/ gradient step (<i>Ours</i>)	0.772\pm0.063	0.490\pm0.080
Spleen	w/o gradient step	0.476 \pm 0.165	0.289 \pm 0.130
	w/ gradient step (<i>Ours</i>)	0.501\pm0.169	0.316\pm0.115
Hepatic Vessels	w/o gradient step	0.281 \pm 0.102	0.306 \pm 0.100
	w/ gradient step (<i>Ours</i>)	0.307\pm0.120	0.364\pm0.129
Pancreas	w/o gradient step	0.186\pm0.137	0.123\pm0.076
	w/ gradient step (<i>Ours</i>)	0.153 \pm 0.138	0.102 \pm 0.078
Brain Edema	w/o gradient step	0.288 \pm 0.191	0.194 \pm 0.126
	w/ gradient step (<i>Ours</i>)	0.326\pm0.208	0.217\pm0.150
Heart	w/o gradient step	0.457 \pm 0.053	0.254 \pm 0.035
	w/ gradient step (<i>Ours</i>)	0.500\pm0.119	0.281\pm0.071

The DSC and NSD for both configurations are reported in [Tab. 4](#). For DSC, we observe a consistent increase in performance across most tasks, with the largest gain for the liver (0.772 vs. 0.673). In contrast, the pancreas task shows a slight decrease when applying the gradient step.

For NSD, performance also improves for all tasks except the pancreas. Compared to the moderate DSC gains, NSD exhibits substantial improvements for the liver (0.490 vs. 0.171) and a notable increase for the hepatic vessels (0.364 vs. 0.306) when the gradient step is applied.

A.2. Without normalization

The mask-averaged pooled encoder feature map F_{enc} serves as our global task encoding z^{task} , which adapts positional encodings z^{pos} and generates the weights and bias for the final $1 \times 1 \times 1$ convolutional layer.

We compare two [HYPERUNET](#) variants: one employing normalization ($z^{\text{task}} = \text{LayerNorm}(\text{MAP}(F_{\text{enc}}))$) and one using the raw pooled features ($z^{\text{task}} = \text{MAP}(F_{\text{enc}})$).

Table 5: **Ablation with vs. without normalization.** DSC \uparrow and NSD \uparrow across all CT and MRI tasks. Best configuration per task in **bold**.

Task	Model	DSC \uparrow	NSD \uparrow
Liver	w/o norm	0.844\pm0.056	0.411 \pm 0.070
	w/ norm (<i>Ours</i>)	0.772 \pm 0.063	0.490\pm0.080
Spleen	w/o norm	0.383 \pm 0.131	0.196 \pm 0.070
	w/ norm (<i>Ours</i>)	0.501\pm0.169	0.316\pm0.115
Hepatic Vessels	w/o norm	0.014 \pm 0.041	0.017 \pm 0.048
	w/ norm (<i>Ours</i>)	0.307\pm0.120	0.364\pm0.129
Pancreas	w/o norm	0.022 \pm 0.049	0.018 \pm 0.031
	w/ norm (<i>Ours</i>)	0.153\pm0.138	0.102\pm0.078
Brain Edema	w/o norm	0.015 \pm 0.036	0.025 \pm 0.239
	w/ norm (<i>Ours</i>)	0.326\pm0.208	0.217\pm0.150
Heart	w/o norm	0.053 \pm 0.130	0.137 \pm 0.025
	w/ norm (<i>Ours</i>)	0.500\pm0.119	0.281\pm0.071

We report the DSC and NSD of both configurations in [Tab. 5](#). For liver and spleen, both relatively large and homogeneous organs, DSC and NSD remain of roughly similar magnitude across both configurations. For the remaining tasks, however, we observe a substantial degeneration of performance when no [normalisation](#) is applied. On the hepatic vessels, pancreas, brain [oedema](#), and heart tasks, both DSC and NSD drop to values close to zero, indicating diffuse predictions that fail to capture underlying anatomical patterns.

A.3. Task-dependent positional encodings

We [hypothesise](#) that enriching positional encodings with task-specific information enables the [HYPERUNET](#) to assign task-dependent roles and importance to different convolutional filters in the final network.

Concretely, we define

$$z^* = \text{MLP}(\text{Concat}[z^{\text{task}}, z^{\text{pos}}]),$$

where z^{task} is the global task encoding and z^{pos} the positional encoding. To test this hypothesis, we compare two [HYPERUNET](#) variants: (i) one employing task-dependent positional encodings z^* as defined above, and (ii) one using purely positional encodings $z^* = z^{\text{pos}}$ without any task information.

Table 6: **Ablation with vs. without task-specific positional encodings.** DSC \uparrow and NSD \uparrow across all CT and MRI tasks. Best results per task in **bold**.

Task	Model	DSC \uparrow	NSD \uparrow
Liver	Regular PE	0.654 \pm 0.151	0.158 \pm 0.055
	Task-specific PE (<i>Ours</i>)	0.772\pm0.063	0.490\pm0.080
Spleen	Regular PE	0.376 \pm 0.165	0.198 \pm 0.081
	Task-specific PE (<i>Ours</i>)	0.501\pm0.169	0.316\pm0.115
Hepatic Vessels	Regular PE	0.130 \pm 0.130	0.170 \pm 0.156
	Task-specific PE (<i>Ours</i>)	0.307\pm0.120	0.364\pm0.129
Pancreas	Regular PE	0.116 \pm 0.118	0.082 \pm 0.069
	Task-specific PE (<i>Ours</i>)	0.153\pm0.138	0.102\pm0.078
Brain Edema	Regular PE	0.295 \pm 0.197	0.246\pm0.132
	Task-specific PE (<i>Ours</i>)	0.326\pm0.208	0.217 \pm 0.150
Heart	Regular PE	0.292 \pm 0.130	0.204 \pm 0.059
	Task-specific PE (<i>Ours</i>)	0.500\pm0.119	0.281\pm0.071

The DSC and NSD for both configurations are reported in [Tab. 6](#).

Across all tasks, using task-specific positional embeddings gives a higher DSC than using task-agnostic ones. The performance gain is most notable for Heart (0.500 vs 0.292) and Hepatic Vessels (0.307 vs. 0.130), and smallest for Brain Edema (0.326 vs 0.295). For the NSD, we see a similar trend. The NSD is higher in almost all tasks, except the brain edema where the difference, however, is small (0.217 vs. 0.246). Similar to our previous ablation on the gradient step, we again see a significant improvement of NSD for the Liver (0.490 vs. 0.158), Hepatic Vessels (0.364 vs 0.170) and Spleen (0.316 vs. 0.198) tasks.

A.4. MAML vs HyperUNet for 5 Reference pairs per Organ

Model	Spleen	Liver	R Kidney	L Kidney
MAML (Alsaleh et al., 2024)	0.839	0.903	0.775	0.870
HYPERUNET (<i>Ours</i>)	0.821	0.865	0.753	0.801

Table 7: DSC of our HyperUNet and an MAML approach using 5 reference pairs per organ.

In Tab. 7, MAML achieves slightly higher scores on a small set of closely related abdominal organs, a setting known to favour gradient-based meta-learning. HYPERUNET is trained across more than 100 heterogeneous tasks spanning multiple organs and modalities, and does not rely on narrow task similarity, highlighting its broader applicability and complementary strengths.

A.5. Sensitivity Analysis

To assess HYPERUNET’s sensitivity to the sampled reference pair, we conduct a small-scale study on a held-out test set of 20 samples from the MSD Spleen dataset. Specifically, we evaluate the test set using eight different HYPERUNET instances, each synthesised from a distinct reference pair. The resulting DSC values range from 0.409 to 0.557, with a mean performance of 0.491 ± 0.042 across reference pairs. While this confirms an expected dependence on the chosen reference, the observed variability remains moderate and does not raise immediate concerns.

A.6. Inference time

Table A.6 reports the inference time of HYPERUNET and related approaches.

Method	Inference Time (s)
UniverSeg	659.4
SAM-Med2D	648.4
SAM-Med3D	15.2
Iris	2.0
HYPERUNET (<i>Ours</i>)	1.77 (Generation) / 0.06 (Inference)

Table 8: Empirical measurements of inference time on one NVIDIA A100 GPU. The image size is processed to $128 \times 128 \times 128$ for inference. For HYPERUNET the time for U-Net generation and U-Net inference is reported.