

# Implicit Anatomical Rendering for Medical Image Segmentation with Stochastic Experts

Chenyu You <sup>1</sup>(✉), Weicheng Dai<sup>2</sup>, Yifei Min<sup>4</sup>, Lawrence Staib<sup>1,2,3</sup>, and  
James S. Duncan<sup>1,2,3,4</sup>

<sup>1</sup>Department of Electrical Engineering, Yale University  
chenyu.you@yale.edu

<sup>2</sup>Department of Radiology and Biomedical Imaging, Yale University

<sup>3</sup>Department of Biomedical Engineering, Yale University

<sup>4</sup>Department of Statistics and Data Science, Yale University

**Abstract.** Integrating high-level semantically correlated contents and low-level anatomical features is of central importance in medical image segmentation. Towards this end, recent deep learning-based medical segmentation methods have shown great promise in better modeling such information. However, convolution operators for medical segmentation typically operate on regular grids, which inherently blur the high-frequency regions, *i.e.*, boundary regions. In this work, we propose MORSE, a generic implicit neural rendering framework designed at an anatomical level to assist learning in medical image segmentation. Our method is motivated by the fact that implicit neural representation has been shown to be more effective in fitting complex signals and solving computer graphics problems than discrete grid-based representation. The core of our approach is to formulate medical image segmentation as a rendering problem in an end-to-end manner. Specifically, we continuously align the coarse segmentation prediction with the ambiguous coordinate-based point representations and aggregate these features to adaptively refine the boundary region. To parallelly optimize multi-scale pixel-level features, we leverage the idea from Mixture-of-Expert (MoE) to design and train our MORSE with a stochastic gating mechanism. Our experiments demonstrate that MORSE can work well with different medical segmentation backbones, consistently achieving competitive performance improvements in both 2D and 3D supervised medical segmentation methods. We also theoretically analyze the superiority of MORSE.

**Keywords:** Medical Image Segmentation · Implicit Neural Representation · Stochastic Mixture-of-Experts.

## 1 Introduction

Medical image segmentation is one of the most fundamental and challenging tasks in medical image analysis. It aims at classifying each pixel in the image into an anatomical category. With the success of deep neural networks (DNNs), medical image segmentation has achieved great progress in assisting radiologists in contributing to a better disease diagnosis.

Until recently, the field of medical image segmentation has mainly been dominated by an encoder-decoder architecture, and the existing state-of-the-art (SOTA) medical segmentation models are roughly categorized into two groups: (1) convolutional neural networks (CNNs) [25,4,19,1,29,38,11,12,6,34,33], and (2) Transformers[2,5,35]. However, despite their recent success, several challenges persist to build a robust medical segmentation model: ❶ Classical deep learning methods require precise pixel/voxel-level labels to tackle this problem [37,36,30,31,32]. Acquiring a large-scale medical dataset with exact pixel- and voxel-level annotations is usually expensive and time-consuming as it requires extensive clinical expertise [16,14,13,20,10,34]. Prior works [15,7] have used point-level supervision on medical image segmentation to refine the boundary prediction, where such supervision requires well-trained model weights and can only capture discrete representations on the pixel-level grids. ❷ Empirically, it has been observed that CNNs inherently store the discrete signal values in a grid of pixels or voxels, which naturally blur the high-frequency anatomical regions, *i.e.*, boundary regions. In contrast, implicit neural representations (INRs), also known as coordinate-based neural representations, are capable of representing discrete data as instances of a continuous manifold, and have shown remarkable promise in computer vision and graphics [22,27,28]. Several questions then arise: *how many pixel- or voxel-level labels are needed to achieve good performance? how should those coordinate locations be selected? and how can the selected coordinates and signal values be leveraged efficiently?*

Orthogonally to the popular belief that the model architecture matters the most in medical segmentation (*i.e.*, complex architectures generally perform better), this paper focuses on an under-explored and alternative direction: *towards improving segmentation quality via rectifying uncertain coarse predictions*. To this end, we propose a new INR-based framework, MORSE (implicit anatOmical Rendering with Stochastic Experts). The core of our approach is to formulate medical image segmentation as a rendering problem in an end-to-end manner. We think of building a generic implicit neural rendering framework to have fine-grained control of segmentation quality, *i.e.*, to adaptively compose coordinate-wise point features and rectify uncertain anatomical regions. Specifically, we encode the sampled coordinate-wise point features into a continuous space, and then align position and features with respect to the continuous coordinate.

We further hinge on the idea of mixture-of-experts (MoE) to improve segmentation quality. Considering our goal is to rectify uncertain coarse predictions, we regard multi-scale representations from the decoder as experts. During training, experts are randomly activated for features from multiple blocks of the decoder, and correspondingly the INRs of multi-scale representations are separately parameterized by a group of MLPs that compose a spanning set of the target function class. In this way, the INRs are acquired across the multi-block structure while the stochastic experts are specified by the anatomical features at each block.

In summary, our main contributions are as follows: (1) We propose a new implicit neural rendering framework that has fine-grained control of segmentation quality by adaptively composing INRs (*i.e.*, coordinate-wise point features)

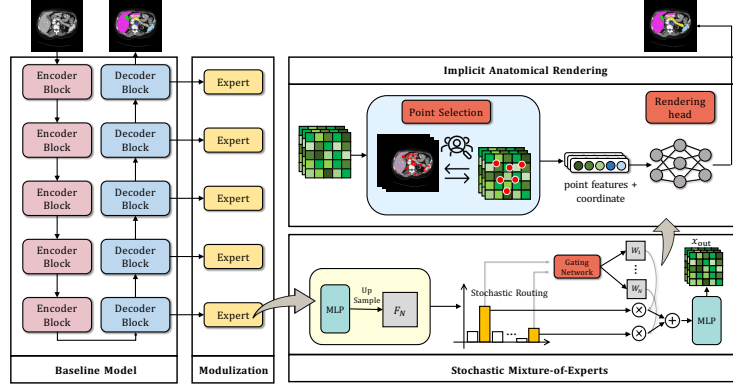


Fig. 1. Illustration of the MORSE pipeline.

and rectifying uncertain anatomical regions; (2) We illustrate the advantage of adopting mixture-of-experts that endows the model with better specialization of features maps for improving the performance; (3) Extensive experiments show that our method consistently improves performance compared to 2D and 3D SOTA CNN- and Transformer-based approaches; and (4) Theoretical analysis verifies the expressiveness of our INR-based model. Code is released at [here](#).

## 2 Method

Let us assume a supervised medical segmentation dataset  $\mathcal{D} = \{(x, y)\}$ , where each input  $x = x_1, x_2, \dots, x_T$  is a collection of  $T$  2D/3D scans, and  $y$  refers to the ground-truth labels. Given an input scan  $x \in \mathbb{R}^{H \times W \times d}$ , the goal of medical segmentation is to predict a segmentation map  $\hat{y}$ . Fig. 1 illustrates the overview of our MORSE. In the following, we first describe our baseline model  $f$  for standard supervised learning, and subsequently present our MORSE. A baseline segmentation model consists of two main components: (1) encoder module, which generates the multi-scale feature maps such that the model is capable of modeling multi-scale local contexts, and (2) decoder module that makes a prediction  $\hat{y}$  using the generated multi-block features of different resolution. The entire model  $M$  is trained end-to-end using the supervised segmentation loss  $\mathcal{L}_{\text{sup}}$  [35] (*i.e.*, equal combination of cross-entropy loss and dice loss).

### 2.1 Stochastic Mixture-of-Experts (SMoE) Module

**Motivation** We want a module that encourages inter- and intra-associations across multi-block features. Intuitively, multi-block features should be specified by anatomical features across each block. We posit that due to the specialization-favored nature of MoE, the model will benefit from explicit use of its own anatomical features at each block by learning multi-scale anatomical contexts with adaptively selected experts. In implementation, our SMoE module follows an MoE design [21], where it treats features from multiple blocks of the decoder as

experts. To mitigate potential overfitting and enable parameter-efficient property, we further randomly activate experts for each input during training. Our approach makes three major departures compared to [21] (*i.e.*, SOTA segmentation model): (1) implicitly optimized during training since it greatly trims down the training cost and the model scale; (2) using features from the *decoder* instead of the *encoder* tailored for our refinement goal; and (3) empirically showing that “self-slimmable” attribute delivers sufficiently exploited expressiveness of the model.

**Modulization** We first use multiple small MLPs with the same size to process different block features and then up-sample the features to the size of the input scans, *i.e.*,  $H \times W \times d$ . With  $N$  as the total number of layers (experts) in the decoder, we treat these upsampled features  $[F_1, F_2, \dots, F_N]$  as expert features. We then train a gating network  $\mathcal{G}$  to re-weight the features from activated experts with the trainable weight matrices  $[W_1, W_2, \dots, W_N]$ , where  $W \in \mathbb{R}^{H \times W \times d}$ . Specifically, the gating network or router  $\mathcal{G}$  outputs these weight matrices satisfying  $\sum_i W_i = \mathbf{1}^{H \times W \times d}$  using a structure depicted as follows:

$$W_i = [\text{Softmax}(\text{Conv}([F_1, F_2, \dots, F_N]))]_i, \quad \text{for } i \in [N]. \quad (1)$$

The gating network first concatenates all the expert features along channels and uses several convolutional layers to get  $\text{Conv}([F_1, F_2, \dots, F_N]) \in \mathbb{R}^{C \times H \times W \times d \times N}$ , where  $C$  is the channel dimension. A softmax layer is applied over the last dimension (*i.e.*,  $N$ -expert) to output the final weight maps. After that, we feed the resultant output  $x_{\text{out}}$  to another MLP to fuse multi-block expert features. Finally, the resultant output  $x_{\text{out}}$  (*i.e.* the coarse feature) is given as follows:

$$x_{\text{out}} = \text{MLP}\left(\sum_{i=1}^N W_i \cdot F_i\right), \quad (2)$$

where  $\cdot$  denotes the pixel-wise multiplication, and  $x_{\text{out}} \in \mathbb{R}^{C \times H \times W \times d}$ .

**Stochastic Routing** The prior MoE-based model [21] are densely activated. That is, a model needs to access all its parameters to process all inputs. One drawback of such design often comes at the prohibitive training cost. Moreover, the large model size suffers from the representation collapse issue [26], further limiting the model’s performance. Our proposed SMOE considers *randomly activated* expert sub-networks to address the issues. In implementation, we simply apply standard dropout to multiple experts with a dropping probability  $\alpha$ . For each training iteration, there are dropout masks placed on experts with the probability  $\alpha$ . That is, the omission of experts follows a  $\text{Bernoulli}(\alpha)$  distribution. As for inference, there is no dropout mask and all experts are activated.

## 2.2 Implicit Anatomical Rendering (IAR)

The existing methods generally assume that the semantically correlated information and fine anatomical details have been captured and can be used to obtain high-quality segmentation quality. However, CNNs inherently operate the discrete signals in a grid of pixels or voxels, which naturally blur the high-frequency

anatomical regions, *i.e.*, boundary regions. To address such issues, INRs in computer graphics are often used to replace standard discrete representations with continuous functions parameterized by MLPs [28,27]. Our key motivation is that the task of medical segmentation is often framed as a rendering problem that applies implicit neural functions to continuous shape/object/scene representations [22,27]. Inspired by this, we propose an implicit neural rendering framework to further improve segmentation quality, *i.e.*, to adaptively compose coordinate-wise point features and rectify uncertain anatomical regions.

**Point Selection** Given a coarse segmentation map, the rendering head aims at rectifying the uncertain boundary regions. A point selection mechanism is thus required to filter out those pixels where the rendering can achieve maximum segmentation quality improvement. Besides, point selection can significantly reduce computational cost compared to blindly rendering all boundary pixels. Therefore, our MORSE selects  $N_p$  points for refinement given the coarse segmentation map using an uncertainty-based criterion. Specifically, MORSE first uniformly randomly samples  $k_p N_p$  candidates from all pixels where the hyper-parameter  $k_p \geq 1$ , following [9]. Then, based on the coarse segmentation map, MORSE chooses  $\rho N_p$  pixels with the highest uncertainty from these candidates, where  $0.5 < \rho < 1$ . The uncertainty for a pixel is defined as  $\text{SecondLargest}(\mathbf{v}) - \max(\mathbf{v})$ , where  $\mathbf{v}$  is the logit vector of that pixel such that the coarse segmentation is given as  $\text{Softmax}(\mathbf{v})$ . The rest  $(1 - \rho)N_p$  pixels are sampled uniformly from all the remaining pixels. This mechanism ensures the selected points contain a large portion of points with uncertain segmentation which require refinement.

**Positional Encoding** It is well-known that neural networks can be cast as universal function approximators, but they are inferior to high-frequency signals due to their limited learning power [23,18]. Unlike [9], we explore using the encoded positional information to capture high-frequency signals, which echoes our theoretical findings in Appendix A. Specifically, for a coordinate-based point  $(x, y) \in [H] \times [W]$ , the positional encoding function is given as:

$$\begin{aligned} \psi(x, y) = & [\sin(2\pi(w_1\tilde{x} + v_1\tilde{y})), \dots, \sin(2\pi(w_L\tilde{x} + v_L\tilde{y})), \\ & \cos(2\pi(w_1\tilde{x} + v_1\tilde{y})), \dots, \cos(2\pi(w_L\tilde{x} + v_L\tilde{y}))], \end{aligned} \quad (3)$$

where  $\tilde{x} = 2x/H - 1$  and  $\tilde{y} = 2y/W - 1$  are the standardized coordinates with values in between  $[-1, 1]$ . The frequency  $\{w_i, v_i\}_{i=1}^L$  are trainable parameters with Gaussian random initialization, where we set  $L = 128$  [3]. For each selected point, its position encoding will then be concatenated with the coarse features of that point (*i.e.*,  $x_{\text{out}}$  defined in Sec. 2.1), to output the fine-grained features.

**Rendering Head** The fine-grained features are then fed to the rendering head whose goal is to rectify the uncertain predictions with respect to these selected points. Inspired by [9], the rendering head adopts 3-layer MLPs design. Since the rendering head is designed to rectify the class label of the selected points, it is trained using the standard cross-entropy loss  $\mathcal{L}_{\text{rend}}$ .

**Adaptive Weight Adjustment** Instead of directly leveraging pre-trained weights, it is more desirable to train the model *from scratch* in an *end-to-end*

**Table 1.** Quantitative comparisons for multi-organ segmentation on the Synapse multi-organ CT dataset. The best results are indicated in **bold**.

Method	Average											
	DSC $\uparrow$	Jaccard $\uparrow$	95HD $\downarrow$	ASD $\downarrow$	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
UNet (Baseline) [25]	70.11	59.39	44.69	14.41	84.00	56.70	72.41	62.64	86.98	48.73	81.48	67.96
+ PointRend [9]	71.52	61.34	43.19	13.70	85.74	57.14	75.42	63.27	87.32	50.16	81.82	71.29
+ Implicit PointRend [3]	67.33	59.73	52.44	22.15	76.32	51.99	70.28	70.36	81.69	43.77	77.18	67.05
+ Ours (MoE)	72.83	62.64	40.44	13.15	86.11	59.51	75.81	67.10	87.82	52.11	83.48	70.86
+ Ours (SMoE)	74.86	64.94	37.69	12.66	86.39	63.99	77.96	68.93	88.88	53.62	86.12	72.98
+ Ours (IAR)	73.11	62.98	34.01	12.67	86.28	60.25	76.58	65.34	88.32	52.12	83.47	72.51
+ Ours (IAR+MoE)	75.37	65.65	33.34	11.43	87.00	64.45	78.14	70.13	89.32	52.33	85.20	76.40
+ Ours (MORSE)	<b>76.59</b>	<b>66.97</b>	<b>32.00</b>	<b>10.67</b>	<b>87.28</b>	<b>64.73</b>	<b>80.58</b>	<b>71.87</b>	<b>90.04</b>	<b>54.60</b>	<b>86.67</b>	<b>76.93</b>
TransUnet (Baseline) [2]	77.49	64.78	31.69	8.46	87.23	63.13	81.87	77.02	94.08	55.86	85.08	75.62
+ PointRend [9]	78.30	65.88	34.17	8.62	87.93	63.96	83.47	77.23	94.86	56.45	85.76	76.75
+ Implicit PointRend [3]	71.92	60.62	41.42	18.55	78.39	61.64	79.59	73.20	89.61	50.01	80.17	62.75
+ Ours (MoE)	77.85	65.30	32.75	7.90	87.40	63.46	82.34	77.88	94.14	56.12	85.24	76.25
+ Ours (SMoE)	78.68	65.98	31.86	7.00	87.60	66.21	82.62	78.12	94.88	57.59	85.97	76.48
+ Ours (IAR)	79.37	66.50	30.13	7.25	88.63	66.76	83.70	79.50	95.26	57.10	86.90	77.10
+ Ours (IAR+MoE)	79.60	66.99	27.59	6.54	88.73	66.83	83.85	80.19	95.98	57.12	86.92	77.21
+ Ours (MORSE)	<b>80.85</b>	<b>68.53</b>	<b>26.61</b>	<b>6.46</b>	<b>88.92</b>	<b>67.53</b>	<b>84.83</b>	<b>81.68</b>	<b>96.83</b>	<b>59.70</b>	<b>87.73</b>	<b>79.58</b>

way. For instance, we empirically observe that directly using coarse masks by pre-trained weights to modify unclear anatomical regions might lead to suboptimal results (See Sec. 3.1). Thus, we propose to modify the importance of  $\mathcal{L}_{\text{rend}}$  as:

$$\lambda_t = \lambda_{\text{rend}} \cdot \left[ \mathbb{1}\{t > T/2\} \cdot \left( \frac{t - T/2}{T} \right) \right], \quad (4)$$

where  $t$  is the index of the iteration,  $T$  denotes the total number of iterations, and  $\mathbb{1}\{\cdot\}$  denotes the indicator function.

**Training Objective** As such, the model is trained in an *end-to-end* manner using total loss  $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{sup}} + \lambda_t \times \mathcal{L}_{\text{rend}}$ .

### 3 Experiments

**Dataset** We evaluate the models on two important medical segmentation tasks.

(1) **Synapse multi-organ segmentation**<sup>1</sup>: Synapse multi-organ segmentation dataset contains 30 abdominal CT scans with 3779 axial contrast-enhanced abdominal clinical CT images in total. Each volume scan has variable volume sizes  $512 \times 512 \times 85 \sim 512 \times 512 \times 198$  with a voxel spatial resolution of  $([0.54 \sim 0.54] \times [0.98 \sim 0.98] \times [2.5 \sim 5.0])\text{mm}^3$ . For a fair comparison, the data split<sup>2</sup> is fixed with 18 (2211 axial slices) and 12 patients’ scans for training and testing, respectively. The entire dataset has a high diversity of aorta, gallbladder, spleen, left kidney, right kidney, liver, pancreas, spleen, and stomach.

(2) **Liver segmentation**: Multi-phasic MRI (MP-MRI) dataset is an in-house dataset including 20 patients, each including T1 weighted DCE-MRI images at three-time phases (*i.e.*, pre-contrast, arterial, and venous). Here, our evaluation is conducted via 5-fold cross-validation on the 60 scans. For each fold, the training and testing data includes 48 and 12 cases, respectively.

<sup>1</sup> <https://www.synapse.org/#!/Synapse:syn3193805/wiki/217789>

<sup>2</sup> [https://github.com/Beckschen/TransUNet/tree/main/lists/lists\\_Synapse](https://github.com/Beckschen/TransUNet/tree/main/lists/lists_Synapse)

**Table 2.** Quantitative comparisons for liver segmentation on the Multi-phasic MRI dataset. The best results are indicated in **bold**.

Method	Average				Method	Average			
	DSC $\uparrow$	Jaccard $\uparrow$	95HD $\downarrow$	ASD $\downarrow$		DSC $\uparrow$	Jaccard $\uparrow$	95HD $\downarrow$	ASD $\downarrow$
3D-UNet (Baseline) [4]	89.19	81.21	34.97	10.63	UNETR (Baseline) [5]	89.95	82.17	24.64	6.04
+ PointRend [9]	89.55	81.80	30.88	10.12	+ PointRend [9]	90.49	82.36	21.06	5.59
+ Implicit PointRend [3]	88.01	79.83	37.55	12.86	+ Implicit PointRend [3]	88.72	80.18	26.63	10.58
+ Ours (MoE)	89.81	82.06	29.96	10.15	+ Ours (MoE)	90.70	82.80	15.31	5.93
+ Ours (SMoE)	90.16	82.28	28.36	9.79	+ Ours (SMoE)	91.02	83.29	15.12	5.64
+ Ours (IAR)	91.22	83.30	27.84	8.89	+ Ours (IAR)	91.63	83.83	14.25	4.99
+ Ours (IAR+MoE)	92.77	83.94	26.57	7.51	+ Ours (IAR+MoE)	93.01	84.70	13.29	4.84
+ Ours (MORSE)	<b>93.59</b>	<b>84.62</b>	<b>19.61</b>	<b>6.57</b>	+ Ours (MORSE)	<b>93.85</b>	<b>85.53</b>	<b>12.33</b>	<b>4.38</b>

**Implementation Details** We use AdamW optimizer [17] with an initial learning rate  $5e^{-4}$ , and adopt a polynomial-decay learning rate schedule for both datasets. We train each model for 30K iterations. For Synapse, we adopt the input resolution as  $256 \times 256$  and the batch size is 4. For MP-MRI, we randomly crop  $96 \times 96 \times 96$  patches and the batch size is 2. For SMoE, following [21], all the MLPs have hidden dimensions  $[256, 256]$  with ReLU activations, the dimension of expert features  $[F_1, F_2, \dots, F_N]$  are 256. We empirically set  $\alpha$  as 0.7. Following [9],  $N_p$  is set as 2048, and 8192 for training and testing, respectively, and  $k_p, \rho$  are 3, 0.75. We follow the same gating network design [21], which includes four  $3 \times 3$  convolutional layers with channels  $[256, 256, 256, N]$  and ReLU activations.  $\lambda_{rend}$  are set to 0.1. We adopt four representative models, including UNet [25], TransUnet [2], 3D-UNet [4], UNETR [5]. Specifically, we set  $N$  for UNet [25], TransUnet [2], 3D-UNet [4], UNETR [5] with 5, 3, 3, 3, respectively. We also use Dice coefficient (DSC), Jaccard, 95% Hausdorff Distance (95HD), and Average Surface Distance (ASD) to evaluate 3D results. We conduct all experiments in the same environments with fixed random seeds (Hardware: Single NVIDIA RTX A6000 GPU; Software: PyTorch 1.12.1+cu116, and Python 3.9.7).

### 3.1 Comparison with State-of-the-Art Methods

We adopt classical CNN- and transformer-based models, *i.e.*, 2D-based {UNet [25], TransUnet [2]} and 3D-based {3D-UNet [4], UNETR [5]}, and train them on {2D Synapse, 3D MP-MRI} in an end-to-end manner<sup>3</sup>.

**Main Results** The results for 2D synapse multi-organ segmentation and 3D liver segmentation are shown in Tables 1 and 2, respectively. The following observations can be drawn: (1) Our MORSE demonstrates superior performance compared to all other training algorithms. Specifically, Compared to UNet, TransUnet, 3D-UNet, and UNETR baselines, our MORSE with all experts selected obtains 3.36%~6.48% improvements in Dice across two segmentation tasks. It validates the superiority of our proposed MORSE. (2) The stochastic routing policy shows consistent performance benefits across all four network backbones on 2D and 3D settings. Specifically, we can observe that our SMoE framework improves all the baselines, which is within expectation since our model is implicitly “optimized” given evolved features. (3) As is shown, we can observe that IAR consistently

<sup>3</sup> All comparison experiments are using their released code.

outperforms PointRender across all the baselines (*i.e.*, UNet, TransUnet, 3D-UNet, and UNETR) and obtain {1.59%, 1.07%, 2.03%, 1.14%} performance boosts on two segmentation tasks, highlighting the effectiveness of our proposal in INRs. (4) With Implicit PointRender [3] equipped, all the models’ performances drop. We find: adding Implicit PointRender leads to significant performance drops of -2.78%, -5.57%, -1.18%, and -1.23% improvements, compared with the SOTA baselines (*i.e.*, UNet, TransUnet, 3D-UNet, and UNETR) on two segmentation tasks, respectively. Importantly, we find that: [3] utilizes INRs for producing different parameters of the point head for each object with point-level supervision. As this implicit function does not directly optimize the anatomical regions, we attribute this drop to the introduction of additional noise during training, which leads to the representation collapse. This further verifies the effectiveness of our proposed IAR. In Appendix Figs. 2 and 3, we provide visual comparisons from various models. We can observe that MORSE yields sharper and more accurate boundary predictions compared to all the other training algorithms.

**Visualization of IAR Modules** To better understand the IAR module, we visualize the point features on the coarse prediction and refined prediction after the IAR module in Appendix Fig. 4. As is shown, we can see that IAR help rectify the uncertain anatomical regions for improving segmentation quality.

**Table 3.** Effect of stochastic rate  $\alpha$  and expert number  $N$ .

$\alpha$	DSC[%]↑	ASD[voxel]↓	$N$	DSC[%]↑	ASD[voxel]↓
0.1	75.41	11.96	1 (No MoE)	75.11	11.67
0.2	75.68	11.99		75.63	11.49
0.5	76.06	<b>10.43</b>		75.82	11.34
0.7	<b>76.59</b>	10.67		76.16	11.06
0.9	74.16	11.32		<b>76.59</b>	<b>10.67</b>

**Table 4.** Ablation studies of the Adaptive Weight Adjustment (AWA).

Method	DSC[%]↑	ASD[voxel]↓
w/o AWA & train w/ $\mathcal{L}_{\text{rend}}$ from scratch	70.56	14.89
w/o AWA & train w/ $\mathcal{L}_{\text{rend}}$ in $\frac{T}{2}$	75.42	12.00
w/ AWA	<b>76.59</b>	<b>10.67</b>

### 3.2 Ablation Study

We first investigate our MORSE equipped with UNet by varying  $\alpha$  (*i.e.*, stochastic rate) and  $N$  (*i.e.*, experts) on Synapse. The comparison results of  $\alpha$  and  $N$  are reported in Table 3. We find that using  $\alpha = 0.7$  performs the best when the expert capacity is  $N = 5$ . Similarly, when reducing the expert number, the performance also drops considerably. This shows our hyperparameter settings are optimal.

Moreover, we conduct experiments to study the importance of Adaptive Weight Adjustment (AWA). We see that: (1) Disabling AWA and training  $\mathcal{L}_{\text{rend}}$  from scratch causes unsatisfied performance, as echoed in [9]. (2) Introducing AWA shows a consistent advantage compared to the other. This demonstrates the importance of the Adaptive Weight Adjustment.

## 4 Conclusion

In this paper, we proposed MORSE, a new implicit neural rendering framework that has fine-grained control of segmentation quality by adaptively composing coordinate-wise point features and rectifying uncertain anatomical regions. We



also demonstrate the advantage of leveraging mixture-of-experts that enables the model with better specialization of features maps for improving the performance. Extensive empirical studies across various network backbones and datasets, consistently show the effectiveness of the proposed MORSE. Theoretical analysis further uncovers the expressiveness of our INR-based model.

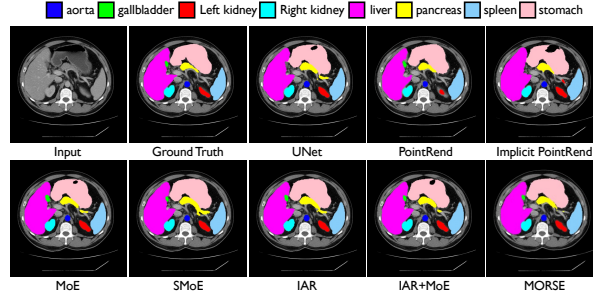
## References

1. Chen, C., Qin, C., Qiu, H., Ouyang, C., Wang, S., Chen, L., Tarroni, G., Bai, W., Rueckert, D.: Realistic adversarial data augmentation for mr image segmentation. In: MICCAI (2020)
2. Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y.: Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 (2021)
3. Cheng, B., Parkhi, O., Kirillov, A.: Pointly-supervised instance segmentation. In: CVPR (2022)
4. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: MICCAI (2016)
5. Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H., Xu, D.: Unetr: Transformers for 3d medical image segmentation. In: WACV (2022)
6. He, Y., Lin, F., Tzeng, N.F., et al.: Interpretable minority synthesis for imbalanced classification. In: IJCAI (2021)
7. Huang, R., Lin, M., Dou, H., Lin, Z., Ying, Q., Jia, X., Xu, W., Mei, Z., Yang, X., Dong, Y., et al.: Boundary-rendering network for breast lesion segmentation in ultrasound images. *Medical Image Analysis* (2022)
8. Jacot, A., Gabriel, F., Hongler, C.: Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems* **31** (2018)
9. Kirillov, A., Wu, Y., He, K., Girshick, R.: Pointrend: Image segmentation as rendering. In: CVPR (2020)
10. Lai, Z., Oliveira, L.C., Guo, R., Xu, W., Hu, Z., Mifflin, K., Decarli, C., Cheung, S.C., Chuah, C.N., Dugger, B.N.: Brainsec: Automated brain tissue segmentation pipeline for scalable neuropathological analysis. *IEEE Access* (2022)
11. Lai, Z., Wang, C., Cheung, S.c., Chuah, C.N.: Sar: Self-adaptive refinement on pseudo labels for multiclass-imbalanced semi-supervised learning. In: CVPR. pp. 4091–4100 (2022)
12. Lai, Z., Wang, C., Gunawan, H., Cheung, S.C.S., Chuah, C.N.: Smoothed adaptive weighting for imbalanced semi-supervised learning: Improve reliability against unknown distribution data. In: ICML. pp. 11828–11843 (2022)
13. Lai, Z., Wang, C., Hu, Z., Dugger, B.N., Cheung, S.C., Chuah, C.N.: A semi-supervised learning for segmentation of gigapixel histopathology images from brain tissues. In: EMBC. IEEE (2021)
14. Lai, Z., Wang, C., Oliveira, L.C., Dugger, B.N., Cheung, S.C., Chuah, C.N.: Joint semi-supervised and active learning for segmentation of gigapixel pathology images with cost-effective labeling. In: ICCV (2021)
15. Li, H., Yang, X., Liang, J., Shi, W., Chen, C., Dou, H., Li, R., Gao, R., Zhou, G., Fang, J., et al.: Contrastive rendering for ultrasound image segmentation. In: MICCAI. Springer (2020)

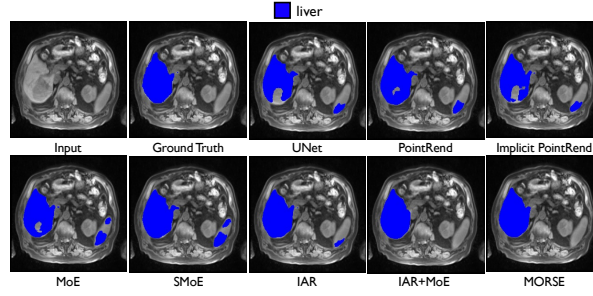
16. Lin, F., Yuan, X., Peng, L., Tzeng, N.F.: Cascade variational auto-encoder for hierarchical disentanglement. In: ACM CIKM (2022)
17. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
18. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* (2021)
19. Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., et al.: Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999* (2018)
20. Oliveira, L.C., Lai, Z., Siefkes, H.M., Chuah, C.N.: Generalizable semi-supervised learning strategies for multiple learning tasks using 1-d biomedical signals. In: *NeurIPS 2022 Workshop on Learning from Time Series for Health* (2022)
21. Ou, Y., Yuan, Y., Huang, X., Wong, S.T., Volpi, J., Wang, J.Z., Wong, K.: Patcher: Patch transformers with mixture of experts for precise medical image segmentation. In: *MICCAI*. Springer (2022)
22. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: *CVPR* (2019)
23. Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F., Bengio, Y., Courville, A.: On the spectral bias of neural networks. In: *ICML*. PMLR (2019)
24. Rahimi, A., Recht, B.: Random features for large-scale kernel machines. *Advances in neural information processing systems* **20** (2007)
25. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: *MICCAI* (2015)
26. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* (2017)
27. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. In: *NeurIPS* (2020)
28. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. In: *NeurIPS* (2020)
29. Xue, Y., Xu, T., Zhang, H., Long, L.R., Huang, X.: Segan: Adversarial network with multi-scale l1 loss for medical image segmentation. *Neuroinformatics* (2018)
30. You, C., Dai, W., Liu, F., Su, H., Zhang, X., Staib, L., Duncan, J.S.: Mine your own anatomy: Revisiting medical image segmentation with extremely limited labels. *arXiv preprint arXiv:2209.13476* (2022)
31. You, C., Dai, W., Min, Y., Liu, F., Zhang, X., Clifton, D.A., Zhou, S.K., Staib, L.H., Duncan, J.S.: Rethinking semi-supervised medical image segmentation: A variance-reduction perspective. *arXiv preprint arXiv:2302.01735* (2023)
32. You, C., Dai, W., Min, Y., Staib, L., Sekhon, J., Duncan, J.S.: Action++: Improving semi-supervised medical image segmentation with adaptive anatomical contrast. *arXiv preprint arXiv:2304.02689* (2023)
33. You, C., Dai, W., Staib, L., Duncan, J.S.: Bootstrapping semi-supervised medical image segmentation with anatomical-aware contrastive distillation. In: *IPMI* (2023)
34. You, C., Yang, J., Chapiro, J., Duncan, J.S.: Unsupervised wasserstein distance guided domain adaptation for 3d multi-domain liver segmentation. In: *Interpretable and Annotation-Efficient Learning for Medical Image Computing* (2020)
35. You, C., Zhao, R., Liu, F., Chinchali, S., Topcu, U., Staib, L., Duncan, J.S.: Class-aware generative adversarial transformers for medical image segmentation. In: *NeurIPS* (2022)

36. You, C., Zhao, R., Staib, L.H., Duncan, J.S.: Momentum contrastive voxel-wise representation learning for semi-supervised volumetric medical image segmentation. In: MICCAI (2022)
37. You, C., Zhou, Y., Zhao, R., Staib, L., Duncan, J.S.: Simcvd: Simple contrastive voxel-wise representation distillation for semi-supervised medical image segmentation. *IEEE Transactions on Medical Imaging* (2022)
38. Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J.: Unet++: A nested u-net architecture for medical image segmentation. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support* (2018)

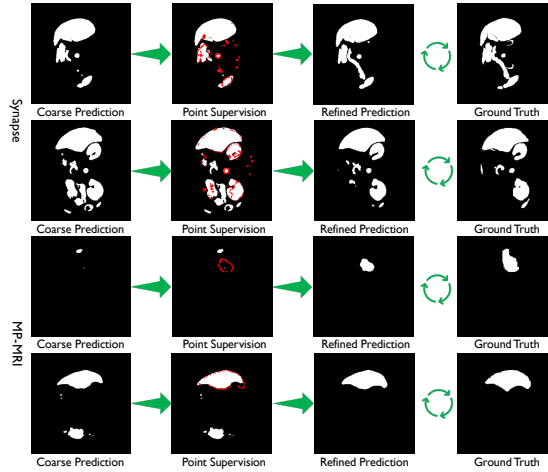
## Appendix



**Fig. 2.** Visualization results on Synapse. MORSE yields more accurate predictions, especially for small regions.



**Fig. 3.** Visualization results on MP-MRI. MORSE outputs more accurate segmentation results, especially for boundaries.



**Fig. 4.** Visualization of sampled point location. As is shown, IAR significantly improves the segmentation quality.

## A Theoretical Analysis

In this section, we theoretically analyze the expressiveness of MORSE, and demonstrate the approximation power of INR features combined with MLPs.

We first introduce the kernel method which is commonly used in analyzing neural networks. We define the kernel function  $\kappa: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ , which is a symmetric function measuring the similarity between two vectors in  $\mathbb{R}^d$ . Kernel functions are used to approximate unknown functions from data. Specifically, given a training dataset  $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^n$  where  $\mathbf{y}_i = \mu(\mathbf{x}_i)$  for some unknown function  $\mu(\cdot)$ , an estimate  $\hat{\mu}$  can be constructed using kernel function as  $\hat{\mu}(\mathbf{x}) = \sum_{i=1}^n (\mathbf{K}^{-1} \mathbf{Y})_i \kappa(\mathbf{x}_i, \mathbf{x})$ , where  $\mathbf{K}$  is the  $n \times n$  matrix with  $\mathbf{K}_{i,j} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ , and  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^\top$ .

The kernel method is related to MLPs through a kernel function called Neural Tangent Kernel [8]. For an MLP  $\Pi_\theta$  with trainable parameters  $\theta$ , it has been shown that, under certain conditions, the model  $\Pi_\theta$  trained with stochastic gradient descent will converge to the estimate generated by the kernel method through Neural Tangent Kernel, which is defined as  $\kappa_{\text{NTK}}(\mathbf{x}_i, \mathbf{x}_j) = \mathbb{E}_{\theta \sim \mathcal{N}} \left\langle \frac{\partial \Pi_\theta(\mathbf{x}_i)}{\partial \theta}, \frac{\partial \Pi_\theta(\mathbf{x}_j)}{\partial \theta} \right\rangle$ . Importantly, for  $\mathbf{x}_i, \mathbf{x}_j$  on the unit sphere, NTK is an inner product kernel, *i.e.*,

$$\kappa_{\text{NTK}}(\mathbf{x}_i, \mathbf{x}_j) = k_{\text{NTK}}(\mathbf{x}_i \cdot \mathbf{x}_j), \quad (5)$$

for some function  $k_{\text{NTK}}$ .

With the kernel function explained, we now show how positional encoding helps with expressiveness. By construction, the positional encoding maps pixel coordinates into sinusoidal vectors. This is closely related to random Fourier features which are provably able to approximate the family of shift-invariant kernel functions [24]. Specifically, a kernel function  $\kappa$  is shift-invariant if for any  $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^d$ , it holds that  $\kappa(\mathbf{x}_1, \mathbf{x}_2) = \kappa(\mathbf{x}_1 - \mathbf{x}_2)$  with a slight abuse of notation. In other words, the value of  $\kappa(\mathbf{x}_1, \mathbf{x}_2)$  only depends on the difference  $\mathbf{x}_1 - \mathbf{x}_2$ . It is clear that shift-invariance is an ideal property in imaging tasks.

To understand how the Fourier features generated by positional encoding can approximate shift-invariant kernel, we rewrite positional encoding (Eqn. 3) as:

$$\psi(\mathbf{x}) = [\sin(2\pi \mathbf{w}_1 \cdot \mathbf{x}), \dots, \sin(2\pi \mathbf{w}_L \cdot \mathbf{x}), \cos(2\pi \mathbf{w}_1 \cdot \mathbf{x}), \dots, \cos(2\pi \mathbf{w}_L \cdot \mathbf{x})],$$

where  $\mathbf{x} = (\tilde{x}, \tilde{y})$ , and  $\mathbf{w}_i = [w_i, v_i]$  for  $i = 1, \dots, L$ . Given any shift-invariant kernel  $\kappa(\mathbf{x}_1, \mathbf{x}_2) = \kappa(\mathbf{x}_1 - \mathbf{x}_2)$ , we define a distribution  $\mathcal{P}$  over  $\mathbf{w}$  as  $\mathcal{P}(\mathbf{w}) = \frac{1}{2\pi} \int e^{-2\pi i \mathbf{w}^\top \mathbf{x}} \kappa(\mathbf{x}) d\mathbf{x}$ , which is the Fourier transform of the kernel  $\kappa$ . Suppose that  $\mathbf{w}_1, \dots, \mathbf{w}_L$  are *i.i.d.* samples from  $\mathcal{P}$ . Then it holds that [24]:

$$\Pr \left[ \sup_{\mathbf{x}_1, \mathbf{x}_2} \left| \frac{1}{L} \psi(\mathbf{x}_1)^\top \psi(\mathbf{x}_2) - \kappa(\mathbf{x}_1, \mathbf{x}_2) \right| \geq \epsilon \right] \leq 2^8 \frac{\sigma_{\mathcal{P}}^2}{\epsilon^2} \exp \left( -\frac{L\epsilon^2}{16} \right),$$

where  $\sigma_{\mathcal{P}}^2 = \mathbb{E}_{\mathcal{P}}(\mathbf{w}^\top \mathbf{w})$ . The above result indicates that with high probability, any shift-invariant kernel function can be approximated with Fourier features. Therefore, this demonstrates the expressive power of INR features.

Finally, we show that  $\psi(\cdot)$  combined with the MLP  $\Pi_\theta$  forms a shift-invariant kernel. We define the positional encoding kernel  $\kappa_{\text{pe}}$  as  $\kappa_{\text{pe}}(\mathbf{x}_1, \mathbf{x}_2) = \psi(\mathbf{x}_1)^\top \psi(\mathbf{x}_2)$ . It can be shown that  $\kappa_{\text{pe}}(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^L \cos(2\pi \mathbf{w}_j \cdot (\mathbf{x}_1 - \mathbf{x}_2))$  [28], which is shift-invariant. Combining with Eqn. 5, our positional encoding combined followed by MLP approximately equals to  $k_{\text{NTK}}(\kappa_{\text{pe}}(\mathbf{x}_1 - \mathbf{x}_2))$ , which is a shift-invariant kernel. This demonstrates the expressiveness of MORSE.