공학석사 학위논문

Transformer-UNet 단계적 네트워크를 통한 조영 증강 CT 영상 기반 대동맥 박리 자동 분할 연구

Automatic Segmentation of Aortic Dissection in Contrast-Enhanced CT Images Using Transformer-UNet Cascade Network

울산대학교대학원

의과학과

정지훈

Transformer-UNet 단계적 네트워크를 통한 조영 증강 CT 영상 기반 대동맥 박리 자동 분할 연구

지도교수 정진홍, 이준구

이 논문을 공학석사 학위논문으로 제출함

2024년 8월

울산대학교대학원

의과학과

정지훈

정지훈의 공학석사 학위논문을 인준함

심사위원 양동현 인

심사위원 정진홍 인

심사위원 이준구 인

울산대학교대학원

2024년 8월

Abstract

Aortic Dissection (AD) is a severe condition caused by a tear in the aortic inner wall, allowing blood to flow between the layers of the aortic wall and potentially leading to life-threatening complications. Managing AD involves imaging, medical treatment, and sometimes surgery. CT scans, which produce high-resolution images quickly, are commonly used for AD diagnosis and prognosis evaluation. Accurate segmentation of the True Lumen (TL), False Lumen (FL), and Thrombosis (TH) is crucial, but manual measurement is time-consuming and variable. To address this, computer vision, machine learning, and deep learning methods have been introduced. Although CNN-based models have played a significant role in medical image analysis, they have limitations in comprehensively understanding anatomical structures. To overcome these limitations, Transformer-based models have been introduced, excelling in extracting global context information but being less effective in capturing local texture details. Therefore, this study proposes a model that combines the strengths of CNNs and Transformers. We designed a two-stage model: the first stage uses a 3D Transformer UNet to learn the aorta's global information, while the second stage uses a 3D UNet to learn the detailed textures of TL, FL, and TH. Additionally, a multi-scale patch extraction method is applied to effectively capture both the aorta's global information and detailed textures. This model's two-step approach—using a 3D Transformer UNet for global context and a 3D CNN UNet for local texture—has been validated in ablation studies. The model's performance was evaluated using the dataset from Asan Medical Center and compared with existing models such as nnUNet and nnFormer. Our method achieved Dice Similarity Coefficients (DSC) of 0.917, 0.888, and 0.630 for TL, FL, and TH, respectively, demonstrating the highest segmentation accuracy. The model's robustness and generalizability were further assessed using external datasets, showing potential for improving AD diagnosis and treatment across various clinical settings.

Contents

Contents of Tables

Contents of Figures

1. Introduction

Aortic dissection (AD) is a critical condition caused by a tear in the aorta's inner lining. This tear leads to blood flowing between the aorta's layers, potentially disrupting its function and becoming life-threatening [1]. The management of AD includes imaging tests, medical treatments, and sometimes surgery. Timely diagnosis and treatment are crucial to prevent severe complications [2]. For effective management, it is necessary to accurately segment the aorta into the true lumen (TL), false lumen (FL), and thrombosis (TH) on CT images. The TL represents the normal part of the aorta, the FL arises from the separation and is filled with blood that has seeped between the aortic wall's layers, and the TH is a clot within the FL that can lead to severe complications, such as organ ischemia [3].

Computed tomography (CT) scans are frequently employed for AD diagnosis due to their capability to quickly generate high-resolution images. One of the essential roles of CT in AD is to assess the patient's prognosis or evaluate the disease's progression using the aortic dimension or the size of the true or false lumen measured on the CT scans. AD segmentation can be accomplished by experts measuring the diameter or the area of each section. However, manually measuring these parameters on CT images can be time-consuming and may have significant variability between measurements [4]. Therefore, considering the urgency required for AD diagnosis, there have been attempts in the field to improve the time-consuming procedure by adopting computer vision, machine learning (ML), or deep learning (DL) methodologies.

ML and DL models have played crucial roles in advancing AD segmentation. Notably, UNet [5], which is based on convolutional neural networks (CNNs), has revolutionized the domain of medical imaging analysis. UNet is particularly well-suited for biomedical image segmentation due to its encoder-decoder architecture. The encoder captures the context in the image through a series of convolutional and pooling layers, while the decoder precisely localizes and classifies each pixel by a series of up-convolutions and concatenations with high-resolution features from the encoder. This architecture allows for efficient feature extraction and pixel-wise segmentation, which is essential for delineating complex structures like the TL, FL, and TH in AD. Introduced by Olaf Ronneberger et al. in 2015, UNet has since become a foundational model in medical imaging due to its ability to handle small datasets effectively and produce precise segmentations even with limited annotated data. Its success in various medical applications, from brain tumor segmentation to liver analysis, has set a new standard for DL models in the healthcare domain. However, conventional CNN-based UNet models have a notable limitation. These models rely on convolution kernels for feature extraction, which inherently emphasizes the local regions of an image. Although the receptive field can expand through pooling in deeper layers, it does not completely address the inherent constraints of CNN. These limitations are particularly pronounced in AD segmentation, where distinguishing among TL, FL, and

TH based on brightness differences alone is difficult and requires an understanding of anatomy.

To overcome the shortcomings of these CNN-based models, transformer-based models [6] originating from Natural Language Processing (NLP) have emerged in the field of computer vision, with the Vision Transformer (ViT) [7] being a pioneering example. ViT introduces a novel approach to image analysis by treating images as sequences of patches, analogous to the tokens in NLP. ViT divides an image into fixed-size patches, flattens them, and linearly projects them into a sequence of embeddings. These embeddings are then processed by a standard Transformer encoder, which uses self-attention mechanisms to capture long-range dependencies and global context. This approach contrasts with CNNs, which focus on local feature extraction through convolutional operations. ViT's ability to model global relationships between different parts of an image makes it highly effective for tasks requiring an understanding of overall structure. The development of ViT marked a significant shift from traditional convolution-based methods to transformer-based architectures, highlighting the versatility and power of attention mechanisms initially designed for language processing. Following ViT, transformer-based models have been developed and applied to image analysis, leading to the creation of segmentation models such as the Swin Transformer [8] and nnFormer [9]. These models excel in extracting global context information; however, they are less effective at extracting local text features compared with CNNs. Despite the strengths of transformer-based models, solely relying on transformers may not capture all the necessary details, particularly in complex medical imaging tasks such as AD segmentation. This highlights the importance of leveraging both the global context extraction capabilities of transformers and the local feature extraction strengths of CNNs.

Previous studies have explored various approaches for AD segmentation. For instance, Long Cao et al. (2019) [10] applied a two-step process first segmenting the whole aorta and then the TL and FL. Zeyang Yao et al. (2021) [11] used a three-step approach for segmenting the whole aorta, TL/FL, and TL/FL/TH. Wobben, Liana D., et al. (2021) [12] employed a similar strategy, segmenting the whole aorta followed by TL/(FL+TH) and finally FL/TH. These studies, however, faced limitations in capturing the global structure due to the reliance on UNet. Lewis D. Hahn et al. (2020) [13] aimed to address these limitations by leveraging the anatomical structure of the aorta, using a train-free ML technique to straighten the long tubular shape of the aorta before segmenting subregions. Xiang Dongqiao et al. (2023) [14] utilized a transformer model to extract global information about the aorta, combining it with local information from UNet.

In this paper, we propose a model that combines the advantages of both approaches. We developed a cascade network benchmarked on experts' procedures, integrating both transformer- and CNN-based networks. The overall model architecture is shown in Figure 1. First, we designed a “3D transformer for panoptic context-aware” model using a transformer to capture the overall anatomical position of the

TL, FL, and TH of the AD. In the next step, we designed a "3D UNet for localized texture refinement" model using the UNet model, which is highly capable of extracting local features, to capture the detailed texture of the TL, FL, and TH. Our contribution can be categorized into two aspects:

1. **Cascade network**: This network is designed to learn the anatomical structure and detailed texture information of AD. We applied a two-stage cascade method. The Stage 1 model, a 3D transformer for panoptic context-aware, learns the anatomical structure (global information) of the TL, FL, and TH of the AD. In Stage 2, a 3D UNet for localized texture refinement learns the detailed texture of the TL, FL, and TH based on the anatomical structure learned in Stage 1.
2. **Multi-scale patch extraction**: Using this scheme, the overall structure of the aorta is first analyzed in a broader context (large patches), and then segmentation is performed in a more focused context (smaller patches). This method effectively extracts both anatomical structures and detailed texture information by applying different patch sizes for each step.

In summary, our proposed approach improves the performance of AD segmentation, potentially aiding in the diagnosis and treatment planning for patients with this condition.

Internal dataset	DICOM parameter	Overall (n = 253)	Training Set (n = 173)	Testing Set (n = 80)
Patient Characteristics				
	Sex (Male / Female) [a]	166 / 87	110 / 63	56 / 24
	Age range (y) [b]	25 – 85	28 – 85	25 – 77
	Mean age (y) [c]	58 ± 13	59 ± 13	55 ± 14
CTA Parameters				
	CT tube current (mA) [c]	391.84 ± 215.80	417.52 ± 213.58	339.56 ± 210.80
	Peak tube voltage range (kVp) [c]	120.89 ± 8.33	119.31 ± 7.10	124.00 ± 9.62
	Pixel spacing range (mm) [b]	0.54 – 0.86	0.54 – 0.86	0.54 - 0.83
	Pixel spacing (mm) [c]	0.69 ± 0.06	0.70 ± 0.06	0.69 ± 0.05
	Slice thickness range (mm) [b]	2.5 – 5.0	2.5 – 5.0	2.5 – 5.0
	Slice thickness (mm) [c]	4.19 ± 1.03	4.2 ± 1.0	4.16 ± 1.04
	Voxel size range (mm^{3}) [b]	0.86 – 3.68	0.86 – 3.68	0.86 – 3.41
	Voxel size (mm^{3}) [c]	2.04 ± 0.64	2.05 ± 0.62	2.00 ± 0.66
	Voxel value range (HU) [b]	-1024 – 3072	-1024 – 3072	-1024 – 3072
Manufactures				
	GE Medical Systems [a]	97	60	37
	Siemens [a]	151	110	41
	Philips [a]	4	3	1
	Toshiba [a]	1	0	1

Table 1: CT protocols of the internal dataset. Here, the superscript 'a' denotes a specific number, the superscript 'b' indicates the range from minimum to maximum, and the superscript 'c' represents the mean ± standard deviation.

2. Materials and Method

2.1. Datasets

In this study, we utilized two comprehensive datasets to evaluate the efficacy of our proposed algorithm in segmenting Aortic Dissection (AD) in 3D CT images. The datasets are as follows:

2.1.1. Internal Dataset

This dataset was sourced from the Asan Medical Center and comprises a total of 253 AD cases collected over a period spanning from December 2000 to August 2017. These cases include a mix of type A and type B aortic dissections. Out of the 253 cases, 173 were designated for training and validation, while the remaining 80 cases were reserved for internal testing purposes. To ensure the reliability and accuracy of the dataset, the ground truth (GT) labels were meticulously created by a radiologist with 10 years of experience in the field. These labels were subsequently verified by a senior radiologist possessing 20 years of experience, further validating the accuracy of the dataset. Key characteristics of this dataset include the presence of the FL in 233 cases and TH in 196 cases, providing a rich and diverse set of examples for training and evaluation. The patient characteristics in this dataset showed a mean age of 58 years with a standard deviation of 13 years, and an age range spanning from 25 to 85 years. There were 166 male patients and 87 female patients. The CT scan parameters for this dataset were as follows: The CT tube current had a mean of 391.84 mA with a standard deviation of 215.80 mA. The peak tube voltage ranged from 120.89 ± 8.33 kVp. The pixel spacing ranged from 0.54 mm to 0.86 mm, with a mean of 0.69 ± 0.06 mm. The slice thickness varied between 2.5 mm to 5.0 mm, averaging at 4.19 ± 1.03 mm. The voxel size ranged from 0.86 mm^3 to 3.68 mm^3, with an average size of 2.04 ± 0.64 mm^3. The voxel value range was from -1024 HU to 3072 HU. Regarding the manufacturers of the CT scanners used, GE Medical Systems accounted for 97 of the scans, Siemens for 151, Philips for 4, and Toshiba for 1.

2.1.2. External Dataset

The external dataset, referred to as imageTBAD [11], is an open collection of 98 3D CTA images of AD patients, obtained from Guangdong Provincial People’s Hospital. This dataset spans a collection period from 2013 to 2019 and exclusively includes cases of type B aortic dissection. The utilization of this external dataset was crucial in demonstrating the generalizability and robustness of our proposed model across different clinical settings and patient populations. Among the cases in this dataset, 68 featured a FL, and 32 included TH. The inclusion of this external dataset allowed for a comprehensive evaluation of our model's performance, ensuring that it is not only effective on the internal dataset but also maintains high performance when applied to data from different sources. This cross-dataset validation is essential for establishing the practical applicability and reliability of our segmentation algorithm in real-world clinical scenarios.

In summary, the internal dataset from Asan Medical Center and the external imageTBAD dataset from Guangdong Provincial People's Hospital collectively offer a robust and diverse basis for evaluating the performance of our proposed segmentation model. The careful curation and validation of these datasets, along with detailed documentation of CT protocols, underscore the rigor and thoroughness of our study.

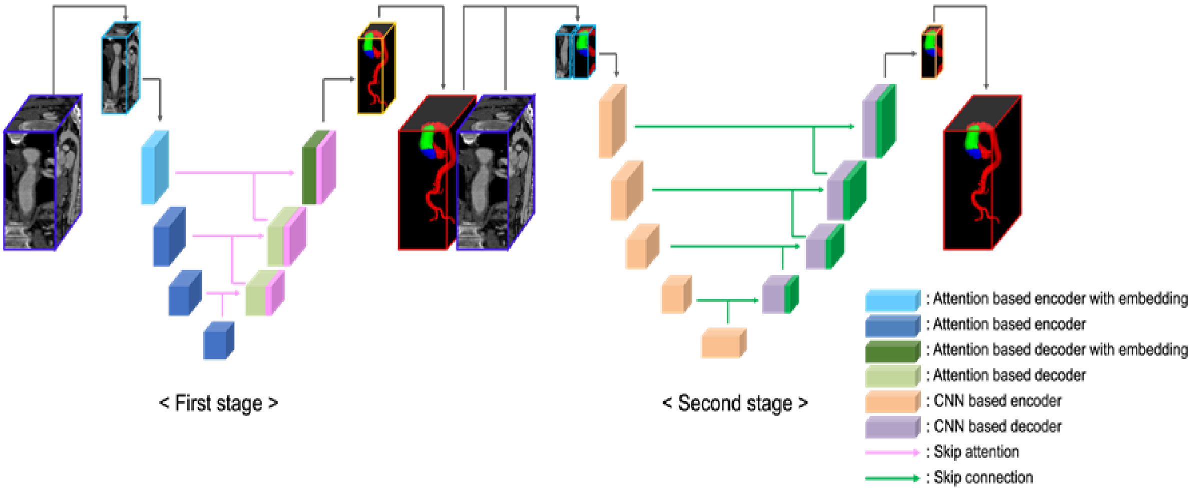
< First stage >
< Second stage >
: Attention based encoder with embedding
: Attention based encoder
: Attention based decoder with embedding
: Attention based decoder
: CNN based encoder
: CNN based decoder
: Skip attention
: Skip connection

Figure 1: The architecture overview that incorporates a two-stage UNet structure. The architecture utilizes a transformer block in Stage 1, configuring the patch size to encompass the entire foreground when extracting patches. For Stage 2, the model adopts 3D UNet architecture, adjusting the patch size to ensure a minimum of 25% foreground inclusion during patch extraction.

2.2. CNNs vs Transformer

2.2.1. CNNs

Convolutional Neural Networks (CNNs) and Transformers are two prominent architectures used in deep learning, each with distinct strengths and weaknesses, particularly in the context of medical image segmentation. CNNs have been foundational in the field of computer vision and medical image analysis. They are designed to automatically and adaptively learn spatial hierarchies of features from input images. The key components of CNNs include convolutional layers, pooling layers, and fully connected layers. One of the primary strengths of CNNs is their ability to effectively extract local features due to their convolutional layers, which apply filters to local regions of the image. This capability makes them well-suited for tasks where local texture and detail are crucial. Additionally, CNNs can process images efficiently due to the shared weights in convolutional layers, reducing the number of parameters and computational complexity. Models like UNet, which are based on CNNs, have achieved significant success in medical image segmentation tasks by leveraging their encoder-decoder architecture to capture both context and precise localization . However, CNNs have limitations as well. Despite pooling layers that expand the receptive field, CNNs can struggle with capturing global context due to their inherent focus on local features. This limitation becomes apparent when dealing with complex anatomical structures that require an understanding of long-range dependencies, which is challenging for CNNs .

2.2.2. Transformer

Transformers, originally developed for NLP, have been adapted for image analysis with models like the ViT. Transformers leverage self-attention mechanisms to capture long-range dependencies and global context within the data. The main strength of Transformers lies in their ability to model global relationships between different parts of an image, making them highly effective for tasks that require an understanding of the overall structure. Transformers can process variable-length inputs and are not restricted by the fixed grid structure of CNNs, allowing for more flexibility in handling different types of data. Models like ViT and Swin Transformer have demonstrated the potential of transformer-based architectures in achieving state-of-the-art performance in various vision tasks. However, Transformers typically require more computational resources and larger datasets to train effectively, which can be a limitation in some medical imaging applications. Additionally, while Transformers excel at capturing global context, they can be less effective at extracting finegrained local details compared to CNNs.

2.2.3. Hybrid Approach

Given the complementary strengths and weaknesses of CNNs and Transformers, various studies have explored combining these methods to maximize the advantages of each approach [7], [8]. These research efforts aim to leverage the local feature extraction capabilities of CNNs and the global context modeling strengths of Transformers. Our proposed method is also designed to maximize the benefits of both techniques.

2.3. Method

We present our approach, a cascade network integrating transformer and UNet principles, designed specifically for AD segmentation in enhanced CT images. Our model uses nnUNet [15] and nnFormer as benchmarks, which serve as our baselines. The figure 1 is our architecture overview.

2.3.1. Preprocessing

We adhere to the nnUNet preprocessing protocol due to its demonstrated robustness and significant benefits in managing a variety of medical imaging datasets. Its efficiency in minimizing computational overhead, maintaining data uniformity, and improving model performance has led us to implement this preprocessing strategy. The nnUNet preprocessing framework is adept at handling diverse medical imaging data with different attributes. By standardizing the data, it ensures that segmentation models perform reliably across various datasets, making it particularly effective when dealing with data from different scanners or acquisition methods. The preprocessing stages of nnUNet significantly enhance data consistency and model efficacy through a series of well-defined steps. Initially, data is cropped to include only regions with nonzero values, reducing data size and computational load while preserving critical areas, such as the liver in CT scans or the brain in skull-stripped MRI scans. Next, to address voxel size variability from different scanners or acquisition protocols, all images are resampled to a common voxel spacing using the median voxel spacing of the dataset as the standard, with third-order spline interpolation for image data and nearest neighbor interpolation for segmentation masks. Finally, CT scan images are normalized based on the dataset's statistical properties, where intensity values within the segmentation masks are gathered, clipped to the 0.5th and 99.5th percentiles, and z-score normalized. If cropping significantly reduces data size, normalization is confined to the mask of nonzero elements, setting all values outside this mask to zero. These steps collectively ensure consistent data formatting, which is crucial for effective CNN training and enhances the overall performance and generalization capabilities of the segmentation model. By implementing these preprocessing steps, the data is consistently formatted, thereby enhancing the performance and generalization capabilities of our segmentation model.

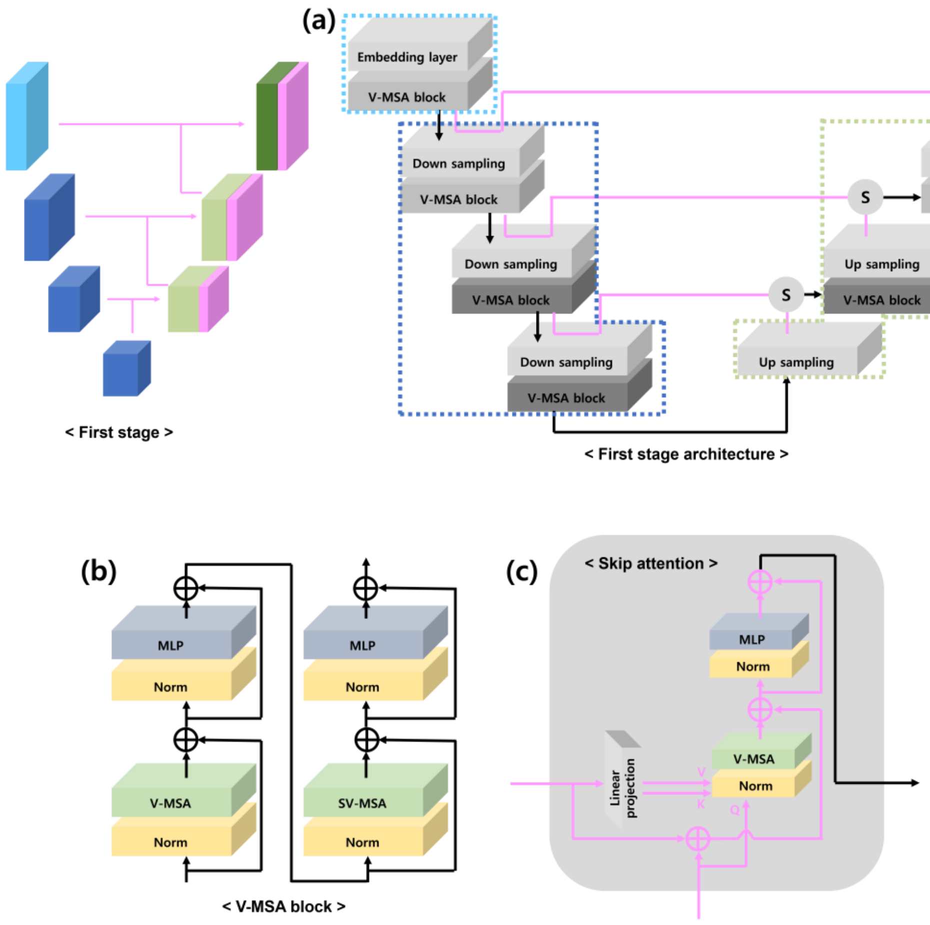

Figure 2: The Stage 1 model architecture and modules; (a) is the overall Stage 1 model architecture, (b) is the V-MSA block, and (c) is the skip attention module.

2.3.2. Stage 1: 3D transformer for panoptic context-aware

In designing Stage 1, we prioritized the effective learning of global information. When experts segment the TL, FL, and TH of an AD, they consider the entire CT image holistically rather than concentrating on individual sections. We incorporated this expert approach into Stage 1. The overall architecture of Stage 1 is depicted in Figure 2.

2.3.2.1. Encoder

The input to Stage 1 is a 3D patch $\mathbf{X} \in \mathbb{R}^{H \times W \times D}$, where H, W, and D represent the height, width, and depth of each input tensor.

- **Embedding Layer.** The embedding layer translates the raw pixels of the image into a higher dimensional space while enriching the features. This transformed information allows the network to better capture complex patterns and produce accurate results. Particularly in structures such as transformers, embedding captures information across multiple dimensions and incorporates contextual data. This process enables the network to analyze image information in greater detail. Specifically, the layer is responsible for converting each input scan $\mathbf{X}$ into a high-dimensional tensor $\mathbf{X}_{er}$ where the subscript e stands for 'embedded', signifying the transformation of the input data into an embedded space. The resulting tensor $\mathbf{X}_e \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times\frac{D}{2}\times C}$, where $\frac{H}{4}, \frac{W}{4}$, and $\frac{D}{2}$ denote the number of patch tokens and C represents the sequence length. We designed our embedding layer by adapting the method used in a Swin transformer. By using a small kernel instead of a large one, we can encode pixel-level spatial information more accurately while reducing computational complexity. The embedding block comprises four convolutional layers with a kernel size of three. After each convolutional layer (except the last one), we add a GELU [16] and a layer normalization [17].

- **Volume-Based Multihead Self-Attention.** Following the embedding layer, the high-dimensional tensor $\mathbf{X}_e$ is forwarded to the volumebased multihead self-attention (V-MSA) and multi-layer perceptron (MLP). The primary objective of this block is to seamlessly integrate the captured long-term dependencies with the multiscale features obtained from either the downsampling layer or the high-resolution spatial information from the embedding layer. This integration clarifies the relationships between structural features at various depths and sizes within the image, thereby enabling more precise segmentation. Unlike the Swin transformer, we compute self-attention within a 3D volume instead of a 2D local window. Given $\mathbf{X}_V \in \mathbb{R}^{L\times C}$ as the input to a transformer block, where the subscript V stands for 'volume', $\mathbf{X}_V$ is initially restructured into $\mathbb{R}^{N_V\times N_T\times C}$ in $\hat{\mathbf{X}}_V$, where N_V denotes the number of predefined 3D volumes; $N_T = S_H \times S_W \times S_D$, representing the number of patch tokens in each volume, and the subscript T stands for 'tokens'; and $\{S_H, S_W, S_D\}$ represent the volume dimensions. As illustrated in Figure 2 b, we perform two consecutive transformer layers in each block, defining the second layer as the shifted volume-based multihead self-attention (SV-MSA) of the first layer. The key distinction from conventional transformer blocks lies in our computation, which is executed on 3D volumes instead of 2D windows. The computational steps are summarized as follows:

$$
\begin{aligned}
\hat{\mathbf{x}}_V^l &= \mathrm{V-MSA}\left(\mathrm{Norm}\left(\hat{\mathbf{x}}_V^{l-1}\right)\right) + \hat{\mathbf{x}}_V^{l-1} \\
\mathbf{X}_V^l &= \mathrm{MLP}\left(\mathrm{Norm}\left(\hat{\mathbf{x}}_V^{l}\right)\right) + \hat{\mathbf{x}}_V^{l} \\
\hat{\mathbf{X}}_V^{l+1} &= \mathrm{SV-MSA}\left(\mathrm{Norm}\left(\mathbf{X}_V^{l}\right)\right) + \mathbf{X}_V^{l} \\
\mathbf{X}_V^{l+1} &= \mathrm{MLP}\left(\mathrm{Norm}\left(\hat{\mathbf{X}}_V^{l+1}\right)\right) + \hat{\mathbf{X}}_V^{l+1}
\end{aligned}
$$

where l denotes the layer order. The computational complexity of V-MSA for a patch size of $h \times w \times d$ is given by:

$$\Omega(\mathrm{V-MSA}) = 4hwdC^2 + 2S_H S_W S_D hwdC$$

SV-MSA shifts the 3D volumes used by V-MSA by $\left(\frac{S_H}{2}, \frac{S_W}{2}, \frac{S_D}{2}\right)$ to increase interactions between different volumes. In practice, SV-MSA has a similar computational complexity to VMSA: the query-key-value attention in each 3D volume is computed as follows:

$$\text{Attention }(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + B\right)V$$

where $Q, K, V \in \mathbb{R}^{N_T \times d_k}$ denote the query, key, and value matrices, and $B \in \mathbb{R}^{N_T}$ denotes the information representing the relative positions. For computational efficiency, we first initialize a small-sized location matrix $\hat{\mathbf{B}} \in \mathbb{R}^{(2S_{H-1})\times(2S_{W-1})\times(2S_{D-1})}$ and create a larger location matrix B by extracting corresponding values from $\hat{\mathbf{B}}$.

- **Downsampling Layer.** Instead of utilizing the patch merging operation of the Swin transformer, we applied a direct stride convolution. The rationale behind using convolutional downsampling is to create a hierarchical representation that effectively models feature information at different resolutions.

2.3.2.2. Decoder

The transformer's decoder structure mirrors that of the encoder. Stage 1 employs stride deconvolution in the decoder to produce high-resolution output and incorporates skip attention to inform the decoder about the relationships between features in the encoder and those in the decoder's previous layer. Skip attention is designed based on cross-attention between the two aforementioned features. Consequently, the decoder can generate the desired segmentation mask while recognizing complex image patterns and structures. The decoder's overall structure is identical to the encoder's.

Skip Attention

Instead of using the skip connection typical of UNet, we employed V-MSA, where upsampling features are queried, and encoder features serve as the key or value. This method facilitates deeper information integration and finer feature alignment by extracting features, enhancing the interaction between features at different scales within the decoder, and contributing to more accurate identification and reconstruction of complex structures and patterns in the image. The overall skip attention structure is depicted in Figure 2c.

2.3.3. Stage 2: 3D UNet for localized texture refinement.

We used the well-defined nnUNet's 3D Full Resolution U-Net in Stage 2 because the 3D U-Net Full Resolution network structure of nnUNet is a deep learning model primarily designed for medical image segmentation tasks. This model is optimized to process high resolution 3D volumetric data and is based on the U-Net architecture. Here, we will describe the components and working principles of this network in detail.

2.3.3.1. Encoding

The input to the nnUNet 3D U-Net Full Resolution model is a 3D volumetric dataset of size $C \times D \times H \times W$, where C is the number of channels, D is the depth, H is the height, and W is the width. Medical imaging data typically have multiple channels, each representing different sequences or modalities. The encoding path constitutes the lower half of the network, progressively compressing the input data to extract high-level features. The encoding path consists of multiple stages of convolutional blocks and downsampling layers. Each stage includes the following:

- **Convolutional Block.** Each stage employs two consecutive 3D convolutional layers, each followed by batch normalization and a ReLU activation function. These convolutional blocks extract spatial features and learn the correlations between neighboring voxels.
- **Downsampling.** At the end of each stage, a 3D max pooling layer is used to halve the spatial resolution of the feature maps, preserving essential information while reducing their size.

This encoding process is repeated multiple times, typically involving 4-5 stages of downsampling. The number of convolutional filters doubles at each stage, allowing the network to learn more features in deeper layers.

2.3.3.2. Bottleneck

The final stage of the encoding path is the bottleneck, which operates at the lowest resolution and extracts the highest-level features of the network. The bottleneck consists of two 3D convolutional layers, batch normalization, and ReLU activation functions. The bottleneck processes the high-level features extracted by the encoding path before passing them to the decoding path.

2.3.3.3. Decoding

The decoding path constitutes the upper half of the network, restoring the compressed features to their original resolution. The decoding path is also divided into multiple stages, each consisting of the following components:

- **Upsampling**. A transposed convolution (or up-convolution) layer is used to double the spatial resolution of the feature maps. Upsampling helps restore the spatial resolution lost in the encoding path.
- **Skip Connections.** At each stage of the decoding path, feature maps from the corresponding resolution in the encoding path are concatenated. Skip connections transfer the detailed information from the encoding path to the decoding path, ensuring that the restored feature maps contain precise and detailed information.

- **Convolutional Block.** After upsampling and skip connections, two 3D convolutional layers, batch normalization, and ReLU activation functions are applied again. These convolutional blocks fine-tune the upsampled feature maps and contribute to generating the final output.

In the decoding path, the number of convolutional filters is halved at each stage, symmetrically to the encoding path. This helps maintain the complexity of the feature maps while restoring the original resolution. After the final stage of the decoding path, a 1x1x1 3D convolutional layer is used to produce the final output. This layer generates outputs corresponding to the number of classes, calculating the class probability for each voxel. The output size is $C \times D \times H \times W$, where C represents the number of classes. Following the output layer, a softmax activation function is applied to calculate the class probabilities for each voxel. The softmax function outputs the probability that each voxel belongs to a specific class.

2.3.4. Multi-Scale Patch Extraction

In the proposed model, the input data for stages 1 and 2 are generated by creating patches from the original images. The size of these patches is essential for effectively capturing global or local information. To determine the appropriate patch size, we referenced the methods employed by medical imaging professionals. These professionals typically begin with a general structure analysis of the aorta using a wide view, followed by a more detailed segmentation process using a close-up view. For Stage 1, the patch size was selected to ensure 100% inclusion of the foreground, enhancing the model's ability to extract global information, as all foreground details contribute to selfattention computations. For Stage 2, the patch size was chosen to include 25% of the foreground, enabling the model to capture detailed local features efficiently while managing computational resources. By varying the patch sizes across stages, the proposed model effectively learns both global and local features.

2.3.5. Loss: Dice Similarity Coefficient + Cross-Entropy

To improve the model's performance in AD segmentation, we used a loss function that combines cross-entropy (CE) and Dice similarity coefficient (DSC) losses. This combination is aimed at reducing the difference between the network's output and the goal while increasing the accuracy of the segmentation. The loss function L is defined as follows:

$$L(\text{ output, target}) = \omega_{\text{ce}} \cdot \text{CE (output, target)} + \omega_{\text{dice}} \cdot \text{DSC (output, target)}$$

where ω_{ce} and ω_{dice} denote the weights of the CE and the DSC losses, respectively. The default value for both ω_{ce} and ω_{dice} is 1 . The CE loss is used to measure the difference between the model's predicted probability distribution and the actual label distribution and is calculated as follows:

$$\mathrm{CE}(\text{ output, target }) = -\sum_{i} \text{ target }_{i} \log\left(\text{ output }_{i}\right)$$

The cross-entropy loss is a widely used loss function for classification tasks, including segmentation. It quantifies the difference between two probability distributions - the predicted distribution and the true distribution (ground truth). For each pixel or voxel in the image, the cross-entropy loss is calculated as shown above. This formula computes the negative log likelihood of the predicted probabilities, penalizing the model more when it is confident about an incorrect prediction. This loss encourages the model to produce probability distributions that are close to the actual label distributions.

In contrast, the DSC loss is used to measure the similarity between the two samples and is defined as follows:

$$\text{DSC(output, target)} = \frac{2\sum_{i} \text{ output }_{i} \times \text{ target }_{i} + \epsilon}{\sum_{i} \text{ output }_{i} + \sum_{i} \text{ target }_{i} + \epsilon}$$

The Dice similarity coefficient is a measure of overlap between two samples. In segmentation tasks, it is used to evaluate the accuracy of the predicted segmentation against the ground truth. The DSC is defined as shown above. Here, the numerator represents the intersection of the predicted and true segmentation masks, while the denominator represents the union. The small constant ϵ is added to avoid division by zero. The DSC loss is particularly useful in scenarios where the classes are imbalanced, as it directly optimizes the overlap between the predicted and true masks, leading to better segmentation performance.

By using this combined loss function, the model is better equipped to learn both the detailed local features and the global structure of the segmentation targets, leading to improved performance in medical image segmentation tasks. Combining cross-entropy and Dice similarity coefficient losses allows the model to benefit from the advantages of both loss functions. Crossentropy loss provides a strong gradient signal for individual pixel-wise classification, ensuring that the model learns to assign high probabilities to the correct classes. Dice loss, on the other hand, optimizes for overall overlap between the predicted and true segmentations, addressing issues related to class imbalance and ensuring that the segmentation mask is as accurate as possible. With this loss function, the model can capture the various features of the image well while maintaining the precision of the segmentation. These advantages motivated us to apply this loss function to our model.

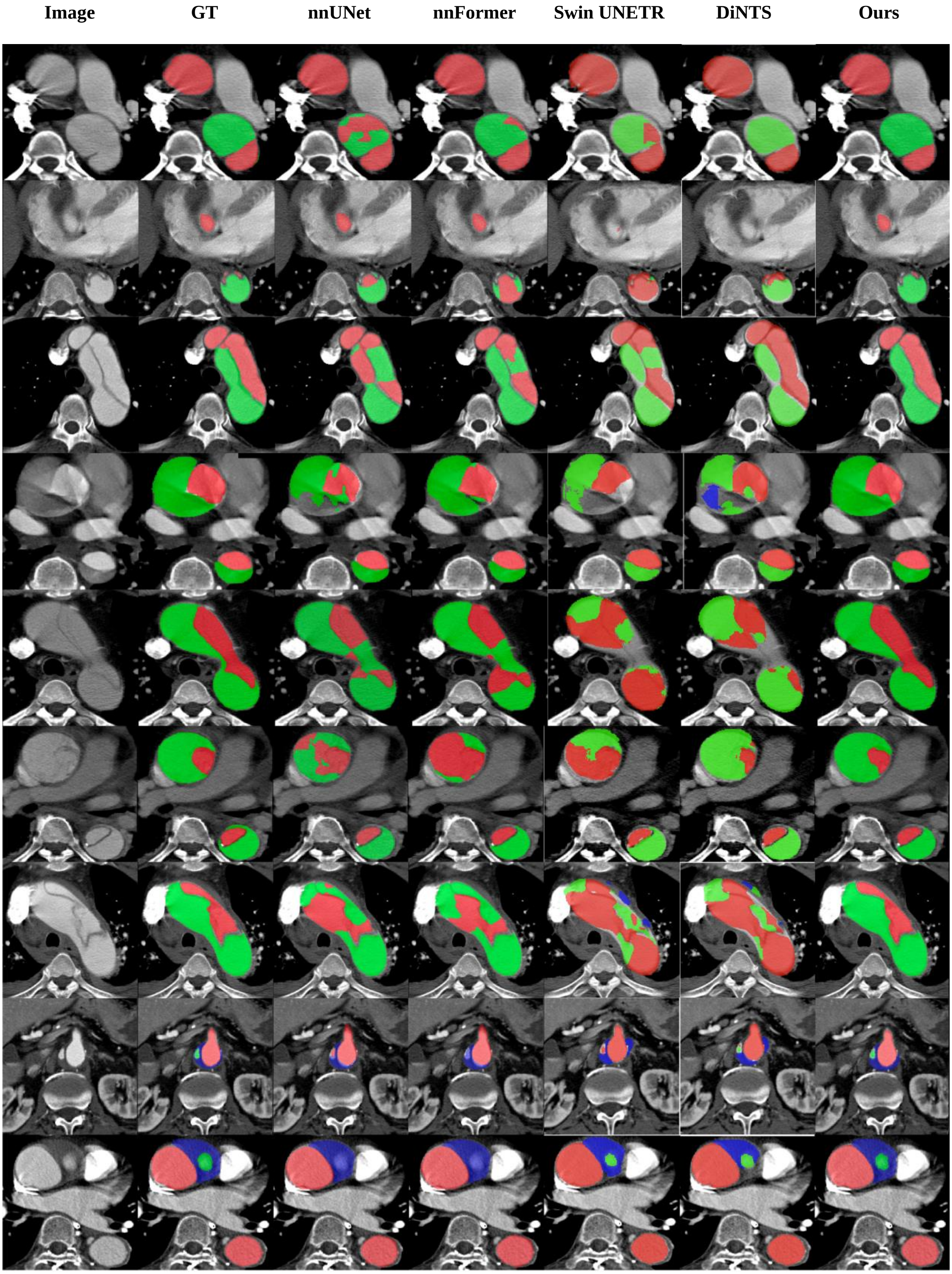

Figure 3: Qualitative evaluation compared to State-Of-The-Art models. Red, green, and blue represent the TL, FL, and TH regions, respectively.

3. Experiments

Model	TL (μ ± σ)	FL (μ ± σ)	TH (μ ± σ)
nnUNet [15]	0.907 ± 0.101	0.854 ± 0.211	0.589 ± 0.300
nnFormer [9]	0.906 ± 0.102	0.857 ± 0.206	0.615 ± 0.278
Swin transformer [8]	0.815 ± 0.114	0.720 ± 0.220	0.576 ± 0.296
DiNTS [18]	0.834 ± 0.099	0.750 ± 0.228	0.578 ± 0.291
Ours	**0.917 ± 0.097**	**0.882 ± 0.167**	**0.630 ± 0.304**

Table 1: This table presents the performance of the SOTA models including nnUNet, nnFormer, Swin Transformer, and DiNTS in the Medical Image Segmentation Decathlon, using the internal test dataset. The table shows the mean and standard deviation values of the metric Dice similarity coefficient. This metric measures the similarity among the datasets for the three substructures: TL, FL, and TH.

Model	CV (per case)
nnUNet [15]	6,005 mm^3
nnFormer [9]	6,434 mm^3
Swin transformer [8]	23,193 mm^3
DiNTS [18]	15,016 mm^3
Ours	**2,733** mm^3

Table 2: This table presents the performance of the SOTA models including nnUNet, nnFormer, Swin Transformer, and DiNTS in the Medical Image Segmentation Decathlon, using an internal test dataset. The table shows the mean and standard deviation values of the metric confusion volume (CV).

3.1. Performance Comparison with Baseline Models

In this study, the proposed model was compared with the state-of-the-art (SOTA) models nnUNet, nnFormer, Swin transformer, and DiNTS from the Medical Image Segmentation Decathlon. Using the internal test dataset, the proposed model demonstrated superior performance over the other models in terms of Dice Similarity Coefficient (DSC) and Confusion Volume (CV) metrics.

Based on the quantitative metrics in Table 1, the four SOTA models already perform well, with nnUNet achieving a DSC of 0.907 ± 0.101 for TL and 0.854 ± 0.211 for FL, while nnFormer scores

0.906 ± 0.102 for TL and 0.857 ± 0.206 for FL. Swin transformer and DiNTS also show strong performance with DSC scores of 0.815 ± 0.114 for TL and 0.720 ± 0.220 for FL, and 0.834 ± 0.099 for TL and 0.750 ± 0.228 for FL, respectively. Our model's scores of $0.917 \pm 0.097, 0.882 \pm 0.167$, and 0.630 ± 0.304 may not seem significantly different from the baseline models in numerical terms. However, Figure 3 demonstrates that evaluating with DSC alone does not provide a sufficiently objective assessment of the models. As shown in Figure 3, even when the boundaries are clearly visible in the image, $nnUNet$ and nnFormer tend to overpredict or underpredict the true labels (TL) and false labels (FL), resulting in incorrect boundaries.

To quantitatively evaluate this, we introduced a metric called confusion volume (CV). The CV is designed to measure the volume of regions where the model incorrectly labels the segmentation, either by predicting false labels where there should be true labels or vice versa. The CV was calculated as follows:

For the ground truth (GT) true label volume T and the predicted false label volume F' :

$$V_1 = \text{Volume}\ (T \cap F') \ \ \text{if Volume}\ (T \cap F') \geq 100\ \text{mm}^3$$

Here, T represents the volume of the true labels in the ground truth segmentation, and F' represents the volume of the predicted false labels in the segmentation output. The intersection $T \cap F'$ measures the volume where the true labels in the ground truth overlap with the false labels in the prediction. If the volume of this intersection is $100\ \text{mm}^3$ or greater, it is considered significant, and this volume is assigned to V_1.

For the GT false label volume F and the predicted true label volume T' :

$$V_2 = \text{Volume}\ (F \cap T') \ \ \text{if Volume}\ (F \cap T') \geq 100\ \text{mm}^3$$

Similarly, F represents the volume of the false labels in the ground truth segmentation, and T' represents the volume of the predicted true labels in the segmentation output. The intersection $F \cap T'$ measures the volume where the false labels in the ground truth overlap with the true labels in the prediction. If the volume of this intersection is $100\ \text{mm}^3$ or greater, it is considered significant, and this volume is assigned to V_2.

The final CV is calculated as follows:

$$|\text{CV}| = V_1 + V_2$$

The CV is measured in volume (mm^3), showing how much of the predicted volume is incorrect compared to the actual volume. This sum represents the combined volume of significant regions where the model's predictions are incorrect. A higher CV indicates more significant discrepancies, while a lower CV indicates better alignment between the predicted and actual segmentations.

Table 2 presents the results of the CV evaluation. The proposed model achieved a CV of $2{,}733\ \text{mm}^3$

per case, significantly lower than nnUNet's 6,005 mm^3, nnFormer's 6,434 mm^3, Swin transformer's 23,193 mm^3, and DiNTS's 15,016 mm^3. From Table 2 and Figure 3, it can be seen that nnUNet and nnFormer do not accurately find the boundaries of the labels and missegment different regions in TL and FL. The proposed model, however, has addressed these issues to some extent, as indicated by the evaluation results with the CV in Table 2 and Figure 3. This demonstrates the effectiveness of the proposed model in providing more accurate segmentation results by reducing the confusion volume.

	TL (μ ± σ)	**FL (μ ± σ)**	**TH (μ ± σ)**	**Train (n)**	**Test (n)**
Long Cao et al. [10]	0.930 ± 0.010	0.910 ± 0.020	-	246	30
Zeyang Yao et al. [11]	0.850 ± 0.070	0.780 ± 0.210	0.520 ± 0.400	67	33
Lewis D. Hahn et al. [13]	0.884 ± 0.042	0.906 ± 0.027	-	125	28
Wobben, Liana D., et al. [12]	0.860 ± 0.055	0.850 ± 0.015	0.500 ± 0.230	125	22
Xiang, Dongqiao, et al. [14]	0.911 ± 0.039	0.884 ± 0.062	-	68	20
Zhang, Jinhui, et al. [19]	0.910 ± -	0.892 ± -	-	80	20
Ours	**0.942 ± 0.035**	**0.922 ± 0.089**	**0.593 ± 0.307**	-	98

Table 3: This table presents a comparative analysis of the proposed model with those in previous studies for AD segmentation. The table shows the mean (μ) and standard deviation (σ) values of the DSC in the segmentation of the aortic dissection into three substructures: TL, FL, and TH. In addition, it presents the combined number of training and validation samples (Train) as well as the number of testing samples (Test) used in each study. Zhang Jinhui et al. is a semi-supervised study applied with insufficient data, so please note that there is a slight difference from our research direction.

3.2. Performance Comparison with Previous Studies

As can be seen from Table 3, each study has a different composition of training and testing datasets, and some studies do not include TH segmentation. Furthermore, the number of testing data used varied from study to study.

These differences make it difficult to perform a direct comparison between our proposed method and previous studies. Nevertheless, we used an external dataset, which provided us with information about the generalizability and robustness of the model. The results are summarized in Table 3, showing that the proposed model maintains its performance under various datasets and conditions. In particular, the performance of DSC is higher than the models used in previous studies. Through this experiment, we demonstrated that our proposed method can perform consistently on external datasets.

For instance, Long Cao et al. achieved a DSC of 0.930 ± 0.010 for TL, 0.910 ± 0.020 for FL, and did not include TH segmentation. Their study used 246 training samples and 30 testing samples. Zeyang Yao et al. reported a DSC of 0.850 ± 0.070 for TL, 0.780 ± 0.210 for FL, and 0.520 ± 0.400 for TH, with 67 training samples and 33 testing samples. Lewis D. Hahn et al. showed results of 0.884 ± 0.042 $\text{for TL}, 0.906 \pm 0.027$ for FL, and did not include TH segmentation, using 125 training samples and 28 testing samples.

Wobben, Liana D., et al. achieved 0.860 ± 0.055 for TL, 0.850 ± 0.015 for FL, and 0.500 ± 0.230 for TH, with 125 training samples and 22 testing samples. Xiang, Dongqiao, et al. reported a DSC of 0.911 ± 0.039 $\text{for TL}, 0.884 \pm 0.062$ for FL, and did not include TH segmentation, with 68 training samples and 20 testing samples. Zhang, Jinhui, et al. achieved $0.910 \pm -$ $\text{for TL}, 0.892 \pm -$ for FL, and did not include TH segmentation, using 80 training samples and 20 testing samples.

In comparison, our proposed method achieved a DSC of 0.942 ± 0.035 $\text{for TL}, 0.922 \pm 0.089$ for FL, and 0.593 ± 0.307 for TH, using 98 training samples and 20 testing samples. These results show that our proposed method outperforms the previous studies in most cases, particularly in the TL and FL segments.

By using an external dataset, we demonstrated that our model not only generalizes well but also maintains robustness across various conditions. These findings underline the effectiveness of our proposed method in providing accurate and consistent segmentation results, which is crucial for clinical applications.

Model	TL (μ ± σ)	FL (μ ± σ)	TH (μ ± σ)
Transformer based 3D UNet with close-up view patch	0.906 ± 0.102	0.857 ± 0.206	0.615 ± 0.278
Transformer based 3D UNet with wide view patch	0.908 ± 0.097	0.882 ± 0.145	0.626 ± 0.261
Transformer based 3D UNet with wide view patch + CNN based 3D UNet with close-up view patch (Ours)	**0.917 ± 0.097**	**0.882 ± 0.167**	**0.630 ± 0.304**

Table 4: This table shows a comparison experiment between the Transformer based 3D UNet with the wide view patch extraction. The dataset used for evaluation is the internal test dataset. This table presents the mean and standard deviation values for the DSC metric.

3.3. Ablation Study

In Stage 1, we proposed nnFormer as the standard. The standard patch size of nnFormer is designed to include 33.3% of the foreground. However, our wide view patch size is designed to include 100% of the foreground. Therefore, to evaluate the contribution of wide view in Stage 1, we conducted an ablation study for patch size, and the results are presented in Table 6. Stage 1 with the wide view patch consistently performed better in all classes (TL, FL, and TH) than Stage 1 with the standard patch of nnFormer. Specifically, Stage 1 with the standard patch of nnFormer achieved a DSC of 0.906±0.102 for TL, 0.857±0.206 for FL, and 0.615±0.278 for TH. In contrast, Stage 1 with the wide view patch achieved a DSC of 0.908±0.097 for TL, 0.882±0.145 for FL, and 0.626±0.261 for TH. This improvement is particularly evident in the FL and TH categories, indicating the effectiveness of wide view in these regions. Based on these results, we can prove the effectiveness of wide view, i.e., this experiment proves that using wide view in Stage 1 to ensure that the patch contains the whole aorta can better learn global information (anatomical structure). Additionally, the reason why we did not do the patch size experiment for Stage 2 is that we used the standard patch of nnUNet, which includes 25% of the foreground. We defined this as a close-up view patch. The combination of Stage 1 with the wide view patch and Stage 2 with the standard patch of nnUNet further improved the performance, achieving a DSC of 0.917±0.097 for TL, 0.882±0.167 for FL, and 0.630±0.304 for TH. These results demonstrate that our proposed method effectively combines the strengths of both patch strategies to provide superior segmentation results.

3.4. Hyperparameters and Development Environments

The hyperparameters for our proposed method are automatically determined by the heuristic rules of nnU-Net, while data-independent parameters are consistent with nnU-Net. Table 4 shows the hyperparameters of our proposed method, and the development environments are provided in Table 5. For both stages 1 and 2, we used a batch size of 2 and trained the model for a total of 1000 epochs. The learning rate schedule followed the PolyLRScheduler, with an initial learning rate set at 0.01 and a weight decay of 3×10^{-5}. The optimizer used for training was SGD, and the loss function combined Dice and Cross-Entropy. The number of pooling layers per axis was set to [5, 4, 4], and the pooling kernel size was [[2,2,2], [2,2,2], [2,2,2], [2,2,2], [2,2,2], [2,2,2], [2,2, 2]]. The convolution kernel size for all layers was [[3,3,3], [3,3,3], [3,3,3], [3,3,3], [3,3,3], [3, 3,3], [3,3,3]].

The development environment for our proposed method included Ubuntu 18.04 LTS as the operating system. The CPU used was an Intel(R) Xeon(R) Gold 5217 CPU @ 3.00GHz32 − Core Processor, and the GPU was a single NVIDIA TITAN RTX with 24GB of memory. The CUDA version was 12.2, and the deep learning framework used was PyTorch version 1.7.

Component	**Stages 1 & 2**
Batch size	2
Total epochs	1000
Learning rate schedule	PolyLRScheduler [18]
Initial learning rate	0.01
Weight decay	3e-5
Optimizer	SGD [19]
Loss function	Dice and Cross-Entropy
Number of pools per axis	[5, 4, 4]
Pooling kernel size	[[2, 2, 2], [2, 2, 2], [2, 2, 2], [2, 2, 2], [2, 1, 1]]
Convolution kernel size	[[3, 3, 3], [3, 3, 3], [3, 3, 3], [3, 3, 3], [3, 3, 3], [3, 3, 3]]

Table 5: This table shows the hyperparameters of the Stage 1 and 2 models.

Windows/Ubuntu version	Ubuntu 18.04 LTS
CPU	Intel(R) Xeon(R) Gold 5217 CPU @ 3.00GHz 32-Core Processor
GPU (Number and type)	1 NVIDIA TITAN RTX (24G)
CUDA version	12.2
Deep learning framework	Pytorch (1.7)

Table 6: This table shows the development environments.

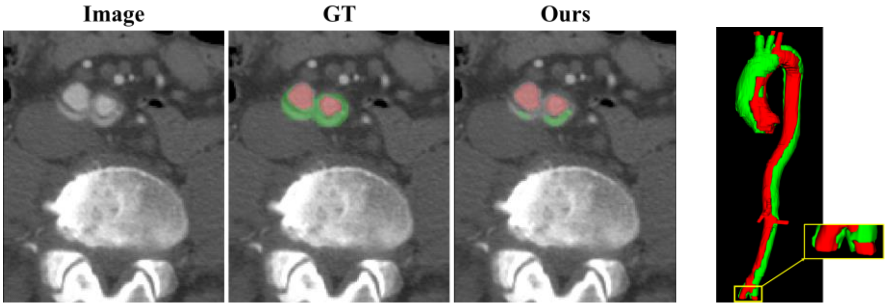

Figure 4: Axial images, GT labels, and prediction results of our method. Red and green represent the TL and FL, respectively. The image on the right is the 3D rendering image of the GT. We marked the part corresponding to the axial on the left with the yellow box.

4. Discussion

Quantifying AD is crucial for clinical decision-making. Clinicians rely on extracting the centerline from the TL region and generating curved planar images to measure the diameter of each region. This measurement helps determine the need for surgical intervention or assess disease progression during follow-up. However, differentiating between TH and FL is challenging due to their overlapping characteristics. Thrombus can form within or adjacent to the FL, and delayed contrast enhancement in the FL can mimic the low HU values of thrombus, complicating consistent labeling. This inconsistency leads to lower DSCs and higher standard deviations in various studies and our results.

Traditional AD segmentation methods often rely on CNNs with inherent biases, struggling to distinguish between TL and FL due to their similar HU values. This similarity can result in inconsistent segmentation, affecting accurate centerline extraction and quantitative analysis. To address these challenges, our proposed method was designed to conceptualize the segmentation procedure of a medical imaging analyst for AD. Initially, the analyst identifies the aorta's shape in the overall view of the CT image to locate the TL and FL. This step was implemented in Stage 1 using a 3D transformer with a multi-scale patch extraction scheme. By using the multi-scale patch extraction, our model can analyze the aorta comprehensively, enabling accurate segmentation of TL and FL based on their overall positions.

In Stage 2, we combined the input from the previous stage with the CT image and applied the multi-scale patch extraction scheme again, this time with a 3D UNet, to refine local features. This approach leverages both global and local contexts, enhancing segmentation accuracy.

Our experiments highlighted several key points:

1 We compared our proposed method with baseline models such as nnUNet and nnFormer, and our experiments demonstrated superior performance, as evidenced by better evaluation metrics.

2 The multi-scale patch extraction approach in Stage 1 significantly improved model performance. Using the entire CT image as input for global feature extraction proved more effective.

3 We evaluated our proposed method on an external dataset to confirm its generalizability and robustness.

Despite these positives, there are areas for improvement:

1 The performance of AD segmentation at the bottom of the aorta needs enhancement. Figure 4 shows that segmentation accuracy decreases at the bottom end of the aorta as blood vessels become branched and thinner. Anatomical challenges in this region may degrade performance, especially if FL or TH are located in the thinned areas of the vessels. Another factor is the variability in label consistency at the bottom of the aorta due to longterm annotation processes. We plan to address this through further study and refinement of labeling in the aorta's lower branches.

2 Our segmentation technique does not separately segment the carotid artery, iliac branch, and abdominal branches extending toward the brain, complicating centerline extraction. One approach to mitigate this issue is to assume these branches are smaller than the true lumen and use morphological operations to remove them before extracting the centerline. However, this method is challenging when aortic dissection is severe, and the true lumen is significantly narrowed. Future research should focus on integrating the segmentation of these branches into the overall process and providing them as separate outputs. This will enhance centerline extraction accuracy and improve overall segmentation performance.

3 The segmentation performance for the TH region is slightly lower relative to other regions. This is primarily due to the inherent difficulty in clearly delineating the TH region in enhanced CT images. The variability in the appearance of the TH region—sometimes brighter than the FL depending on the timing of contrast injection and blood flow, and at other times darker—poses a significant challenge in accurate segmentation. Consequently, this variability leads to decreased performance in TH segmentation compared to other regions. Furthermore, the smaller size of the TH region exacerbates the issue, as the DSC score is more susceptible to small false positives. This sensitivity to minor inaccuracies represents a secondary cause of the lower segmentation performance for the TH region. To address this challenge, we plan to apply various evaluation metrics to objectively assess small regions as a future study.

Moreover, we believe that our proposed method holds potential beyond aortic dissection and can be adapted for various medical segmentation tasks. Our ongoing research efforts are aimed at expanding and generalizing this method to a broader array of medical applications.

5. Conclusion

Despite the identified limitations, our proposed method has demonstrated significant value through extensive quantitative evaluations, contributing substantially to AD segmentation research. The method's ability to accurately segment the TL, FL, and TH under various conditions highlights its robustness and potential for clinical application. The combined use of multi-scale patch extraction techniques effectively captures global and local features, leading to improved segmentation performance. Future work will focus on addressing the current limitations, such as enhancing segmentation accuracy at the aorta’s lower end and incorporating the segmentation of additional branches. By continuously refining our approach, we aim to advance the state-of-the-art in AD segmentation, providing valuable tools for clinical decision-making and pushing the boundaries of research in this field.

References

[1] Khan, I. A., & Nair, C. K. (2002). Clinical, Diagnostic, and Management Perspectives of Aortic Dissection. Chest, 122(1), 311–328. doi:10.1378/chest.122.1.311.

[2] M. Bednarska, E. Stolarz, M. Stopiński, A. Biederman, J. Polkowski, O. Kapuściński, Early diagnosis of aortic dissection. The key to successful surgical treatment., Medical Science Monitor (1996) CS348–351.

[3] Heggie, J., & Karski, J. (2006). The Anesthesiologist's Role in Adults with Congenital Heart Disease. Cardiology Clinics, 24(4), 571–585. doi:10.1016/j.ccl.2006.08.006.

[4] 2014 ESC Guidelines on the diagnosis and treatment of aortic diseases. (2014). European Heart Journal, 35(41), 2873–2926. doi:10.1093/eurheartj/ehu281.

[5] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 234–241. doi:10.1007/978-3-319-24574-4_28.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In NIPS, 2017.

[7] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[8] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 10012-10022).

[9] Zhou, H. Y., Guo, J., Zhang, Y., Yu, L., Wang, L., & Yu, Y. (2021). nnformer: Interleaved transformer for volumetric segmentation. arXiv preprint arXiv:2109.03201.

[10] Cao, L., Shi, R., Ge, Y., Xing, L., Zuo, P., Jia, Y., … Guo, W. (2019). Fully Automatic Segmentation of Type B Aortic Dissection from CTA Images Enabled by Deep Learning. European Journal of Radiology, 108713. doi:10.1016/j.ejrad.2019.108713.

[11] Z. Yao, W. Xie, J. Zhang, Y. Dong, H. Qiu, H. Yuan, Q. Jia, et al., ImageTBAD: A 3D Computed Tomography Angiography Image Dataset for Automatic Segmentation of Type-B Aortic Dissection, Computational Physiology and Medicine 12 (2021). doi:10.3389/fphys.2021.732711.

[12] Wobben, L. D., Codari, M., Mistelbauer, G., Pepe, A., Higashigaito, K., Hahn, L. D., … & Willemink, M. J. (2021, November). Deep learning-based 3D segmentation of true lumen, false lumen, and false lumen thrombosis in type-B aortic dissection. In 2021 43rd Annual

International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 3912-3915). IEEE.

[13] Hahn, L. D., Mistelbauer, G., Higashigaito, K., Koci, M., Willemink, M. J., Sailer, A. M., … Fleischmann, D. (2020). CT-based True- and False-Lumen Segmentation in Type B Aortic Dissection Using Machine Learning. Radiology: Cardiothoracic Imaging, 2(3), e190179. doi:10.1148/ryct.2020190179.

[14] Xiang, D., Qi, J., Wen, Y., Zhao, H., Zhang, X., Qin, J., ... & Zheng, C. (2023). ADSeg: A flap-attention-based deep learning approach for aortic dissection segmentation. Patterns, 4(5).

[15] Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nature methods, 18(2), 203-211.

[16] CHEN, Liang-Chieh, et al. Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv 2014. arXiv preprint arXiv:1412.7062, 2014.

[17] RUDER, Sebastian. An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747, 2016.

[18] He, Y., Yang, D., Roth, H., Zhao, C., & Xu, D. (2021). Dints: Differentiable neural network topology search for 3d medical image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5841-5850).

[19] Zhang, J., Liu, J., Wei, S., Chen, D., Xiong, J., & Gao, F. (2023). Semi-supervised aortic dissections segmentation: A time-dependent weighted feedback fusion framework. Computerized Medical Imaging and Graphics, 106, 102219.

[20] Jung, Ji-Hoon, et al. "ZOZI-Seg: A transformer and UNet cascade network with Zoom-Out and Zoom-In scheme for aortic dissection segmentation in enhanced CT images." Computers in Biology and Medicine 175 (2024): 108494.

Abstract (with Korean)

대동맥 박리(AD)는 대동맥 내벽의 파열로 인해 대동맥 벽 층 사이로 혈액이 흐르게 되어 생명을 위협하는 합병증을 초래하는 심각한 상태입니다. AD 관리에는 영상 촬영, 의학적 치료, 때로는 수술이 포함됩니다. 고해상도 이미지를 빠르게 생성하는 CT 스캔은 AD 진단 및 예후 평가에 일반적으로 사용됩니다. 진성 내강(TL), 가성 내강(FL) 및 혈전(TH)의 정확한 분할은 필수적이지만 수동 측정은 시간이 많이 걸리고 변동성이 큽니다. 이를 해결하기 위해 컴퓨터 비전, 기계 학습 및 딥 러닝 방법이 도입되었습니다. CNN 기반 모델은 의료 영상 분석에서 중요한 역할을 했지만, 해부학적 구조를 종합적으로 이해하는 데 한계가 있습니다. 이러한 한계를 극복하기 위해 전역 컨텍스트 정보를 추출하는 데 뛰어나지만 로컬 텍스처 세부 정보를 캡처하는 데 덜 효과적인 Transformer 기반 모델이 도입되었습니다. 따라서 본 연구에서는 CNN과 Transformer의 장점을 결합한 모델을 제안합니다. 우리는 두 단계 모델을 설계했습니다. 첫 번째 단계는 대동맥의 전역 정보를 학습하기 위해 3D Transformer UNet을 사용하고, 두 번째 단계는 TL, FL 및 TH의 세부 텍스처를 학습하기 위해 3D UNet을 사용합니다. 추가적으로, 다중 스케일 패치 추출 방법을 적용하여 대동맥의 전역 정보와 세부 텍스처를 효과적으로 캡처합니다. 이 모델의 두 단계 접근 방식—전역 컨텍스트를 위한 3D Transformer UNet과 로컬 텍스처를 위한 3D CNN UNet 사용—은 절제 연구에서 검증되었습니다. 모델의 성능은 아산병원 데이터셋을 사용하여 평가되었고, nnUNet 및 nnFormer와 같은 기존 모델과 비교되었습니다. 제안된 방법은 TL, FL 및 TH에 대해 각각 0.917, 0.888 및 0.630의 Dice 유사 계수를 달성하여 최고의 분할 정확도를 입증했습니다. 모델의 견고성과 일반화 가능성은 외부 데이터셋을 사용하여 추가로 평가되었으며, 이는 다양한 임상 환경에서 AD 진단 및 치료를 개선할 가능성을 보여줍니다.