Leveraging Transformers for Vessel Tracing

Zhiyu An, Numi Sveinsson, Zihan Wang, Austin Zane

Introduction

Cardiovascular diseases are the leading cause of death in the world. Cardiovascular bloodflow simulations have become an important part of cardiovascular research, giving key insights into blood velocity, pressure and wall shear stress of healthy and diseased blood vessels and hearts. In order to set up a blood flow simulation, a 3D geometric model of the blood vessels must first be constructed. One way to determine these patient-specifically is by using medical image scans of patients, i.e. CT or MR images. The most common way to construct 3D models of vasculature is to represent the vessels as connecting 1D lines in 3D space. These lines are often referred to as centerlines. The full 3D models can then be built by estimating the width of the vessel at each point along the centerline. The generation of centerlines is called vessel tracing and is the core problem of vasculature 3D model generation.

Developing efficient vessel-tracing algorithms is crucial for imaging-based diagnosis and treatment of vascular diseases. This task often requires multiple steps, the first of which being image segmentation. The image segmentation problem is formulated with an input 3D medical image and an output being the segmentation of the vessels. The segmentation can then be post-processed to determine the direction of the vessel at that point. Current work for vessel segmentation consists of pixel-wise CNNs, U-Net, etc. In this project, we leverage Transformers for vessel tracing[1].

Our contributions are two-fold:

1. We were the first to introduce Transformers to general 3D vessel image data;
2. We proposed a new task defining how to trace vessels using machine learning algorithms.

Related Work and Context

Medical image segmentation involves isolating objects of interest in 2D or 3D images. In our case, detecting blood vessels in data obtained from magnetic resonance imaging (MRI) and computed tomography (CT). Unlike typical image tasks, segmentation requires prediction on a pixel-by-pixel scale. This means that our network must accept input of arbitrary dimension and output an image of the same shape. The fully convolutional neural network (FCNN) was an important step in addressing this challenge. FCNNs are comprised solely of locally connected layers, e.g. convolution, pooling, and upsampling.

This model has two main shortcomings: the network must be run separately for each pixel and there is a tradeoff between localization accuracy and the use of context. This is largely due to FCNNs only having the ability to "contract" the image data. The advent of U-Net in 2015 was a watershed moment in medical image segmentation [1]. It addresses these problems by adding an "expansion" path after the contraction, which allows the network to learn localized classification information without sacrificing the use of context. These two parts of the model are almost symmetrical and form a "U" shape, hence the name.

For decades, convolution-based networks like U-Net were the standard in image classification

[1] Our code can be found **here** and here and our dataset can be found **here**.

tasks. However, the overwhelming success of self-attention-based architectures on NLP tasks quickly motivated researchers to begin exploring its potential on images. The Vision Transformer (ViT) model is the first to demonstrate that Transformers can replace standard convolutions in deep neural networks on large-scale image datasets rather than simply augmenting them [2]. Although Transformer models tend to be more data-hungry than their convolutional counterparts, they are capable of learning more long-range relationships.

With the success of ViT, U-Net was revisited and improved upon to eventually produce UNetFormer and UNetFormer+, which use a Transformer encoder and CNN-based and Transformer-based decoders, respectively [3]. In both cases, the encoder portion undergoes self-supervised pretraining with a temporary lightweight CNN decoder. Both achieve state-of-the-art performance on common segmentation benchmarks, with UNetFormer performing better on smaller organs and UNetFormer+ performing better on larger organs. However, UNetFormer+ is smaller and much more computationally efficient.

A member of our team is currently working on a pipeline for automatic vasculature tracing using local vessel segmentations from a U-Net model. The segmentations are post-processed to choose the next point to move to and the next sub-volume to segment. Taking these local steps allows for the construction of extensive vascular networks by only looking at one sub-volume at a time. Previous work has sought to automate this process using convolution-based models, but nothing has been done in the way of Transformer-based models [4].

This is related to a larger project in the Prof. Shadden lab at UC Berkeley, website here. The lab focuses on finite-element modeling for fluid dynamic simulations of blood vessels and the heart. As a part of this project, an open-source software package has been developed called SimVascular, website here, as well as an open and free repository of vascular models called the Vascular Model Repository, available here [5].

Problem Statement and Goal

Vessel Tracing: The goal of blood vessel tracing is to construct the vasculature centerlines. There exist many ways to approach this problem, but for this project a sub-volume step-wise method is chosen.

This method only looks at one small image volume within the global vasculature tree at a time. This sub-volume containing a vessel segment is processed to choose the next point in the centerline by returning two variables, Vessel Direction and Vessel Size, at the current point. The direction is used to choose the next point in the centerline and the vessel size is used to determine the size of the next sub-volume to extract. Once the new point is determined, a new sub-volume is extracted and processed, and so forth.

In this project, two different processing methods will be explored via deep learning:

- **Approach 1: Classification**

  The sub-volume is segmented into voxels belonging to vessel (1) or not belonging to vessel (0). The binary segmentation can then be converted into a surface mesh using marching cubes and a centerline calculated using the open-source Vascular Model Tool Kit (VMTK) method, which can be found here. This approach is highly dependent on accurate segmentations, and that is where neural networks will be applied. The post-processing steps are deterministic using geometric mathematical calculations, hence no learning is currently utilized.

- **Approach 2: Regression**

  Instead of the multiple steps involved in approach 1, the direction and size are directly predicted from the image data using regression. Note that there is not always a single vessel in a sub-volume, i.e. a bifurcation may be present. For vessel tracing, our ultimate goal is to create a model that can give different directions to all branches present. For simplicity, we additionally predict a binary bifurcation label.

Goal: The project aims to use a Transformer-based model to 1) segment local vasculature from 3D medical image volumes and 2) predict the vessel orientation, size and bifurcation at a given point.

Methods

Model

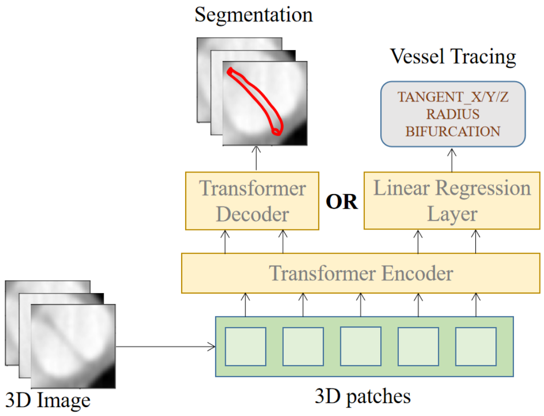

Figure 1: *Our model architecture.*

We propose a two-step procedure to predict the relevant vascular information from the medical images. First, we start from a local region of the 3D model and use a Transformer encoder to output the hidden state of the patched medical image. For segmentation task, we use a Transformer decoder to decode the hidden states to the size of the patches image, each pixel with a probability to be the segmented vessel zone. For the regression and classification tasks, we use a hidden size×5 linear layer after the Transformer encoder to regress on the direction of x, y, z, and the radius of the vessel, and classify on bifurcation. We sum the loss functions of regression and classification to train vessel tracking model, but train the segmentation and vessel tracking models separately.

Data Augmentation

We adopt conventional data augmentation methods for training. We augment each image by rotating, mirroring and adding Gaussian noise. We then use the original data point and randomly sample three augmented data points for training.

Pre-training

As Transformer models need a lot of data to train, we also attempt to adopt pre-training for the ViT model to overcome data scarcity. Firstly, we perturb the training data with random masking and Gaussian noise, and train a masked-autoencoder Transformer model. We then train a new model, using the pre-trained encoder, for the relevant tasks and report the results below.

Datasets

Our dataset consists of 101 3D models of vascular networks constructed from medical images. 39 models were contructed from CT image stacks and 62 from MR image data. They all come from the free and open Vascular Model Repository (VMR)[5]. An example of an image volume with its respective 3D vascular model and centerline can be seen in Figure 2.

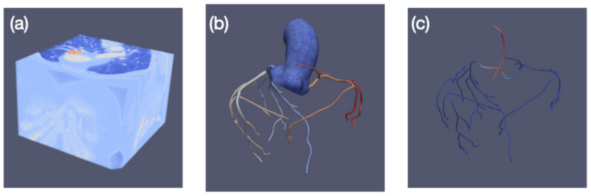

Figure 2: *An example of an (a) image volume, (b) 3D vasculature model and its (c) centerline.*

Data Curation

Since the step-wise algorithm relies on taking local steps, the training data must reflect that. Firstly, the global volumes are sampled along the known centerlines to obtain sub-volumes containing local vascular segments. This sampling procedure is shown in Figure 3(a). These samples of raw image data are the inputs to our model. All sub-volumes are then resampled to be

64x64x64 in resolution to keep the inputs standardized. There are key differences in the data curation of labels based on the approaches we compared:

- **Task 1 - Classification**: Identical global volumes are obtained with binary labels, 1 for voxels within vessels and 0 outside. To generate groundtruth for training, these global volumes are sampled identically to the raw image data. That way matching pairs of raw image and binary labelled images are created.

- **Task 2 - Regression**: The ground truth labels for this task are threefold:

  1. The three components of the tangent vector, normalized so the vector has a length of one.
  2. The size of the vessel as the maximum enscribed radius (cm).
  3. Binary label if the volume contains a bifurcation (1) or not (0).

  For each raw image sampling, the respective information from the centerline is kept for training.

In total, Table 1 shows the number of samples obtained based on image modality.

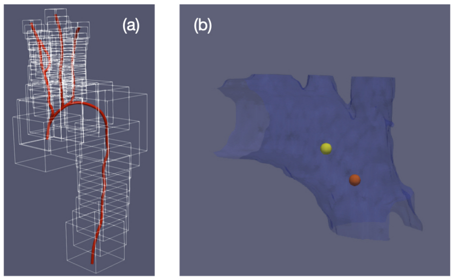

Figure 3: *How sampling is done along vasculature to generate thousands of vessel segments to train on, (a) samples for one vasculature and (b) an example of a sample containing two points representing the tangent vector in the center.*

Sample Type	Amount
CT - Training	11322
CT - Validation	1760
MRI - Training	16312
MRI - Validation	2902

Table 1: *Number of samples obtained from the 101 vasculature models.*

Loss Functions

A binary cross entropy (BCE) loss function is used for the classification task. For the regression task, mean squared error (MSE) is used during training.

Experiments

For the segmentation task, we compare the result of our method with a 3D U Net [6]. Both are trained and tested on the same data. For the regression task, since this is a newly proposed task and there have not been other methods released for the task, we report our results alone as a baseline for future work.

Implementation Details

We use a starting learning rate of 1e-4, batch size of 128, 0.01 stdev Gaussian noise, and train for 400 epochs. We shrink the learning rate to 1e-5 when the performance plateaus. For other hyper-parameters, we tune in terms of number of layers, attention heads, and hidden dimensions sizes. Because traversing all setting takes an extremely long time, we searched on 8 different settings, comparing every hyperparameter and using each variable's best setting, which is: 6 layers, 4 heads, 2048 feed forward hidden size, using data augmentation, using pre-training. Our experiments are run on Nvidia-V100 GPUs.

Results

Results for can be seen in Table 2 and Table 3. Table 2 shows our best Dice score result for vessel segmentation compared to an implementation of a 3D fully convolutional U-Net architecture [6]. Table 3 shows the results for the regression task, the accuracy of predicting bifurcations, and mean squared error for the vessel size and tangent vector predictions. Plots for the performance on the validation dataset during training can be seen in Figure 4.

Method	Average Dice
UNet-3D	0.761
Ours	**0.806**

Table 2: *Comparison of Dice score for our method and a 3D U-Net implementation on the same data.*

P(%)	R(%)	RMSE$_{rad}$	RMSE$_{vec}$
60.2	76.5	0.11	0.39

Table 3: *Results for the regression tracing task. P / R is for precision and recall on the bifurcated image as positive. RMSE computes the RMSE errors of the vessel radius and tangent vector. We directly adopted the settings in Table 4 which gave the best result.*

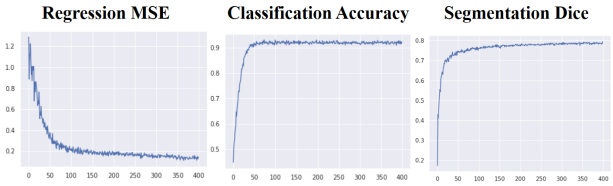

Figure 4: *Plots for experiment results on validation set.*

Ablation Study

We tune hyperparameters and control data augmentation / pre-training on different settings. The results can be seen in Table 4. Results show that data augmentation and pre-training can consistently enhance model performance, but the pre-training does not help significantly. This might due to the fact that we do not use unlabelled data from outside of our dataset, and the patterns the model learn from our current dataset is valuable enough for generalization.

Settings	Result
L/H/FF size,/ViT size/Aug/Pt	Dice Score (%)
4, 4, 1024, 256, False, True	79.2
4, 4, 1024, 256, True, True	80.0
6, 4, 1024, 512, False, True	79.6
6, 4, 1024, 512, False, False	79.4
6, 4, 2048, 512, False, True	80.0
6, 4, 2048, 512, True, True	**80.6**
6, 4, 2048, 512, True, False	80.4
6, 8, 1024, 512, False, True	79.5
6, 8, 1024, 512, False, False	79.4

Table 4: *Results for the segmentation task. L: number of layers, H: attention heads, FF size: hidden size for the feed forward layer, ViT size: hidden size for the ViT encoder/decoder, Aug: data augmentation, Pt: pre-training.*

Discussion

In this paper, we demonstrate the first-ever use of Transformers for general 3D tracing and the first application of a Transformer-based 3D segmentation model on general blood vessel data. Our work is far more general than any previous methodology in the area of vascular tracing.

While much attention has been paid to segmentation, domain experts seeking to perform tracing must still spend a significant amount of time manually connecting the resultant subvolumes. Our work is an important first step in automating this process. Eventually, methods like our could save countless of man-hours that would instead be spent performing research. These results will serve as a benchmark for future researchers investigating the potential of Transformers in centerline tracing.

Our segmentation results are also novel in that they achieve better performance than popular convolution-based models like U-Net. While there are few other papers addressing 3D segmentation on vascular data, our method beats state-of-the-art models that have been trained on liver data, specifically (Dice scores of 80.6% vs 76.5%) [7]. The data that our model is trained and evaluated on is more general; the arteries come from various locations in the body. This suggests that our Transformer-based segmentation model is more accurate and more robust than current methods.

Potential Future Steps

Our model is entirely Transformer-based. The creators of UNetFormer found that utilizing a CNN-based decoder achieved better results on small-scale organs, possibly because CNN layers recover localized information that may not be captured in Transformer layers. Modifying our architecture in a similar manner may improve segmentation on our very small sub-volumes. This would still be an improvement over current models in this area that are solely convolution-based.

When performing vessel tracing, the true labels for the orientation of the tangent vector assume that the origin is in the center of the vessel. Using this tangent vector to determine our next step can be problematic when the center of the 3D image is not the same as the center of the vessel. If we run into this issue for several steps, it is possible that our algorithm can completely lose sight of the vessel. In practice, the very small size of the steps helps mitigate this problem, but making the centerline algorithm more robust to non-centered data is an interesting future direction.

Utilizing reinforcement learning methods or deep learning training tricks could enable the algorithm to self-correct when it begins to violate the centered vessel assumption. Instead of estimating the tangent vector from the center of the vessel, the model would estimate the vector from the center of the 3D image that would best capture the next sub-volume along the vessel.

Another potential improvement to predicting the tangent vector is changing the loss function to account for the fact that an exact opposite tangent vector (parallel to ground truth but with opposite direction) is just as correct for predicting purposes. The current implementation regards an opposite vector as wrong. For the classification task a Dice loss functions can be tested, it has proven efficient for ill-balanced datasets (such as medical image segmentation).

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015, pp. 234–241.

[2] Alexey Dosovitskiy et al. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *International Conference on Learning Representations* (2021).

[3] Ali Hatamizadeh et al. “UNetFormer: A Unified Vision Transformer Model and Pre-Training Framework for 3D Medical Image Segmentation”. In: *ArXiv* abs/2204.00631 (2022).

[4] Jelmer M. Wolterink et al. “Coronary artery centerline extraction in cardiac CT angiography using a CNN-based orientation classifier”. In: *Medical Image Analysis* 51 (2019), pp. 46–60. ISSN: 1361-8415.

[5] Nathan M Wilson, Ana K Ortiz, and Allison B Johnson. “The Vascular Model Repository: A Public Resource of Medical Imaging Data and Blood Flow Simulation Results”. In: *J. Med. Devices* 7(4), 040923 (2013).

[6] Fabian Isensee and Klaus H. Maier-Hein. *An attempt at beating the 3D U-Net*. 2019. DOI: 10.48550/ARXIV.1908.02182. URL: https://arxiv.org/abs/1908.02182.

[7] Wei Yu et al. “Liver Vessels Segmentation Based on 3d Residual U-NET”. In: *2019 IEEE International Conference on Image Processing (ICIP)*. 2019, pp. 250–254.