U-Shaped Transformer Architectures for Medical Image Segmentation

Abstract

Medical image segmentation, which involves identifying regions of interest in medical images, is an important tool in the diagnosis, treatment, and monitoring of various ailments. For much of the past decade, convolutional neural networks (CNN's) have been used with great success for segmentation tasks. In particular a U-shaped encoder/decoder CNN architecture called U-net has become popular for medical image segmentation. Although U-net performs very well for many image segmentation tasks, its performance is constrained by one inherent limitation: it struggles to capture long-range dependencies. Inspired by the recent success of the Vision Transformer, a new type of segmentation network has appeared in the literature which combines aspects of U-net and Transformer architectures. In this paper, we analyze three such hybrid U-net/Transformer architectures: UTNet, UCTransNet, and Swin-Unet. Inspired by the Channel Transformer module proposed in UCTransNet, we also propose an extension to UTNet which replaces the network's skip connections with a Convolutional Fusion (ConFuse) module. Our experiments show that the proposed network, which we've called ConFUsenet, can significantly improve the segmentation performance of UTNet, particularly when limited training data is available.

1. Introduction

1.1. Medical Image Segmentation

Broadly speaking, the phrase "medical imaging" describes processes for imaging the interior of the body. Several modalities of medical image exist, many of which can be further divided into several sub-modalities. Magnetic resonance imaging (MRI) is a common medical imaging modality which uses magnets and radio waves to produce cross-sectional images of soft tissue. MRI can be used to diagnose a variety of ailments related to soft tissue, including strokes and inflammation, cancerous tumors, and damage to muscles or tendons. Computed tomography (CT) scanning is another common medical imaging modality, which uses rotating X-rays to create cross-sectional images of bone, organs, and soft tissue. CT can be used to diagnose traumatic brain injuries and tumors, bone fractures, and injuries to organs.

Image segmentation is the task of extracting objects or regions of interest from an image. For example, many lane detection algorithms use image segmentation to extract lane markers from images of roads. Medical image segmentation is the task of identifying regions of interest in medical images, such as MRI and CT scans. In medical image segmentation, the regions of interest are typically anatomical structures, such as organs, or abnormalities, including tumors and ulcers. A variety of approaches to image segmentation exist; in this paper we'll focus on semantic segmentation. Semantic segmentation involves assigning a class label to each pixel in an image, typically using deep neural networks (DNN). For example, performing semantic segmentation on an abdominal CT cross section could involve labeling each pixel as belonging to either "heart", "lung", or "other". Medical image segmentation is a powerful tool that can help medical professionals more effectively diagnose ailments, plan treatments, monitor disease progression, and perform surgeries.

1.2. U-net

Semantic medical image segmentation is typically performed using a particular type of DNN, called a convolutional neural network (CNN). CNN's use layers of small sliding filters (also called kernels) to extract features, such as edges, from an image. A CNN can possess an arbitrary number of these layers, and the size and behavior of each layer can be modified by varying several hyper parameters. Unique arrangements of layers are referred to as "architectures", and different architectures perform better than others at certain tasks.

Introduced in 2015, the U-net architecture (Ronneberger et al., 2015) has become a cornerstone in medical image segmentation, due to its elegant design and its effectiveness at semantic segmentation tasks. Named for its distinct "U" shape, the U-net consists of two portions: a contracting path (also called the encoder) and an expanding path (also called the decoder). The encoder takes in an image and down-samples it to a spatially compact latent representation. The decoder then up-samples the latent representation back

to the dimension of the input image, producing a map which highlights the regions of interest (typically referred to as a segmentation map). The encoder and decoder are bridged by several so-called "skip connections", which simply concatenate the feature map from a particular encoder layer to the feature map in the corresponding decoder layer. This technique helps propagate spatial information that gets lost during down-sampling, and is one of the defining characteristics that has made U-net popular for image segmentation.

1.3. Transformer

The Transformer architecture (Vaswani et al., 2017) was developed in 2017 for natural language processing (NLP) tasks, such as language translation. Like the U-net, the Transformer consists of an encoder and a decoder; however, unlike the U-net, it doesn't use any convolution operations to extract features from the input. Instead, the Transformer breaks the input sequence down into tokens, and uses a mechanism called self attention to compare each token to one another. For each token pair, a so-called "attention score" is computed, which measures the relevance that the two tokens have to each other. These attention scores are used to generate an abstract representation of the input sequence at the output of the encoder. This representation is fed as input to the decoder, which works similarly to the encoder, but produces a sequence in the target language.

Inspired by the Transformer's success in NLP, the Vision Transformer (Dosovitskiy et al., 2020) was developed in 2020, with the goal of bringing the Transformer into the computer vision domain. Like the original Transformer, the Vision Transformer (commonly referred to as "ViT") does not use any convolution operations, instead leveraging self attention to learn relationships between fixed-size patches of an image. The Vision Transformer has demonstrated impressive success in computer vision tasks including image classification and object detection, and inspired many new architectures in doing so.

1.4. Hybrid U-net/Transformer Architectures

Although U-net performs very well for many image segmentation tasks, its performance is constrained by one inherent limitation: a limited receptive field. While U-net, and CNN's more generally, do very well identifying features at a local level, they often struggle to capture long-range dependencies. In other words, U-net tends to have a hard time associating features that are far apart in an image. This is due to the convolutional kernels used in CNN's, which typically map very small regions of the input data onto the output.

Transformers address this shortcoming with their self attention mechanism. By dividing images into small patches and comparing each patch to every other patch, the Transformer is able to effectively learn long-range associations in the image. However, Transformers are notoriously difficult to train, generally requiring larger datasets than U-net, which is able to generalize well with relatively little data. Furthermore, because self attention involves comparing each image patch to every other image patch, the time complexity of Transformers is generally quadratic with respect to the number of patches, making them more time-consuming to train than U-net. Thus, when it comes to medical image segmentation, there exists a resource-performance trade-off between U-net and Transformer-based architectures.

For this reason, much attention has recently been dedicated to blending U-net and the Transformer into a single hybrid U-net/Transformer architecture. Such an architecture would ideally inherit the Transformer's ability to learn global dependencies while maintaining the U-net's simplicity of training, resulting in an architecture that is superior to either of its constituent components. In the next section, we provide an overview and analysis of three recent proposals for hybrid U-net/Transformer architectures.

2. Related Work

In this section, we analyze three different medical image segmentation networks, which draw from U-net and Transformer architectures.

2.1. UTNet

The first network, aptly named UTNet (Gao et al., 2021), attempts to combine the strengths of U-net and the Transformer by incorporating computational primitives from both architectures. In particular, the authors seek to inherit the Transformer's ability to learn long-range dependencies, while maintaining U-net's relative ease of training.

UTNet inherits its distinctive "U" shape from U-net, comprising four stacked encoder layers, four stacked decoder layers, and a bottleneck layer bridging the two sides. Like U-net, each encoder and decoder layer (as well as the bottleneck layer) consist of a so-called "Residual Basic Block", composed of convolution and batch normalization operations, with ReLU as an activation function. Unlike U-net, UTNet includes an additional Transformer Block in three of the four encoder and decoder layers, as well as the bottleneck layer. Only the top-most encoder and decoder layers (the ones which receive the input image and output the segmentation map, respectively) are purely convolutional, containing an additional Residual Basic Block instead of a Transformer block.

At first glance, these Transformer blocks may resemble the residual basic block, consisting of two batch normalization operations, a ReLU activation, and a convolution. However, they also contain within them a modified multi-headed self

attention module. The authors show that by incorporating self attention into the layers of the network, it is better able to learn long-range dependencies than U-net. Figure 1 comes from the UTNet paper (Gao et al., 2021), and depicts the architecture of UTNet (figure 1(a)), the composition of the Residual Basic Block (figure 1(b)) and the composition of the Transformer Block (figure 1(c)).

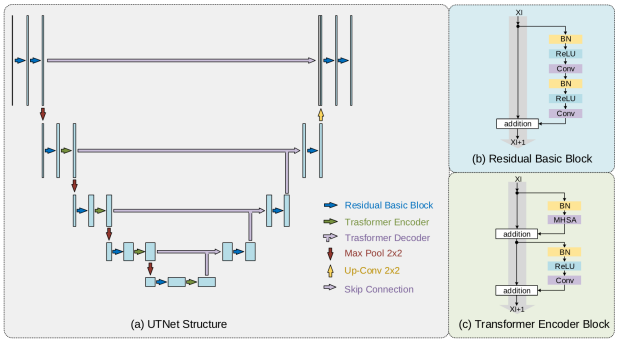

Figure 1. (a) The architecture of UTNet. (b) The composition of the Residual Basic Block. (c) The composition of the Transformer Block. This image comes from the UTNet paper (Gao et al., 2021).

As mentioned previously, the quadratic time and space complexity of the self attention operation can complicate the training of Transformer-based networks. To minimize the effect that the Transformer Blocks have on the training of UTNet, the authors propose a new formulation for self attention. Traditional self attention projects the input $X \in \mathbb{R}^{\mathrm{C}\times\mathrm{H}\times\mathrm{W}}$ (where C is the number of channels, H is number of vertical patches, and W is the number of horizontal patches) onto query, key, and value embeddings $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{\mathrm{d}\times\mathrm{H}\times\mathrm{W}}$. These embeddings are then flattened and used to compute the attention as follows:

$$Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}})\boldsymbol{V} \quad (1)$$

After flattening the query, key and value, the resulting embeddings are of size $n \times d$, where $n = HW$. Thus, computing the dot product between two embeddings has a quadratic time complexity with respect to the number of patches in the image. While in certain cases it may be sufficient to simply reduce the number of patches (thereby increasing the size of each patch), medical images are highly fine-grained structures and increasing the patch size may lead to a loss of spatial information. Therefore, a new approach is needed to reduce the time complexity of self attention without increasing the patch size.

To address this issue, the authors propose a modified efficient self attention mechanism. Instead of directly using the key and value embeddings $\boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{\mathrm{d}\times\mathrm{H}\times\mathrm{W}}$ in the attention computation, the authors first use down-sampling operations (e.g., pooling, strided convolutions) to project K and V into low-dimensional embeddings $\bar{\boldsymbol{K}}, \bar{\boldsymbol{V}} \in \mathbb{R}^{\mathrm{d}\times\mathrm{h}\times\mathrm{w}}$, where $h, w \ll H, W$. By doing so, the time complexity is reduced by a factor of $(H-h)(W-w)$.

2.2. UCTransNet

Like UTNet, the second network, titled UCTransNet (Wang et al., 2022) tries to combine the strengths of U-net and the Transformer by incorporating computational primitives from both architectures, albeit in a slightly different fashion. The authors posit that U-net's main limitation is not the encoder, but rather its skip connections, which naively concatenate the feature maps of same-scale encoder and decoder layers. By analyzing U-net's performance on the MoNuSeg and GlaS datasets, the authors found that in certain situations, specific skip connections do not improve, and may even degrade, the performance of the network. Furthermore, same-scale encoder and decoder layers may not be semantically similar, creating a "semantic gap" between encoder and decoder. Thus, to improve the performance of U-net, the authors propose a Transformer-based alternative to traditional skip connections, which they call the Channel Transformer (CTrans) module. Simply put, the CTrans module aims to bridge the semantic gap between encoder and decoder by first fusing together each of the encoder outputs (i.e., "feature maps"), and then selectively applying this information to the appropriate decoder layer.

The CCT module is composed of two distinct sub-modules: the Channel-wise Cross Fusion Transformer (CCT) and the Channel-wise Cross Attention (CCA). The purpose of the CCT is to fuse together each of the four feature maps that are output by the encoder layers. It does this by passing the feature maps through a "Multi-head Channel-wise Cross-attention" mechanism and then a Multi-Layer Perceptron. Unlike traditional self-attention mechanisms which apply attention in the image dimension, the cross-attention mechanism applies attention in the channel dimension.

After being passed through the CCT module, the transformed encoder feature maps and the corresponding decoder feature maps are passed into the CCA as inputs. Rather than naively concatenating, the CCA uses global average pooling and linear transformations to fuse the i^{th} CCT output to the i^{th} decoder feature map, resulting in a semantically stronger representation. Figure 2, which comes from the UCTransNet paper (Wang et al., 2022), depicts the architecture of UCTransnet, as well as the compositions of CCT

and CCA.

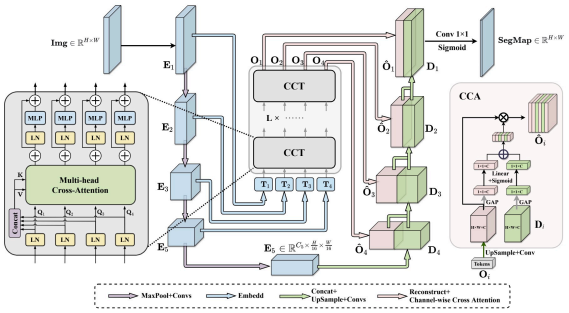

Figure 2. (center) The architecture of UCTransNet. (left) The composition of the Channel-wise Cross Fusion Transformer (CCT) module. (right) The composition of the Channel-wise Cross Attention (CCA) module. This image comes from the UCTransNet paper (Wang et al., 2022).

2.3. Swin-Unet

The final network that we analyze in this section is titled Swin-Unet (Cao et al., 2022). Like UTNet and UCTransNet, Swin-Unet is a U-net based architecture; however, it does not incorporate convolution operations as UTNet and UCTransNet do. Instead, Swin-Unet relies exclusively on so-called "Swin Transformer Blocks" in its computation, modeling only its encoder-decoder structure after U-net.

Like U-net, Swin-Unet consists of a contracting encoder path and an expanding decoder path, bridged by a bottleneck layer and several skip connections. Unlike U-net, Swin-Unet does not use any convolution operations, instead relying solely on Swin Transformer Blocks. The Swin Transformer Block consists of two back-to-back sub-blocks, each containing layer normalization, multi-headed self attention, and a multi-layer perceptron. The multi-headed self attention mechanism in the first sub-block is window-based, meaning that the input is divided up into non-overlapping windows (i.e., collections of patches) and self attention is computed per-window, without regard to other windows. The multi-headed self attention mechanism in the second sub-block is shifted-window-based, meaning that it operates the same as window-based, but shifts the original windows slightly before computing the self attention. Figure 3 comes from the Swin-Unet paper (Cao et al., 2022), and depicts the composition of the Swin Transformer Block.

Between each Swin Transformer Block in the encoder is a so-called "Patch Merging" layer, which fuses adjacent patches, increasing their spatial and channel dimensions. Inversely, between each Swin Transformer Block in the decoder is a "Patch Expanding" layer, which breaks apart patches, decreasing their spatial and channel dimensions.

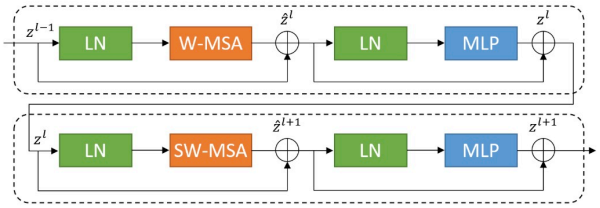

Figure 3. Composition of the Swin Transformer Block. This image comes from the Swin-Unet paper (Cao et al., 2022).

Figure 4 comes from the Swin-Unet paper (Cao et al., 2022), and shows the Swin-Unet architecture.

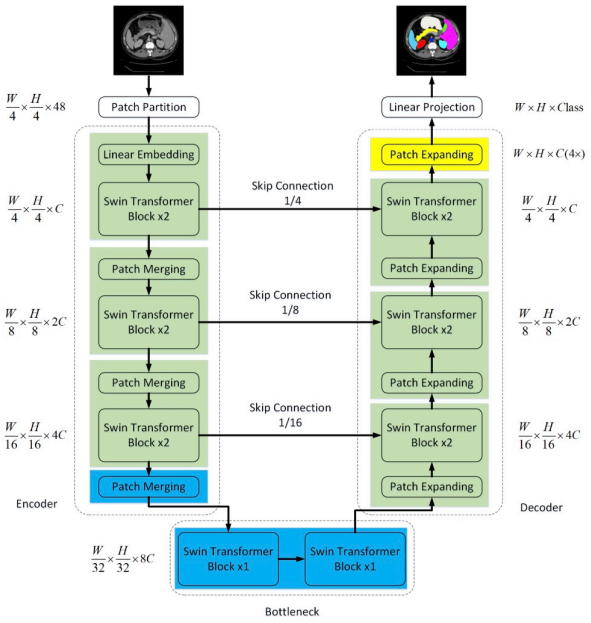

Figure 4. Architecture of Swin-Unet. This image comes from the Swin-Unet paper (Cao et al., 2022).

3. Problem Definition

The three aforementioned papers, as well as the experiment described in the next section, are concerned with the problem of 2D medical image segmentation. Informally, medical image segmentation involves partitioning a 2D medical image into regions corresponding to anatomical structures or abnormalities. Mathematically, it can be described as:

$$S = f(I) \tag{2}$$

where $I \in \mathbb{R}^{m \times n}$ is a 2D medical image and $S \in \mathbb{R}^{m \times n}$ is

the partitioned image (i.e., "segmentation map"). The goal is to learn the mapping function f that predicts S given I.

To achieve such a mapping function, we aim to minimize a suitable objective function $\mathcal{L}$, which quantifies the dissimilarity between S and the ground truth segmentation map $\hat{S}$. In other words, the objective function should penalize deviations and encourage similarity between S and $\hat{S}$.

Two commonly-used objective functions in image segmentation are Dice Loss, defined as:

$$\mathcal{L}_{Dice} = 1 - \frac{2\sum_{i=1}^{N}\sum_{j=1}^{C} S_{i,j}\hat{S}_{i,j}}{\sum_{i=1}^{N}\sum_{j=1}^{C}(S_{i,j} + \hat{S}_{i,j})} \tag{3}$$

and Cross-Entropy Loss, defined as:

$$\mathcal{L}_{CE} = -\sum_{j=1}^{C} \hat{S}_j log(S_j) \tag{4}$$

where N is the number of pixels in the image and C is the number of classes. In our experiments, we use Cross-Entropy Loss to train our network and Dice Loss as an evaluation metric.

4. Methodology

Inspired by UCTransNet, here we propose an extension to UTNet which uses a Convolutional Fusion (ConFuse) module to help bridge the semantic gap between encoder and decoder. We call the proposed architecture ConFUse-net, and in the next section we show that it greatly outperforms UTNet with limited training.

Despite improving U-net's ability to learn long-range dependencies, UTNet does little to address the limitations of traditional skip connections. While the network does incorporate additional transformations between the encoder and decoder, it still effectively concatenates feature maps from same-scale encoder and decoder layers. If these same-scale feature maps are semantically dissimilar, this approach could degrade the performance of the network, as shown by UCTransNet. To address this shortcoming, we introduce the ConFuse module, which combines feature maps from every scale to produce globally-aware feature maps for the decoder.

Like the CTrans module in UCTransNet, the ConFuse module takes each of the four encoder feature maps as input and combines their information to create four globally-aware feature maps. It does so by first concatenating each feature map in the channel dimension, using bi-linear interpolation to match their spatial dimensions. It then passes the unified feature map through a single Residual Basic Block (the same type used in the rest of the network), preserving its dimensions in the process. The transformed feature map is then divided into four output maps along the channel dimension, such that each output map $FM_{out,i}$ has the same number of channels as the corresponding input map $FM_{in,i}$. Bi-linear interpolation is then used to match the height and width of each output map $FM_{out,i}$ to the corresponding input map $FM_{in,i}$. At this point, the dimensions of each output map $FM_{out,i}$ should exactly match one input map $FM_{in,i}$. The output maps are output from the ConFuse module, where they are concatenated to the corresponding decoder feature map. Figures 5 and 6 show the composition of the ConFuse module and the ConFUse-net architecture, respectively.

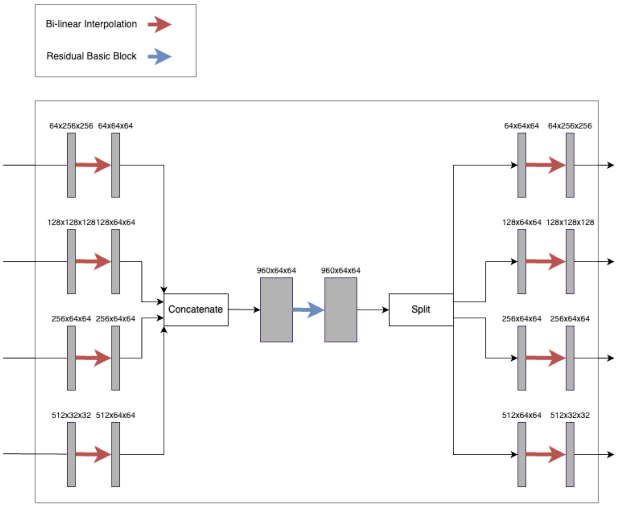

Figure 5. Composition of the ConFuse module.

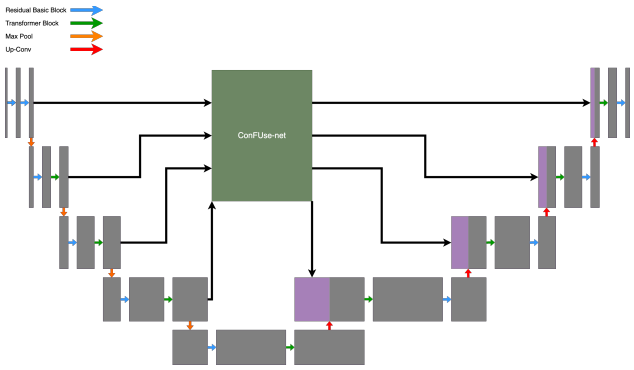

Figure 6. Architecture of ConFUse-net.

5. Experimental

In this section we describe how we trained and evaluated ConFUse-net, beginning with our experimental setup before

getting into our results.

5.1. Experimental Setup

Dataset To train and evaluate our network, we use the popular Synapse dataset for multi-organ segmentation. The dataset consists of 50 abdominal CT scans (30 for training, 20 for testing), with 13 different classes (spleen, right kidney, left kidney, gallbladder, esophagus, liver, stomach, aorta, inferior vena cava, portal vein and splenic vein, pancreas, right adrenal gland, and left adrenal gland). Because this dataset is used for competitions, it does not include labels in the test set. To address this issue, we split the designated training set of 30 scans into a 20-scan impromptu training set and a 10-scan impromptu testing set. Each scan contains between 85 and 198 512x512 cross sectional images.

Data Pre-processing Due to memory and compute constraints, we discard the first 60 percent of the images in each scan, as these were mostly blank (i.e., didn't contain any segmented areas). We also resize each image from 512x512 to 256x256 and use min-max normalization to scale the pixel values to the range [0, 1] before passing it to the network.

Model Structure To ensure fair comparison, we re-implement UTNet in addition to ConFUse-net. The encoder, decoder, and bottleneck layer of our ConFUse-net are identical to our UTNet. The only difference is that ConFUse-net replaces UTNet's skip connections with a ConFuse module.

Hyper-parameters and Metrics We train each of our our models on a single Tesla T4 GPU with 15GB of RAM. We train the models from scratch for 30 epochs, using batches of eight cross-sectional images and Adam with a learning rate of 1e-4 as our optimizer. Cross-entropy loss is used as our loss function during training and we evaluate the models using Dice loss.

5.2. Results

The average Dice loss and number of parameters of U-net, UTNet, and ConFUse-net are reported in table 1. As can be seen, U-net significantly outperforms both UTNet and ConFUse-net with respect to Dice loss. We believe that this is due to the relatively small dataset and short training time used. Transformers are notoriously data-hungry, requiring large datasets and long training time. For example, in the UTNet paper, the authors trained their model for 150 epochs. Conversely, U-net is generally able to perform relatively well with less training. Due to memory and compute constraints, we were only able to train our models for 30 epochs with 20 scans. This would explain why the hybrid-transformer models performed worse than U-net.

In our experiment, ConFUse-net outperforms UTNet by a fairly wide margin of 0.0546. This result suggests that our ConFuse module is effective in bridging the semantic gap between encoder and decoder, helping the model more effectively learn long-range dependencies. This claim is empirically supported by the segmentation maps generated by ConFUse-net and UTNet. One example of a segmentation map generated by ConFUse-net is shown in figure 7.

Table 1. Performance of ConFUse-net compared to U-net and UTNet.

MODEL	DICE LOSS	PARAMETERS
U-NET	0.1214	31092142
UTNET	0.4023	39115754
CONFUSE-NET	0.3477	55708394

Figure 7. (left) CT image. (middle) Ground-truth segmentation map. (right) Segmentation map generated by ConFUse-net.

6. Conclusion and Future Direction

In this paper, we have analyzed several approaches to medical image segmentation that draw from U-net and Transformer architectures. UTNet incorporates Transformer blocks into the encoder and decoder layers of U-net. UCTransNet replaces U-net's skip connections with a Channel Transformer module that fuses encoder and decoder feature maps. Swin-Unet replaces the convolutional blocks in U-net with Swin Transformer blocks.

Additionally, we have proposed ConFUse-net, a novel hybrid U-net/Transformer architecture for medical image segmentation that draws from UTNet and UCTransNet. ConFUse-net uses a convolutional fusion (ConFuse) module to fuse encoder feature maps, resulting in better cross-network feature propagation. Our experimental results show that ConFUse-net outperforms UTNet with limited training. More work will need to be done to compare ConFUse-net and UTNet when both networks are sufficiently trained. In the future we would like to explore how the addition of the ConFuse module could improve other U-shaped networks, for example U-net and Swin-Unet.

References

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., and Wang, M. Swin-unet: Unet-like pure transformer for medical image segmentation. In *European conference on computer vision*, pp. 205–218. Springer, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Gao, Y., Zhou, M., and Metaxas, D. N. Utnet: a hybrid transformer architecture for medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part III 24*, pp. 61–71. Springer, 2021.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234–241. Springer, 2015.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wang, H., Cao, P., Wang, J., and Zaiane, O. R. Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In *Proceedings of the AAAI conference on artificial intelligence*, pp. 2441–2449, 2022.