Applying TransUNet to Renal Tumor Segmentation

Matthew Nguyen, Neel Karsanbhai

Paper ID 2

1 Introduction

1.1 Motivation

The analysis of medical images (MRIs, CT scans, PET images) for cancer diagnosis, and cancer analysis on patients is currently a high level analysis done by human radiologists [1]. This is inefficient from a cost perspective, and data extraction perspective. Radiomics is a practice based on the idea that medical imaging in general can provide more insights into a patient than just visual information. Medical imaging can also provide doctors and other healthcare providers with context. Radiomics can provide correlations between genetics and tissue type, and confirmation of the findings of a radiologist among may other uses [1]. This is a new and promising field of medical imaging that can help tumor boards create more effective and cost efficient treatment plans for their patients. Analysis of this type if leading the way due to support from the National Cancer Institute, and the Quantitative Imaging Network. After collecting image data, and identifying data that is useful, tumors must be identified, and segmented. Automating image segmentation will aid radiologists and largely tumor boards in the process of radiomics, while lowering the costs of cancer care for patients. The current annual cost of cancer treatment in the United States is $109,727 [2]

1.2 Why Current Solutions are Inadequate

The TransUNet architecture proposed by J. Chen et al. showed the efficacy of the architecture on multi-organ abdominal CT scans as well as MR images of the heart. TransUNet achieved superior performance over state of the art architectures (V-Net, DARR, U-Net, and AttnUNet) based on the Dice similarity coefficient [3]. This architecture has been proven to work on multi-organ segmentation, which is within the larger domain of medical image segmentation, however, it has not been proven on the renal cancer segmentation task. Global context and spatial information should be helpful in the renal cancer segmentation task, because both abilities should be able to work together to determine the location of a tumor, as well as the change in tissue type from renal tissue to tumor tissue.

Berbera et al. proposed a network that used three sequential modules and spatial transformers [4]. The network was able to reduce training time while increasing the Dice score for renal tumor segmentation from 85.52% to 87.12%; however, the model was only valid for a small set of data [5].

1.3 Proposed Idea

Our proposed network will perform renal tumor segmentation on 2D CT scan images using U-Net with visual transformers (ViT), an adaptation of the transformer, proposed by Dosovitskiy et al. [6], that applies global attention to 16x16 patches of an image. The benefit of using this type of transformer is that it is the the best at incorporating global context in the image features without compromising the computational efficiency [5]. Our proposed network will trained and tested using the KiTS19 dataset [7].

2 Related Work

2.1 Combining UNet with ViT for Renal Cancer Segmentation

To the best of out knowledge, this is the first application of ViT and UNet to renal cancer segmentation. This architecture, however, has been shown to outperform the state of the art on multi-organ segmentation. TransUNet itself has performed well on multiple medical applications such as cardiac segmentation, organ segmentation, and polyp segmentation on multiple forms of imaging [3]. The purpose of TransUNet is to overcome the inability of Convolutional Neural Networks (CNN) to model long range relationships. CNNs more specifically have difficulties identifying textures, shapes, and sizes of objects that are highly variable between patients [3]. TransUNet is well suited to medical applications, because it requires less training due to the fact that it uses UNet. UNet is able to do this by up sampling images [8]. By applying the TransUNet architecture to the renal tumor segmentation task, we hope to identify how hyper parameters of the network should change to best fit the task, if changes need to be made at all. This experiment can possibly identify characteristics of the data set that human radiologists may have a difficult time identifying due to bias, inconspicuousness, etc.

2.2 UNet with Spatial Transformer Network for Renal Tumor Segmentation

While the network proposed by Berbera et al. was trained and tested on a dataset of renal CT scans, their dataset is composed of CT scans from pediatric patients. Additionally Berbera et al. used a Spatial Transformer Network (SPN) proposed by Jaderberg et al. [9] while we plan on using visual transformers. SPNs transform image inputs to decrease the noise in an image, locate the area of interest in an image, and increase the invariance of a network [9]. ViT, however applies attention mechanisms to patches of the input image. The data set we will be using contains abdominal CT scans from adult patients. Berbera et al. uses a normalization layer in addition to a localization layer, and segmentation layer to account for the high variability in kidney size of pediatric patients. Adult patients most likely do not have high inter patient variability with regard to kidney size reducing the need for a normalization layer. Thus our architecture will not include a normalization layer.

3 Methods

3.1 Architecture

Our architecture will exactly replicate the TransUNet architecture proposed by Chen et al. [3].

Input. Our model starts off with an input layer that takes in images with a resolution of 224 × 224. While results from Chen et al. showed that increasing this input resolution to 512 × 512 would increase performance, it comes at the expense of a much larger computational cost [3].

CNN. The input images are passed into a CNN that is used as a feature extractor in order to create a feature map. A CNN is used because without it, low-level details would be lost due to the shape of encoded features being much smaller than the original image. Including a CNN, allows us to use the intermediate high resolution feature maps generated by the CNN in the decoding path. Along with this, Chen et al. found that using a CNN and Transformer to encode features performed better than just using a Transformer as the encoder [3].

Patch Embedding. The feature maps from the CNN are tokenized by reshaping them into a sequence of 2D patches $\{x_p^i \in \mathbb{R}^{P^2 \cdot C} | i = 1, ..., N\}$, of size $P \times P$, where $N = \frac{HW}{P^2}$ is the number of patches. For our model, we will be using a patch size of 16 × 16. Using a trainable linear projection, the vectorized patches x_p are mapped into a latent D-dimensional embedding space. The equation that is used to learn the specific position embeddings that are to the patch embeddings is:

$$z_0 = [x_p^1 E; x_p^1 E; ...; x_p^N E] + E_{pos},$$

where the patch embedding projection is $E \in \mathbb{R}^{(P^2 \cdot C) \times D}$ and the position embedding is denoted by $E_{pos} \in \mathbb{R}^{N \times D}$.

Visual Transformer Encoder. The embedding sequence is passed through an encoder which 12 visual transformer layers. Each of the layers contain Multi-head Self-Attention (MSA) and Multi-Layer Perception (MLP) blocks, and the output of each layer ℓ can be written as:

$$z'_\ell = MSA(LN(z_{\ell-1})) + z_{\ell-1},$$

$$z_\ell = MLP(LN(z'_\ell)) + z'_\ell,$$

where z_L is the encoded image representation and $LN(\cdot)$ is the layer normalization operator.

Cascaded Upsampler. The sequence of hidden feature $z_L \in \mathbb{R}^{\frac{HW}{P^2} \times D}$ from the encoder is reshaped to the shape of $\frac{H}{P} \times \frac{W}{P} \times D$, and passed through a 1×1 convolution to reduce the channel size of the reshaped feature to the number of classes. After reshaping, it is passed through four cascaded upsampling blocks, consisting of a $2\times$ upsampling operator, a 3×3 convolution layer, and a ReLU layer successively, in order to get from resolution $\frac{H}{P} \times \frac{W}{P}$ to $H \times W$ for predicting the final segmentation result.

The architecture used with the CNN, visual transformers, and cascaded upsampler allows using skip-connections for feature aggregation at different resolution levels.

3.2 Training Overview

We trained our model using the KiTS19 dataset. We had originally planned on using 146 samples (70%) for training, but we were only able to use 73 samples due to time and RAM limits. 51 of these samples were used for the actual training of the model while the other 22 were used for validation. Each sample contained 610 images, with a resolution of 512×512, which represented a layer of a CT scan of a subject's torso. In order to train our model, we split these images up into separate cases, resized them to a resolution of 224×224, and converted them to numpy arrays. As mentioned in the architecture section, we used be using 16×16 for our patch size. The model was trained using a SGD optimizer, a momentum of 0.9, weight decay of 1e-4, and a batch size of 24. We originally tried using a learning rate of 0.01, but this resulted in a NaN loss when training. We kept lowering the learning rate until this problem disappeared, which occurred at a learning rate of 1e-5. Also, while we originally planned on training our model over 150 epochs, we were only able to train for 2 due to time constraints. Our model was trained on a device with a Ryzen 7 2700X 3.7 GHz Eight-Core AM4 Processor, a NVIDIA GeForce GTX 1060 graphics card, and 16GB of RAM. We were not able use GPU as a hardware accelerator while training due to a lack of RAM.

4 Experimental Design

4.1 Main Purpose

The goal of this experiment is to determine how a successful medical image segmentation neural network architecture will perform in the new medical image segmentation task of renal tumor segmentation. Furthermore, our experiment will provide us with insights on how to improve the TransUNet architecture for better performance on KiTS19, and a deeper understanding of the characteristics of our data set [7].

4.2 Design

We had originally planned on diving the KiTS19 dataset into the same splits as what was used on TransUNet in multi-organ segmentation [3]; however, due to time constraints, we were only able to use half of the samples for each split. This ended up being 51 samples for training, 22 samples for validation, and 5 samples for testing. Using the Captum library, we will extract the information that our neural net learns. Based on these insights, we will determine what the neural network has learned. By understanding what patterns in the data set the neural network was able to learn we can improve our understanding of the data set, and more importantly information about renal cancer and abdominal CT scans that were not apparent from human observation.

4.3 Evaluation Metrics

Integrated Gradients will show how invariant the network is to our dataset as well as how much certain features influence the prediction of the neural network. This will tell us the most relevant features of renal CT scans that predict cancer at a pixel level. The metric may also reveal a new way of reasoning about tumor segmentation that most human radiologists have not discovered. [10]. Furthermore Sundararajan et al. showed that integrated gradients by way of feature importance provided retina specialists with a level of trust in the neural network's predictions that they would not have had without the metric.

The second metric we will be using is the Dice Similarity Coefficient (DSC). The DSC measures the fraction number of pixels in the same location across two images that are exactly the same relative to the number of pixels in both images. The use of this metric is to measure how close our neural networks predictions are to the segmentation produced by a human radiologist, which is currently the gold standard in radiology [1].

5 Experimental Results

5.1 Dice Similarity Coefficient

For the five samples that we tested on from the KiTS19 dataset, our model scored an average dice coefficient of 0.0172. We believe that the main reason behind this low dice coefficient is due to the model not having enough time to train. As mentioned in the previous sections, we had to reduce the number of samples that was used to train the model by half, and we were only able to train the model over 2 epochs instead of 150.

When looking at the predictions from our model, one of the things that we noticed is that our model considered 99% of the pixels to be part of the segmentation, assuming a pixel is considered part of the segmentation if the RGB values are not equal to each other. One of the possible reasons behind this could be the way segmentation is represented in our dataset versus the one originally used by TransUNet. In the TransUNet dataset, the segmentated organs are represented with a solid color; however, in the KiTS19 dataset, the segmentation color is more translucent resulting in pixels with RGB values like [199 122 122] instead of [255 0 0]. This slight difference in outputs may be confusing our model a little bit, especially with the reduced training time.

Another thing that we noticed when looking at the prediction data was that many of the RGB values were over 255. One of the possible reasons why this may be happening is because the model may have been expecting RGB values of 0-1 instead of the 1-255 that we inputted into it.

5.2 Integrated Gradients

Looking at the images below, Figure 1 shows the integrated gradients of one of the layers when the model had only been trained on a few scan layers. As seen in the figure, the model still considers almost all the parts of the image to be important, even the ones outside of the torso. Figure 2 and 3 show the integrated gradients for two different scan layers. As seen in these figures, the model now only really considers things within the torso as important. Based on the integrated gradients from our model, it seems like it is considering any pixel that is not black important. While this is an improvement from the model in figure 1, ideally, we want the model to focus around the area of the image where the kidneys are supposed to be. Essentially, the model should learn to be more efficient by ignoring all areas of a scan except to the areas where kidney's are assuming the scan contains kidneys. If the input scan does not contain kidneys, the model should output the original scan. We can see that the model started to ignore larger areas of a torso when the input image did not contain kidneys compared to images that did contain kidneys. We believe that the reason why our model was not doing that is because it did not have enough time to train.

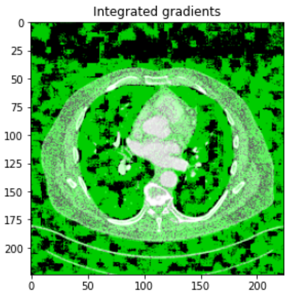

Fig. 1. Early Training

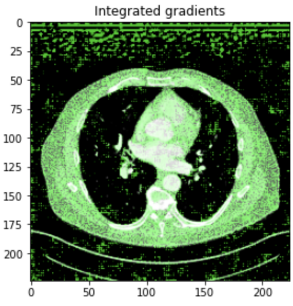

Fig. 2. After Training

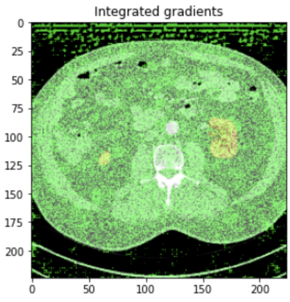

Fig. 3. After With Kidney

5.3 Next Steps

The question of whether or not the TransUNet architecture proposed by Chen et al. is useful for the renal cancer segmentation task is still left unanswered after our research. In order to get a definitive answer on this, one of the things that needs to be done is to train the model using all of the data from the KiTS19 dataset and over the 150 epochs that we originally planned. Once the original TransUNet experiment has been applied to the full KiTS19 dataset, further investigation into the interpretability of TransUNet should be made. When dealing in high risk areas such as medicine, especially oncology where financial and patient health are strained, interpretability of models for doctors is of utmost importance. To this end, it is worth using more metrics and methods that will help researchers, and eventually doctor's, understand why our model is making the predictions it is making. In order for doctors to make clear diagnoses, they must not use tools that are black boxes. More research also has to be done using other datasets in order to validate the findings of the model trained using the KiTS19 datset.

6 Conclusion

Through this experiment, we found that TransUNet has the potential to perform well on renal tumor segmentation. This is based on the output of Integrated Gradients. The Dice Coefficient on the other hand did not provide any insight as to how the model may perform with more data, and time to train. Having the ability to interpret the model through Integrated Gradients is a sign that TransUNet can be useful in the process of Radiomics. With powerful a technology such as neural networks being applied to high stakes areas such as medicine, it is necessary to consider the ethical implications of our experiment.

The most important ethical consideration of this work is how much trust society as a whole should put into algorithms, and if we should limit the amount of trust we put into algorithms. There are multiple stakeholders in this issue. Patients, doctors, and hospital administrators are among the most important stakeholders. Considering the Hippocratic Oath that Doctors swear by, Doctors and patients most likely would come to the same conclusion that an algorithm such as TransUNet cannot harm any patients, and that an algorithm will always come second to a doctor. Hospital administrators on the other hand have aspects of their jobs that may conflict with the goals of doctors and patients. Among these aspects is maintaining profitability, and efficiency of a hospital. Profitability and efficiency are problems that can be solved by deep learning. In the US, it is currently largely up to the government to force the medical industry to use neural nets in a way that serves the interests of American society. Doctors tend to be hesitant to adopt new technologies, because of strict rules that the healthcare industry must follow [11]. This will hopefully allow society to gain more understanding through real world experience of the ethical implications of deep learning, before we apply deep learning to the extent that we have begun to in other industries.

References

1. Gillies, R.J., Kinahan, P.E., Hricak, H.: Radiomics: Images are more than pictures, they are data. Radiology **278**(2) (2016) 563–577 PMID: 26579733.
2. NIH: Financial burden of cancer care. (2021)
3. Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y.: Transunet: Transformers make strong encoders for medical image segmentation. (2021)
4. Barbera, G.L., Gori, P., Boussaid, H., Belucci, B., Delmonte, A., Goulin, J., Sarnacki, S., Rouet, L., Bloch, I.: Automatic size and pose homogenization with spatial transformer network to improve and accelerate pediatric segmentation. (2021) 1773–1776
5. Parvaiz, A., Khalid, M.A., Zafar, R., Ameer, H., Ali, M., Fraz, M.M.: Vision transformers in medical computer vision – a contemplative retrospection (2022)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. CoRR **abs/2010.11929** (2020)
7. Heller, N., Sathianathen, N., Kalapara, A., Walczak, E., Moore, K., Kaluzniak, H., Rosenberg, J., Blake, P., Rengel, Z., Oestreich, M., et al.: The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. arXiv preprint arXiv:1904.00445 (2019)
8. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015)
9. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. CoRR **abs/1506.02025** (2015)
10. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks (2017)
11. Topol, E.J.: Deep medicine: how artificial intelligence can make healthcare human again. (2019)