Deep Learning for Head and Neck CT Image Segmentation

Pengyun Ding
New York University
pd1341@nyu.edu

Tianyu Zhang
New York University
tz904@nyu.edu

Abstract

Radiotherapy is a common treatment option for head and neck cancer. In the treatment, an important step is delineating organs-at-risk (OARs) based on head and neck CT images. However, manual delineation is very time-consuming because each slice of CT images needs to be individually examined and each CT scan contains hundreds of slices. Automating OARs segmentation can increase the efficiency and standardization of radiation treatment planning for head and neck cancers. In this paper we propose end-to-end deep learning neutral networks for OARs segmentation automation. We implement Res3DUnet and UNEt TRansformers (UNETR) for model architecture and construct dynamic weighted loss as well as multiple loss functions. Our ensemble model's performance is competitive with state-of-the-art methods.

1 Introduction

Radiotherapy is one of the cornerstone therapies for head and neck cancers, and it works by delivering targeted high dose radiation to patients' tumors to shrink and kill cancer cells. However, this treatment can be deleterious to the healthy tissues surrounding the tumor and misguided radiation can cause treatment related side effects, which may increase the risk of treatment failure. Therefore, avoiding the surrounding organs at risk (OARs) is a critical part in radiation therapy. In current clinical practice, OARs are outlined manually on CT scans of the patient's head and neck, which is very time consuming and can be inaccurate. Automation of CT image segmentation can greatly alleviate doctors' manual workload if the performance is accurate enough with a reasonable amount of required time.

Deep Learning is a promising method to automate segmentation because neural networks are able to extract features of different organs and perform segmentation tasks. There have been several public competitions based on this task and many relevant data sets are available. Our project is a continuation of last year's project[1], which leveraged 2D U-Net for the segmentation task. In our project, we use data from the MICCAI imaging challenge in 2015[3]. We implement Res3DUnet and UNETR to build deep learning models trained end-to-end. The models take a 3D image as input and return a segmentation mask for all of the OARs. Our main contributions are leveraging 3D U-Net and its refined versions, constructing a 3-stage dynamically weighted loss function and implementing data augmentation techniques. As for the results, our model performance is competitive with the state-of-art model performance in terms of Dice Score. The ensemble model's performance beats the champion of the MICCAI competition. Comparing with last year's project, our model has improvement in most of the nine organs (Optic Nerve, Submandibular, etc) in terms of Dice Score.

The remainder of this report is structured as follows:

- We give an overview of related research work that inspired our project in Section 2.

- In Section 3, Section 4.1, and Section 4.2 we formalize our task, describe the data set that is used in this project and introduce our approach, ranging from data processing to model training.
- We present our final results and compare them with the state-of-the-art in Section 4.3 and Section 4.4.

2 Related Work

Head and Neck CT image segmentation has many proposed approaches which can be roughly split into atlas-based methods and learning-based methods. Before the success in using Deep Learning for medical segmentation, atlas-based methods were the most popular approach. Basically speaking, new images are aligned to a fixed set of examples and some transformation will be applied to the input so that it is compatible with the atlas. A main drawback of this method is that it has trouble with anatomical variations because the atlas is fixed. Learning-based methods became the mainstream approach after U-Net was introduced. The main idea of U-Net is applying a contracting path and an expansive path, which form a U-shaped network. The architecture is usually combined with additional data augmentation techniques.

The MICCAI competition is a very popular competition in this field held every few years, aiming to provide a principled way of evaluating segmentation algorithms. The latest one in 2015 was the MICCAI Head and Neck Auto Segmentation Challenge[3] and our project uses the provided data set as well.

In 2018, a learning-based algorithm called AnatomyNet[7] was proposed, which implemented a 3D U-Net combined with squeeze-and-excitation residual blocks and formed a new loss function using a combination of Dice loss and Focal loss.The model has state-of-the-art performance and gives our project inspiration for multiple loss functions and model architecture.

In last year's project[1], a residual 2D U-Net is implemented with a weighted mix-up training strategy, and the model performance is close to the state-of-the art models. Continuing from last year's project, we expand 2D U-Net to 3D U-Net with residual blocks.

In 2021, a novel architecture called UNEt TRansformers (UNETR)[5] was introduced. This model architecture is inspired by the success of Transformers for long-range sequence learning tasks in the field of Natural Language Processing (NLP). The model utilizes a Transformer as the encoder and still follows the "U-shaped" network architecture of U-Net. The model's performance is competitive to state-of-the-art. We are also inspired to implement the architecture as a second model for our specific task, Head and Neck CT image segmentation.

3 Problem Definition and Algorithm

3.1 Task

In a typical image segmentation task, a binary mask with multiple regions of interest segmented out is generated based on an input image. In our case, we have 3-dimensional CT images of the head and neck region as our input (data set description is in section 4.1), and our task is to generate segmentation masks that indicate the presence/absence of 9 different OARs.

3.2 Algorithm

The overall structure of the proposed training framework is illustrated in Figure1. It can be summarized as follows:

1. Given an CT scan volume, crop it into the uniform size (96,128,224) according to the region of interest. Apply random affine and random elastic deformation to the input volume for data augmentation.

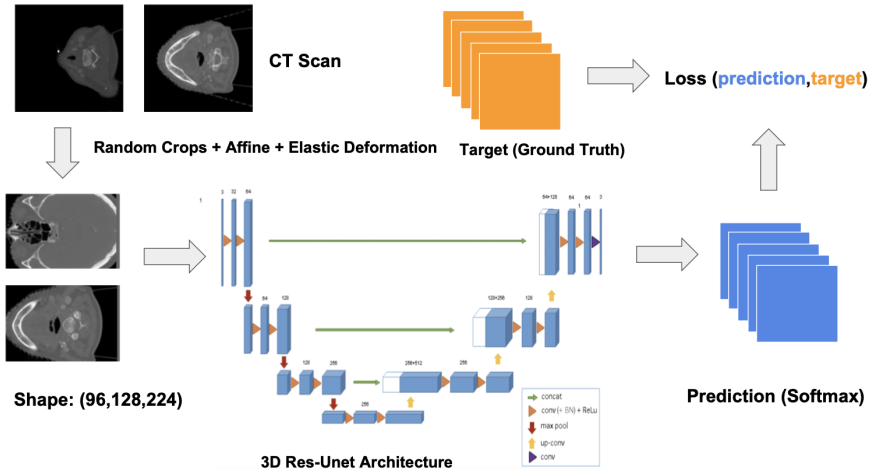


Figure 1: Overview of Our Training Framework

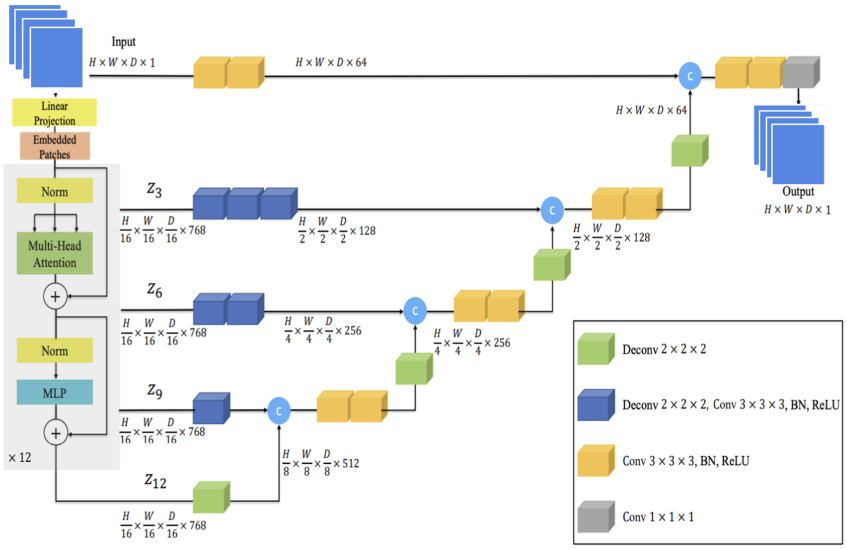


Figure 2: UNEt TRansformers (UNETR) Architecture

2. Generate prediction of the preprocessed volume with our model. We train Res3DUnet and UNEt TRansformer (UNETR) models separately. The architecture of UNETR is shown in Figure 2.

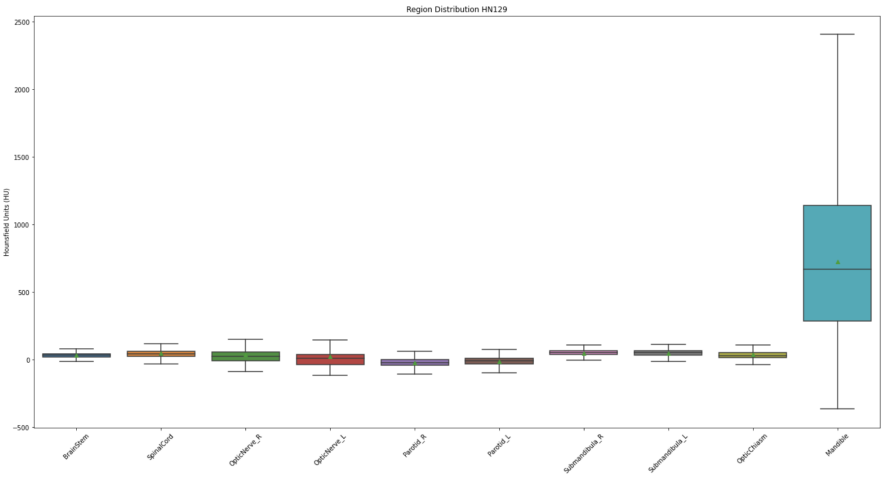
3. Compute the loss between the ground truth and the prediction using the chosen loss functions. Feed the loss back to iteratively update our model using gradient decent method.

4. Ensemble the predictions of Res3DUNet and UNETR model for the best prediction on test data set.

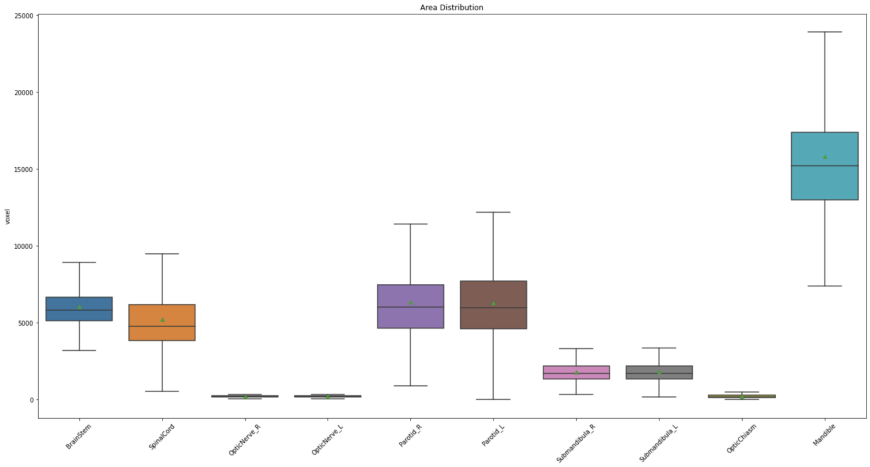
4 Experimental Evaluation

4.1 Data

We use the data from MICCAI 2015 Head and Neck Auto-segmentation Challenge[3]. The data set consists of 48 CT images corresponding to 48 patients. The images are manually segmented to annotate nine OARs: brain stem, mandible, chiasm, optic nerves left & right, parotid left & right, and



(a) Hounsfield Unit Distribution



(b) Area Distribution

Figure 3: Region Distribution

submandibular left & right. 15 images were set as test set and the remaining images were randomly split into training (25 images) and validation (8 images). Each CT image is a 3D volume of dimension N x 512 × 512, having a height and width of 512 pixels and N (140 ± 30) slices. Each OAR has a binary segmentation mask of the same dimensions as the CT image. A value of 1 indicates a specific structure's presence and 0 otherwise. To see the distribution of the OARs, we use both Hounsfield Unit (HU) and volume. Conventionally, the numerical values of the voxels in CT scans are represented using the Hounsfield Unit (HU), measuring the density of the organs. For our data set, these HU values range from -1024 to 2000. As shown in Figure 3a, all the structures except the mandible, which is a bone with high density, have HU values in the range of $(-200, 200)$. Apart from HU, we also measure the volume of each OARs by counting the number of voxels they take up in Figure 3b. Mandible, being the largest organ, takes up about 15000 voxels per image, while optic nerves and chiasm barely take up any space. As we can see, the data set has a serious imbalance problem, in the following parts we will elaborate on the techniques applied in this project in order to tackle such a problem.

Windowing Windowing is a way to strengthen the contrast ratio of an image in order to highlight certain structures. We use this technique in this project because it would be hard for the model to distinguish gray scale levels. We leverage the standard soft-tissue windowing heuristic3[12] and clip the pixel values to the range of [-155, 195].

Table 1: Comparison of different loss functions

Name	Loss Function	Dice Score
Res3DUnet	Dice Loss	0.3180
Res3DUnet	Dice Loss + Focal Loss	0.3963
Res3DUnet	Dice Loss + Weighted Cross-Entropy	0.3560
Res3DUnet	Generalized Dice Loss + Focal Loss+ Weighted Cross-Entropy	**0.5744**

Cropping The original volume has the dimension of (N,512,512), which is too expansive in terms of computational cost for a U-Net based model (N ranges from 110 to 190). We convert the original 3D volume to the size of (96,128,224) through cropping the region of interest.

Normalize We normalize the CT volumes to have zero mean and unit standard variance for the ease of convergence.

4.2 Methodology

This section discusses the details about the methodologies we use to achieve the final solution. In general, we have experimented with the following techniques:

- Data augmentation including random affine and random elastic transformation.
- Combine multiple loss functions and dynamically weighted loss functions along training procedure.
- Assemble predictions of models with different structures.

4.2.1 Loss Functions and Evaluation Metrics

We have explored different combinations of the following loss functions and use Dice Score as the evaluation metrics. Various loss functions can provide our model with multiple angles to better segment different organs. A combination of Generalized Dice Loss, Focal Loss and Weighted Cross-Entropy provides us with the best performance. See Table 1

Weighted Cross Entropy[14] Cross Entropy is traditionally used for classification problems and we add weight based on pixel counts to the equation to resolve the data imbalance problem.

$$WeightedCrossEntropy = \begin{cases} -w \log p & y = 1 \\ -\log(1-p) & otherwise \end{cases} ;\ w = c * \frac{N - \sum_n p_n}{\sum_n p_n} \tag{1}$$

note: w will be scaled by constant c such that weight for all classes sum up to 1.

Focal Loss[11] Focal loss is an refined version of cross entropy and it puts less weight on "easy" samples and more on "hard" ones.

$$FocalLoss = \begin{cases} -(1-p)^\gamma \log p & y = 1 \\ -(p)^\gamma \log(1-p) & otherwise \end{cases} ; \gamma = 2 \tag{2}$$

Dice Loss[10] Dice loss measures the similarity of two samples and is essentially a ratio of intersection over union.

$$DiceLoss = 1 - \frac{2 \sum_{pixels} p_i \cdot y_i}{\sum_{pixels} p_i + \sum_{pixels} y_i} \tag{3}$$

Generalized Dice Loss[14] Generalized Dice Loss is an extension of Dice Loss. It is tailored to tackle data imbalance problem by adding pixel weight to the equation.

$$GeneralizedDiceLoss = 1 - 2\frac{\sum_{labels} w_l \sum_{pixels} y_{li} p_{li}}{\sum_{lables} w_l \sum_{pixels} y_{li} + p_{li}}; w_l = 1/\left(\sum_{pixels} y_{li}\right)^2 \tag{4}$$

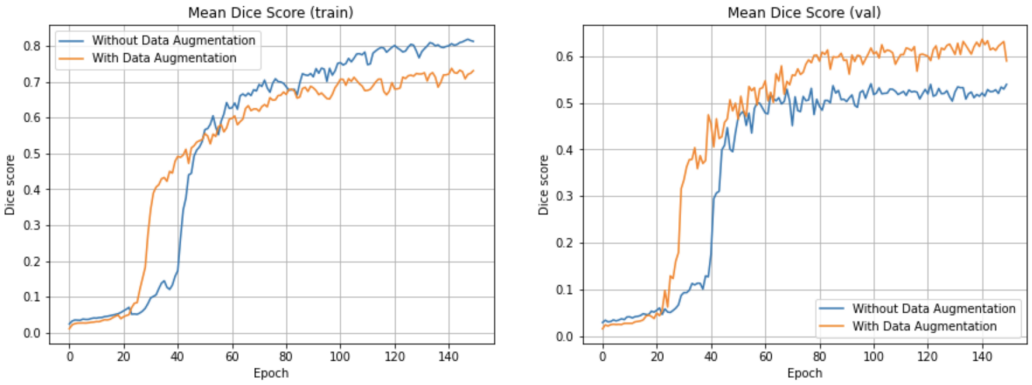


Figure 4: Improvement provided by data agumentation

Table 2: Grid search result of data augmentation. P stands for the applied probability.

Name	Data Augmentation	Dice Score
Res3DUnet		0.6077
Res3DUnet	RandomAffine(P=0.5)	0.6706
Res3DUnet	RandomAffine(P=1)	0.6928
Res3DUnet	RandomElasticDeformation(P=0.5)	0.6318
Res3DUnet	RandomElasticDeformation(P=1)	0.6412
Res3DUnet	RandomElasticDeformation(P=0.5) +RandomAffine(P=0.5)	0.7155
Res3DUnet	RandomElasticDeformation(P=0.3) +RandomAffine(P=0.7)	**0.7324**

4.2.2 Dynamically Weighted Loss Function

To further deal with the class imbalance issue and strengthen our model's performance on small organs, we replaced the fixed combination of loss functions with dynamically weighted loss functions. We decide to put more weight on Weighted Cross-Entropy at the beginning of training to help roughly locate organs' locations. In order to improve performance on small organs, we switch to the mode with more weight on Focal Loss utilizing its feature of focusing on hard examples, when the average performance increases on the validation set. The best setting we find is as follows:

- Stage1: Generalized Dice Loss + 1 * Weighted Cross Entropy + 0.1 * Focal Loss
- Stage2: Generalized Dice Loss + 0.1 * Weighted Cross Entropy + 0.5 * Focal Loss
- Stage3: Generalized Dice Loss + 0.01 * Weighted Cross Entropy + 1 * Focal Loss

The mean Dice Score improves to 0.6077 after using dynamically weighted loss functions, benefiting mainly from better performance on small organs.

4.2.3 Data Augmentation

Random affine A geometric transformation that changes distances and angles while keeping lines and parallelism.

Random elastic deformation A random displacement assigned to a coarse grid of control points around and inside the image voxels using cubic B-splines.

According to Ronneberger et al.[6], Spatial transformation like random affine and random elastic deformation is an extremely useful data augmentation tool for 3D segmentation tasks with a small data set. It boosts the model's performance on test set tremendously. Random affine and random elastic deformation augment the data in various ways, thus, a combination of them provides us with a better ability of generalization. Through grid searching results (check Table 2) and considering the relatively long processing time of random elastic deformation, we apply 30% probability for random elastic transformation and 70% probability for random affine. Improvement is visualized in Figure 4.

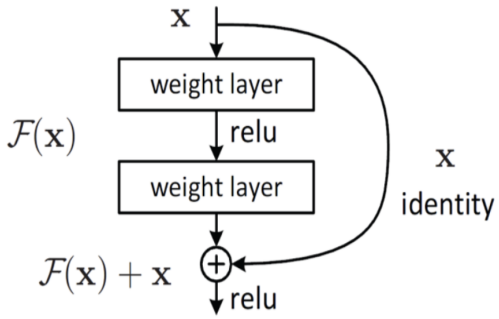


Figure 5: Residual block

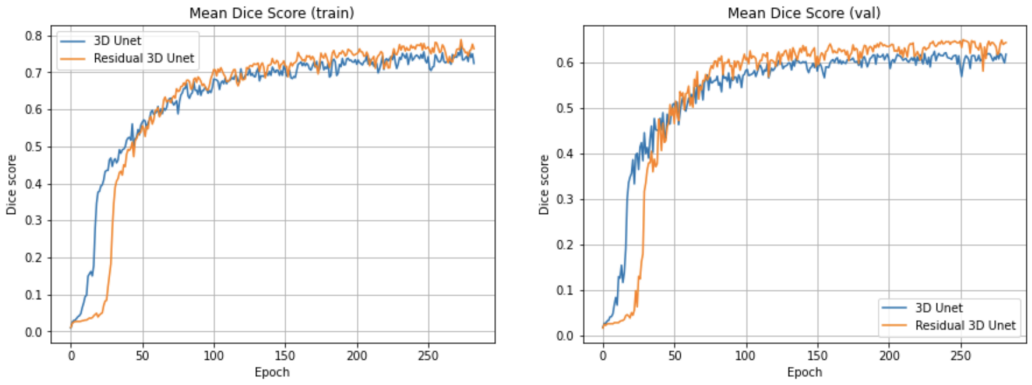


Figure 6: Improvement provided by adding residual connection

4.2.4 Model Architecture

Res3DUnet Res3DUnet is a variant of 3D U-Net with residual blocks replacing the traditional convolutional blocks. Residual blocks[4] enable layers to learn F(x) = H(x) - x instead of approximating H(X). The original function becomes F(x) + x. Residual connection is significantly important for training giant neural networks. It increases the model's generalization ability. (Figure 5)

UNETER We use UNEt TRansformers (UNETR) (Figure 2) as our second model. It is a novel architecture which leverages a Transformer as the encoder to learn sequence representations of the input CT volume, while following the 'U-shaped' structure for the encoder and decoder. The Transformer encoder is directly connected to a CNN-based decoder through skip connections at different levels of resolution to compute the final semantic segmentation. Similar to its applications in NLP, UNETR creates a 1D sequence of a 3D input volume (H, W, D, C) with resolution (H, W, D) and C input channels via dividing it into flattened uniform non-overlapping patches. The resolution of each patch is (P, P, P) and the length of sequence is $N = (H * W * D)/P^3$. UNETR makes use of a linear layer to project the patches into a K dimensional embedding space, adding a 1D learnable positional embedding $(N * K)$ at the end to reserve the spatial information of the current patches (Figure 7). After the embedding layer, UNETR utilizes a couple of Transformer layers consisting of multi-head self-attention (MSA) and multilayer perceptron (MLP) sublayers to create feature maps of different levels. Following the 'U-shaped' methodology, features from various resolutions of the encoder are merged into the decoding process for final segmentation. UNETR is able to capture long-range dependencies effectively and outperforms Res3DUnet in some small and symmetric tasks while keeping similar performance on the general Dice Score. Figure 8 provides detailed statistics of nine organs compared with Res3DUnet and Figure 9 shows a visualization of performance comparison on small organs OpticNerve-Left and OpticNerve-Right.

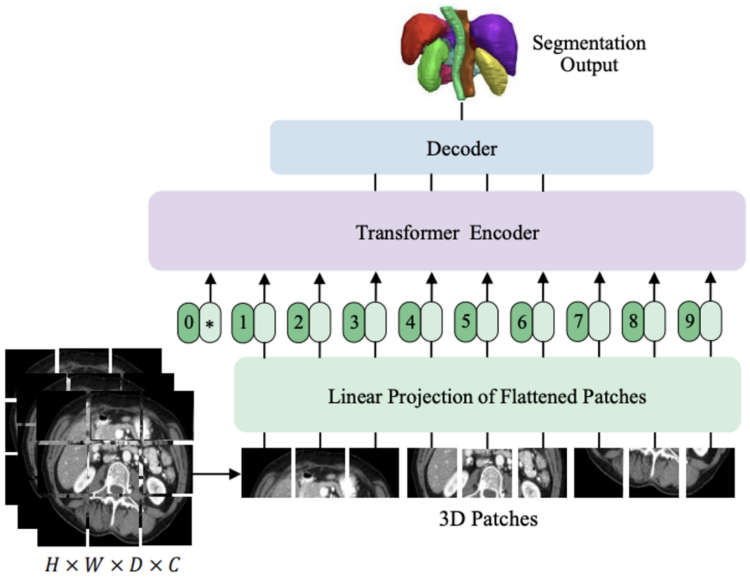


Figure 7: Overview of UNETR

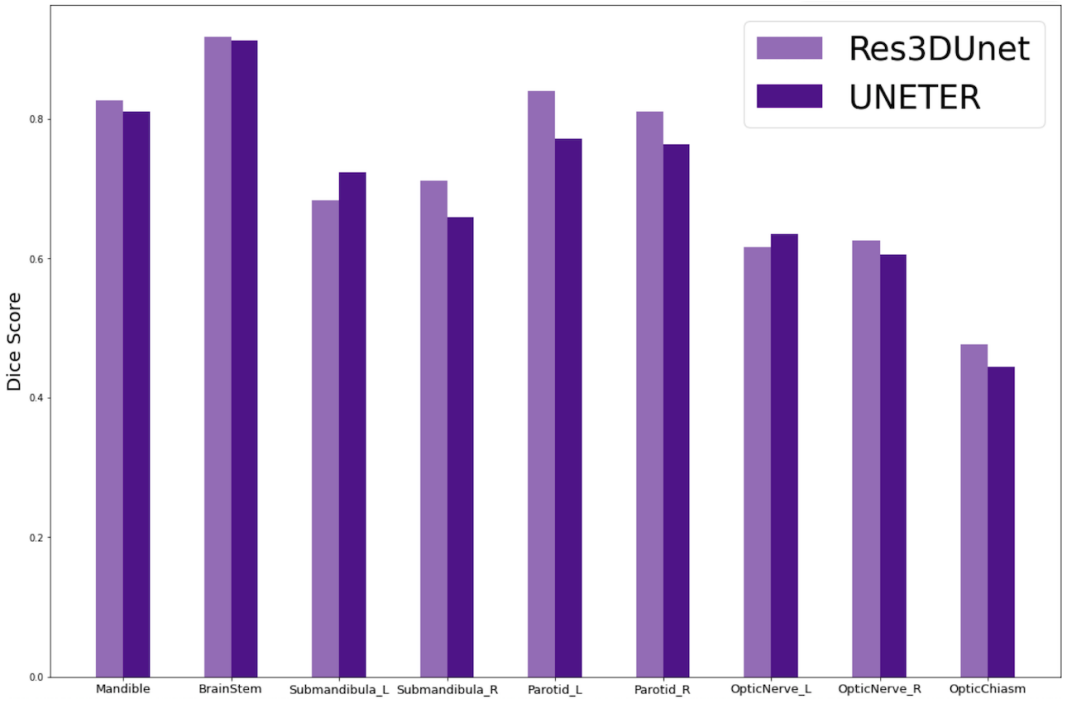


Figure 8: Res3DUnet vs UNETR test set performance

Figure 9: Visualization of segmentation on OpticNerve-Left and OpticNerve-Right. Ground Truth (left), UNETR (middle), Res3DUnet (right) OpticNerve-Left (green) ,OpticNerve-Right (yellow)

Table 3: Model configuration

	Res3DUnet	UNETR
Parameters	31.7M	93.1M
Loss Functions	Generalized Dice Loss+Focal Loss +Weighted Cross-Entropy	Generalized Dice Loss+Focal Loss +Weighted Cross-Entropy
Training Method	Dynamically weighted loss function	Dynamically weighted loss function
Learning Rate	0.001	0.0001

Table 4: Model Comparison

	Res3DUnet	UNETR	Ensemble	2D U-Net	MICCAI 2015
Chiasm	50.3	44.4	50.2	55.1	**55.7**
Mandible	91.6	91.2	**93.8**	83.8	93.0
BrainStem	82.7	81.0	83.8	85.5	**88.0**
OpticNerve-L	65.2	63.5	**67.6**	65.9	64.4
OpticNerve-R	68.7	60.6	**68.7**	64.1	63.9
Parotid-L	81.6	77.2	**82.9**	80.2	82.7
Parotid-R	81.9	77.6	**82.0**	79.8	81.4
Submandibular-L	69.5	72.4	**72.4**	70.1	72.3
Submandibular-R	67.7	65.9	**74.0**	64.3	72.3
Average	73.2	70.3	**75.0**	72.2	74.9

4.2.5 Ensemble

Although our two best trained models have similar Dice Score performance, they contour differently for each organ due to their distinct U-Net Structures. It provides us with flexibility to simply assemble these two models for better prediction. We utilize the two ensemble techniques "Maximizing margin" and "Minimizing margin" separately for each organ, achieving 4% increase on performance.

Maximizing margin Maximizing margin method takes the union of predictions for organs from two models in order to contour as much voxels as possible. This method works best for large organs and organs with very different contours from two models. (Mandible, Brainstem, Submandibular-L, Submandibular-R)

Minimizing margin Minimizing margin method takes the conservative prediction probability for organs from the two models and apply softmax for final predictions. Compared with taking the intersection of predictions, this method contours larger areas while keeping conservative predictions, since we favor larger contouring area in practice for organs-at-risk. This method works best for small organs.

4.3 Results

Table 3 shows the configuration of our best models. Both models are trained until the validation performance does not improve after 50 consecutive epochs.

Comparison with State-of-the-Art Table 4 compares our models with the previous capstone project 2D U-Net [1] and the winner of MICCAI 2015 Challenge [3]. Our Ensemble model is able to outperform in most of the tasks among all the models.

4.4 Discussion

It is hard to train giant neural networks like 3D U-Net for multiple classes segmentation. Utilizing various loss functions and dynamic weighting techniques can be very helpful in alleviating the class imbalance issue. It is beneficial to choose the combination of loss functions with different focuses and change their weight accordingly during training. Meanwhile, we observe a boost in performance after applying data augmentation. As the data set is small, data augmentation can provide our

models with more training samples and assist it to focus more on invariant features. However, data augmentation is tricky for small size classes, the model's performance can easily collapse due to too much displacement of original data. Additionally, variants of 3D U-Net like Res3DUnet and UNETR can also improve model's performance, benefiting from their inner refinement of U-Net structures. Ensemble of models with different U-Net structures can further improve our segmentation performance.

From Table 4, our best ensemble model achieves a bit higher mean Dice Score than the winner of MICCAI 2015 (75.0 vs 74.9). Through absorbing advantages of our two best models, our ensemble model outperforms other models in all symmetric tasks (Mandible is symmetric by itself, see Appendix). The explanation is that UNETR has a unique U-net structure adopting the idea of Transformer which can capture long distance relations effectively. Although UNETR model's Dice Score performance of symmetric tasks is similar to Res3DUnet, their predictions are very different (Figure 9). Therefore, ensemble of the two models will show superiority in tasks which benefit from comprehensive symmetric relation features.

5 Conclusions

In conclusion, we propose an end-to-end solution with 3D U-Net, which outperforms the winner of MICCAI 2015 (state-of-the-art). We leverage novel refinement to the U-Net architecture by using residual blocks and the Transformer structure, respectively constructing two models. We also implement a couple of data augmentation techniques to resolve overfitting issues and a dynamically weighted loss function to tackle the data imbalance problem.

For future work, we consider training our models on a larger data set, which may improve the performance. We would also like to modify the inner structure of 3D U-Net to fit the head and neck segmentation task specifically. Furthermore, we believe that UNETR's inner structure can be improved to reduce the number of parameters of the model. For instance, In Google's newly published paper[13], a module called TokenLearner is introduced. The implementation of TokenLearner into the Transformer architecture not only decreases computational cost, but also increases the accuracy of the model. Lastly, we plan to apply more evaluation metrics to train our models in a more comprehensive way.

6 Lessons Learned

- **Multiple imbalance classes segmentation is tricky.** Before utilizing combinations of various loss functions, our model suffered from failure in learning small organ tasks.

  It is important to prevent overfitting for small data sets like 3D volume data sets. 3D volume data sets are usually small due to its giant sample size. Spacial transformation including random affine and random elastic deformation can augment the data in various ways to help the model better generalize. However, small organ tasks are fragile for data augmentation, we spent a large amount of time searching for suitable parameters for transformations.

  Ensemble of different models helps. Utilizing ensemble model of Res3DUnet and UNETR can further improve performance on a high basis.

  3D model is expensive. 3D model is computationally expensive in terms of training time and memory. Tryout of different parameters and settings is limited due to time concern.

  Visualization is important for model training and debugging. We utilized Tensorboard and PyTorch-Lighting to set up a real time visualization of learning curve which allowed us to quickly diagnose problems of the model. This is extremely helpful for time consuming 3D Models. We also created a pipeline for output visualization through Matplotlib. This makes it easier for us to consult industry experts about model's pros and cons for future improvements.

Acknowledgement

We would like to thank Doctor Ye Yuan for mentoring and guiding us throughout the project. We also want to thank Doctor Narges Razavian, Doctor David Barbee, Doctor Ting Cheng for helping us in the project. Lastly, we want to thank Wenda Zhou for giving us advice and keeping track of the whole progress.

References

[1] Jain & Virtanen (2020) "Head and Neck CT Image Segmentation".

[2] Çiçek, Özgün, et al. "3D U-Net: learning dense volumetric segmentation from sparse annotation." *International conference on medical image computing and computer-assisted intervention*. Springer, Cham, 2016.

[3] Raudaschl, P. F., Zaffino, P., Sharp, G. C., Spadea, M. F., Chen, A., Dawant, B. M., ... Jung, F. (2017).Evaluation of segmentation methods on head and neck CT: Auto-segmentation challenge 2015.Medical Physics, 44(5), 2020-2036.

[4] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[5] Hatamizadeh, Ali, et al. "Unetr: Transformers for 3d medical image segmentation." *arXiv preprint arXiv:2103*.10504 (2021).

[6] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

[7] Zhu, Wentao, et al. "AnatomyNet: deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy." *Medical physics* 46.2 (2019): 576-589.

[8] Raudaschl, Patrik F., et al. "Evaluation of segmentation methods on head and neck CT: auto-segmentation challenge 2015." *Medical physics* 44.5 (2017): 2020-2036.

[9] Gao, Yunhe, et al. "Focusnet: Imbalanced large and small organ segmentation with an end-to-end deep neural network for head and neck ct images." *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, Cham, 2019.

[10] Milletari, Fausto, Nassir Navab, and Seyed-Ahmad Ahmadi. "V-net: Fully convolutional neural networks for volumetric medical image segmentation." *2016 fourth international conference on 3D vision (3DV)*. IEEE, 2016.

[11] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *Proceedings of the IEEE international conference on computer vision*. 2017.

[12] Murphy, A., Baba, Y. Windowing (CT). Reference article, Radiopaedia.org. (accessed on 08 Dec 2021) https://doi.org/10.53347/rID-52108

[13] Ryoo, Michael S., et al. "TokenLearner: What Can 8 Learned Tokens Do for Images and Videos?." *arXiv preprint arXiv:2106.11297* (2021).

[14] Sudre, Carole H., et al. "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations." *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, Cham, 2017. 240-248.

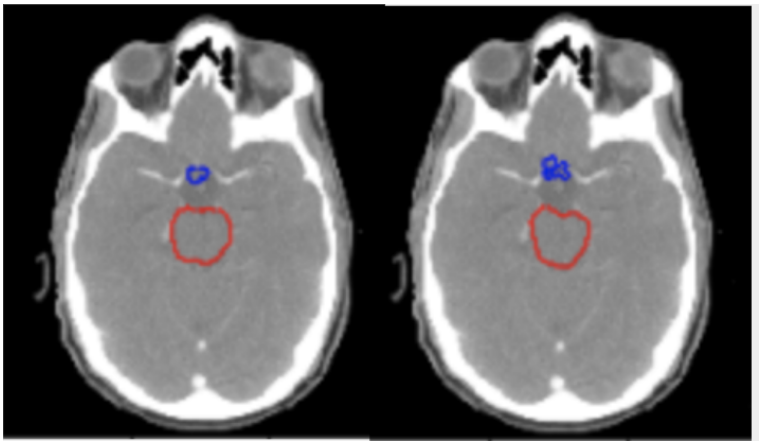
Student contributions

Pengyun: data preprocessing and visualization pipeline, report

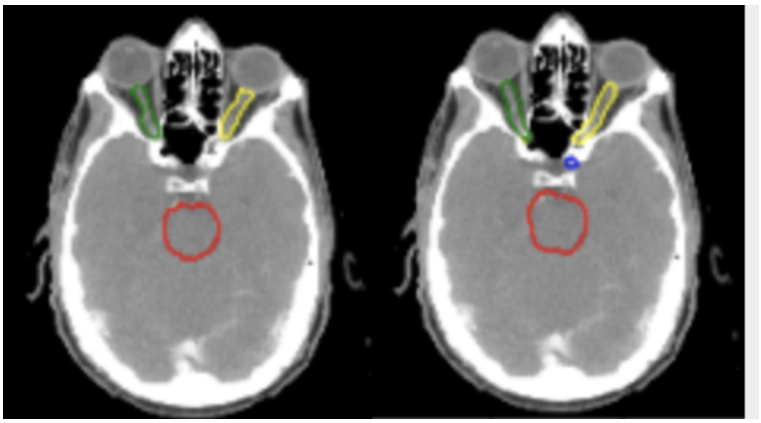
Tianyu: Implement models and end-to-end training, report

Appendices

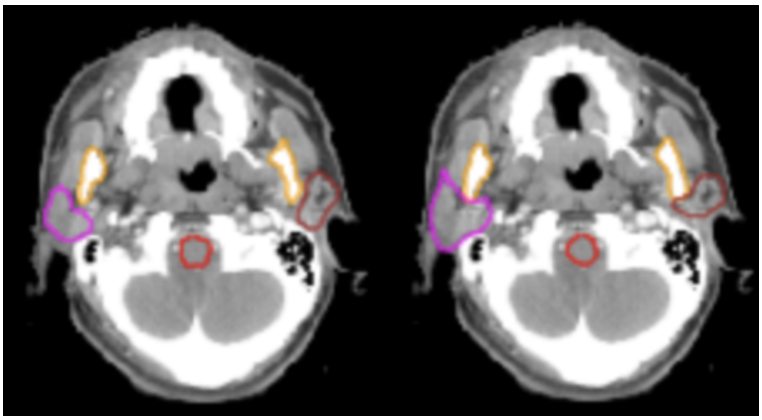
Ground-truth (left) and Prediction (right)



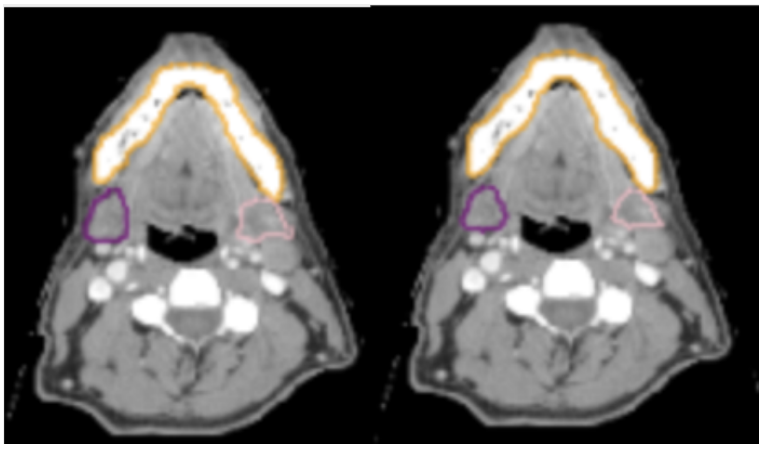
BrainStem (red), OpticChiasm (blue)



OpticNerve Left (green), OpticNerve Right (yellow), BrainStem (red)



Parotid Left (margenta), Parotid Right (brown), BrainStem (red), Mandible (orange)



Mandible (orange), Submandibula Left (purple), Submandibula Right (pink)