Kindey Tumor Image Segmentation using 3D-UNet Neural Network

Abdullah Al-Hayali 0969687
School of Engineering
University of Guelph
Guelph, Ontario
aalhayal@uoguelph.ca

Megan Govers 0956716
School of Engineering
University of Guelph
Guelph, Ontario
mgovers@uoguelph.ca

***Abstract*—Annually, 7500 Canadians will be diagnosed with kidney and renal pelvis cancer, and 1950 of them will succumb to the disease. In efforts to contribute to better detection and diagnosis, a CT scan image semantic segmentation algorithm will be implemented to help to detect and classify tumors in the kidneys. As a part of the KiTS 19 challenge, the provided kidney tumor and segmentations were utilized and segmented using 3D UNet. For the test size and computational power available, the algorithm yielded promising results that can be improved upon in order to translate these algorithms to clinical applications.**

***Index Terms*—Kidney Tumor, KiTS-19 , Semantic Segmentation, 3D U-NET**

I. Introduction

Each year in Canada, approximately 7500 Canadians will be diagnosed with kidney and renal pelvis cancer, and 1950 Canadians will succumb to the disease[1]. Of those 7500 cases, approximately 65% will be male[1]. Kidney cancer is the ninth most common cancer in men and fourteenth most common cancer in women, and the sixteenth leading cause of death worldwide [4]. In 2018, 400 000 cases were diagnosed worldwide[5]. With the increase in abdominal imaging for various unrelated indications, the incidental detection of asymptomatic renal masses has become increasingly common. This has increased the proportion of tumors that are small and localized when treated, which is thought to be a contributing factor to the disease's increased overall survival [8].

KiTS19 challenge is to promote and improve development of methodologies used in kidney and kidney tumor segmentation. This challenge has a specific focus on surgical outcomes. Kidney cancer is often treated initially using surgery [7], which makes surgical outcome prediction of the utmost importance. The data set for this problem contains CT scan data for 300 cancer patients who underwent partial or radical nephrectomy at the University of Minnesota Medical Center. A radical nephrectomy consists of removal of the entire kidney and additional surrounding tissue if necessary, while the partial nephrectomy removes the tumor and a margin of healthy tissue surrounding it[7].

This paper attempts to implement semantic segmentation utilizing 3D U-Net as a neural network architecture. The performance and accuracy of the architecture will be assessed on low to mid range power computers with optimization in mind. Overall goal: In terms of image segmentation, the 3D UNet architecture is designed to utilize semantic segmentation, where the object desired will be highlighted as a painted pixel, and the background and unnecessary image detail will be painted a different colour pixel, creating a distinct separation in the segmented image. This type of segmentation is also referred to as multiclass semantic segmentation due to the presence of multiple classes in the segmentation.

II. Methods

A. Hardware Specifications

To fit and evaluate the model a custom PC rig was used. The PC used contains a quad-core i7-7700 3.6 GHz processor, 32 GB of DDR4-2400 RAM and 2 graphics processing units - an Nvidia GeoForce RTX 2070 8GB video card and an AMD R9 380 4GB video card. The model was trained using 5 training data samples, 3 validation data samples, and 5 test data samples, for 10 epochs and required over two hours to fit.

B. CT Scan Image Pre-Processing

The KiTS 19 challenge provided data for 300 patients, each image contained original image arrays of the CT scan, along with their ground truth segmentations attached inside the same folder. After investigating the dataset, it was decided that 13 cases were to be used instead of the 300 cases due to computational power availability. Isensee and Mair-Hein stated that the dataset required modifications. Case IDs 23, 68, 125, and 133 were excluded due to their networks being in disagreement, therefore, to avoid potential discrepancy in the data quality given the relatively small sample, they were excluded in the model training and analysis. Furthermore, case IDs 15 and 37 were excluded due to them being faulty data samples, as per the KiTS 19 organisers [10].
When loading the data, the utility function provided by the organizers was used to load the data. The code is as follows:

```
from starter_code.utils import load_case
volume, segmentation = load_case("case#")
```

The CT scan images extracted are Nifty images, with volume being a NumPy float 32 data type, and the segmentation being a NumPy unsigned integer 8 (uint8).

The images were normalized based on the CT volume minima of -1000 Hounsfield Unit (HU), which is the standard quantification unit for CT scans. Standard convention to normalize CT scans is by setting a minimum bound of -1,000 and a maximum bound of 400[6]. Following that, the images and ground truth segmentations were resized to size (64,128,128), with 64 representing the depth pixel value (number of slices), and 128 representing width and height pixel value respectively. 64 slices was chosen since the minimum slices number in the data used was 67.

The given ground truth segmentations were reshaped to three channels such that they could be easily compared to the output produced by the model which also has 3 channels. This allowed an comparison of categorical accuracy, cross-entropy and categorical and generalized dice scores.

The ground truth segmentations were also one-hot encoded, as they contain mask data embedded in the three classifications (0 = background, 1 = kidney, 2 = tumor). The one-hot encoding process converts the the integer classes to binary variables unique to each class.

C. Neural Network Architecture

The neural network selected to train the data in is 3D UNet. 3D UNet was designed to learn to generate dense volumetric segmentations, but only requires some annotated slices to train the network. The main two applications for 3D UNet is to densify sparsely annotated data sets, and the other learns from multiple sparsely annotated data sets to generalize new data [9].

Prior to introducing the network, it is worth defining two terms that are crucial in the functionality of 3D UNet, which are max pooling and dropout. Max pooling is a sample based discretization process, in which the input is down-sampled, reducing its dimensionality and allowing for assumptions to be made about features contain in the sub-regions. Dropout is a regularization technique where randomly selected neurons are ignored, or ”dropped out”, during training, resulting in less overfitting[3]. The dropout value is a randomised probability value that was set to be between 0 and 1.

The network contains analysis (left) and synthesis (right) paths each with four resolution steps. In the analysis portion of the architecture, each layer contains two 3x3x3 convolutions followed by a rectified linear unit (ReLu), followed by a 2x2x2 max pooling with strides of two. In the synthesis section, each layer consists of upconvoluting 2x2x2 by strides of two, followed by two 3x3x3 convolutions each followed by a ReLu. The last layer consists of a 1x1x1 convolution, reducing the number of output channels to the number of labels. Batch normalization was also introduced before each ReLU instance, with each batch being normalized during the training with its mean, standard deviation. The global statistics are then updated using the output values[9]. The architecture is shown visually in Figure 1.

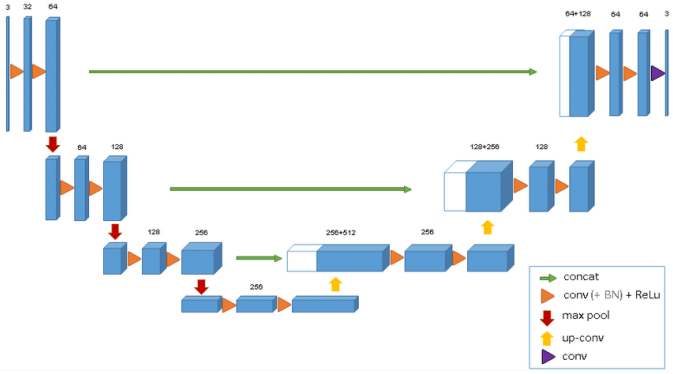


Fig. 1: 3D UNet Architecture [9]

Highlighted by Cicek et. al., the important section of the architecture that allows us to train on sparse annotations is the weighted softmax loss function, with setting the weights of unlabelled pixels to zero, making it possible to learn from only the labelled ones, hence generalizing the whole volume[9].

I. Inputs, Outputs and Parameters

3D UNet's architecture results in 10,816,259 total parameters, where 10,814,851 of which are trainable and 1,408 are not. The input to the 3D UNet model consists of the dimensions of the pre-processed CT scans, as well as the validation and training data sets that have been pre-processed. The model outputs a final segmentation with the same dimensions as the input data, with 3 channels instead of 1. Where, each channel represents a different category of the segmentation - background, kidney and tumor respectively.

The output given by the UNET architecture consists of a numpy matrix of probabilities. Thus in order to generate the segmentation, the label with the highest probability is assigned to said pixel value, where the label is equal to the channel number-1.

D. Training the Model

I. Optimization -

The Adam optimizer was used for optimization of the model. Adam is a stochastic optimizer, and is commonly used in CT scan segmentation [21], [22], [14], [15], [23], [16]. More specifically, Adam only requires first order gradients to optimize, thus presenting a benefit of reduced memory requirements, and computational efficiency [21], [13]. Adam is an adaptive moment estimation technique where the learning rate for all parameters is estimated and used to train [21]. Throughout its application Adam creates adaptive learning rates for all trainable parameters

within the model by estimation of first and second moments [13]. In this model, the Adam optimizer was implemented with a learning rate of 0.001. During training several metrics were considered during the optimization process. The 3D UNet model created considered four metrics: dice coefficients, class-wise dice coefficients, categorical accuracy and categorical cross entropy. The loss of the model was Tversky loss.

II. Tversky Loss Index -
Tversky index loss is often chosen for use in 3D convolution neural networks, such as 3D UNet [16]. The main advantage of Tvserky index loss is its optimal performance in data with imbalanced classes compared to other loss metrics [16], [17]. In this particular case, the class imbalance comes from the very large number of pixels classed as background compared to those classed as kidney or tumour. This large class imbalance can create issues when using other metrics, leading to low recall and high precision [16], [17], [18]. However false positives are preferred over false negatives in medical applications, thus it is very important to overcome this challenge by implementing Tverksy index loss [16].
The Tversky index quantifies similarity by using a combination of dice coefficient and Jaccard index, where the dice coefficient quantifies the area of overlap between pixels in 2 images and the Jaccard index is a statistic measure that quantifies the similarity between sample sets [18]. Choosing the weights built into this function allows the user to increase penalization of false negatives [18].

III. Metrics
   i. Dice Coefficients -
   Two types of dice coefficients were used as metrics in this implementation - generalized and class-wise dice. The generalized dice coefficient determines the area of overlap between the two images being compared - the given segmentation and the predicted segmentation [19]. Dice coefficient is calculated as

$$Dice = \frac{2 * (A \bigcap B)}{|A| + |B|} \tag{1}$$

   Where A is the predicted segmentation and B is the given ground-truth segmentation and $|A|$ and $|B|$ are the number of pixels in each respective segmentation [19].
   However, if the classes present a large imbalance, the model can score highly on the dice coefficient, while having poor segmentation performance [20]. Thus, class-wise dice coefficients were also considered. Class-wise dice considers the area of overlap within specific classes [20]. Therefore, if a model is being dominated by one class, the class-wise dice will be low while the generalized dice will be high. This metric allows insight into finer details of the non-dominant classes within the segmentation, and is therefore preferred since this is a class-imbalanced problem [20].

   ii. Categorical Accuracy -
   When training the model the built in metric of categorical accuracy from the Keras library was utilized. Since the output of the model is a 3 channel NumPy array, with each channel corresponding to a class, categorical accuracy was used for the multi-class segmentation. This output is one hot encoded thus allowing the use of categorical accuracy. However, it important to consider that this metric is likely to present a very high value, even in the case of poor segmentation due to the large dominance of the background class over the kidney and tumor classes.

   iii. Categorical Cross-Entropy -
   Categorical cross entropy, from the Keras library was also used as a metric. Since the segmentation is multi-class, each channel with have a value of 0 or 1 for the ground truth, or given segmentation. Thus the probabilities that are output can be compared to those given in the ground truth. Categorical cross entropy consists of a combination of a soft-max activation function and a cross entropy loss. Thus training our model to output the probability for each class for the given input image. Binary cross-entropy is commonly used when segmenting into 2 classes, where one is of interest and the other is not, thus it was logical to employ categorical cross entropy as the loss function.
   However, once again, this metric may show an unusually high value even in the case of poor segmentation, due to class imbalance, since the built in categorical cross entropy function is not balanced.

IV. Data Augmentation -
Data augmentation is a technique used to diversify the training set by applying random slight transformations to the original image. Such image transformations can be rotations, zooming in an out, flipping the image in the horizontal and vertical axes. The type of data augmentation performed is on-the-fly augmentation during training, which applies transformations on mini-batches fed to the model[2].
The data augmentations performed on the data set were random rotations from -20 to 20 degrees with 5 degree increments.

E. Predicting Using the Model

Since the output of the model is a 3 channel image, further steps must be taken to create a predicted segmentation. Within each channel, each pixel location contains the probability of said pixel belonging to that channel or class. Therefore to create the segmentation, channel with the highest probability is assigned to the corresponding pixel to that class. This results in a segmentation with integer values, with 0 corresponding to background, 1 corresponding to kidney and 2 corresponding to tumor. This segmentation can then be compared to the ground truth segmentation and performance can be evaluated.

F. Evaluation using KiTS 19 Dice Scores

The KiTS 19 challenge uses two different dice scores to quantify the performance of submissions. These dice scores are tumor dice and tumor+kidney dice. Essentially, they are class-wise dice scores, where tumor dice considers only the tumor class, while tumor+kidney dice considers the average dice of the tumor and kidney classes. The leader board is ranked on the basis of mean kidney + tumor dice. Thus, this metric was considered when evaluating the network's segmentation performance, but it was not included in training, since class-wise dice was already considered.

III. Results and Discussion

It is worth noting that due to computational power limitations, 5 training data samples, 3 validation data samples, and 5 testing data samples. The prediction output of the model (ie. masks) were then overlayed on its corresponding volumetric image to produce the final semgentations. For training, validation and testing, the average tumor dice and kidney+tumor dice were calculated, and summarized in table I.

TABLE I: Average, and kidney + tumor dice scores for training, validation, and testing data.

Dice	Training	Validation	Testing
Kidney+Tumor	0.00119	0.0	0.000007
Tumor	0.0	0.0	0.0

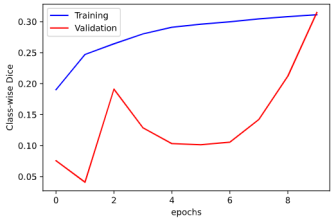
Based on table 1, it is clear that the models performance is much below the level that would be considered adequate. However, it should be considered that a very low number of epochs and training data were used, and that performance is likely to improve if the model receives further training. Thus, at this point the model performs very poorly in its given task, especially when compared to the current challenge leader who has a mean kidney+tumor dice score of 0.9168.

Figure 2 showcases 3 of the metrics used to quantify performance of the model throughout its training epochs. It is important to consider that the general dice score (Figure 2c) has much increased values when compared to the categorical dice score. This is likely caused by the dominance of the background class. From the categorical dice scores (Figure 2a) and the kits 19 dice scores (Table 1) it is clear that the model performance leaves much to be desired. However, Figure 2a shows a increased in categorical dice score that has clearly not converged, thus model performance can likely be improved significantly through the use of larger data sets and increased training epochs.

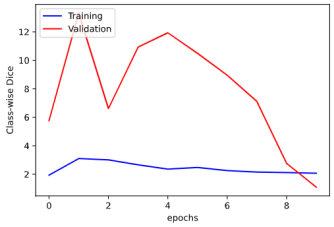
The ideal categorical cross entropy value would be 0, thus it is clear that this value has also not converged to the desired value, therefore the model requires further training.

It can be seen that through the training segmentations compared to the ground truth segmentations are poor, as the kidneys detection is is swayed to the left, compared to the middle positioning of the kidney and slight tumor presence. Along with that, the tumor was segmented to be everywhere surrounding the kidneys except for the kidneys themselves. The validation data did not result with a proper segmentation, with the only classification being background. The test data showed a tumor classifcation that is to the very left of the image, and no kidney classification, despite the groundtruth segmentation being slighltly composed of kidneys with no presence of a tumor.

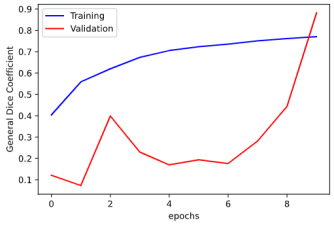
These results are to be expected since the sample size and number of epochs is extremely small relative to what is required to compose proper and meaningful segmentations. The results are, however, promising, since a segmentation process resulted in classes other than background despite the dominance of the background class and the very small sample size and epochs.



(a) Classwise dice plot



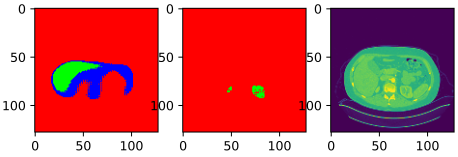
(b) Categorical cross entropy dice score



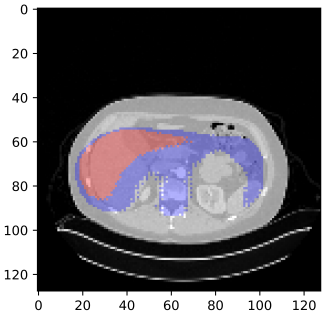
(c) General dice score

Fig. 2: Dice scores plot

Figure 3 presents predicted segmentations from the a single training data sample, compared to the corresponding ground truth segmentations, along with the overlayed segmentation on the volumetric CT image shown.



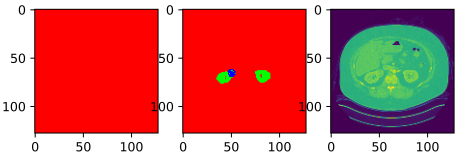
(a) Predicted training segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right). Green = kidney, and Blue = tumor



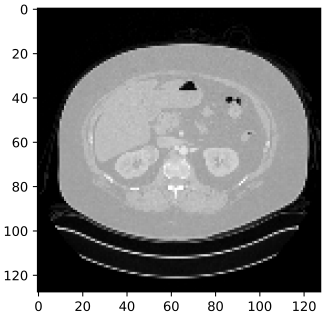
(b) Overlayed segmentation on the volumetric CT image. Red = kidney, Blue = tumor

Fig. 3: Train segmentation results

Figure 4 presents predicted segmentations from the a single validation data sample, compared to the corresponding ground truth segmentations, along with the overlayed segmentation on the volumetric CT image shown.

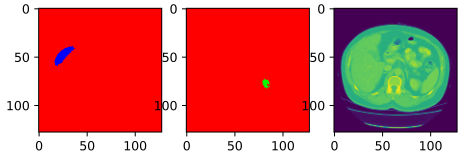


(a) Predicted validation segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right). Red = background, Green = kidney, and Blue = tumor

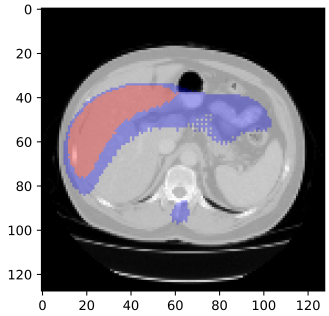


(b) Overlayed segmentation on the volumetric CT image, Red = kidney, Blue = tumor

Figure 5 presents predicted segmentations from the a single testing data sample, compared to the corresponding ground truth segmentations, along with the overlayed segmentation on the volumetric CT image shown.



(c) Predicted testing segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right). Green = kidney, and Blue = tumor



(d) Overlayed segmentation on the volumetric CT image. Red = kidney, Blue = tumor

Fig. 4: Predicted testing segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right). Green = kidney, and Blue = tumor

IV. Conclusion

In conclusion this model shows promising potential, however gives poor initial results due to time and computational constraints. In order to realise the full potential of the model, a much larger training and validation set should be used with at least 100 epochs. Furthermore, having access to higher computational power will allow for a higher number of epochs and the full breadth of the dataset, which is 300 cases, compared to the 13 total cases that were examined in this project. In terms of neural network architecture, it would remain to be 3D UNet, as it seems that the literature shows not a significant statistical difference between frameworks to reconstruct and build new neural networks. Overall, this project was successful in covering the basic to intermediate knowledge required to construct, train, and optimise neural networks, and this knowledge will be carried forward for more relevant projects in the future.

REFERENCES

[1] Canadian Cancer Society, "Kidney cancer: Kidney cancer statistics". [Online]. Available: https://www.cancer.ca/en/cancer-information/cancer-type/kidney/statistics/?region=ab [Accessed Oct. 27. 2020].

[2] A. Ghandi, "How to use Deep Learning when you have Limited Data". [Online]. Available: https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/ [Accessed Dec. 5, 2020].

[3] J. Brownlee, "Dropout Regularization in Deep Learning Models With Keras". [Online]. https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/: :text=Dropout. [Accessed Dec. 5, 2020].

[4] N. Mahdavifar, M. Mohammadian, M. Goncheh, and H. Salehiniya, "Incidence, Mortality and Risk Factors of Kidney Cancer in the World", World Cancer Research Journal, vol. 5, no. 1, 2018. [Online serial]. Available: https://www.wcrj.net/wp-content/uploads/sites/5/2018/03/e1013-Incidence-Mortality-and-Risk-Factors-of-Kidney-Cancer-in-the-World.pdf. [Accessed Oct. 27, 2020].

[5] World Cancer Research Fund, "Kidney cancer statistics". [Online]. Available: https://www.wcrf.org/dietandcancer/cancer-trends/kidney-cancer-statistics. [Accessed Oct. 27, 2020].

[6] Horwich, P.J. "Adrenal Adenoma Imaging". [Online]. Available: https://emedicine.medscape.com/article/376240-overview. [Accessed Dec. 5, 2020].

[7] Mayo Clinic, "Kidney Cancer". [Online]. Available: https://www.mayoclinic.org/diseases-conditions/kidney-cancer/diagnosis-treatment/drc-20352669. [Accessed Oct. 27, 2020].

[8] Y. Homma, "Increased incidental detection and reduced mortality in renal cancer recent retrospective analysis at eight institutions". International Journal of Urology vol. 2, no. 2, pp. 77-80, 1995.

[9] O. Cicek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation". In Lecture Notes in Computer Science: 18th International Conference on Medical Image computing and Computer-Assisted Intervention. MICCAI 2015, Munich, Germany, October 5-9, 2015.

[10] F. Insensee and K. H. Maier-Hein. "An Attempt at beating the 3D U-Net", 2019, arXiv:1908.02182 [eess.IV].

[11] "Canny Edge Detection" [Online]. Available from: https://docs.opencv.org/master/da/d22/tutorialpycanny.html. [Accessed Dec. 5, 2020].

[12] D. Nikitenko. "Lecture Notes Fall 2020", [Online]. Available from: https://moodle.socs.uoguelph.ca/course/view.php?id=173. [Accessed Dec. 5, 2020].

[13] D. P. Kingma, and J. L. Ba. "Adam: a method for Stoachastic Optimization". Proceedings of 3rd International Conference on Learning Representations. ICLR 2015, San Diego CA, USA, May 7-9, 2015.

[14] J. Chang, M. Ye, X. Zhang, C. Huang, P. Wang, and C. Yao. "Brain Tumor Segmentation based on 3D Unet with Multi-class Focal Loss". Proceedings of 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2018, Beijing, China, October 13-15, 2018.

[15] C. I. L. Meija, R. R. Biswal, E. Rodriguez-Tello, and G. Ochoa-Ruiz. "Accurate Identification of Tomograms of Lung Nodules Using CNN: Influence of the Optimizer, Preprocessing and Segmentation". Proceedings of 12th Mexican Conference on Pattern Recognition, CMCPR 2020, Morelia, Michoacán, México, June 24-27, 2020.

[16] S. S. M. Salehi, D. Erdogmus, and A. Ghoplipour. "Tversky loss function for image segmentation using 3D fully convolutional deep networks". Proceedings of 8th International Workshop on Machine Learning in Medical Imaging, MLMI 2017, Quebec City, QC, Canada, September 10, 2017.

[17] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, and S.K. Warefield. "Tversky as a Loss Function for Highly Unbalanced Image Segmentation using 3D Fully Convolutional Deep Networks". Proceedings of 9th International Workshop on Machine Learning in Medical Imaging, MLMI 2018, Granada, Spain, September 16,2018.

[18] N. Abraham, and N. M. Khan. "A NOVEL FOCAL TVERSKY LOSS FUNCTION WITH IMPROVED ATTENTION U-NET FOR LESION SEGMENTATION". Proceedings of 16th International Symposium on Biomedical Imaging, ISBI 2019, Venice, Italy, April 8-11, 2019.

[19] S. R. Hashemi, S. S. M. Salehi, D. Erdogmus, S.K. Warefield, and A. Ghopliour. "Asymmetric Loss Functions and Deep Densely-Connected Networks for Highly-Imbalanced Medical Image Segmentation: Application to Multiple Sclerosis Lesion Detection". IEEE Access. vol. 7, no. 1, pp. 1721-1735, 2019.

[20] D. Mulle, and F. Kramer. "MIScnn: A Framework for Medical Image Segmentation with Convolutional Neural Networks and Deep Learning", 2019, arXiv:1910.09308 [eess.IV].

[21] M. Yaqub, J. Feng, M. S. Zia, K. Arshid, K. Jia, Z. U. Rehman, and A. Mehmood. "State-of-the-Art CNN Optimizer for Brain Tumor Segmentation in Magnetic Resonance Images". Brain Sciences. vol. 10, no. 7, pp. 427, 2020.

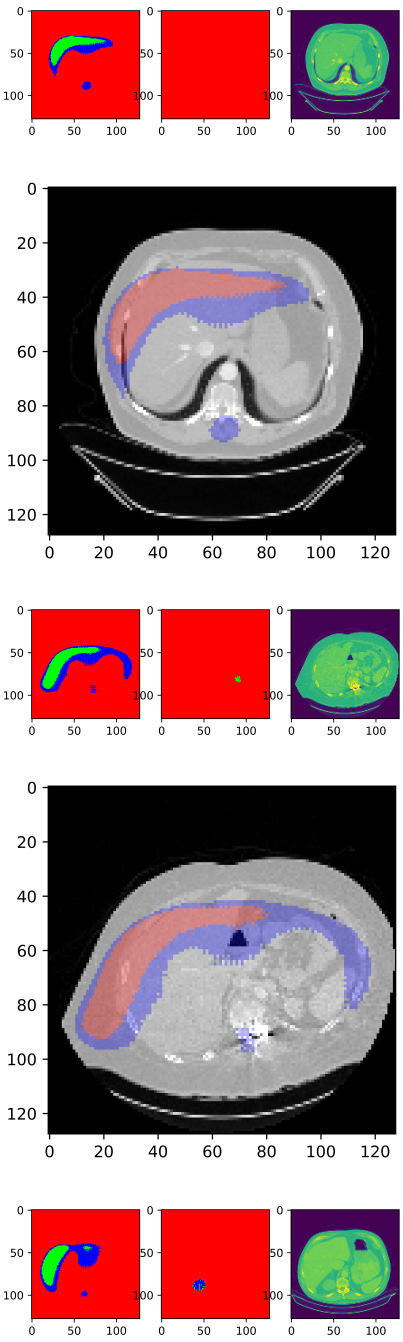
[22] A. Mortazi. "Optimization Algorithms for Deep Learning Based Medical Image Segmentations Segmentation". PhD Dissertation, Dept. of Computer Science, Univ. of Central Florida, Orlando, Florida, USA.

[23] X. Zhou, R. Takayama, S. Wang, X. Zhou, T. Hara, and H. Fujita. "Automated segmentation of 3D anatomical structures on CT images by using a deep convolutional network based on end-to-end learning approach". SPIE Medical Imaging, 2017, Orlando, FL, USA, February 11-16, 2017.

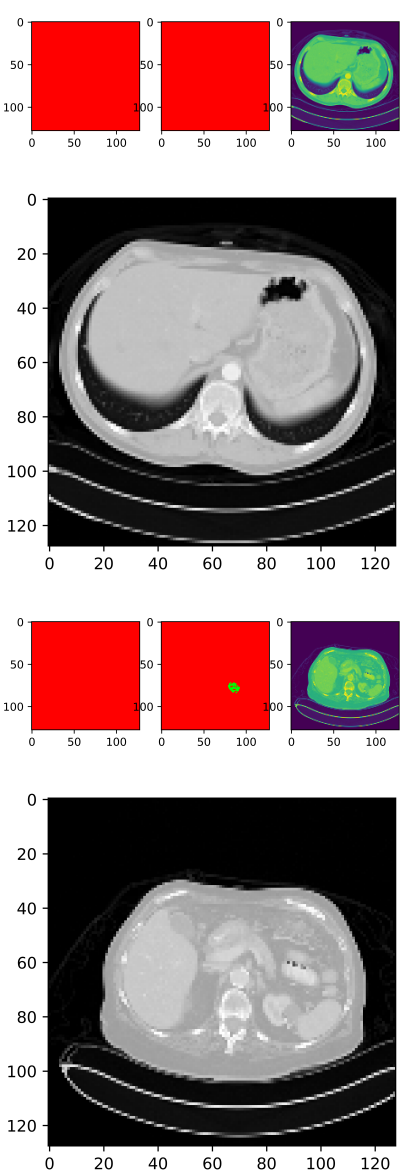
[24] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", 2015, arXiv:1505.04597 [cs.CV].

V. Appendix

Predicted training segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right); Red = background, Green = kidney, and Blue = tumor; along with overlayed segmentation on the volumetric CT image. Red = kidney, Blue = tumor



Predicted validation segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right); Red = background, Green = kidney, and Blue = tumor; along with overlayed segmentation on the volumetric CT image. Red = kidney, Blue = tumor



Predicted testing segmentation (left) compared to ground truth segmentation (middle) and original volumetric CT scan image (right); Red = background, Green = kidney, and Blue = tumor; along with overlayed segmentation on the volumetric CT image. Red = kidney, Blue = tumor