Theses and Dissertations
Graduate School

2021

A Deep Learning U-Net for Detecting and Segmenting Liver Tumors

Vidhya Cardozo
Virginia Commonwealth University

www.manaraa.com

A Deep Learning U-Net for Detecting and Segmenting Liver Tumors

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at Virginia Commonwealth University.

by

Vidhya Cardozo

Bachelor of Science in Biophysics, University of Mary Washington, May 2019

Advisors: Laura Padilla and Lulin Yuan

Professors, Department of Medical Physics

Virginia Commonwealth University

Richmond, VA

August, 2021

Acknowledgement

I would like to thank my entire thesis committee for their guidance and willingness to collaborate as needed. I would like to especially thank my advisors Dr. Padilla and Dr. Lulin for their immense patience, encouragement and for always providing suggestions. Finally, I would like to thank my friends and sister for coding advice, debugging support, and just generally listening to my ideas.

Table of Contents

List of Figures

List of Tables

Abstract

Deep Learning Algorithm (Modified U-Net) for Detecting and Segmenting Liver Tumors

by

Vidhya Cardozo, B.S.

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Medical Physics

Virginia Commonwealth University

Summer 2021

Visualization of liver tumors on simulation CT scans is challenging even with contrast-enhancement, due to the sensitivity of the contrast enhancement to the timing of the CT acquisition. Image registration to magnetic resonance imaging (MRI) can be helpful for delineation, but differences in patient position, liver shape and volume, and the lack of anatomical landmarks between the two image sets makes the task difficult. This study develops a U-Net based neural network for automated liver and tumor segmentation for purposes of radiotherapy treatment planning. Non-contrast simulation based abdominal CT axial scans of 52 patients with primary liver tumors were utilized. Preprocessing steps included HU windowing to isolate livers from the scan and creating masks for liver and tumor using the radiotherapy structure set (RTSTRUCT) DICOM file, and converting the images to a PNG format. The RTSTRUCT file contained the ground truth contours that were manually labelled by the physician for both liver and tumor. The image slices were split into 1400 for training and 600 for validation. Two fully convolutional neural networks with a U-Net architecture were used in this study. The first U-Net segments the livers. The second U-Net segments the tumor from the liver segments produced from the first network. The dice coefficient for liver segmentation was 89.5% and the dice coefficient for liver

7

tumor segmentation was 44.4%. The results showed that the proposed algorithm had good performance in liver segmentation and shows areas for improvement for liver tumor segmentation.

1. Introduction

Cancer is the second leading cause of death worldwide.[1] According to the data from World Health Organization (WHO), liver cancer was the third most common cause of cancer death in 2020.[1] Liver cancers are categorized into primary or secondary malignancies. Primary liver cancer starts within the liver tissue, whereas secondary or metastatic liver cancer develops when cancer cells from another organ spread to the liver through the bloodstream or the lymphatic system.[2] Hepatocellular carcinoma (HCC) is the most common type of primary liver cancer, accounting for approximately 80% of the cases. HCC is statistically much more likely to occur in people who have severe liver damage due to alcohol abuse, hepatitis B and C, and fatty liver disease caused by obesity.[2] The survival rates depend on the stage of the cancer at the time of diagnosis; however, the reported survival rate for all comers is 10% at five years.

Liver cancer, unlike most other malignancies, can be diagnosed based on computed tomography (CT) or magnetic resonance imaging (MRI) findings. A CT scan gives detailed cross-sectional images of the abdominal region. Further processing of the scan is needed to segment the liver and distinguish the tumorous areas from the rest of the CT scan. However, segmentation of the tumor is still a difficult task due to the intensity similarity between the tumor and other tissues in the abdominal CT scan. As a result, the images need to be processed and enhanced to differentiate the cancerous tissue.

Figure 1: Shows the gray level similarities between liver and nearby organs (spleen) even after applying Hounsfield windowing.

9

Recently, there have been many developments in medical imaging that have allowed medical professionals to greatly improve diagnostic and radiation treatment delivery techniques, leading to earlier detection and better treatment of cancers.[3,4] Several imaging procedures with their own advantages and disadvantages have been utilized. These include CT, MRI, positron emission tomography (PET), ultrasound (US), etc. Liver diagnosis usually utilizes CT, MRI or US for characterization of HCC. For treatment planning, on the other hand, CT scans are typically the imaging modality of choice. In this research, simulation CT scans have been used for liver and liver tumor segmentation given their prevalence in radiotherapy treatment planning. [4]

1.1. Radiation Therapy Process

The radiation therapy (RT) process depends on the patient's treatment needs and the type of program their physician chooses. There are usually five steps, including initial consultation with the patient, simulation to localize the target and manage respiratory motion, creating a treatment plan, delivering the treatment, and following-up with the patient after treatment. Provided below are the simulation for radiation therapy including motion management strategies, and planning procedures.

1.1.1. Simulation

The purpose of simulation is to ensure patient immobilization, control or account for tumor motion, and acquire data for treatment planning and image guided treatment delivery. A RT simulation serves to provide visualization of the patient's anatomy and gather three-dimensional data while the patient is in the treatment position.[5] This data acquired is used for designing a treatment plan that will result in appropriate dose coverage of the target while limiting the dose given to adjacent organs at risk (OAR), as dictated by various target coverage requirements and OAR constraints established through clinical trials and other studies. Some commonly used scanning methods in the simulation are MRI, CT, and PET scans. The use of a particular type of scan is dependent on the characteristics of the tissue that is being scanned. Although, CT scan is the most common one, MRI and PET often complement CT scans.

A CT scan utilizes radiation to acquire a volumetric image representation of the patient's anatomy.[6] The acquired information is processed by a computer to construct images that are stored usually in a digital imaging and communications in medicine (DICOM) format. Frequently, contrast agents are used to highlight blood vessels and other regions of interest to achieve better results. In the case of HCC, intravenous contrast timed to arterial phase is important to highlight HCC as it feeds from the arterial blood supply in the liver. Hyper-vascular HCC will be clear in the late arterial phase versus hypo-vascular HCC will be enhanced poorly in the late arterial phase. As can be seen, there are many difficulties with using contrast medium for imaging, including choosing the type and amount of contrast agent, route of administration, and sensitivity of the contrast agent to the time of image acquisition (the tumor will not be highlighted unless the timing of the contrast agent is appropriate).[7]

Even with the use of contrast agents for acquiring a simulation scan, the possibility of misrepresentation of the tumor still exists due to artifacts induced by respiratory motion. In the case of abdominal imaging for identifying and contouring regions of interest for treatment planning, respiratory organ motion causes severe geometric distortion in free breathing CT scans. The extent of distortion along the patient's longitudinal axis can expand or shorten the target. As a result, we need techniques that will aid in reducing the motion artifacts.[8]

1.1.2. Respiratory Motion Management Techniques

Geometric distortions in the scan due to breathing motion can be minimized by using respiratory motion management techniques. This is particularly important for hypo-fractionated treatments such as SBRT, commonly used to treat liver tumors as described later. Some of the categories of methods used in the management of respiratory motion include motion-encompassing methods, respiratory gates techniques, breath-hold techniques, forced shallow-breathing methods and respiratory-synchronized techniques.[8] At VCU, patients are first simulated using free breathing scans. In patients whose breathing amplitude exceeds 5mm, a breath-hold technique utilizing active breathing control (ABC) is attempted. The 4D-CT scan to account for breathing related motion and the ABC technique are described below:

11

1) Four-dimensional computed tomography (4D-CT)

There are two image binning approaches available for 4D-CT imaging: phase binning and amplitude binning. The approach that is commonly used at VCU is the phase binning approach, which will be described here. In a 4D-CT, the scans are acquired continuously during respiration. The scans acquired are reconstructed at specific phases of the breathing cycle for each patient location. The data at the same phase is combined from multiple breathing cycles. The overall result is a series of three-dimensional CT images depicting a different phase in the breathing cycle. The phase at a point on the respiratory cycle is defined as the amount of cycle in percentage that has elapsed compared to the beginning of the cycle, and at each phase, multiple CT volumes are taken. The breathing cycle is divided into ten respiratory phases and ranges from 0% phase from full inspiration to 50% phase from full expiration and back to 100% phase from full inspiration. The position of the abdominal surface and diaphragm act as surrogates for the respiratory phase.[9]

2) Active breathing control (ABC)

Active breathing control (ABC) is a method developed by the radiation oncology department at William Beaumont hospital for the purposes of achieving reproducible breath-hold without reaching maximum inspiration.[10] The goal of this technique is to reduce tumor motion or in case of mostly left-sided breast cancer treatments, to reduce the dose to the heart. In this technique, a mouthpiece connected to the breathing tube is placed between the teeth with the patient's lips tightly grasping the mouthpiece. A nose clamp is placed on the patient's nose to prevent accidental breathing through the nose. The patient is then instructed to take deep breaths, and once the breathing pattern becomes stable, an optimum volume or phase in the breathing cycle is determined. Air volumes with breathing are displayed as a breathing trace by the system, and the appropriate patient specific breath-hold duration (typically 15-30 seconds) and level are determined during simulation.[10,11] The air volume threshold for the breath-hold will be determined based on the patient-specific data and on the breath hold type (exhale, inhale, deep inhale, etc.).

12

When the patient breathes to the predetermined volume, the patient is asked to hold their breath. A small valve closes in the breathing tube to prevent additional air from entering the lungs during the breath-hold duration. During this time, CT images are acquired with minimal respiratory motion artifacts. A timer embedded into the ABC system is used to count down the remaining duration of breath-hold in seconds.[11]

1.1.3. Treatment Planning

The primary process of treatment planning involves determining the exact target/tumor volume that will be treated, defining radiation dose, and designing radiation fields for treatment delivery. The main focus here is to ensure that the target receives the maximum dose, while reducing the dose to organs at risk (OAR). In order to achieve this goal, it is important that we clearly define our target. This includes figuring out the extent of the disease in all the slices, accounting for patient movement and setup uncertainties. The volumes defined by the International Commission on Radiation Units and Measurements (ICRU) 50 and 62 reports are described below, and shown in Figure 2.[12,13]

In the treatment planning, it is critical to evaluate the coverage of the target being treated and the risk to nearby organs to avoid complications. This is done by contouring the targets and all the OAR. The clinical target volume (CTV) is usually proportional to the gross tumor volume (GTV) which allows for a decreased risk to the normal tissue surrounding the tumor. Alternatively, the adjustments suited for tumor motions are done by extending CTV to ITV.[12,13] The three important volumes are gross tumor volume (GTV), internal target volume (ITV) and planning target volume (PTV), and are described below.

i.   Gross Tumor Volume (GTV)

GTV measures the spread and the location of the tumor. It may contain the primary tumor and the metastases. It is possible to delineate the GTV through imaging, depending on the visibility, location, and tangibility of the tumor. If the tumor has been removed, GTV cannot be

13

defined, although preoperative and postoperative images can allow to visualize the outline of the tumor bed.[12,13]
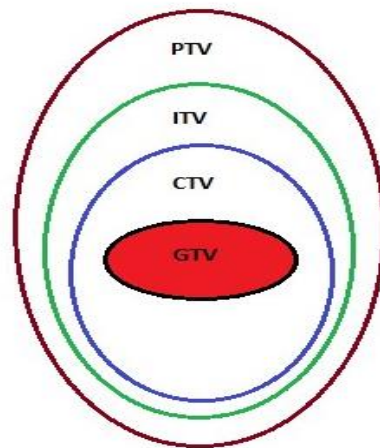
Figure 2: Shows the ICRU volumes used for treatment planning.[12,13]

ii. Clinical Target Volume (CTV)

CTV contains the present tumor and any other microscopic disease that might be malignant or tissue that needs to be eliminated. This overestimate allows for a true representation of the spread and location of the tumor. The CTV delineation assumes that no tumor cells exist outside the CTV boundary. It is critical that the CTV receives sufficient dose for proper treatment.[12,13]

iii. Internal Target Volume (ITV)

ITV is the internal margin (IM) added to the CTV. It accounts for the changes in position, shape and size of the CTV during treatment due to the internal movements resulting from respiration, bladder/bowel contents, etc.[12,13]

14

[ITV = CTV + IM]

    iii.   Planning Target Volume (PTV)

PTV consists of the CTV along with an internal margin (IM) and incorporates a setup margin (SM) to account for any uncertainties in treatment setup and changes in the patient movement. The CTV margin in all directions must account for any possible movements inside the body as well as any possible patient movement and setup inaccuracies.[12,13]

[PTV= CTV + 5 mm for SBRT]

Despite the advances in imaging, liver tumor segmentation on CT for treatment planning still remains a challenging task. This is mainly due to the fact that the liver organ shape and volume differs for each patient, Secondly, as mentioned briefly earlier, liver tumors vary in shape, appearance and location, and in general, there are slight visible differences between normal and tumor tissues.[14] Furthermore, the appearance of each tumor lesion depends on the imaging equipment and settings, and thus may vary for each patient and institution.

In order to overcome this issue and aid radiation oncologists in identifying the GTV, other imaging studies can be used to supplement the information provided by the simulation CT scan. MRI is often used for this purpose because of its superior soft tissue contrast. An image registration between CT and MRI can be used for accurate radiotherapy planning and delivery, however, the lack of calibrated intensity scale, electron density, MRI distortion and other artifacts in MRI and CT makes it challenging to register the images.[15] Image registration can be divided into two types: rigid image registration (RIR) and deformable image registration (DIR). RIR is typically used when there are no anatomical changes expected. In the case of liver and liver tumors, the major disadvantage of the RIR is that it only accounts for whole organ motion of the liver between CT and MRI datasets, but it does not compensate for the deformation caused by breathing, changes in weight, tumor shrinkage and so on.[16]

Deformable image registration (DIR) can be done using an MRI. This process involves calculating the geometric transformation between the CT and MRI, and then mapping them into a

15

common coordinate system. It is called deformable or nonlinear as the transformation includes both rigid transformations such as translation and/or rotation, as well as deformations such as shrinking and/or stretching. However, DIR has certain disadvantages. It is usually difficult to estimate the accuracy of registration when there are no landmarks. It is generally difficult to define anatomical landmarks in some organs, and effective use of large deformations between images with large anatomical differences is limited. Additionally, more complications arise in registration from using images with large differences in appearances from multiple imaging modalities.[15,16]

To understand the importance of correctly identifying the volume of interest, it is beneficial to know about the common type of radiation therapy for treating liver tumors, which is described in the following section.

1.2. Stereotactic Body Radiation Therapy (SBRT)

Stereotactic Body Radiation Therapy (SBRT) is a form of stereotactic radiation therapy procedure that delivers high doses to the target in 5 or less fractions. This treatment modality is typically used for tumors in the liver, lung, pancreas and spine, amongst others. Since SBRT delivers a high dose per fraction, it is important to conform dose to the target and create a rapid dose fall off outside of it to prevent damage to adjacent normal tissue. It needs to be ensured that any foreseen motion of the tumor is taken into account during planning and that the dose is delivered accurately.[5] Hence, respiratory motion management techniques are used for liver SBRT treatments, and other tumors located in the thorax and abdomen, where breathing can lead to tumor location changes. Additionally, it is necessary to properly delineate target volumes such as GTV, ITV and PTV. If the GTV is not delineated accurately then it will affect the ITV and henceforth the PTV as the ITV and PTV are margins added to the GTV to account for motion (in this case, respiratory motion) and setup uncertainties respectively. This will in turn result in the patient being delivered an excess dose to the surrounding normal tissue, or patient not receiving enough dose to adequately cover the target volumes.

16

1.3. Neural Networks

With the advancement in computer vision technology, machine learning and deep learning became common for segmentation problems. In recent years, several deep learning models have been developed for liver tumor segmentation. Convolutional neural networks (CNN) are used commonly today for this purpose. As described in the literature review section below, one of the successful types of CNN is a U-Net based on previous studies conducted for liver and liver tumor segmentation. Therefore, it is beneficial to explain both CNN and U-Net Architecture

1.3.1 Convolutional Neural Network (CNN)

One of the subfields of machine learning is deep learning (DL). DL utilizes layers of representations. The number of layers that are utilized in the model of the data are represented as the depth of the model. These structured representations or layers built upon each other are learned using neural networks. The idea of computational neural networks arose from the biological neurons, where messages are transferred from one neuron to another. Although deep learning requires understanding of the brain, the DL models are not actually part of the human brain. Rather, deep learning serves as an algorithmic framework for learning hierarchy of data.[17]

Before diving more deeply into neural networks, it is necessary to become familiar with two common types of machine learning. First type of learning is called supervised learning. In this type of learning, a model is given an input and an output (or a label). The algorithm learns or makes connections between the given data and is able to apply the knowledge to new data. Supervised learning is most commonly used in machine learning applications. The second type of learning is unsupervised learning. In unsupervised learning, the algorithm learns to find information and hidden patterns from the given unlabeled input data.[17]

Furthermore, it is also beneficial to understand how the computer analyzes images. But first, let us see how humans interpret images. When a human sees an object, the primary visual

17

cortex interprets the signal sent by the light receptors in the eye. It does this by increasing complexity- the first layer differentiates lines, edges and curves. As the layers increase, the brain starts recognizing colors and combines previous information, and is thus able to tell if the image is, for example, a cat or a butterfly.[18]

However, the computer sees an image as an array of numbers (0s and 1s). An image is stored as a combination of pixels, usually a 3-D pixel array. A pixel contains a different number of channels. If the image is grayscale, then it has only one pixel, but if the image is colored, it contains three channels- red, green and blue. For example, red would be (255, 0, 0) and green would be (0, 255, 0). The computer makes sense of the input (image) just like humans as described earlier. First, it detects simple edges and then recognizes complex information, thus learning spatial hierarchies of patterns. This forms the basis of machine learning and computer vision applications.[18],[19]

Now that we know how visual images are processed by computers, it is time to dig deeper into neural networks, specifically the CNN. A CNN is a type of DL model used in computer vision applications. It takes in an input image and assigns importance using weights and biases to differentiate various components in an image. CNN have gained popularity due to their faster processing time and higher generalization when there are few training images. As stated by Chollet in his book, the two main characteristics that make CNN interesting are: the patterns CNN learn are "translation invariant", meaning that the CNN can recognize a pattern anywhere in the image that is learnt from one part of the image. Second characteristic is the network's ability to learn "spatial hierarchies of patterns".[17]

The key characteristic of CNN is the convolution operation. The convolution operation is a linear operation that multiplies an array of input (image) by a set of weights, called a kernel or a filter just as in a regular neural network. Convolutions operate using feature maps, which result from application of the kernel to the input repeatedly, thus creating a map of activations (or feature map).[20]

Much like the convolutional layer, the pooling layer is used for lowering the spatial size of the feature map. This helps to reduce the computational power by the reduction of dimensionality for data processing. Additionally, it also helps in filtering dominant features that are perpetual in their rotation and position, thus allowing to train the model effectively. Max pooling and average pooling are the two types of pooling techniques. Max pooling returns the largest value from the section of the image surrounded by the kernel, and average pooling returns the average value from all the values from the section of the image surrounded by the kernel.[20,21,22]

Max pooling acts as a noise suppressant by eliminating the activations that are noisy and also helps in downsampling by reducing dimensionality, which thereby aids in noise suppression. Contrary to the max pooling, average pooling only helps in downsampling, as a mechanism for noise suppression. Hence, it can be concluded that max pooling is a better pooling technique.

After applying a pooling layer, the feature map needs to be flattened. This process transforms the whole pooled matrix into a single feature column. It performs this by getting the output from previous convolution layers and then flattening the structure into a single vector for classification.[17,21]

The final classification is performed using fully connected layers. The fully connected layers have connections to every activation in the other layers, and therefore a final output is created from data obtained from previous layers.[17]

An activation function is needed to aid the network in learning complex patterns from the training dataset and it also determines the type of predictions the model can make. Without an activation action, a neural network has limited learning power. A commonly used activation function for hidden layers is a rectified linear activation function (ReLU). ReLU helps models to perform better by overcoming the vanishing gradient problem. It returns 0 if it receives a negative input and returns the same value if it receives a positive input. The choice of activation function for output layers depends on the expected output (for example: sigmoid is used for binary classification).[23]

19

Finally, in order to ensure that the network trains properly, it needs to have a loss function, an optimizer and parameters that need monitoring during training and testing. A loss function determines the performance of the network on the training data and basically serves to direct the network in the right direction accordingly. An optimizer is a way the network updates itself based on the information acquired from the loss function.

Furthermore, to ensure best results, tuning of hyperparameters is recommended. The hyperparameters determine the network structure and how well the model can train.

In summary, the aim of a machine learning or a deep learning model is to generalize predictions to new data. CNN are considered one of the best models for computer vision applications as they can train on small datasets and obtain good results. The basic steps involved in a convolutional neural network as can be seen in Figure 3 are choosing an input (and labels for supervised learning), applying a convolution operation and using an activation function, applying a pooling layer, flattening, applying a fully connected layer, compiling CNN by using loss function and optimizer, fitting the CNN and finally obtaining the desired output.
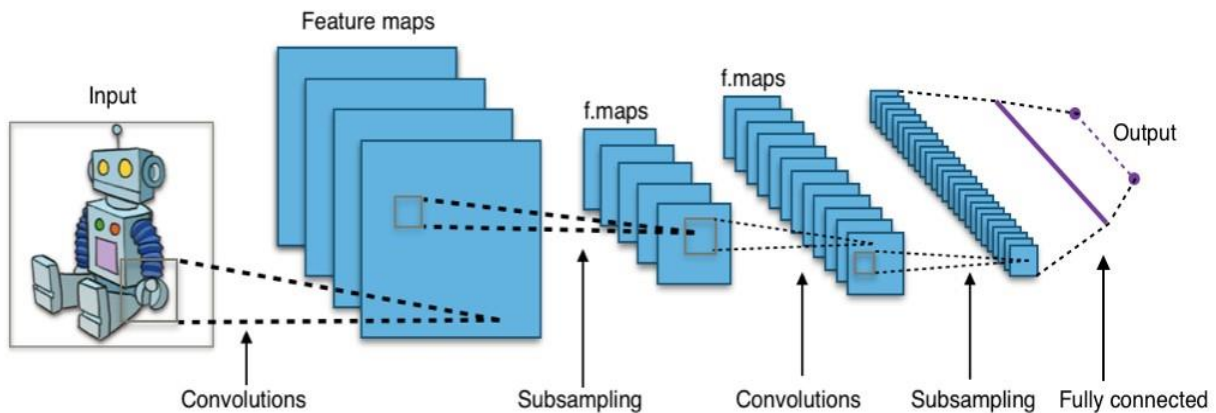

Figure 3: Depicts the architecture of a typical convolutional neural network.[24]

2.   Objective

Visualization of liver tumors on simulation CT scans is challenging even with contrast-enhancement, due to the sensitivity of the contrast enhancement to the timing of CT acquisition.

20

There is a need for tools that will aid in accurate localization to help physicians in contouring target regions, thus facilitating better treatment planning and delivery, and decreasing toxicity to nearby structures. The goal of this research is to develop a neural network that will use simulation based non-contrast CT scans for automated liver and liver tumor segmentation.

3. Literature Review

As mentioned earlier, liver tumor segmentation is a difficult task due to the liver having CT values close to other nearby organs, low contrast between tumor and normal liver tissue, and differences in shape and size of the liver tumors. Given the increasing incidence and emerging role of liver SBRT, there is a need for better automatic liver and liver tumor segmentation algorithms.

In recent years, several types of segmentation algorithms have been developed for automatic liver and liver tumor segmentation.[25,26,27,28] In 2017, a liver segmentation challenge was organized by the liver tumor segmentation benchmark (LiTS).[26] In summary, the data for this challenge was acquired from various clinical sites around the world, and the dataset contained a total of 200 contrast-enhanced CT scans of diagnostic quality with hyper/hypo intense tumor contrast levels and varying amount of lesions. The CT scans had reference annotations (ground truths) manually labelled by radiologists. The training data contained 130 scans and the test data contained 70 scans. The first ranked model in this challenge was built by Han , who used two models with U-Net like architecture consisting of long and short skip connections.[27] The first model was used to segment liver regions as input to the second network, and the second model was trained to segment liver and tumor in the same step. The two models worked in 2.5D by taking a stack of adjacent slices to produce segmentation of the center slice. The average Dice coefficient score was 67% for liver lesion segmentation.[27] Other top-scoring models also used convolutional neural networks with a U-Net-like architecture.[29]

In another study performed by Meng et al.[28], the researchers used a three-dimensional dual path multiscale convolutional neural network (TDP-CNN) for liver and tumor segmentation.[28] They used a dual path to balance the performance of the segmentation network, and make efficient

21

use of computational resources. The feature maps were combined from both paths at the end. In this study, the publicly available contrast-enhanced CT dataset from the liver tumor segmentation (LiTS) was utilized to analyze the proposed model. Liver and liver tumor labels manually labelled by a radiologist were used. For both the segmentation results of the liver tumor and the liver, the quantitative metrics used were Hausdorff distance, Dice, and the average distance. The dice coefficient results for liver and tumor were 96.5% and 68.9% respectively.[28]

A similar study conducted by Alirr[14] also used a fully convolutional neural network with region-based level set function. The proposed framework in this study consisted of three steps: first to use a U-Net FCN to segment liver, second to use another U-Net to segment liver tumors, and then apply localized level set on predicted masks for enhancement to get final liver and tumor masks. Localized level set-based technique was utilized to refine and match the liver and tumor boundaries in the predicted segmentations.[14]

The accuracy of this proposed method was compared against the two datasets: LiTS and 3D image reconstruction for comparison of algorithm databases (3D-IRCAD). The 3D-IRCAD dataset consisted of 20 venous phase-enhanced CT volumes of diagnostic quality, and also contained manual labels created by clinical experts. The number of slices for each patient ranged between 148-260, and the in-plane resolution for this dataset was between 0.56-0.86 mm and inter-slice spacing was between 1.0-4.0mm. The Dice scores for liver segmentation in the LiTS dataset was 95.6% and liver tumor segmentation was 70%. The Dice scores for liver segmentation in the IRCAD dataset was 95.2% and liver tumor segmentation was 76.1%. Alirr[14] concluded that the proposed network performed well for CT scans from various scanners, thus serving as a promising tool for automation methods in liver and liver segmentation in clinical use.[14]

4.   Methodology

The first part of the process was gathering patient data, and identifying and selecting the data to be used based on the type of disease and type of imaging used. The next part of the process included cleaning the selected data to have homogeneity, selecting an appropriate convolutional neural network architecture that would meet the purposes of this project, and then finally

22

evaluating the model's performance and attaining visuals. All of these parts are described later in this section.

Figure 4 shows the proposed workflow for training the network. First, CT volumes were preprocessed using HU windowing and then masked to get liver and tumor ground truths. The data was augmented using rotation and flipping to artificially create more images. The first U-Net segments the liver from the preprocessed and augmented abdominal CT scans. The second U-Net segments the liver tumor from preprocessed and augmented liver segments.
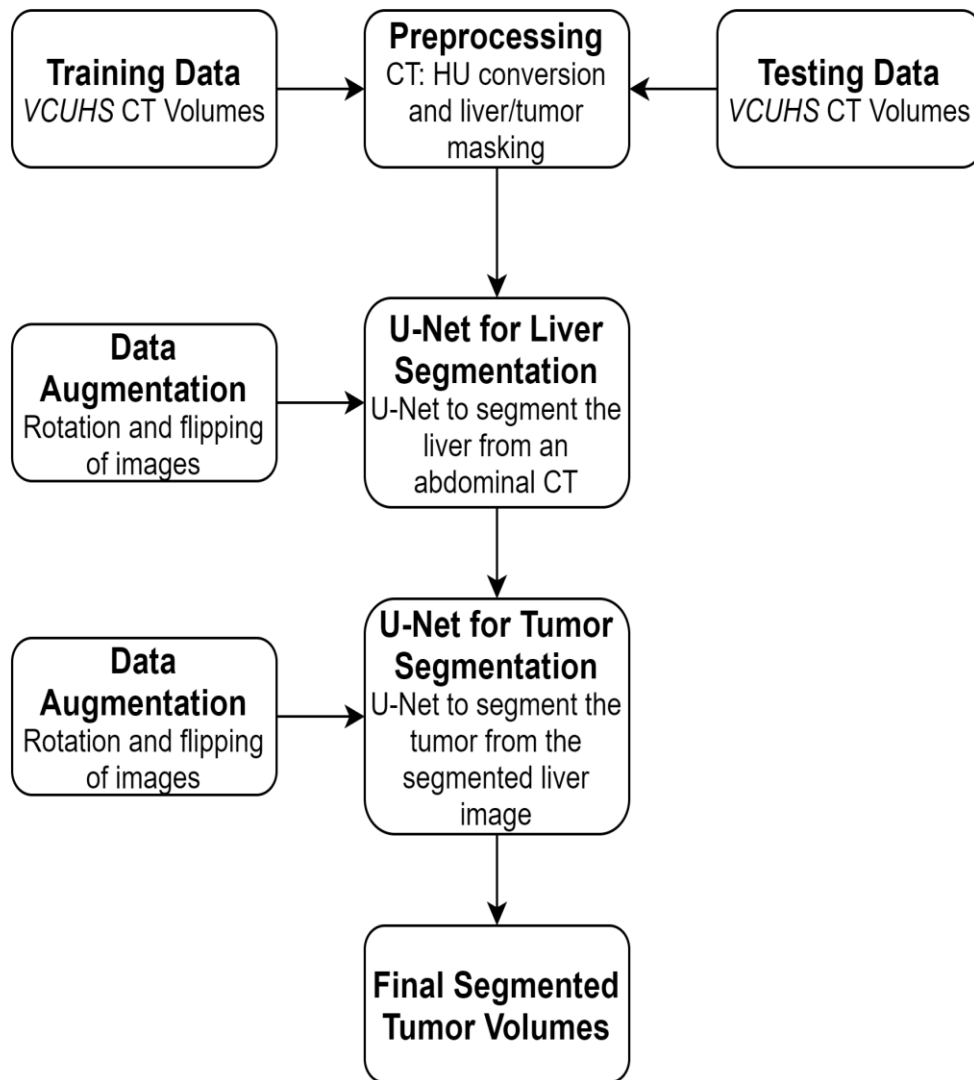

Figure 4: Represents the liver and tumor segmentation workflow for training.

4.1. Patient Data

The images used in this research were obtained from the Virginia Commonwealth University Health System (VCUHS) under institutional review board (IRB) approval. The anonymized images were in DICOM format and were downloaded using the software MIM. Fifty-two patients (Table 1) with HCC treated with SBRT at VCU between 2013 and 2020 were identified. The images consisted of non-contrast simulation abdominal CT scans acquired using either ABC or free breathing. 4D-CT technique was used for free breathing scans. For the 4DCT set, the 30% phase was selected as this is the mid-ventilation phase used for planning as it represents the average target position during the respiratory cycle. Each scan covered the entire abdomen and chest cavity. The dataset also contained several contours of relevant structures--most importantly in this case the contours for liver, planning target volume (PTV), internal target volume (ITV), clinical target volume (CTV), and gross tumor volume (GTV) were stored as coordinates in a radiotherapy structure set (RTSTRUCT) DICOM file.

These contours were generated by the radiation oncology team at VCU and thus might have variability in contouring depending on the creator of the contours. ITV was the chosen structure for ground truth because in an SBRT treatment, a high dose is delivered in a few fractions. ITV encompasses a whole range of tumor and tumor motion due to respiration. A few patients did not have an ITV structure and thus CTV was used in this case.

Total number of HCC patients	52
Original Image Size (px)	512 x 512
Slice Thickness (mm)	3.0
Pixel Spacing Range (mm)	0.95 - 1.36
Type of CT Scanner	Philips (Brilliance Big Bore)

Table 1: Shows VCUHS data characteristics used in this research

24

4.2. Computational Environment

The program was implemented using Python as the programming language and executed using Google Colab, which requires mounting of the Google drive and importing the required packages. Dicomplyer was used to visualize DICOM CT volumes and associated radiotherapy structures. For more details on the versions of each software, refer to Table 2 below.

Environment	Configuration
GPU	Google TPU
Operating System	Ubuntu 18.04.5 LTS
RAM	36 GB
Software tools	GoogleColab; Python 3.7; Dicomplyer 0.4.2

Table 2: Summarizes the computational environment used for the research

4.3. Data Preprocessing

The program was given a folder consisting of an RTSTRUCT file and its corresponding DICOM CT volumes. The CT volumes were converted to pixel-array, and the CT images were enhanced using Hounsfield (HU) windowing. HU windowing was used to exclude irrelevant organs from CT scan images. The HU window in the range of [40, 80] was used to remove the soft tissues around the liver.

The program then iterates through the RTSTRUCT file which contains a list of an array of coordinates for different types of contours. The program searched for the liver contours. Each liver contour contains an array of coordinates that corresponds to a specific DICOM slice.

25

The next step was mapping the coordinates to the image and creating a liver mask. To do this, the program used the Python CV2 library to map out the coordinates, fill in the coordinates, and set the pixel value to be true or false (1 or 0). A copy of the original CT image was made, and then looped through comparing the pixel value to the masked value at the same coordinate. If the masked value was true, then the new image value was set to be the max pixel value of 255. Otherwise, if the value of the masked image pixel was false, then the final value was set to 0, or blank. The original size of the images was 512×512. For better handling and due to storage reasons, the images were resized to 128 x 128.

4.4. Data Augmentation

Data augmentation is a technique used to prevent the issue of overfitting, which is caused by having a few data to train or learn from, and as a result, the network is unable to generalize predictions to new data. Data augmentation is done by creating more data from existing data by applying various transformations such as flipping, padding, and cropping.[17]

In the proposed network, data augmentation was done by using rotation and horizontal flipping. The images were randomly rotated between -25 degrees and +25 degrees. This helps to increase the amount of data, and allows the network to be exposed to more data and therefore generalize better.

4.5. Network Architecture, Network Training and Validation

The U-Net architecture used in this research is based on the original U-Net created by Olaf Ronneberger as described earlier (Figure 5).[30] This architecture was used for liver as well as tumor segmentation. It is a fully convolutional neural network as it contains convolutional layers and no dense layers, and thus can accept images of any size. Figure 4 shows the structure of the U-Net. The major changes to the architecture in this research were the addition of dropout layer, change in learning rate, and addition of batch normalization in each path.

The drop-out layer randomly sets input units to 0 at a given rate. The benefit of adding this layer to the network is to prevent overfitting, which is a strong concern when using a small dataset. The decision to add more convolutional layers than the original U-Net design contains was made in an attempt to increase the accuracy of the network. Batch normalization was added to increase the speed and stability of the network by normalizing the layers inputs.
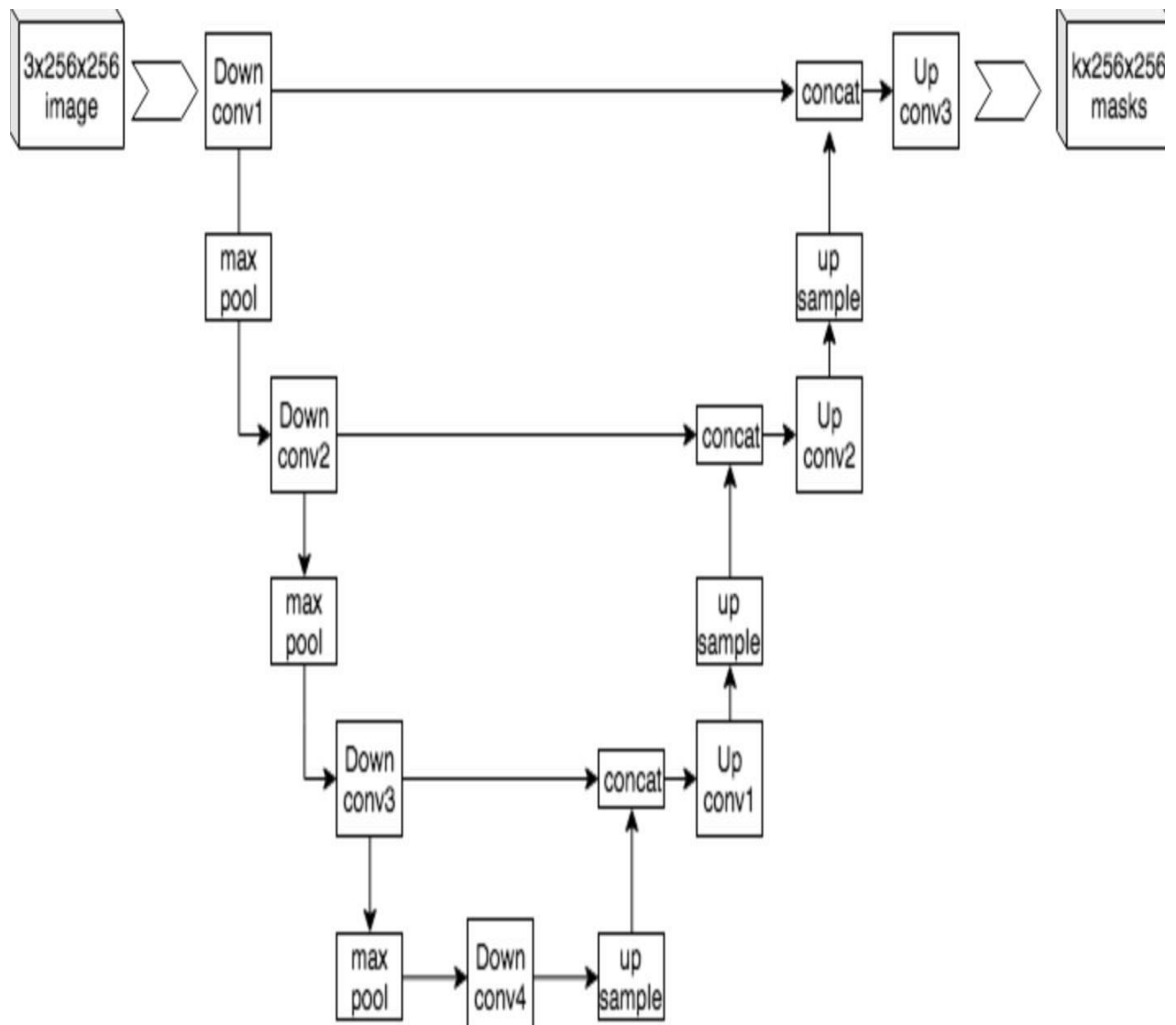
Figure 5: Shows an example U-Net architecture.[31]

The first part of the network is the downsampling path. This path serves to get the feature information of the image or 'what is in the image', while ignoring the spatial information. High level feature information from the image is obtained by reducing the size of the image. It extracts the feature information using the convolution and max pooling layers. In the proposed network, the downsampling path was divided into three blocks. The first block had a ReLU activation layer, followed by a convolution layer, and batch normalization. The second block contained the same components from the first block, but maxpooling layer of poolsize 3 and strides set to 2 was added. In the third block, a residual layer was added. The filter size increases by a multiplier of 2 in each of the three blocks. The first, second and third block had a filter size of 64, 128 and 256 respectively. A dropout layer of 0.25 was also added to minimize overfitting.

The final part of the network is the upsampling part. This path serves to acquire the location or spatial information of the image, and retrieve the original size of the image. It uses deconvolution layers and skip connections to transfer the information from the downsampling path to the upsampling path. In the proposed network, this path consisted of three blocks. The first and second blocks consisted of a ReLU activation layer, a transposed convolution layer (same as deconvolution layer) and batch normalization. An upsampling layer was also added to the second block. The third block consisted of a residual layer.

Furthermore, a Softmax function was used as the last activation function to normalize the output to a probability distribution over predicted output classes.

To thoroughly evaluate the model, it needs to be divided into training, validation and testing sets. The training set is used to train the model, the validation set is used to evaluate the model and the testing set is used to finally test the model.

The proposed network was trained using 2000 2D image slices, out of which 1400 were used for training and 600 were used for validation as shown in Table 3. The input image size was 128 x 128 x 3. Due to the addition of features mentioned earlier, the network had to be trained from scratch. The preprocessed CT images along with their liver masks were used to train the network to segment livers. Secondly, the liver images along with the tumor masks were used to

28

train the network to segment tumors. A batch size of 32 and 50 epochs were used to train the network.

Additionally, the loss function used in this study was sparse categorical cross entropy function. It is a metric used for evaluation of loss and accuracy of the model. Cross entropy measures the distance between ground truth and predictions. It checks to see if the maximum true value in the ground truth is equal to the index of the maximum value in the predicted mask.

The optimizer used in the proposed network was root mean squared propagation (RMSprop) for hidden units. This normalizes the gradients by using the magnitude of the recent gradients. The current gradient is divided over the root mean squared gradients, thus maintaining a moving average. RMSprop solves the issue of gradient vanishing as the normalization balances the momentum by decreasing or increasing the steps for large and small gradients respectively.

Parameter	Value
Input image size	128 x 128 x3
Epochs	100
Dropout	0.25
Learning rate	0.001
Batch size	32
Training Sample Size	1400
Validation Sample Size	600

Table 3: Summarizes the parameters utilized during the liver and tumor segmentation

29

4.6. Evaluation Metrics

For evaluating the performance of the liver and tumor segmentation, dice similarity coefficient (DSC) was used. The DSC is a method used to check similarity between two segmentations. It measures the size of the overlap between two data points (prediction and ground truth), and divides by the total size of the two points. The DSC value ranges from 0-1, where a value of 0 means there is no spatial overlap and a value of 1 means there is perfect overlap.[32]

Dice score The DSC is given by the following formula.[32]

$$\text{DSC} = \frac{2 * A \cap B}{A + B}$$

where A and B = Target masks and ∩ = intersection

5. Results

Two concatenated CNN based U-Nets were trained to segment livers and liver tumors from non-contrast enhanced abdominal simulation CT scans. The first network took input data of patient CT non-contrast abdominal CT images and liver masks and produced a liver segmentation. The second network took the liver segmentation output from the first network along with a tumor mask (ITV, CTV, GTV) and produced a tumor segment as the final result. The results from data preprocessing, network training/validation, liver segmentation, tumor segmentation, evaluation of liver segmentation and tumor segmentation are presented below.

5.1. Data Preprocessing

The first part of the thesis was preprocessing the non-contrast abdominal simulation CT images as described in the methods. The preprocessed images were saved in the PNG format. Figure 6 shows the HU windowing steps used for isolating the liver from the original CT image.
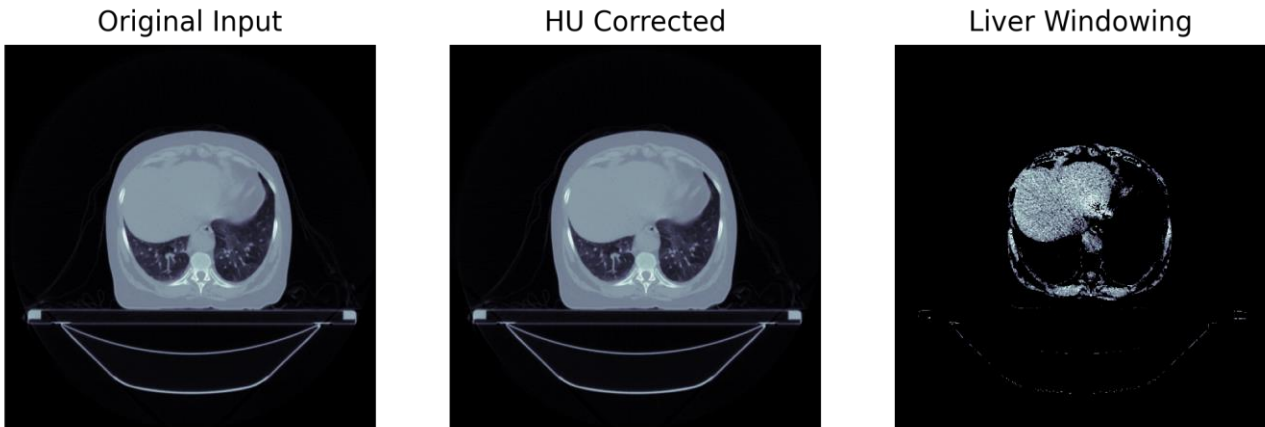
30

Figure 6: HU widowing steps. Any tissue below 40 and above 80 is removed to get an isolated liver.

5.2.   Network Loss and Accuracy

Sparse categorical entropy was used in this study for monitoring the accuracy of the model during training and validation. Figures 7 and 8 shows the sparse categorical loss and accuracy for liver and liver tumor segmentation respectively.

As can be seen in Figure 7, the liver model training loss and validation loss gradually decreases with every epoch which indicates a good learning rate. The liver model training and validation accuracy increases with every epoch. Using more than 100 epochs would be a good indication to see how the model behaves. However, it can also be noticed that the validation accuracy after about 40 epochs is less than training accuracy, which indicates that the model is slightly overfitting. This means that the model is fitting to the training data better, but has a harder time fitting to new data, and thus starting to overfit.

Similarly, as can be seen in Figure 8, the liver tumor model training loss and validation loss decreases with every epoch till it reaches epoch 40, and remains mostly constant after that. The liver tumor model training and validation accuracy increases with every epoch. Using more epochs to train the model would be helpful to decide how the model behaves past 100 epochs. The

31

model appears to not be overfitting as the training and validation accuracy graphs are close to each other, however, more epochs would be needed to decide.
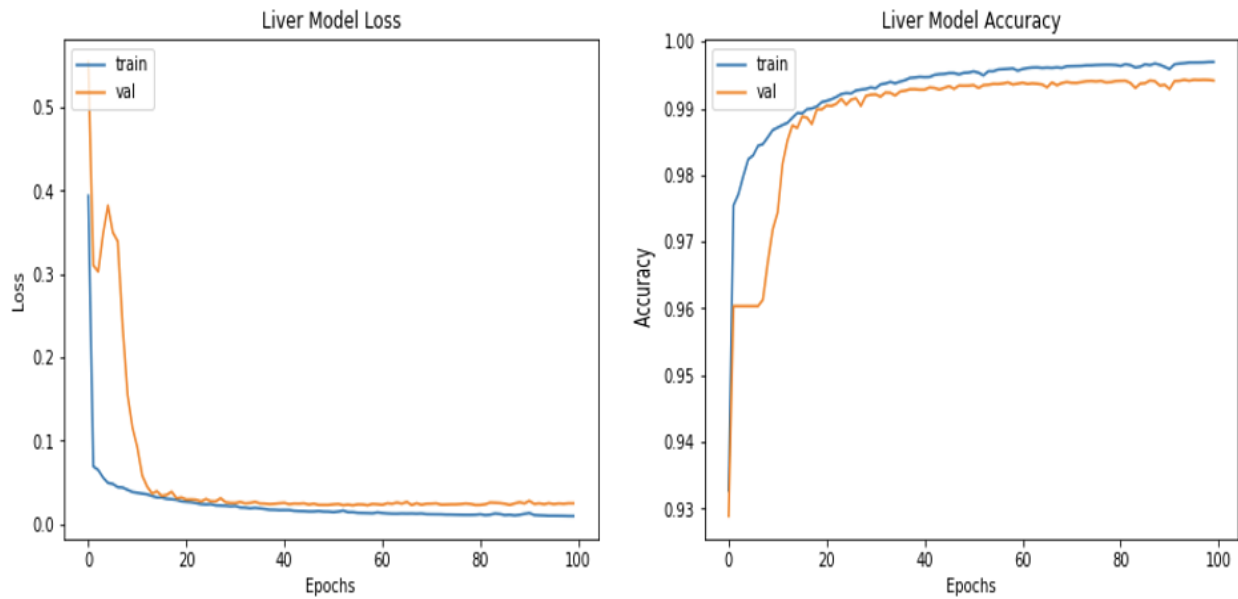
Figure 7: Left plot shows the liver model loss. Right plot shows the liver model accuracy. Blue line shows the training loss/accuracy. Orange line represents the validation loss/accuracy.
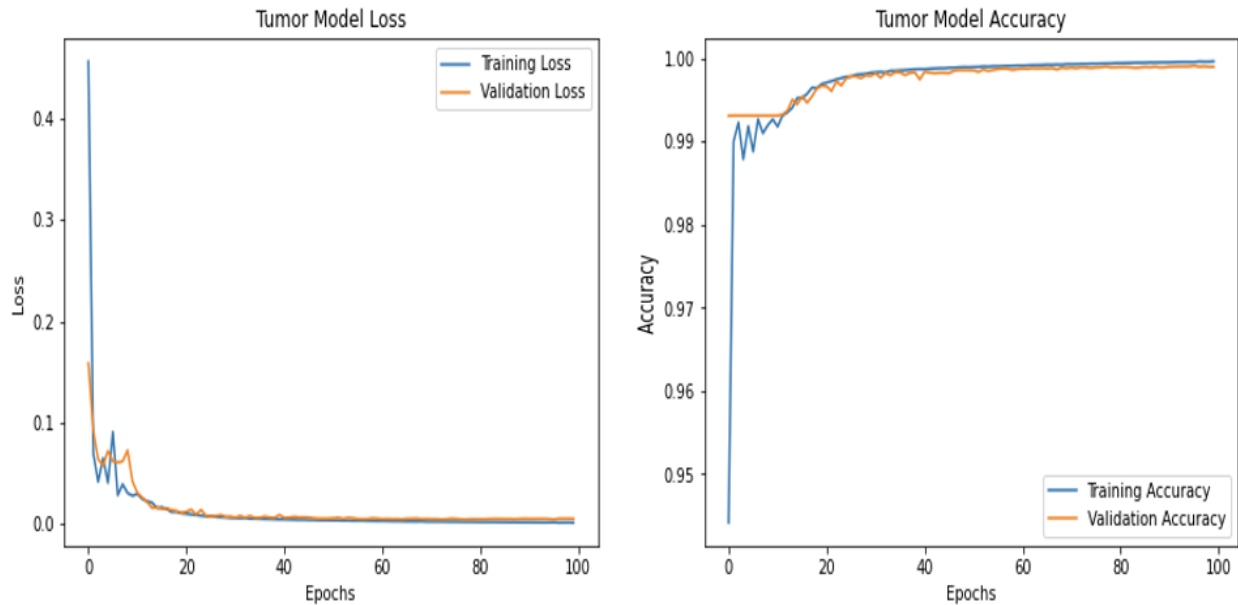
Figure 8: Left plot shows the tumor model loss. Right plot shows the tumor model accuracy. Blue line shows the training loss/accuracy. Orange line represents the validation loss/accuracy.

5.3. Liver and tumor segmentation

Figure 9 shows the liver segmentation results. The first row represents the preprocessed livers from CT scans using HU windowing. The second row shows liver masks used as ground truths obtained from the RTSTRUCT file. The third row represents the predicted liver masks after training and validating the liver network. The fourth row shows the predicted liver segmentation overlaid on the original (first row) preprocessed CT scans.
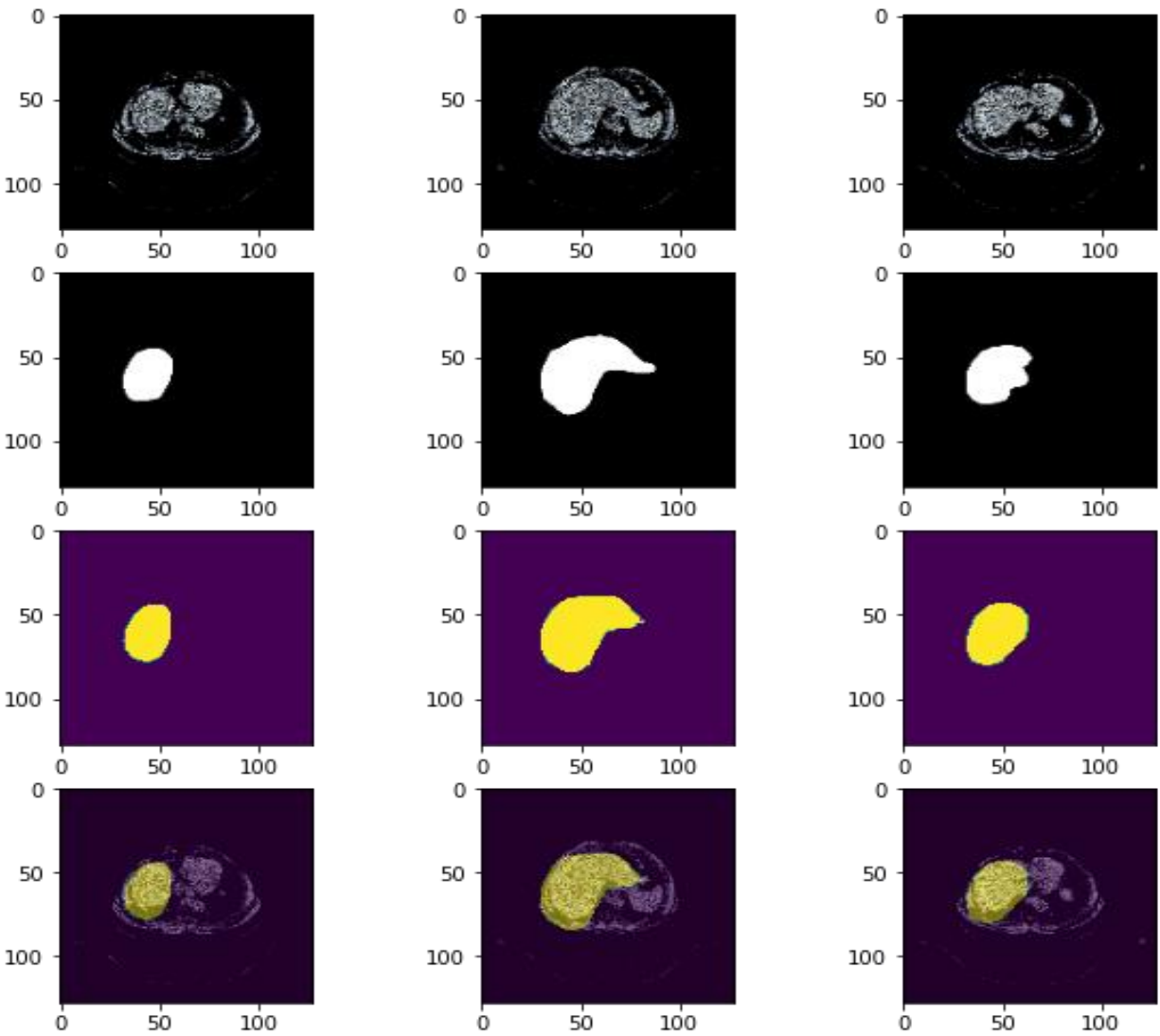
Figure 9: Represents the liver segmentation results from validation data

Similarly, Figure 10 shows the predictions for tumor segmentation network and Figure 11 shows when the network did not perform well in predicting tumors. The first row represents the liver masks filled in to provide segmented livers from the first U-Net. The second row shows the liver tumor masks used as ground truths obtained from the RTSTRUCT file. The third row represents the predicted liver tumor masks after training and validating the second liver tumor network. The fourth row shows the predicted liver tumor segmentation overlaid on the liver segments produced from the first network.
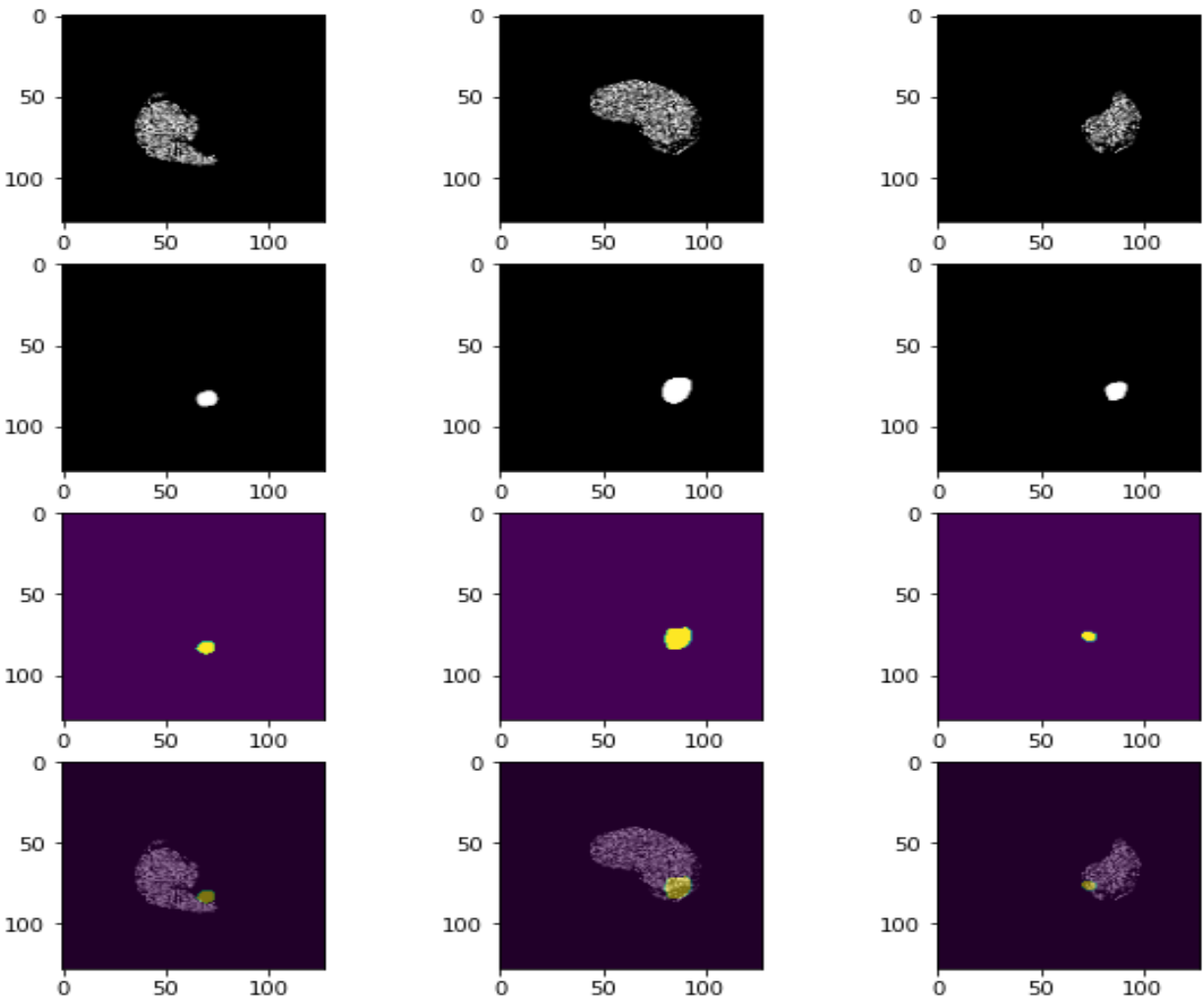
Figure 10: Represents the tumor segmentation results from validation data
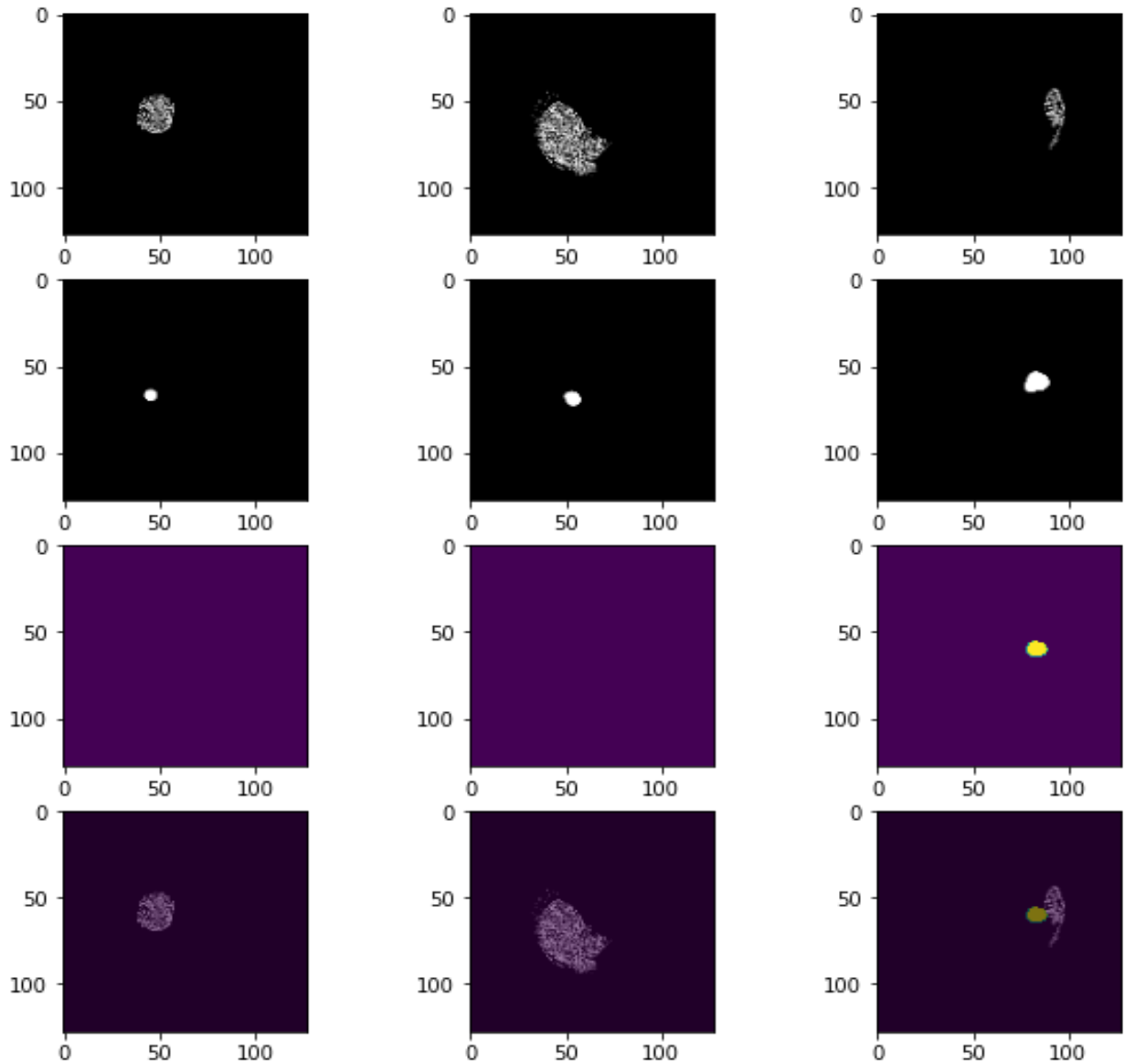
Figure 11: Represents the tumor segmentation results where the actual versus predicted mask did not match well for validation data (in some cases, there was no prediction).

Visually, the liver and tumor segmentation match the ground truths decently. It looks like the model has a difficulty segmenting on the edges. Moreover, the liver segmentations seem to be more accurate than the tumor segmentations. Table 2 depicts the quantitative evaluation of the liver and liver tumor segmentation network performance. The average dice coefficient for liver segmentation is 89.5%. The average dice coefficient for liver tumor segmentation is 44.4%.

35

Network	Average Dice Coefficient
Liver Segmentation	89.5%
Tumor Segmentation	44.4%

Table 4: Dice coefficient results obtained from validation data for liver and liver tumor segmentation

6.   Discussion

In this research, the CT scans used were non-contrast simulation CT scans obtained from the VCUHS health system. The CT scans had liver and liver tumor structures contoured by physicians that were used as ground truths for liver and tumor networks. The simulation images used were taken using 4DCT at 30% phase or breath-hold. Two concatenated CNN based U-Nets that were trained to segment livers and liver tumors from non-contrast enhanced abdominal simulation CT scans. The first network took input data of patient CT non-contrast abdominal CT images and liver masks and produced a liver segmentation. The second network took the liver segmentation mask output from the first network along with a tumor mask (ITV, CTV, GTV) and produced a tumor segment as the final result. The results of the liver segmentation and liver tumor segmentation are discussed below.

The U-Net performance for liver segmentation was evaluated by comparing the predicted results to the ground truths. As can be seen in Figure 8, the qualitative results of liver segmentation seem to be accurate with slight deviations from ground truth liver contours (second row from Figure 8) to predicted results (third row from Figure 8). The dice coefficient value for liver segmentation was 89.5%. This value further supports the qualitative performance of the liver segmentation. In studies performed by Ahmad et al. and Alirr, the liver segmentation dice coefficient using the IRCAD dataset which consists of contrast-enhanced diagnostic CT scans was 91.8% and 95.2% respectively.[33,14] The dice coefficient value obtained from this research compares well with the related works.

The U-Net performance for liver tumor segmentation was evaluated by comparing the predicted results to the ground truths. As can be seen in figure 9, the qualitative results of liver

36

www.manaraa.com

tumor segmentation generally match the ground truth liver tumor contours (second row from figure 9) to predicted results (third row from figure 9). The dice coefficient value for liver tumor segmentation was 44.4%. In the study performed by Christ et al. and Alirr, the liver tumor segmentation dice coefficient results using contrast-enhanced IRCAD diagnostic CT scans were 56% and 76.1% respectively.[34,19] The dice coefficient value obtained from this research is lower than what has been found in the related works.

The results of dice coefficient are lower especially for liver tumor segmentation compared to other related research is possibly due to the following reasons. Most liver tumor works used contrast-enhanced diagnostic CT scans for training and testing, where the contrast between liver and tumors is better. As a result, the network was able to perform much better. If the past segmentation works would test their models on a non-contrast CT scan, then it would be more representative of their network performance. Other common sources of errors in this research include deviations in the intensity of the liver tumors in the abdominal CT scan in every patient. As mentioned earlier, variability in contouring of the livers and tumors affects the ground truth masks used to train the network, and therefore could affect the performance of the network.

Additionally, the contours created by physicians for treatment planning which were used as ground truth also had MRI information. This makes the contours used as ground truths more consistent and accurate. Conversely, the additional information from MRI is not present in the CT images, and this limits the performance of the CT only network. It is not evident in other segmentation works if the ground truths contained information from multiple imaging modalities. This is useful as the network is able to train on images containing information from both CT and MRI and thus learn more about the image. Moreover, this research used simulation CT images instead of diagnostic CT images. This is beneficial because the main goal of this study is to use this liver tumor segmentation network as an automated tool to help physicians in contouring liver tumors. The physicians use simulation images for contouring and treatment planning to deliver a safe radiation treatment.

In this research, the methods used for management of respiratory motion were ABC and 4DCT at 30% phase as this is the phase used for treatment planning, which is the mid-ventilation

phase that is most close to the average motion of the tumor during the entire respiratory cycle. Furthermore, ITV contour, which accounts for respiratory motion was used as the ground truth. The margin used to create an ITV contour will defer from patient to patient based on the movement of the tumor in all the directions in the 4DCT scans. The ITV margin is smaller in ABC scans. The use of ITV contour as the ground truth, and discrepancies in the ITV contour between the two types of scans could have affected the network predictions. Secondly, choosing GTV or CTV contours as ground truths could lead to training the network in underestimating the extent of the tumor as these would not account for tumor motion. Lastly, using a different phase than 30% will also affect the performance of the network as the ITV contour will defer, thus affecting the ground truths.

7.   Other Limitations/Future Recommendations

The conducted research has proven to have valuable results but it also has some limitations. One of the limitations arises from the use of two U-Nets for liver and tumor segmentation respectively. The liver envelope produced from liver segmentation was used as input along with tumor ground truth for the second network, which is the tumor segmentation network. Therefore, the results of the liver segmentation will affect the performance of the liver tumor network.

Secondly, it is important to note that the network was trained with images that contained a liver tumor. In a case, where an image with no tumor is presented, the network performance might be degraded. This was not tested in this research but would be a good future study.

Moreover, the network was not tested using test images to provide a complete unbiased evaluation of the final model. It would be useful to have some test images aside to test the network and get a true evaluation of the model after training and validation.

To overcome some of the limitations expressed above, one possible solution would be to train the liver network using a whole abdominal scan containing all slices of liver in all planes. This would mean that some images will not have a visible tumor. However, this would require a lot of storage and time to train the network. Due to time constraints, this was beyond the scope of

38

this research. The overall performance of the network itself can be improved by having more data or through more data augmentation. Other areas of improvements could be from tuning the hyperparameters such as kernel size, learning rate, activation/loss functions, batch size, epochs as well as experimenting with different architectures.

8.   Conclusion

Several deep learning models have been developed for early detection and segmentation of liver and liver tumors. There is a need for automated algorithms for segmentation in order to come up with a precise radiation therapy treatment plan. Previous research mainly focused on using contrast-enhanced diagnostic CT images as inputs to the neural networks for diagnosing liver tumors. However, contrast enhancement is sensitive to the timing of the CT acquisition, thus making it a challenge to obtain the scan in the right amount of time, and appropriately detect and delineate liver tumors. Secondly, the use of diagnostic images for the purposes of this research has some disadvantages, one being that the images used for treatment planning are simulation CT images. Diagnostic images also lack the motion management strategies used while preparing the radiation treatment plan. As a result, these images might not be appropriate to aid physicians in delineating liver and liver tumors and thus creating radiation therapy treatment plans.

As a contribution to the ongoing research on liver tumor segmentation, this research utilized simulation based non-contrast CT scans for automated liver and liver tumor segmentation for radiotherapy treatment planning. Furthermore, the contours for livers and liver tumors used as ground truths also had MRI information. Two fully convolutional concatenated neural networks with a U-Net architecture originally built by Olaf Ronneberger were used in this study. The proposed method succeeded in liver and liver tumor segmentation. The results for liver segmentation were comparable with related research. The results for tumor segmentation were slightly lower than related studies, but can be improved with changes in the neural network structure (fine tuning parameters, more training iterations, etc.)

Overall, this research takes a significant step in the development of automated liver tumor segmentation using simulation CT scans which are used for delineating regions of interest (liver

39

tumor- GTV, CTV, ITV) and thereby creating an efficient treatment plan. This research also identifies areas of improvement and steps that need to be taken before implementing this work in the clinic.

40

9.   References

1. Cancer. World Health Organization. Published March 3, 2021. Accessed July 18, 2021. https://www.who.int/news-room/fact-sheets/detail/cancer

2. Adult Primary Liver Cancer Treatment (PDQ®)–Patient Version - National Cancer Institute. Published April 12, 2021. Accessed July 18, 2021. https://www.cancer.gov/types/liver/patient/adult-liver-treatment-pdq

3. Rego J, Tan K. Advances in Imaging—The Changing Environment for the Imaging Specialist. *Perm J.* 2006;10(1):26-28.

4. Wang W, Wei C. Advances in the early diagnosis of hepatocellular carcinoma. *Genes Dis.* 2020;7(3):308-319. doi:10.1016/j.gendis.2020.01.014

5. Khan FM, Gibbons JP. *Khan's the Physics of Radiation Therapy*. Fifth edition. Lippincott Williams & Wilkins/Wolters Kluwer; 2014.

6. Computed Tomography (CT). Accessed July 18, 2021. https://www.nibib.nih.gov/science-education/science-topics/computed-tomography-ct

7. Murakami T, Tsurusaki M. Hypervascular Benign and Malignant Liver Tumors That Require Differentiation from Hepatocellular Carcinoma: Key Points of Imaging Diagnosis. *Liver Cancer.* 2014;3(2):85-96. doi:10.1159/000343864

8. Keall PJ, Mageras GS, Balter JM, et al. The management of respiratory motion in radiation oncology report of AAPM Task Group 76. *Med Phys.* 2006;33(10):3874-3900. doi:10.1118/1.2349696

9. Slotman BJ, Lagerwaard FJ, Senan S. 4D imaging for target definition in stereotactic radiotherapy for lung cancer. *Acta Oncol.* 2006;45(7):966-972. doi:10.1080/02841860600902817

10. Wong JW, Sharpe MB, Jaffray DA, et al. The use of active breathing control (ABC) to reduce margin for breathing motion. *Int J Radiat Oncol Biol Phys.* 1999;44(4):911-919. doi:10.1016/s0360-3016(99)00056-5

11. Active Breathing Coordinator (ABC) | Swedish Medical Center Seattle and Issaquah. Accessed July 18, 2021. https://www.swedish.org:443/services/cancer-institute/our-services/radiation-therapy/types-of-radiation-therapy/active-breathing-coordinator

12. Jones D. ICRU Report 50—Prescribing, Recording and Reporting Photon Beam Therapy. *Med Phys.* 1994;21(6):833-834. doi:10.1118/1.597396

13. Landberg T, Chavaudra J, Dobbs J, et al. Report 62. *J Int Comm Radiat Units Meas.* 1999;os32(1):NP-NP. doi:10.1093/jicru/os32.1.Report62

14. Alirr OI. Deep learning and level set approach for liver and tumor segmentation from CT scans. *J Appl Clin Med Phys.* 2020;21(10):200-209. doi:10.1002/acm2.13003

15. Roy S, Carass A, Jog A, Prince JL, Lee J. MR to CT Registration of Brains using Image Synthesis. *Proc SPIE.*
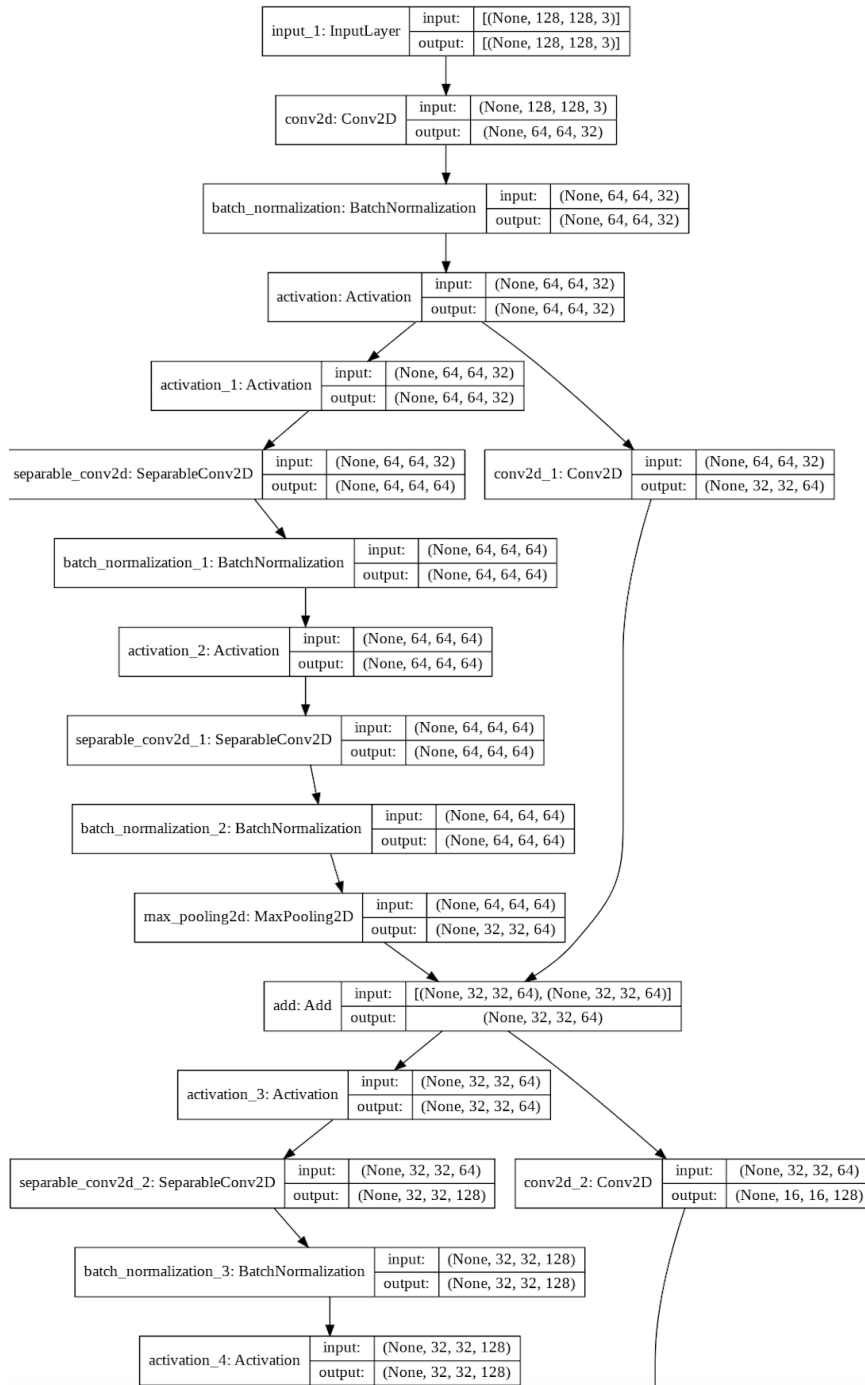
2014;9034:spie.org/Publications/Proceedings/Paper/10.1117/12.2043954. doi:10.1117/12.2043954

16. Oh S, Kim S. Deformable image registration in radiation therapy. *Radiat Oncol J*. 2017;35(2):101-111. doi:10.3857/roj.2017.00325

17. Chollet F. *Deep Learning with Python*. Manning Publications Co; 2018.

18. Kumar T, Verma, Karun. A Theory Based on Conversion of RGB image to Gray image. Published online September 2010.

19. Diaconu A-V, Costea A, Costea M-A. Color Image Scrambling Technique Based on Transposition of Pixels between RGB Channels Using Knight's Moving Rules and Digital Chaotic Map. *Math Probl Eng*. 2014;2014:e932875. doi:10.1155/2014/932875

20. Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. In: *2017 International Conference on Engineering and Technology (ICET)*. ; 2017:1-6. doi:10.1109/ICEngTechnol.2017.8308186

21. Khan S, Rahmani H, Shah SAA, Bennamoun M. A Guide to Convolutional Neural Networks for Computer Vision. *Synth Lect Comput Vis*. 2018;8(1):1-207. doi:10.2200/S00822ED1V01Y201712COV015

22. Kumar K, Rao ACS. Breast cancer classification of image using convolutional neural network. In: *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*. ; 2018:1-6. doi:10.1109/RAIT.2018.8389034

23. Talathi SS, Vartak A. Improving performance of recurrent neural network with relu nonlinearity. *ArXiv151103771 Cs*. Published online June 23, 2016. Accessed August 9, 2021. http://arxiv.org/abs/1511.03771

24. Aphex34. *English: Typical CNN Architecture*.; 2015. Accessed August 9, 2021. https://commons.wikimedia.org/wiki/File:Typical_cnn.png

25. Almotairi S, Kareem G, Aouf M, Almutairi B, Salem MA-M. Liver Tumor Segmentation in CT Scans Using Modified SegNet. *Sensors*. 2020;20(5):1516. doi:10.3390/s20051516

26. LiTS- Liver Tumor Segmentation Challenge. Accessed July 19, 2021. https://competitions.codalab.org/competitions/17094

27. Han X. Automatic Liver Lesion Segmentation Using A Deep Convolutional Neural Network Method. *Med Phys*. 2017;44(4):1408-1419. doi:10.1002/mp.12155

28. Meng L, Tian Y, Bu S. Liver tumor segmentation based on 3D convolutional neural network with dual scale. *J Appl Clin Med Phys*. 2020;21(1):144-157. doi:10.1002/acm2.12784

29. Chlebus G, Schenk A, Moltz JH, van Ginneken B, Hahn HK, Meine H. Automatic liver tumor segmentation in CT with fully convolutional neural networks and object-based postprocessing. *Sci Rep*. 2018;8(1):15497. doi:10.1038/s41598-018-33860-7
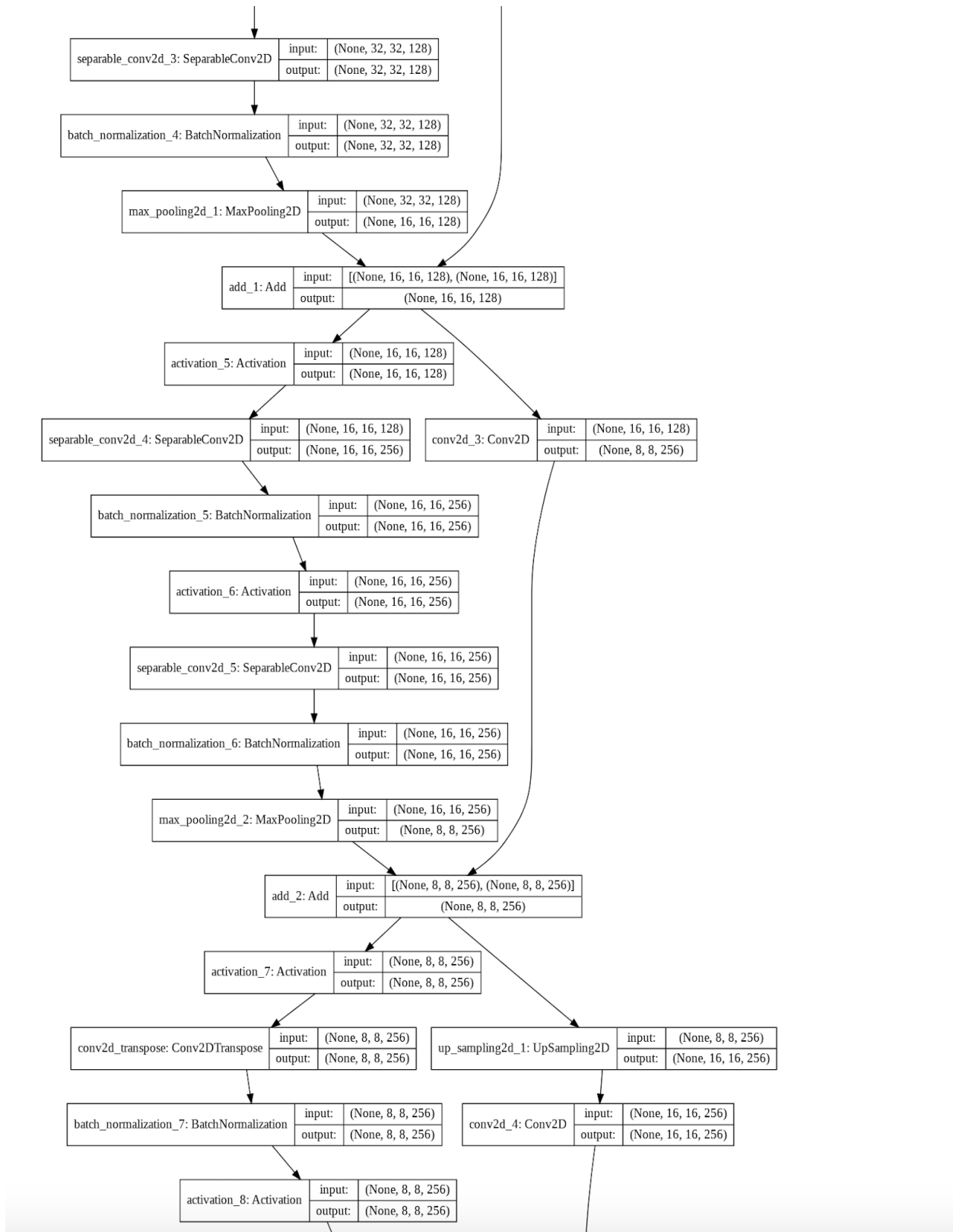
30. Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *ArXiv150504597 Cs.* Published online May 18, 2015. Accessed July 19, 2021. http://arxiv.org/abs/1505.04597

31. Yazdani M. *English: This Is an Example Architecture of U-Net for Producing k 256-by-256 Image Masks for a 256-by-256 RGB Image.*; 2019. Accessed August 3, 2021. https://commons.wikimedia.org/wiki/File:Example_architecture_of_U-Net_for_producing_k_256-by-256_image_masks_for_a_256-by-256_RGB_image.png

32. Zou KH, Warfield SK, Bharatha A, et al. Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index. *Acad Radiol.* 2004;11(2):178-189. doi:10.1016/S1076-6332(03)00671-8

33. Albishri AA, Shah SJH, Lee Y. CU-Net: Cascaded U-Net Model for Automated Liver and Lesion Segmentation and Summarization. In: *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. ; 2019:1416-1423. doi:10.1109/BIBM47256.2019.8983266

34. Christ PF, Ettlinger F, Grün F, et al. Automatic Liver and Tumor Segmentation of CT and MRI Volumes using Cascaded Fully Convolutional Neural Networks. *ArXiv170205970 Cs.* Published online February 23, 2017. Accessed July 19, 2021. http://arxiv.org/abs/1702.05970
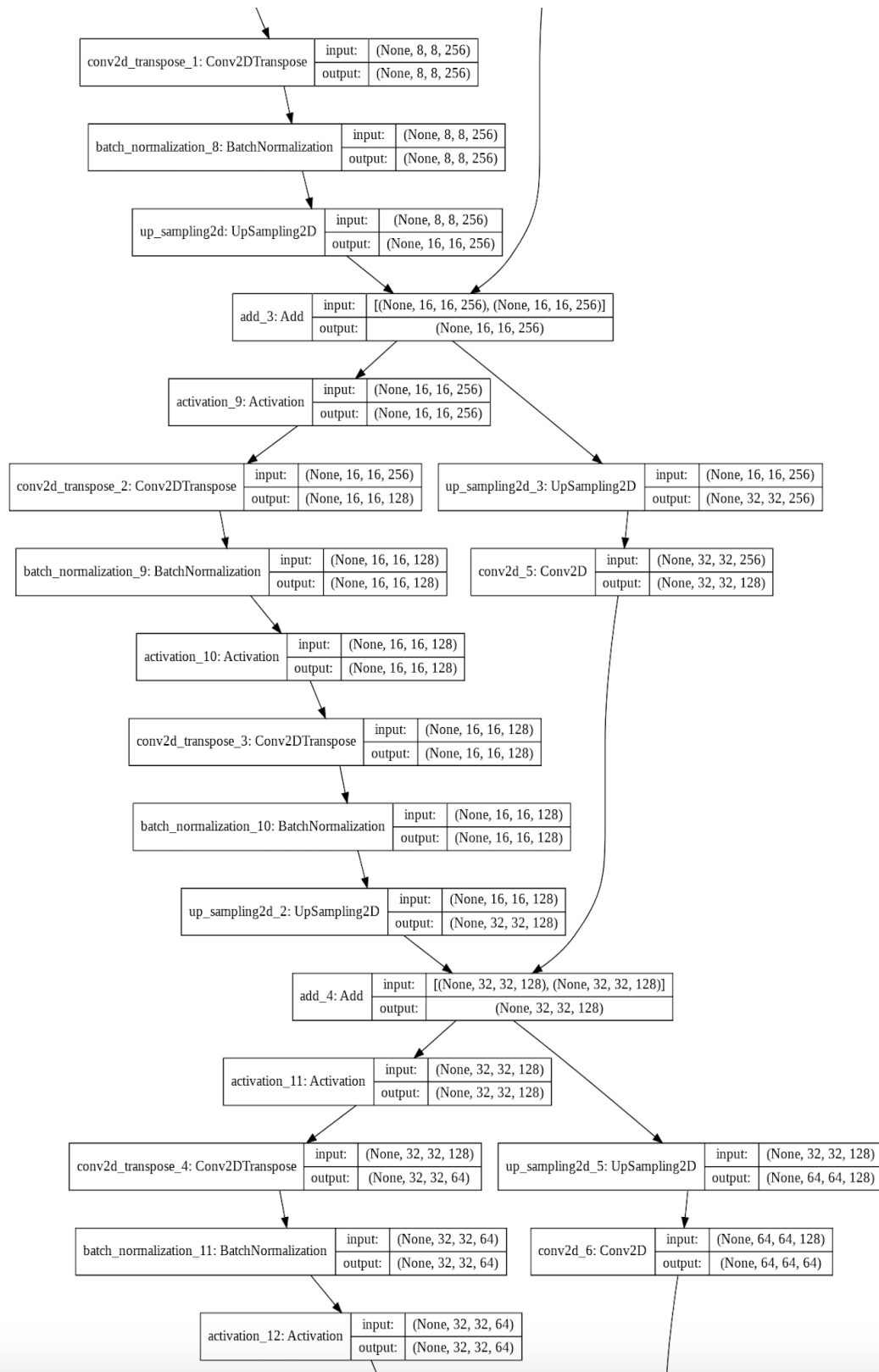
Appendix A.

Shows the summary of the network architecture

Continuation of the summary of the network architecture

Continuation of the summary of the network architecture

Continuation of the summary of the network architecture

activation_12: Activation — input: (None, 32, 32, 64) — output: (None, 32, 32, 64)

conv2d_transpose_5: Conv2DTranspose — input: (None, 32, 32, 64) — output: (None, 32, 32, 64)

batch_normalization_12: BatchNormalization — input: (None, 32, 32, 64) — output: (None, 32, 32, 64)

up_sampling2d_4: UpSampling2D — input: (None, 32, 32, 64) — output: (None, 64, 64, 64)

add_5: Add — input: [(None, 64, 64, 64), (None, 64, 64, 64)] — output: (None, 64, 64, 64)

activation_13: Activation — input: (None, 64, 64, 64) — output: (None, 64, 64, 64)

conv2d_transpose_6: Conv2DTranspose — input: (None, 64, 64, 64) — output: (None, 64, 64, 32)

up_sampling2d_7: UpSampling2D — input: (None, 64, 64, 64) — output: (None, 128, 128, 64)

batch_normalization_13: BatchNormalization — input: (None, 64, 64, 32) — output: (None, 64, 64, 32)

conv2d_7: Conv2D — input: (None, 128, 128, 64) — output: (None, 128, 128, 32)

activation_14: Activation — input: (None, 64, 64, 32) — output: (None, 64, 64, 32)

conv2d_transpose_7: Conv2DTranspose — input: (None, 64, 64, 32) — output: (None, 64, 64, 32)

batch_normalization_14: BatchNormalization — input: (None, 64, 64, 32) — output: (None, 64, 64, 32)

up_sampling2d_6: UpSampling2D — input: (None, 64, 64, 32) — output: (None, 128, 128, 32)

add_6: Add — input: [(None, 128, 128, 32), (None, 128, 128, 32)] — output: (None, 128, 128, 32)

conv2d_8: Conv2D — input: (None, 128, 128, 32) — output: (None, 128, 128, 256)

47