



Liver tumor Segmentation In CT Scans

Faculty of Engineering Ain Shams University

Department of Computer and Systems Engineering

A Thesis Submitted By

Mostafa Hamed Abdelmasoud

Ahmed Ali Mohamed Ahmed

Abdullah Shaker Abdelrahman

Moamen Ahmed Mamoun

Supervised By

Prof. Ahmed Hassan

Dean of Information Technology and Computer Science School at Nile University, a professor in Ain Shams University, Faculty of Engineering, Computers and Systems Engineering Department, IEEE vice-chair for chapter operations in the Egypt section.

Doctor. Mustafa A. Elattar

Assistant Professor, Medical Imaging and Image Processing Group, Center for Informatic Sciences ([Google Scholar](#))

Abstract

Exact automatic liver and tumor segmentation will affect powerfully liver therapy planning procedures, and follow-up reporting, thanks to automation, standardization, and incorporation of full volumetric information. In this work, we propose a fully automatic method for liver tumor segmentation in CT images based on a 2D convolutional deep neural network. We ran our experiments on the (3D-IRCADb 01) datasets and evaluated detection and segmentation performance. Our proposed method achieves segmentation quality for detected tumors comparable to a human expert and is able to detect potentially measurable tumor lesions. To help doctors better diagnose and present personalized curing. in medical practice, it is often required to make segmentation and visualization for live tumor parts. Because of the huge number of slices in the input image, developing an automatic and efficient segmentation technique is very preferred by physicians. Although due to the noise in the scan and the similar pixel weight of liver tumors with surrounding cells, besides both size, position, and shape of tumors also is different for each patient, automatic liver tumor segmentation is still a difficult task. Liver and tumor segmentation is a bottleneck for any system. In this problem, deep learning and image processing techniques are adopted for semantic classification to fit liver CT scan segmentation.

ACKNOWLEDGEMENTS

We would like to express our profound gratitude to everyone who has helped us turn our ideas into reality. The encouragement and support we received were our motives to push forward and give our best.

Special thanks to:

Prof. Ahmed Hassan for his academic guidance and **Dr. Mustafa Elattar** for providing us with the technical support needed. and for the continuous follow-ups and positive feedback they always gave us.

Table of Contents

Abstract	1
ACKNOWLEDGEMENTS	2
Table of Contents	3
1.0 CHAPTER ONE: INTRODUCTION	5
1.1 Problem Definition	5
1.2 Motive	5
1.3 Why liver cancer exactly?	5
1.4 So what will we use?	6
1.5 Dataset	6
1.6 Project goals	9
1.7 Advantages and disadvantages	10
2.0 CHAPTER TWO: RELATED WORK	12
2.1 Comparison between recent papers	13
3.0 CHAPTER THREE: Key Performance Indicators	15
3.1 Evaluation Metrics	15
3.1.1 Intersection over union	16
3.3.2 Dice Loss Function	16
4.0 CHAPTER FOUR: Methodology	18
4.1 Deep Learning	18
4.1.1 Data preprocessing	24
4.1.2 Data augmentation	27
4.1.3 Model selection	31
4.2 Image processing	34
4.2.1 Introduction	34
4.2.2 Liver segmentation	35
4.2.1 Tumor segmentation	42

5.0 CHAPTER FIVE: RESULTS	51
5.1 Deep Learning	51
5.2 Image Processing	54
6.0 CHAPTER SIX: CONCLUSION	55
7.0 CHAPTER SEVEN: Tools Used	57
8.0 CHAPTER EIGHT: Future Work	
58	
9.0 CHAPTER NINE: Bibliography	60

Introduction

Problem Definition

Liver cancer was the second most common cause of cancer-induced deaths in 2015 according to the World Health Organization. Hepatocellular carcinoma (HCC) is the most common type of primary liver cancer which is the sixth most prevalent cancer. Also, the liver is a common site for secondary tumors. Liver therapy planning procedures would profit from an accurate and fast lesion segmentation, that allows for subsequent determination of volume and texture-based information.

Motive

Why Liver Cancer Exactly?



Since 1980, liver cancer incidence in the US has more than tripled, and the disease is about 3 times more common in men than it is in women. For Egypt, it is one of the most cancers in Egypt, if not the most. It ranked second in liver cancer In both Sexes (Age-standardised rate per is 32.5) In Men (Age-standardised rate per is 49) and Women (Age-standardised rate per is 16.7)

So What will we Use?

Artificial intelligence (AI), particularly deep learning algorithms, is gaining extensive attention for its excellent performance in image-recognition tasks. They can automatically make a quantitative assessment of complex medical image characteristics and achieve an increased accuracy for diagnosis with higher efficiency. AI is widely used and getting increasingly popular in the medical imaging of the liver, including radiology, ultrasound, and nuclear medicine. AI can assist physicians to make more accurate and reproductive imaging diagnosis and also reduce the physicians' workload. This Intro illustrates basic technical knowledge about AI, including traditional machine learning and deep learning algorithms and their clinical application in the medical imaging of liver diseases. We conclude that machine-assisted medical services will be a promising solution for future liver medical care.

Dataset

The **3D-IRCADb-01** database is composed of the 3D CT-scans of 10 women and 10 men with hepatic tumors in 75% of cases. The 20 folders correspond to 20 different patients, which can be downloaded individually or conjointly. The table below provides information on the image, such as liver size (width, depth, height) or the location of tumors according to Couninaud's segmentation. It also indicates the major difficulties liver segmentation software may encounter due to the contact with neighbouring organs, an atypical shape or density of the liver, or even artefacts in the image.

These folders are called “**3D-IRCADb-1-number**” (the *number* varying between 01 and 20).. These folders are called “**3D-IRCADb-01-number**” (the *number* varying between 01 and 20). Each “**3D-IRCADb-01-number**” folder contains 4 sub-folders called “**PATIENT_DICOM**”, “**LABELLED_DICOM**”, “**MASKS_DICOM**” and “**MESHES_VTK**”. These folders respectively contain the anonymized patient image in DICOM format, the labelled image corresponding to the various zones of interest segmented in DICOM format, a new set of sub-folders corresponding to the names of the various segmented zones of interest containing the DICOM image of each mask, and finally, all the files corresponding to surface meshes of the various zones of interest in VTK format.

Recently, most studies referenced in a comprehensive review of liver segmentation focus on two types of methods: deformable model based, image segmentation based methods. To preserve liver shape from the adjacent organs.

Some factors in CT images bring some challenges to liver segmentation. First, imaging artifacts and tumor pathologies often result in intensity inhomogeneity. Therefore some standard approaches depending on gray-value only, may not be sufficient for this case. Second, intensities of several adjacent organs like heart and stomach are very similar to liver tissue itself. As shown, some examples of these difficulties are given.

Multi-dimensional features and shape priors can aid to separate the neighboring organs with similar intensities and reach the desired boundaries of the structures.

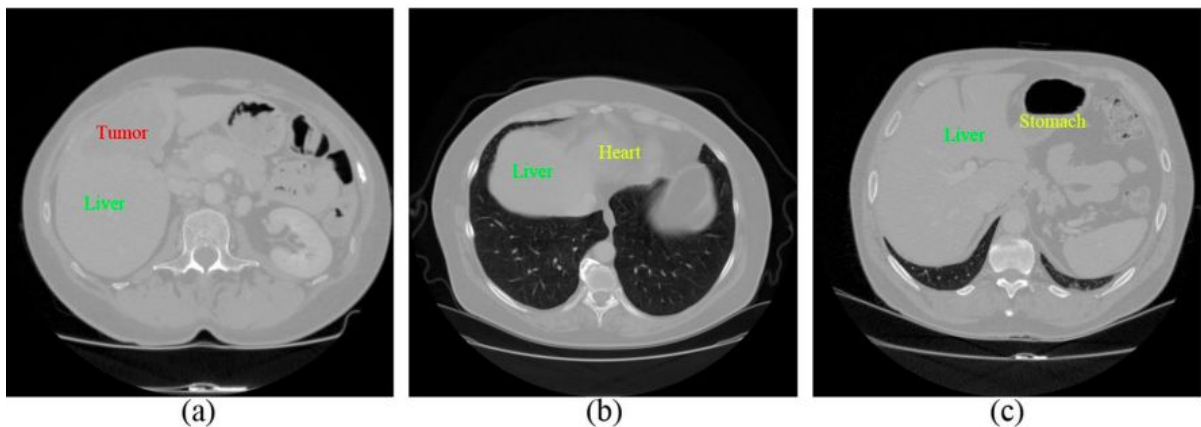
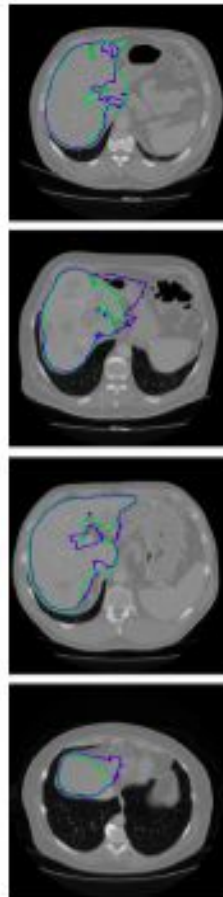


Figure: Examples of the limiting factors for liver segmentation in CT images. (a) Demonstrates intensity inhomogeneity between liver and tumor. (b) and (c) display that the liver, heart, and stomach share similar intensity values.



Liver initial and final
segmentation results with four
difficult cases. The ground truth is
shown in green curves

Data Augmentation

Overview

Data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset.

Training deep learning neural network models on more data can result in more skillful models, and the augmentation techniques can create variations

of the images that can improve the ability of the fit models to generalize what they have learned to new images.

Various techniques like random flipping of the image, random translation, generating random brightness and random shadow on the images through coding are used to expand the collected images from the simulator.

Since our dataset suffered from class imbalance because the number of pixels not containing tumors is much higher than the pixels containing tumors we had to augment the dataset. to enlarge the training dataset to avoid overfitting, x class imbalance and improving training accuracy. Data augmentation incorporates different image transformation techniques, such as reflection, rotation, cropping, translation and addition of noise. In this paper, we used two techniques to augment the data

Project Goals

Liver segmentation is an essential procedure in computer-assisted surgery, radiotherapy, and volume measurement. It is still a challenging task to extract liver tissue from 3D CT images owing to nearby organs with similar intensities. In this document, an automatic approach Deep Learning and Image Processing are developed and validated. Segmentation techniques are utilized to estimate the coarse shape of the liver on the target image. The unsigned distance field based on initial shape is then calculated throughout the whole image, which aims at automatic model construction during the segmentation process. Finally, we draw contours on segmented Liver and Tumor shape. The optimal liver region can be precisely detected with a minimal cost. The proposed technique is evaluated on 20 CT scans, obtained from public database 3Dircadb1. On the dataset 3Dircadb1. The experiment results indicate that the proposed method has the ability to reach the desired boundary of the liver and has potential value for clinical application.

The extraction of liver tissue is very important for hepatic disease diagnosis, function assessment, and computer-assisted surgery . Among the various medical imaging techniques, computed tomography (CT) is often used for these purposes due to higher signal-to-noise ratio and better spatial resolution. However, it is tedious and time-consuming to get liver regions by

manual delineation from several thousand slices. Based on this problem, many researchers have proposed some semi-automatic or automatic methods for liver segmentation [4].

Advantages and disadvantages

1. Deep Learning:

Benefits or advantages	Drawbacks or disadvantages
<ul style="list-style-type: none"> • Features are automatically deduced and optimally tuned for desired outcome. Features are not required to be extracted ahead of time. This avoids time consuming machine learning techniques. • Robustness to natural variations in the data is automatically learned. • The same neural network based approach can be applied to many different applications and data types. • Massive parallel computations can be performed using GPUs and are scalable for large volumes of data. Moreover it delivers better performance results when the amount of data is huge. 	<ul style="list-style-type: none"> • It requires a very large amount of data in order to perform better than other techniques. • It is extremely expensive to train due to complex data models. Moreover deep learning requires expensive GPUs and hundreds of machines. This increases cost to the users. • There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters. As a result it is difficult to be adopted by less skilled people.

2. image Processing:

Benefits or advantages	Drawbacks or disadvantages
<ul style="list-style-type: none">● Important features such as edges can be extracted from images which can be used in industry.● Images can be given more sharpness and better visual appearance.	<ul style="list-style-type: none">● The main limitation is that if the object size is smaller than the pixel size, then it can not be applied efficiently because then one pixel can contain two or more objects...● It's very costly depending on the system used, the number of detectors purchased.● Time consuming● Lack of qualified professional

Related work

The problem of liver tumor segmentation has received a great interest in the medical image computing community. In 2008, the MICCAI 3D Liver Tumor Segmentation Challenge was organized where both manual and automatic methods were accepted. Among the automatic ones, the best method applied an ensemble segmentation algorithm using AdaBoost. Other submitted methods employed adaptive thresholding, region growing or level set methods.

In more recent years, methods using Grassmannian manifolds and shape parameterization were proposed.

Given the variability of liver lesions, the manual design of powerful features is not trivial. Deep neural networks (DNNs) gained rapidly growing attention in the computer vision community over the last years, because of their ability to learn features automatically from the data. Christ et al. applied two cascaded U-net models to the problem of liver and liver tumor segmentation. The approach employed one model solely for the liver segmentation and a separate one for the tumor segmentation within a liver bounding box. The final output was refined using a 3D conditional random field.

More recently, the Liver Tumor Segmentation (LiTS) challenge was organized. All top-scoring automatic methods submitted to the two rounds organized in 2017 used DNNs. Han, the winner of the first round, used two U-Net like models with long and short skip connections, where the first model was used only for coarse liver segmentation allowing the second network to focus on the liver region. The second model was trained to segment both liver and tumors at once. The two models worked in 2.5D, i.e., they received five adjacent slices to segment the middle one, which provided the network with the 3D context information. The best method in the second LiTS round was developed by a group from Lenovo Research, China. Their approach employed two neural network ensembles for the liver and tumor segmentation, respectively. The ensembles consisted of U-Net models working in 2D and 2.5D trained with different hyperparameter settings. Other successful methods proposed to train jointly two networks for liver and tumor segmentation and to exploit 3D information by training a 3D

H-DenseUNet architecture using original image data as well as features coming from a 2D network.

Also The Department of the Software Engineering University of Westminster, UK made research. In this research, the liver region automatically segments from CT abdominal images, using the knowledge of the liver, adaptive threshold decision based on intensity information, and morphological processing. tumor region extracts by using Fuzzy C Means Clustering (FCM). All these technologies and algorithms are well examined and critically evaluated.

with all the mentioned work although some results were promising there is much more can be done but simply the limited resources and small data in our hands it is still a challenge

comparison between recent papers:

Authors	Title	Year	Dataset	Architecture	Scores
Ouhmich, Farid Agus, Vincent Noblet,	Liver tissue segmentation in multiphase CT scans using cascaded convolutional neural networks	2019	data from 7 patients suffering from hepatocellular carcinoma (HCC)	Cascaded CNN based on U-net	Dice similarity coefficient from 80% to 90%
X.Li,H.Chen	H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes	2018	MICCAI 2017 Liver Tumor Segmentation (LiTS) Challenge and 3DIRCADb Dataset	H-DenseUNet	(LiTS) liver dice = 96.5 lesion dice = 82.4 (3DIRCADb) dice = 0.982
N.Grubner , S.Antholier	A Joint Deep Learning Approach for Automated Liver and	2019	LiTS	two modified U-Nets	IoU liver = .93848 IoU tumor = .771

LIVER TUMOR SEGMENTATION IN CT SCANS

	Tumor Segmentation				
G.Chelpus, A.Schenk	Deep learning based automatic liver tumor segmentation in CT with shape-based post-processing	2018	MICCAI 3D Liver Tumor Segmentation Challenge (LiTS)	a modified U-net with fully CNN	dice per case = .72
P.Christ M.Elshaer	Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields	2016	abdominal CT dataset 3DIRCAD	CFCs and 3DCRFs	lesion Dice a score of 82.3%

Key Performance Indicators

Evaluation Metrics and Loss Function

The idea of building AI and machine learning models works on a constructive feedback principle. You build a model, get feedback from **metrics**, make improvements and continue until you achieve a desirable accuracy. Evaluation metrics explain the performance of a model. An important aspect of evaluation metrics is their capability to discriminate among model results.

Simply building a predictive model is not your motive. It's about creating and selecting a model which gives high accuracy on out of sample data. Hence, it is crucial to check the accuracy of your model prior to computing predicted values.

We can consider different kinds of metrics to evaluate our models. The choice of metric completely depends on the type of model and the implementation plan of the model.

Deep learning neural networks are trained using the gradient descent optimization algorithm.

As part of the optimization algorithm, the error for the current state of the model must be estimated repeatedly. This requires the choice of an error function, conventionally called a **loss function**, that can be used to estimate the loss of the model so that the weights can be updated to reduce the loss on the next evaluation.

Neural network models learn a mapping from inputs to outputs from examples and the choice of loss function must match the framing of the specific predictive modeling problem, such as classification or regression. Further, the configuration of the output layer must also be appropriate for the chosen loss function.


In this tutorial, you will discover how to choose a loss function for your deep learning neural network for a given predictive modeling problem.

Intersection over union as evaluation metric

Intersection over Union is simply an evaluation metric. Any algorithm that provides predicted bounding boxes as output can be evaluated using IoU.

More formally, to apply Intersection over Union to evaluate an (arbitrary) object detector we need:

1. The ground-truth bounding boxes (i.e., the hand-labelled bounding boxes from the testing set that specify wherein the image our object is).
2. The predicted bounding boxes from our model.
3. As long as we have these two sets of bounding boxes we can apply Intersection over Union.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


Dice as Loss Function

- Dice Coefficient: it calculates the overlap between the classes of the input A and predicted label B and is calculated as follows see below

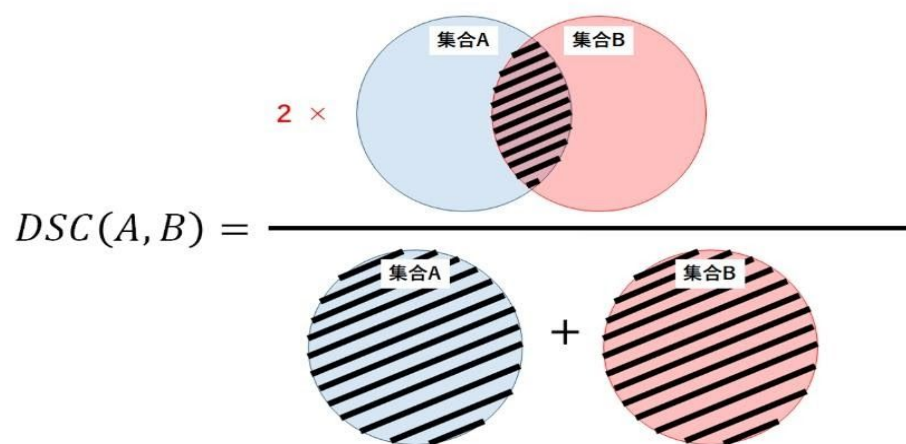


Fig.4: Dice coefficient (set view)

Methodology

1. Deep Learning

Introduction to deep learning

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces. Like other artificial intelligence and machine learning algorithms, it mainly aims at making the computer do a task usually done by humans, with minimum or no human intervention. The following figure illustrates the different layers of a typical neural network and the job of each layer:

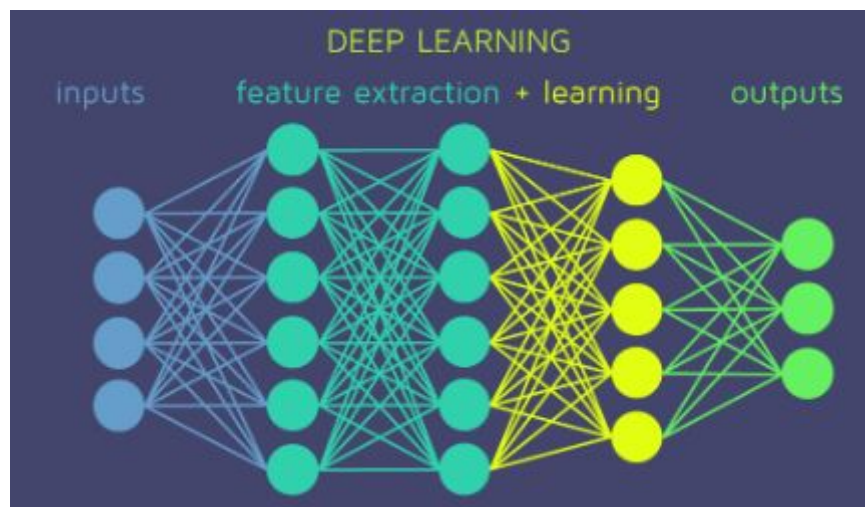


Figure illustrates how a deep network processes the input through different stages to get the output.

The following figure also illustrates the relation between AI, machine learning and deep learning:

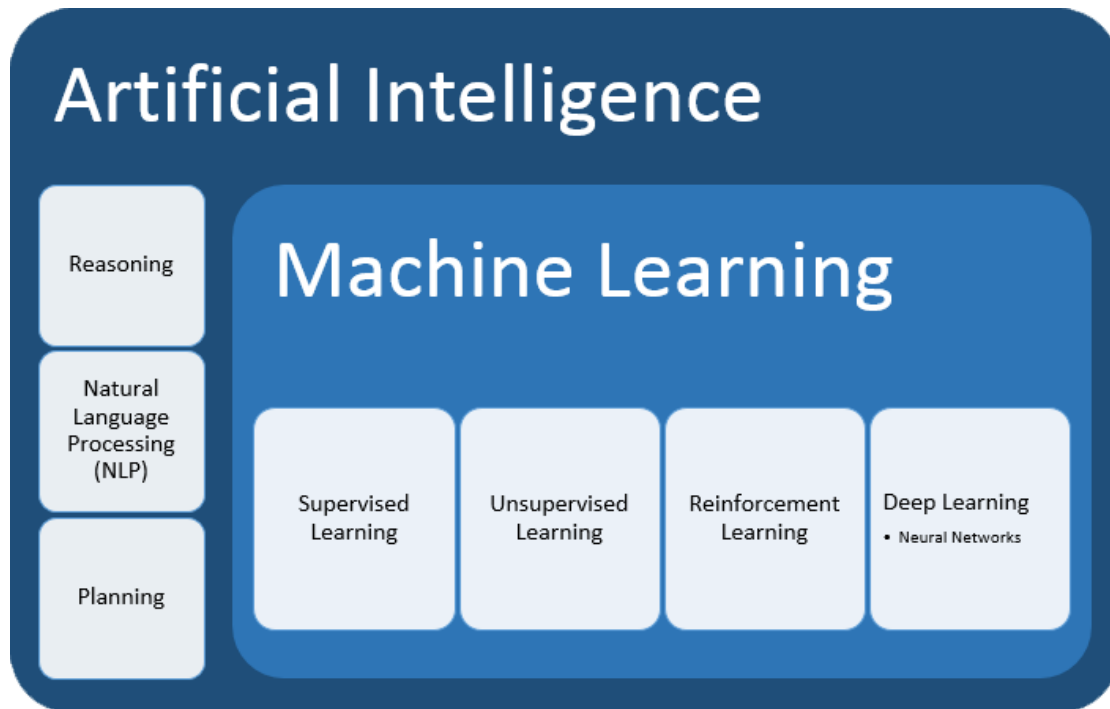


Figure shows the difference between AI, machine learning and deep learning and the types of each.

Deep Learning Models

Models in deep learning are mainly classified into two types: supervised models and unsupervised models. The main difference that distinguishes the two is how these models are trained. While supervised models are trained through examples of a particular set of data, unsupervised models are only given input data and don't have a set outcome they can learn from. As a result, that y-column that we're always trying to predict is not there in an unsupervised model. While supervised models have tasks such as regression and classification and will produce a formula, unsupervised models have clustering and association rule learning. The following figure illustrates the main difference between supervised and unsupervised learning models:

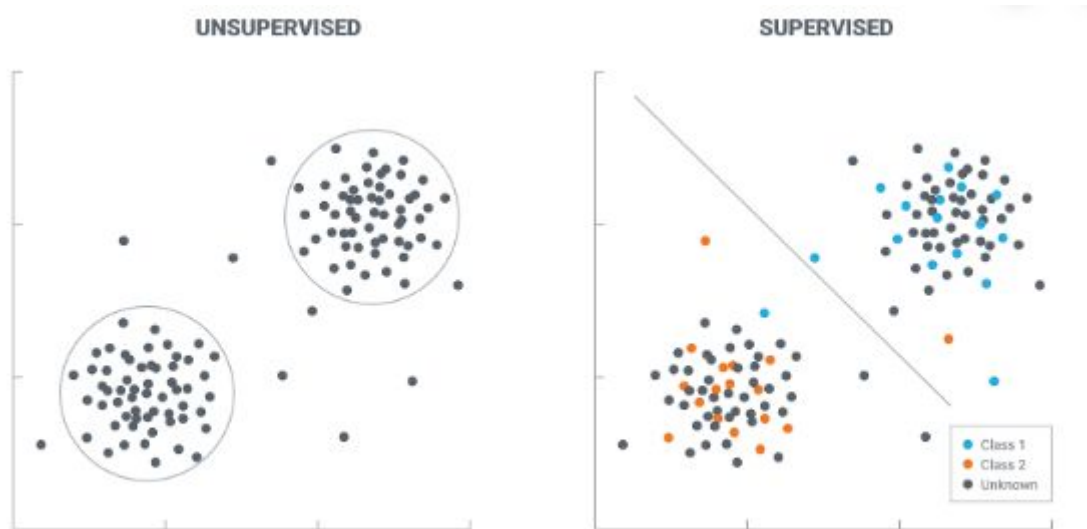


Figure illustrates the dependency of supervised models on classification vs. the dependency of the unsupervised models on clustering.

Classical Neural Networks

Also known as multilayer perceptrons, The perceptron model was created in 1958 by American psychologist Frank Rosenblatt. Its singular nature allows it to adapt to basic binary patterns through a series of inputs, simulating the learning patterns of a human-brain. A Multilayer perceptron is the classic neural network model consisting of more than 2 layers.

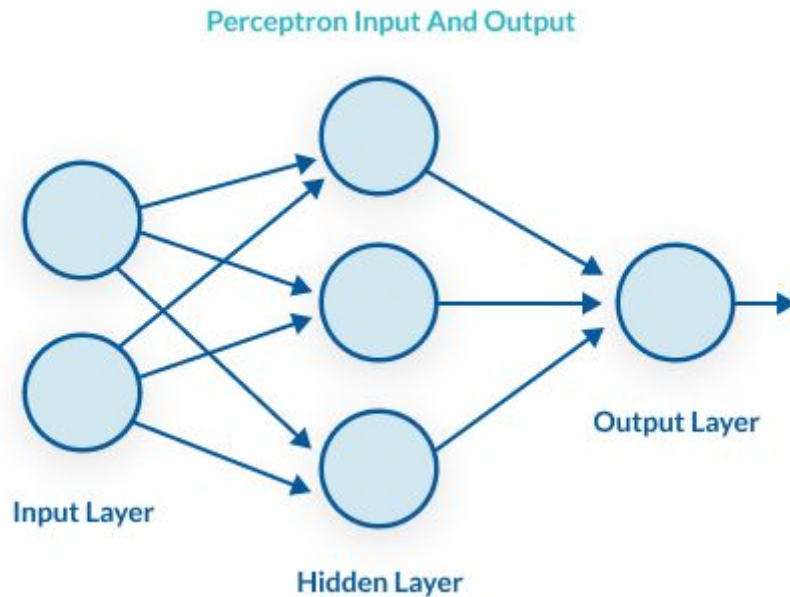


Figure illustrates the structure of a multilayer perceptron(Neural Networks).

Convolution Neural Networks(CNNs)

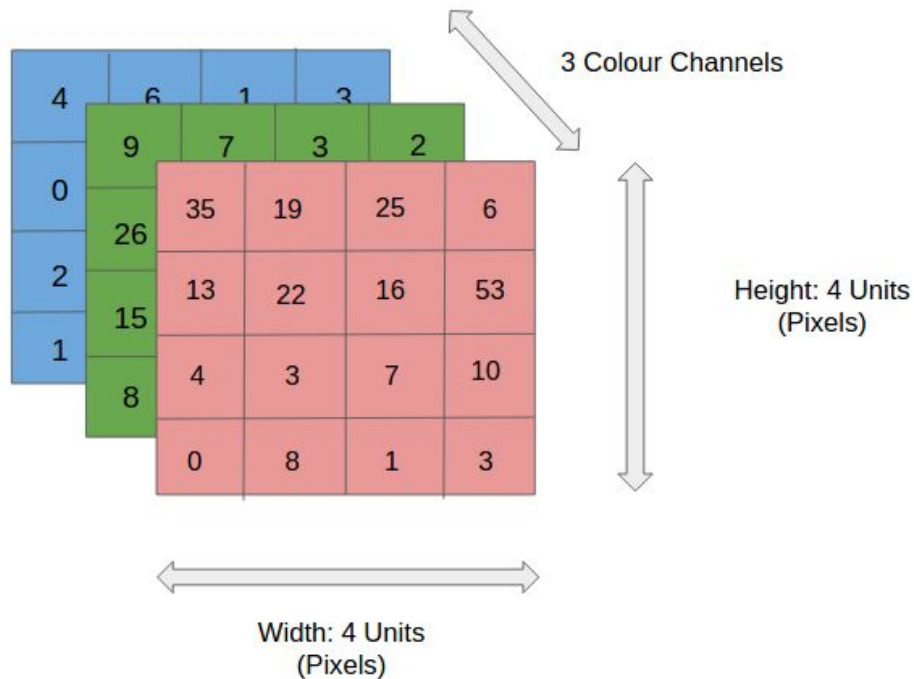
A more capable and advanced variation of classic artificial neural networks, a Convolutional Neural Network (CNN) is built to handle a greater amount of complexity around pre-processing, and computation of data.

CNNs were designed for image data and might be the most efficient and flexible model for image classification problems. Although CNNs were not particularly built to work with non-image data, they can achieve stunning results with non-image data as well.

CNN Terminology

CNNs are usually applied to image data. Every image is a matrix of pixel values. The range of values that can be encoded in each pixel depends upon its bit size. Most commonly, we have 8 bit or 1 Byte-sized pixels. Thus the possible range of values a single pixel can represent is $[0, 255]$. However, with colored images, particularly RGB (Red, Green, Blue)-based images, the presence of separate color channels (3 in the case of RGB images) introduces an additional 'depth' field to the data, making the input 3-dimensional. Hence, for a given RGB image of size, say 255×255 (Width x Height) pixels, we'll have 3 matrices associated with each image, one for each of the color channels. Thus

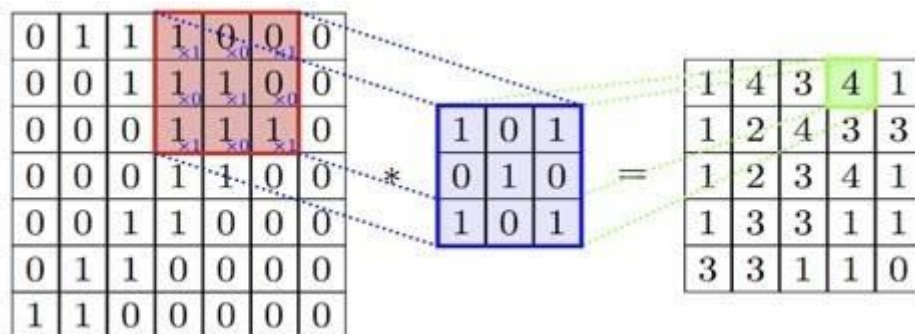
the image constitutes a 3-dimensional structure called the Input Volume (255x255x3).



The cross-section of an input volume of size: 4 x 4 x 3. It comprises the 3 Color channel matrices of the input image.

Kernels(Filters):

A filter or Kernel is a small-sized matrix in comparison to the input image's dimensions in which it is used to extract the features from the input image. A fixed sized kernel extracts the required feature by moving it across the image from top left to bottom right and for each point on the image, a value is calculated based on the filter using a convolution operation.



Example of how convolutions are applied to image.

Convolutional layer:

It is the layer where all the magic is done. Where the kernels are applied to the original image, or to other feature maps in a deep CNN by scanning a few pixels at a time and creating a feature map that predicts the class to which each feature belongs.

Pooling layer:

Pooling layers are similar to convolutional layers. They can perform various functions, such as max pooling which takes the maximum value in a certain filter region. or average pooling which takes the average value in a filter region. They are typically used to reduce the dimensionality of the network.

Fully connected layer:

It takes the output of the previous layers and flattens the results before classification.

Activation function:

As we know, In artificial neurons inputs and weights are given from which the weighted sum of input is calculated, and then it is given to an activation function that converts it into the output. So basically an activation function is used to map the input to the output. This activation function helps a neural network to learn complex relationships and patterns in data. Now the question is what if we don't use any activation function and allow a neuron to give the weighted sum of inputs as it is as the output. Well in that case computation will be very difficult as the weighted sum of input doesn't have any range and depending upon input it can take any value. Hence one important use of the activation function is to keep output restricted to a particular range. Another use of activation function is to add non-linearity in data. We always choose non-linear functions as activation functions. Let's see, why is it important.

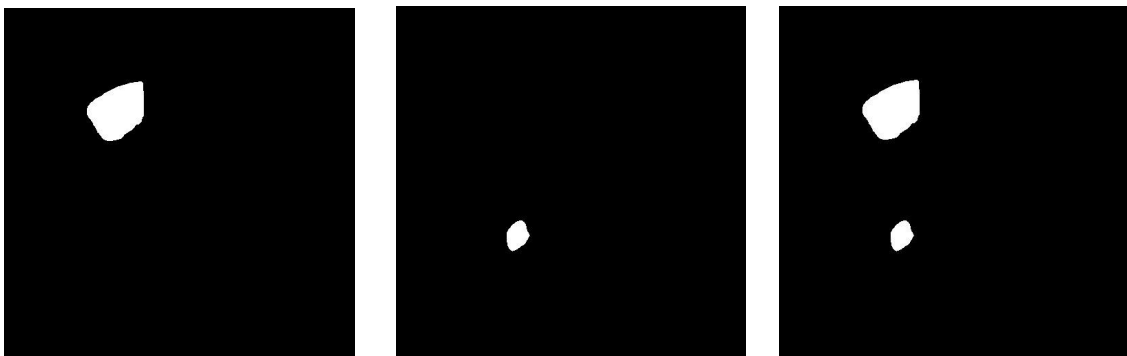
Data Preprocessing

Data preprocessing in Machine Learning is a crucial step that helps enhance the quality of data to promote the extraction of meaningful insights from the data. Data preprocessing in Machine Learning refers to the technique of preparing (cleaning and organizing) the raw data to make it suitable for building and training Machine Learning models. In simple words, data preprocessing in Machine Learning is a data mining technique that transforms raw data into an understandable and readable format

When it comes to creating a Machine Learning model, data preprocessing is the first step marking the initiation of the process. Typically, real-world data is incomplete, inconsistent, inaccurate (contains errors or outliers), and often lacks specific attribute values/trends. This is where data preprocessing enters the scenario – it helps to clean, format, and organize the raw data, thereby making it ready-to-go for Machine Learning models

Merging Liver Tumors

Since the 3D-IRCADb01 Dataset contains tumor masks for every tumor on its own, we needed to merge all the different masks into 1 mask to facilitate the training and Augmentation of data figures



Mask for the first tumor + Mask for the second tumor = Mask for the merged tumor

Hounsfield Windowing

Hounsfield Unit (HU) is the average of the attenuation values of a certain voxel compared to the attenuation value of distilled water at standard temperature and pressure where the HU of water is zero and air is -1000. It is encoded in 12 bits thus has 212 values which is 4096 ranging from -1024 .HU to 3071 HU table 3.1

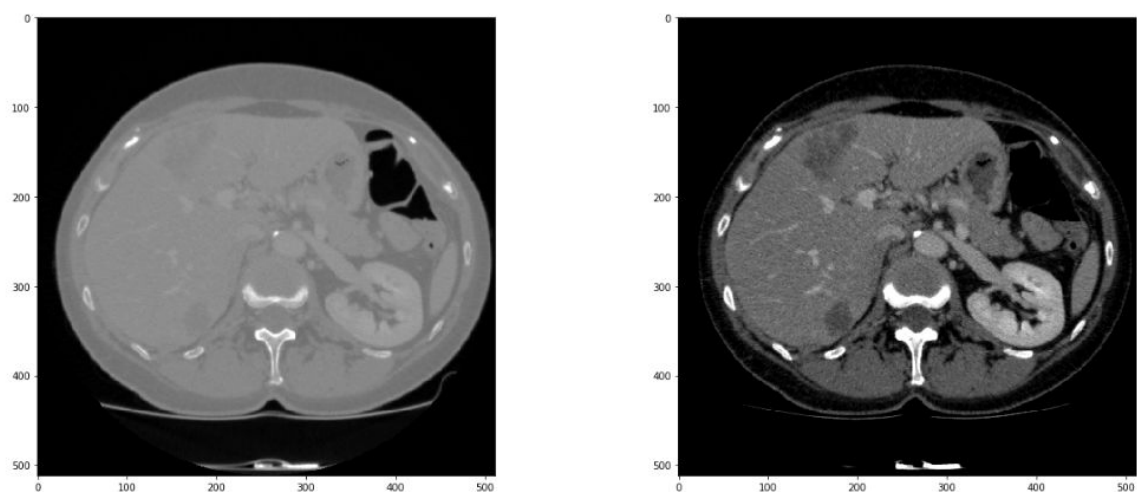
It was named after the inventor of CT-scanning Sir Godfrey Newbold Hounsfield, and it's computed for any tissue as follows where μ is the linear attenuation coefficient

$$HU[9] = \frac{1000 \times (\mu_{tissue} - \mu_{H_2O})}{\mu_{H_2O}}$$

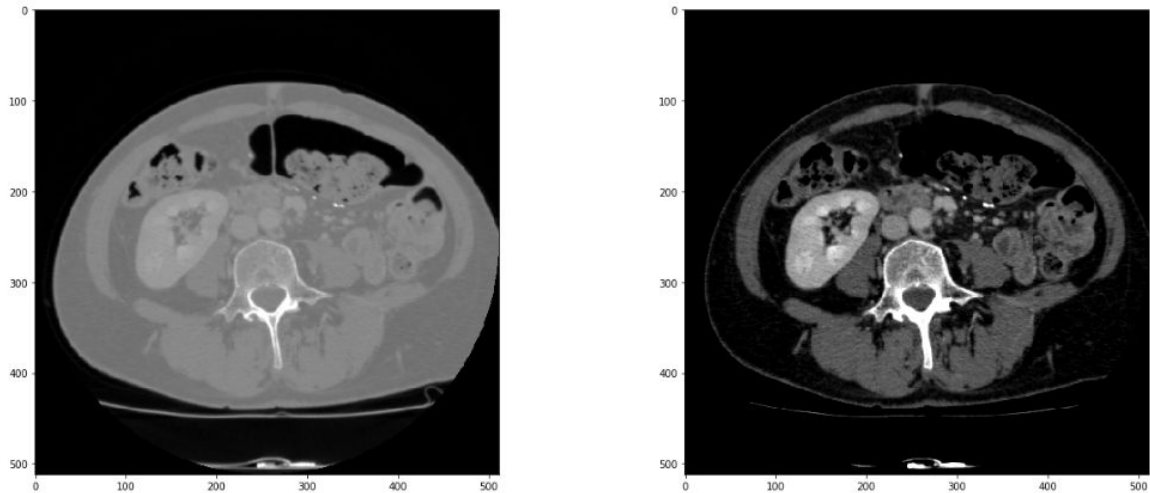
The HU values assigned to each pixel are computed by assigning a grey-scale intensity to each value, a higher value means brighter pixels. After slides are read in DICOM format, Hounsfield Windowing was applied to ranges [-100, 400]. We can see more details after Hu windowing figures

HU values for body organs	
Bone	1000
Liver	40 to 60
White Matter	46
Grey Matter	43
Blood	40
Muscle	10 to 40
Kidney	30
Cerebrospinal	15
Water	0
Fat	-50 to -100
Air	-1000

Table 3.1: Hounsfield Unit values for body organs [7]



CT slice before and after HU windowing



CT slice not containing the liver before and after HU windowing

Resampling

A scan may have a pixel spacing of $[2.5, 0.5, 0.5]$, which means that the distance between slices is 2.5 millimetres. For a different scan, this may be $[1.5, 0.725, 0.725]$, this can be problematic for automatic analysis (e.g. using ConvNets)!

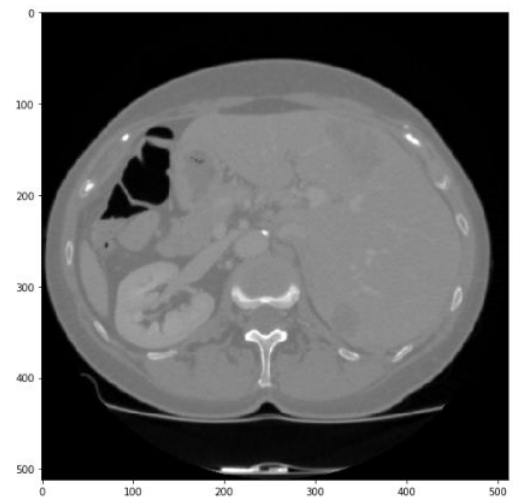
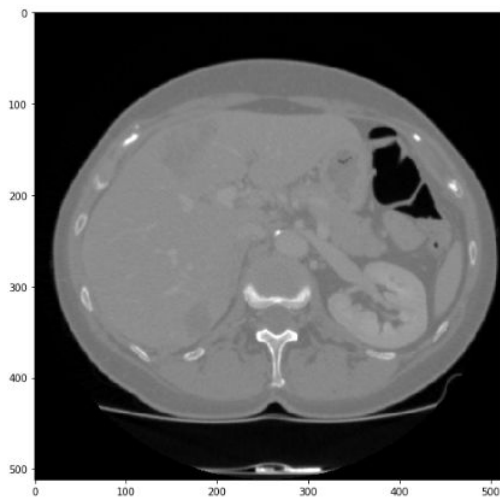
A common method of dealing with this is resampling the full dataset to a certain isotropic resolution. If we choose to resample everything to $1\text{mm}1\text{mm}1\text{mm}$ pixels we can use 3D Conv-Nets without worrying about learning zoom/slice thickness invariance. Whilst this may seem like a very simple step, it has quite some edge cases due to rounding. Also, it takes quite a while [1].

Normalization

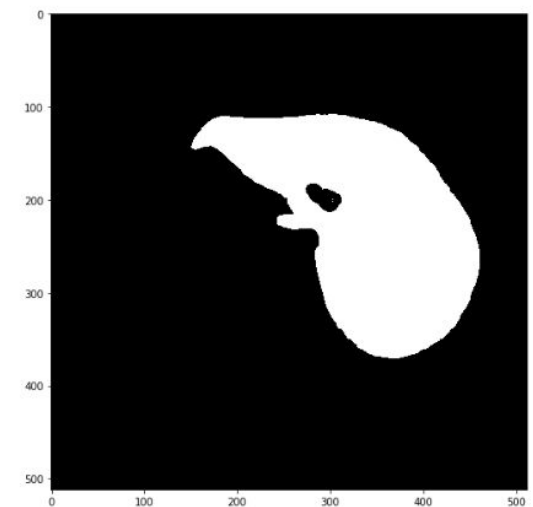
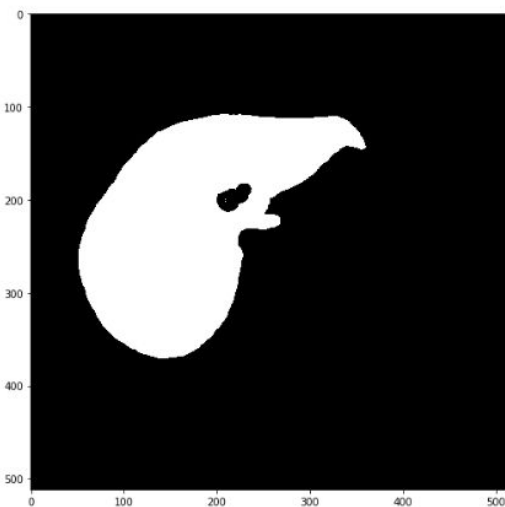
Our values currently range from -1024 to around 2000. Anything above 400 is not interesting to us, as these are simply bones with different radiodensity. A commonly used set of thresholds in the LUNA16 competition to normalize between are -1000 and 400 [1].

Reflection

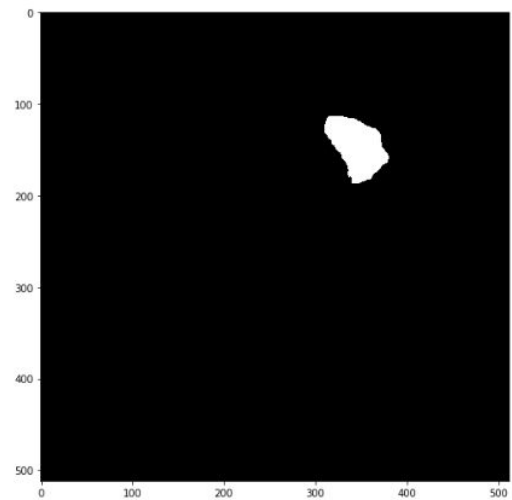
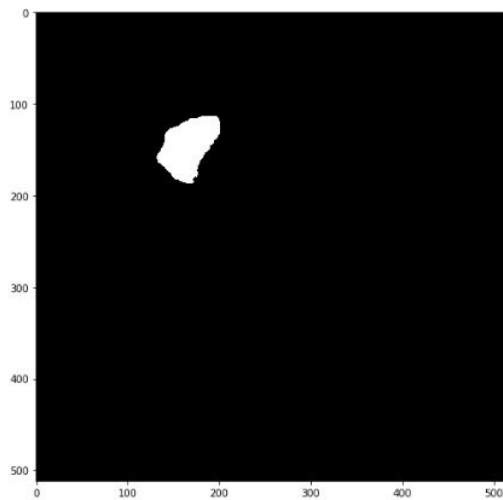
We reflect every slice that contains a tumor along the y-axis along with both the liver mask and tumor mask to increase the number of infected slices figure



Slice before & after reflection



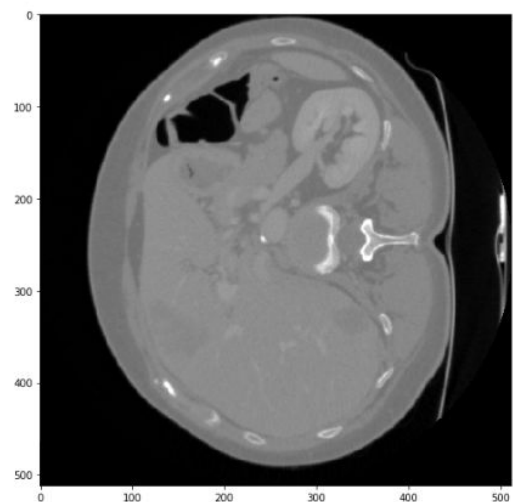
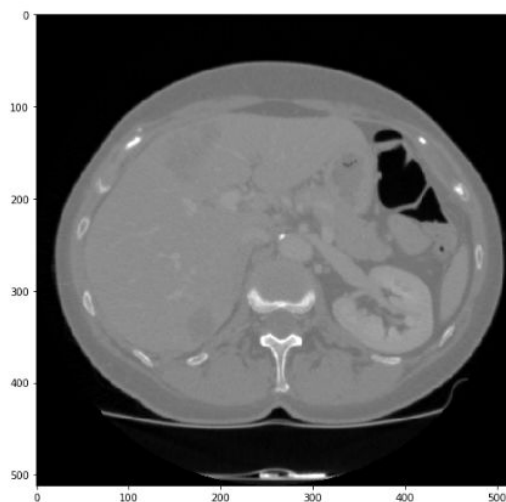
Liver mask before & after reflection



Tumor mask before & after reflection

Rotation

We rotate every slice that contains a tumor along with both the liver mask and tumor mask to increase the number of slices figures



Slice before & after rotation by 90

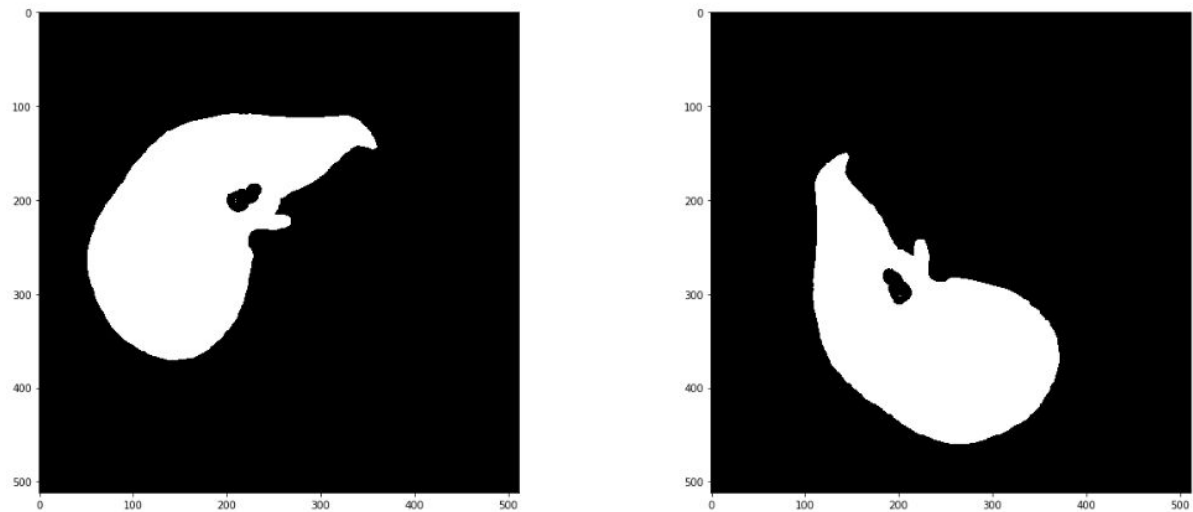


Figure Liver mask before & after rotation by 90

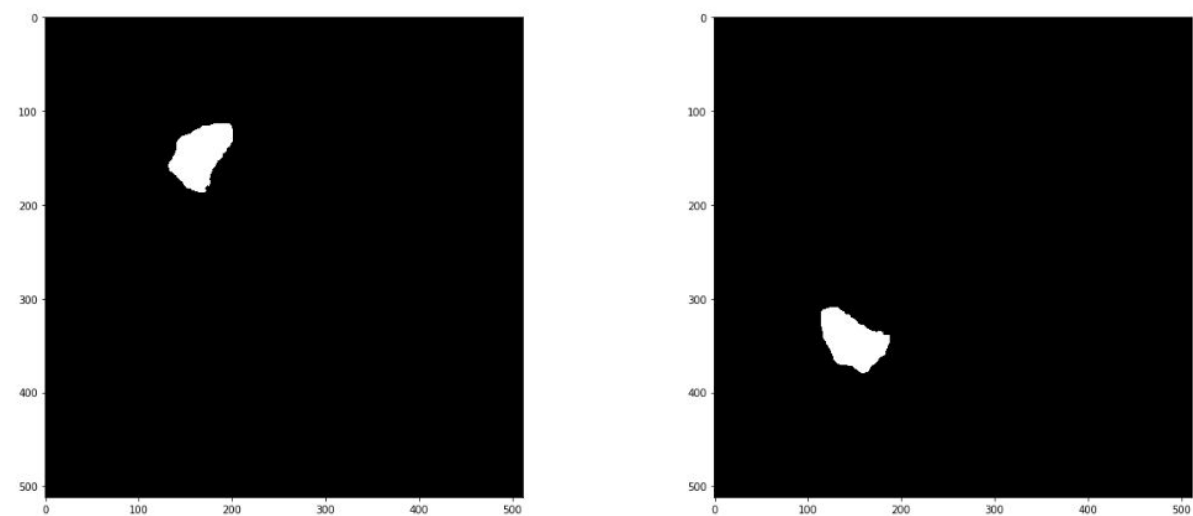


Figure Tumor mask before & after rotation by 90

Liver Segmentation

We use a CNN for the sole purpose of Liver Segmentation. It is trained on CT scans and Liver masks to be able to detect the Region Of Interest (ROI) to segment the liver and mask the neighbouring organs as we have no interest in them. Examples in figures

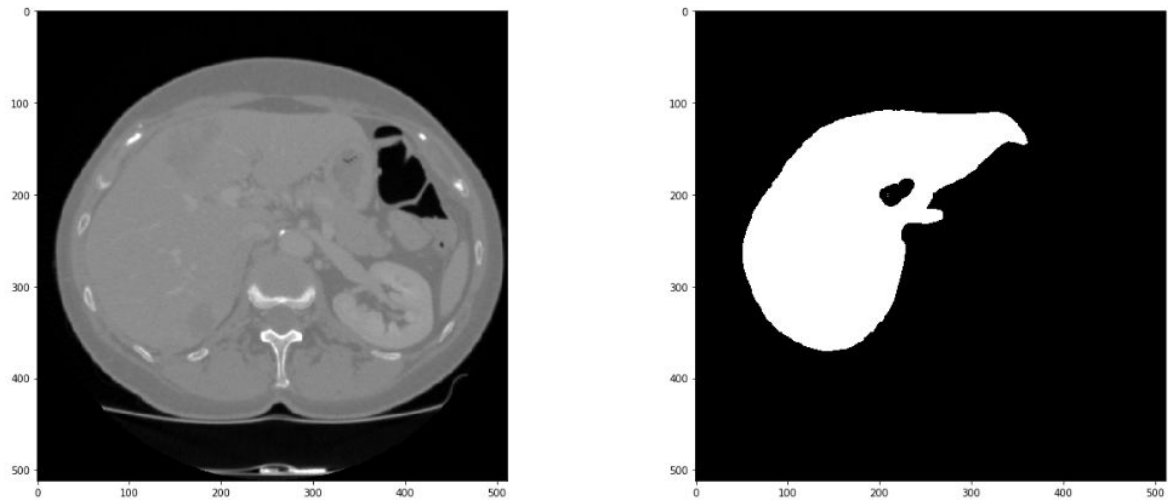


Figure Liver Segmentation example

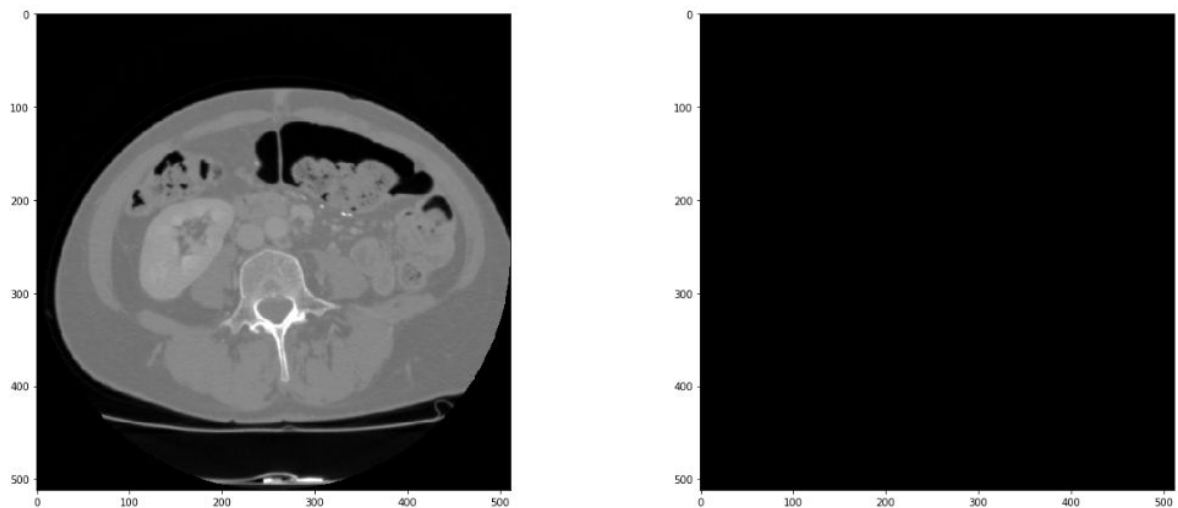


Figure Liver Segmentation without liver

Model Selection

1-U-Net

Overview

U-Net is a Convolutional Neural Network (CNN) developed by the Computer Science Department at the University of Freiburg, Germany especially for the segmentation of bio-medical images.

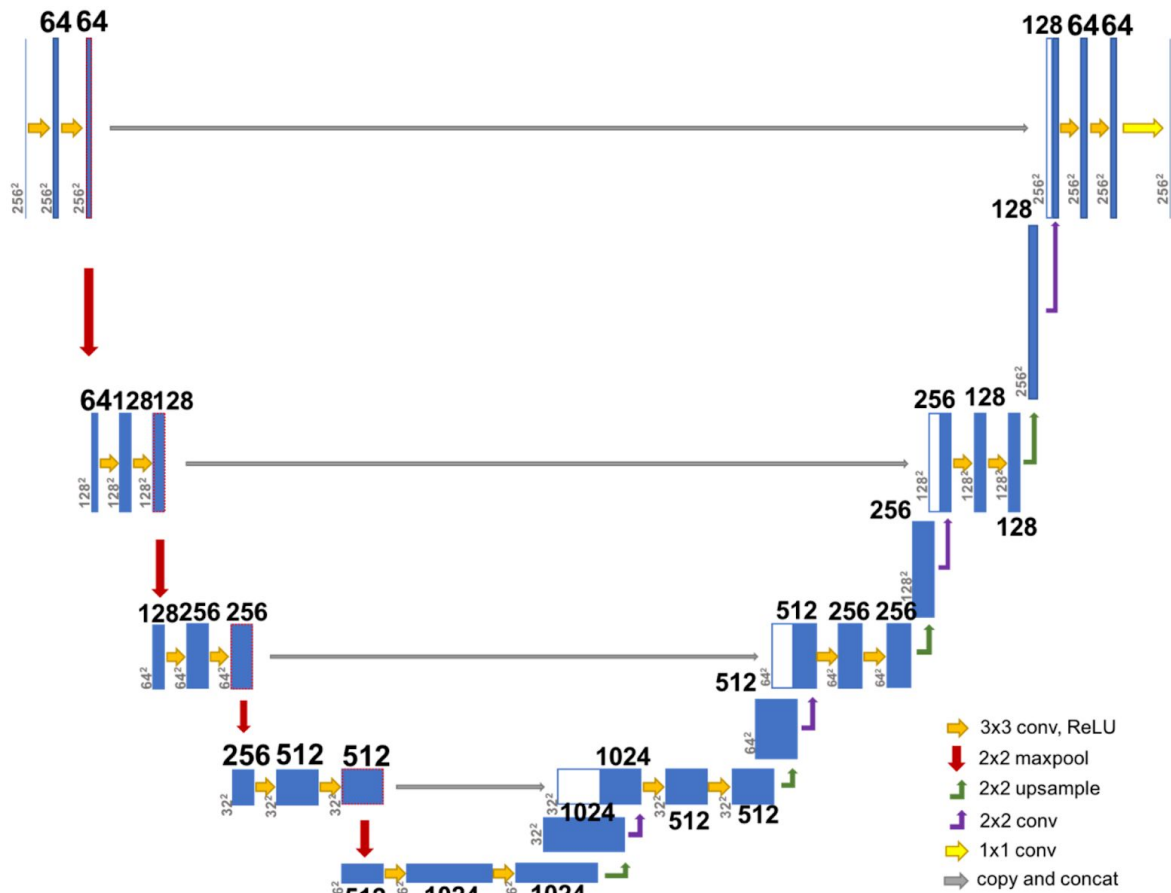
U-Net Architecture

The network architecture is shown in figure, It consists of a contracting path and an expansive path and it consists of a total of 23 convolution layers[16]. Every step in the Contracting Path shares a typical architecture of CNNs which consists of

- Repeated two 3x3 convolutions
- A Rectified Linear Unit (ReLU) refers to following every convolution.
- 2x2 max pooling operation with a stride of 2.
- Downsampling where the number of feature channels is doubled

Every Step in the Expansive Path consists of

- Upsampling of the feature map
- 2x2 convolution which decreases the size of feature maps by half
- Concatenation with the cropped feature map
- Two 3x3 convolutions
- ReLU activation function

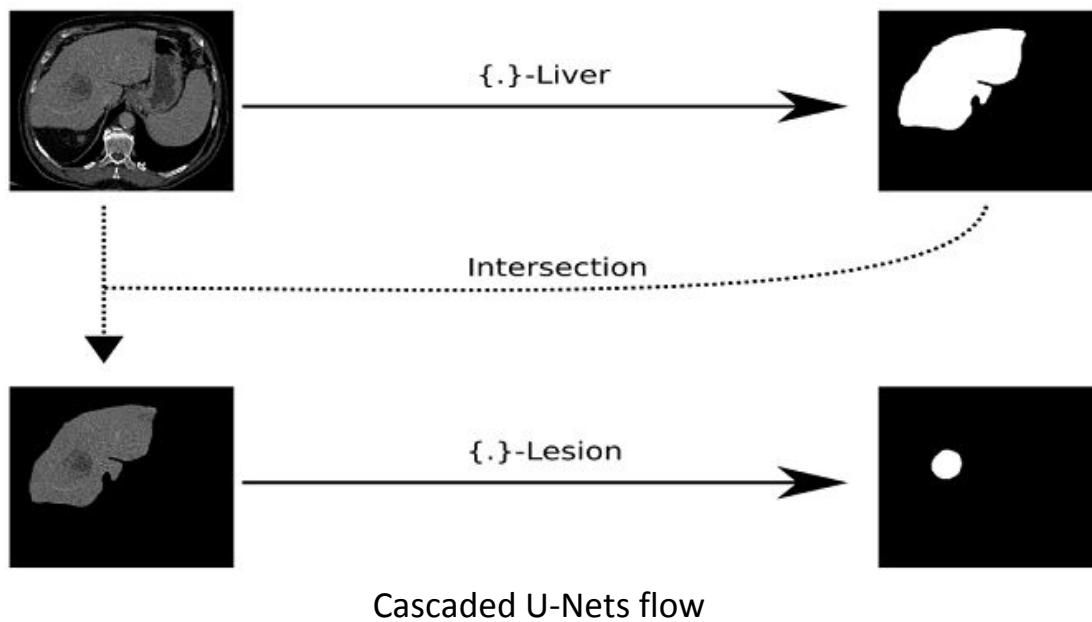


Architecture of the U-Net CNN

Cacaded U-Net

Overview

Cascaded U-Net is just two U-Nets applied after one another basically we segment the liver from the patient first and from the liver we can apply the other U-Net to focus and try to segment the tumor see in figure



2. Image Processing

Introduction

An image processing technique is the usage of a computer to manipulate the digital image. This technique has many benefits such as elasticity, adaptability, data storing, and communication. With the growth of different image resizing techniques, the images can be kept efficiently. This technique has many sets of rules to perform into the images synchronously. The 2D and 3D images can be processed in multiple dimensions. The image processing techniques were founded in the 1960s. Those techniques were used for different fields such as Space, clinical purposes, arts, and TV image improvement. In the 1970s with the development of computer systems, the cost of image processing became less and faster. In the 2000s, the image processing became quicker, inexpensive, and simpler.

it is a method to perform some operations on an image, to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image.

Image processing includes the following three steps:

- Importing the image via image acquisition tools;
- Analyzing and manipulating the image;
- Output in which results can be altered image or report that is based on image analysis.

Liver Segmentation

The process of extracting the liver is a very complex process, with many organs, not just the liver itself. According to many experiments, the location of the liver is in the upper right side of the abdomen and occupies the largest area between the various organs included in the abdomen image.

In the first step, using the basic information of the liver and histogram analyzer, the threshold is fixed to extract the liver pixels from the CT abdominal image which usually contains other organs including the stomach, kidney, spleen, pancreas, etc.

In the second step, Morphological operations like closing and opening operations are used to extract the liver only and not all neighbouring organs and to remove the small fragments of other organs adjacent to the liver with the same intensity as that of the liver.

The following approaches to segment the liver and tumor region with more affluence and accurate manner.

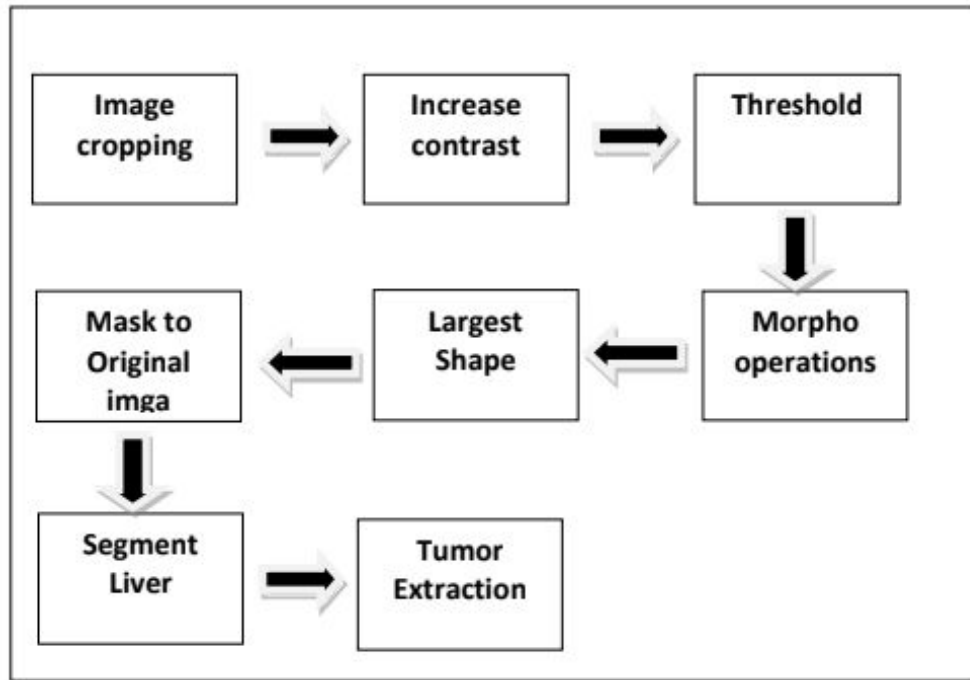
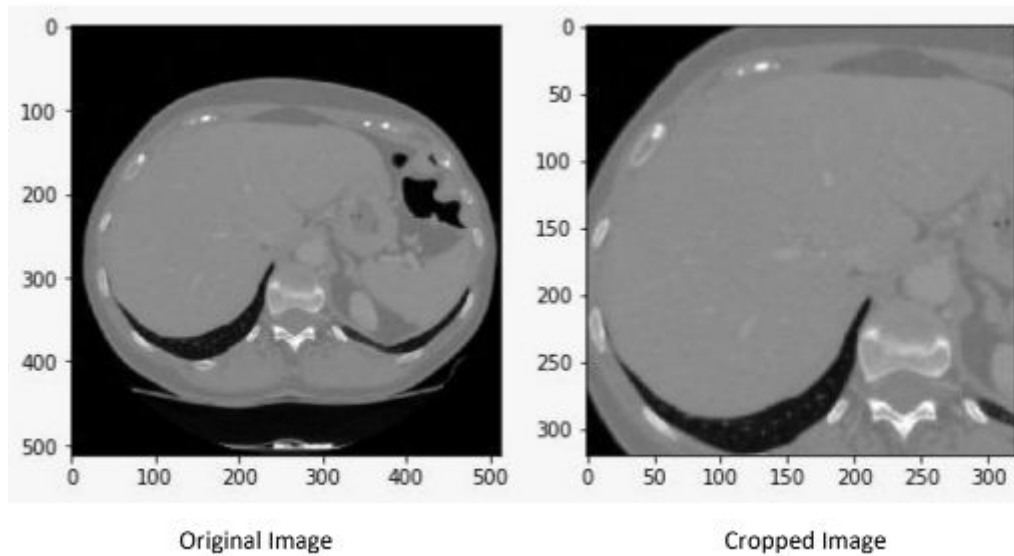


Image Cropping

Cropping an image extracts a rectangular region of interest from the original image. This focuses the viewer's attention on a specific portion of the image and discards areas of the image that contain less useful information.

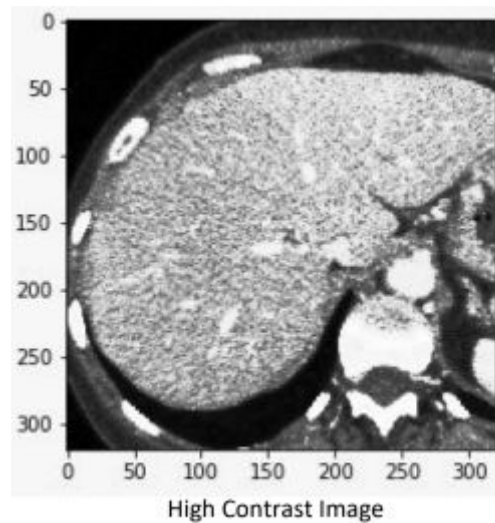
Each image is cropped before any operation on it. The main reason for this is to focus only on the area in which the liver is concentrated and also reduce the time to perform operations using a smaller image so we cropped the image and removed the last 150 columns and the last 100 rows. as shown, the cropped image focuses only on the liver



Histogram Equalization

Histogram equalization makes optimal use of an available grayscale to display an image, and its use could circumvent the problem of selecting specific window settings for each image.

Histogram Equalization works by improving the contrast of images by stretching the intensity distribution of the image. The change is quite noticeable and it can be seen that the image detail is more apparent and that the tumor and blood vessels can be seen much more easily than before.



Threshold

Thresholding is a type of *image segmentation*, where we change the pixels of an image to make the image easier to analyze. In thresholding, we convert an image from color or grayscale into a *binary image*, one that is simply black and white.

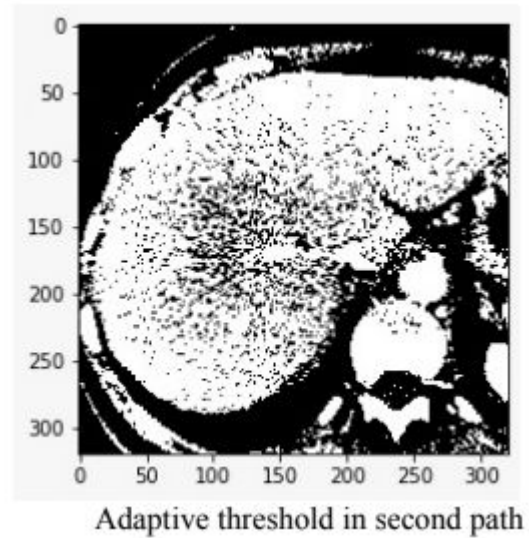
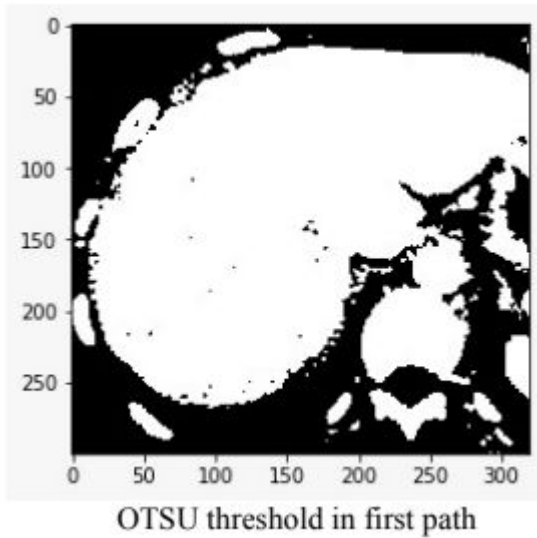
We can't use a fixed threshold because the intensity of the image varies from patient to patient and also depends on the quality of the image, so we use two paths.

In the first path OTSU Threshold :

In global thresholding, we used an arbitrary chosen value as a threshold. In contrast, Otsu's method avoids having to choose a value and determines it automatically.

In second path adaptive Threshold:

adaptive Threshold determines the threshold for a pixel based on a small region around it. So we get different thresholds for different regions of the same image .

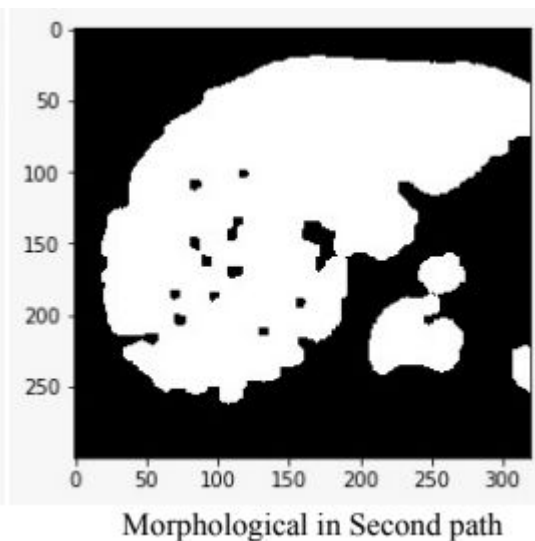
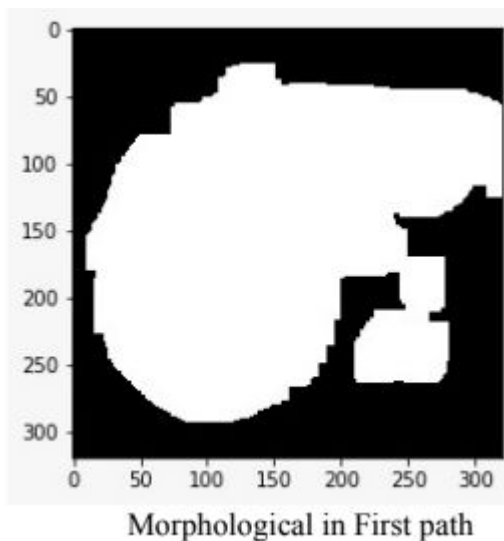


Morphological Operation

Morphological operations are applied to get just the liver and to remove any connected segments to it. We took two paths:

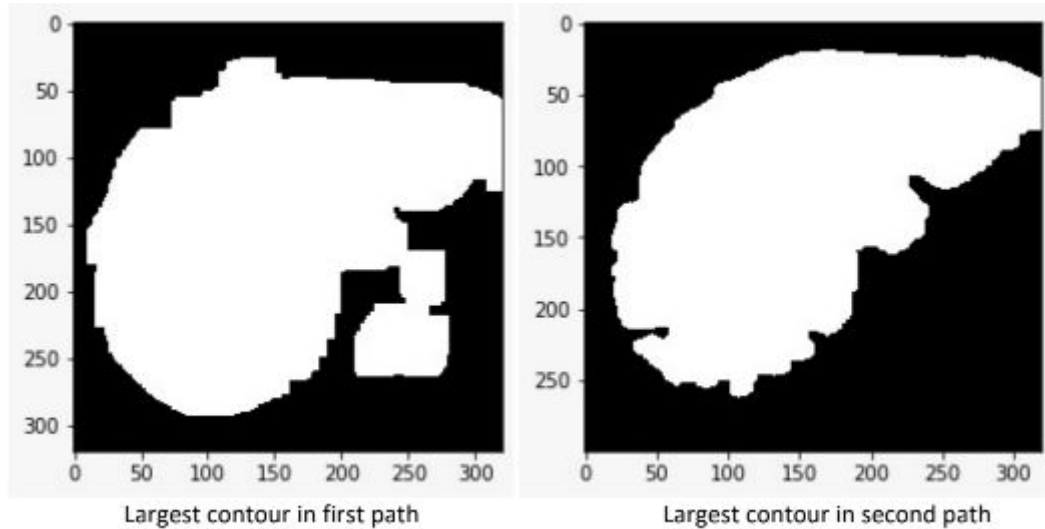
In the first path, we used closing to fill any holes then opening to remove any connections with the liver.

In the second path, we use erosion and dilation with ellipse structure elements.



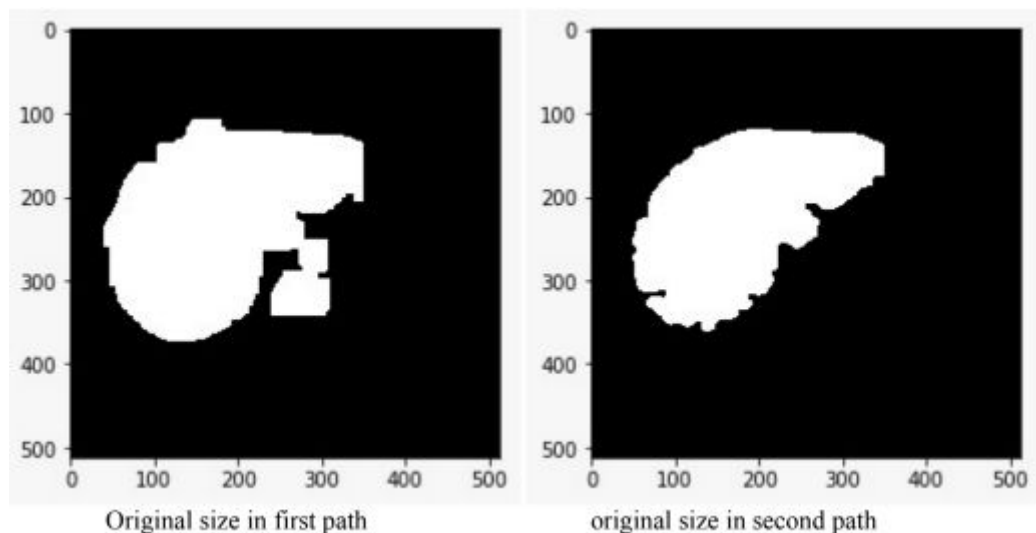
Largest Shape

We used the fine contours function in OpenCV which gets the contours in the image. then we get areas of all contours and get the largest area which is the liver.



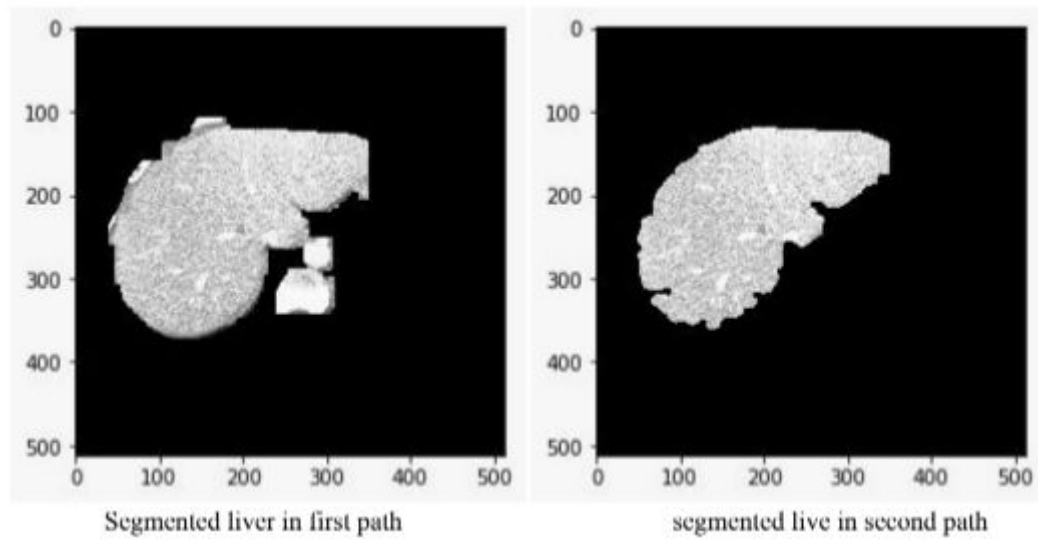
Mask To Original Size

we made a mask with the size of the original image so we can return to the size of the original image

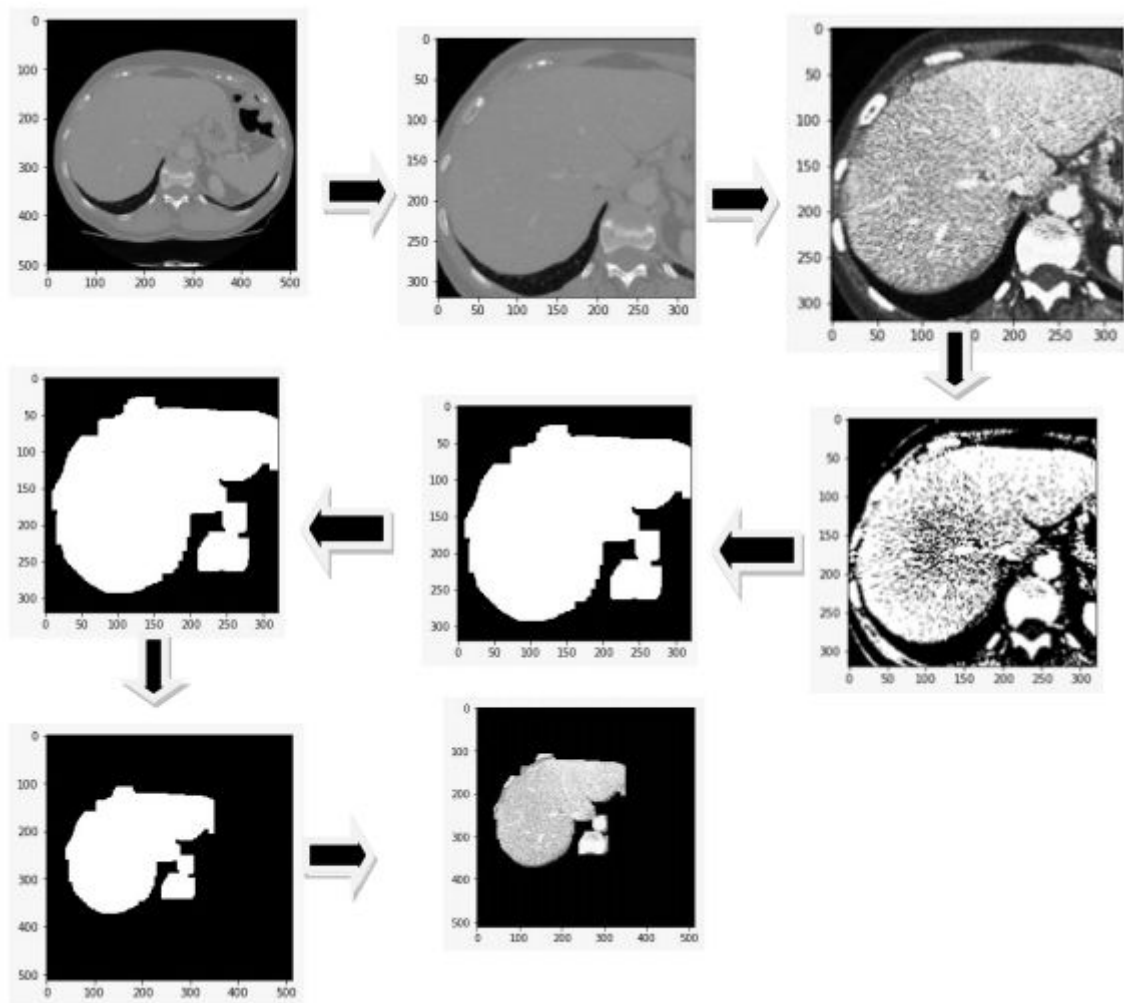


Segment Liver

we multiply the masked image to the original image to get just the liver



Experiment



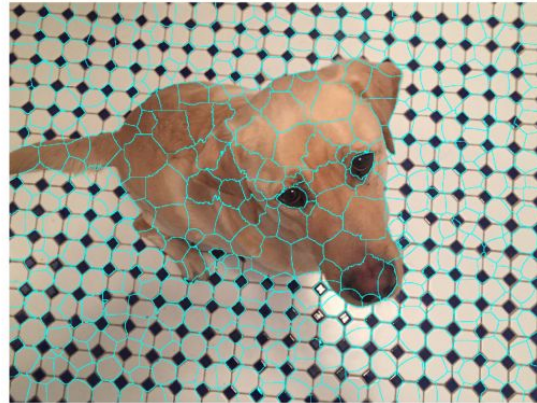
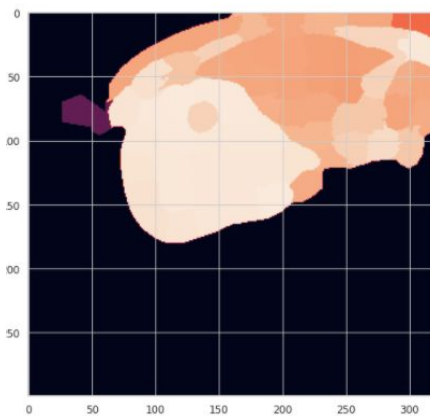
Tumor Segmentation

Introduction

As for liver segmentation the process of extracting the tumor is a very complex process with many steps like cropping, histogram equalization, thresholding and the most important step (Superpixel).

A superpixel can be defined as a group of pixels that share common characteristics (like pixel intensity). Superpixels are becoming useful in many Computer Vision and Image processing algorithms like Image Segmentation, Semantic labeling, Object detection and tracking etc because of the following-

- They carry more information than pixels.
- Superpixels have a perceptual meaning since pixels belonging to a given superpixel share similar visual properties.
- They provide a convenient and compact representation of images that can be very useful for computationally demanding problems.



Superpixels in the context of Image Segmentation

Image Segmentation is the process of partitioning an image into multiple segments(set of pixels or superpixels). The goal is to represent the image as something that is more meaningful and easier to analyze. In other words, image segmentation is the process of assigning a label to every pixel in an

image such that pixels with the same label share certain characteristics. Segmentation can be used to locate objects and boundaries (lines, curves etc.) in images. Some of its practical applications are in -

- Medical Imaging
- Object detection (Face, pedestrian detection)
- Recognition Tasks (Face, Fingerprint recognition)
- Video Surveillance etc.

SLIC (Simple Linear Iterative Clustering) Algorithm for Superpixel generation

This algorithm generates superpixels by clustering pixels based on their color similarity and proximity in the image plane. This is done in the five-dimensional [labxy] space, where [lab] is the pixel color vector in CIELAB color space and xy is the pixel position. We need to normalize the spatial distances in order to use the Euclidean distance in this 5D space because the maximum possible distance between two colors in the CIELAB space is limited whereas the spatial distance in the xy plane depends on the image size. Therefore, In order to cluster pixels in this 5D space, a new distance measure that considers superpixel size was introduced which is described below.

Distance Measure

Some useful notations-

N	Number of pixels in the input image
K	Number of Superpixels used to segment the input image
N/K	Approximate size of each superpixel
$S = \sqrt{N/K}$	For roughly equally sized superpixels there would be a superpixel centre at every grid interval S

Algorithm

This algorithm begins by sampling K regularly spaced cluster centers and moving them to seed locations corresponding to the lowest gradient position in a 3×3 neighborhood (This is done to avoid placing them at an edge and to reduce the chances of choosing a noisy pixel). Image gradients are computed as:

$$G(x,y) = \| I(x+1,y) - I(x-1,y) \|^2 + \| I(x,y+1) - I(x,y-1) \|^2$$

where $I(x,y)$ is the lab vector corresponding to the pixel at position (x,y) , and $\| \cdot \|$ is the L2 norm. This takes into account both color and intensity information. Each pixel in the image is associated with the nearest cluster center whose search area overlaps this pixel. After all the pixels are associated with the nearest cluster center, a new center is computed as the average labxy vector of all the pixels belonging to the cluster.

The process of associating pixels with the nearest cluster center and recomputing the cluster center is repeated until convergence. At the end of this process, a few stray labels may remain, that is, a few pixels in the vicinity of a larger segment having the same label but not connected to it. Connectivity can be enforced in the last step of the algorithm by relabeling disjoint segments with the labels of the largest neighboring cluster.

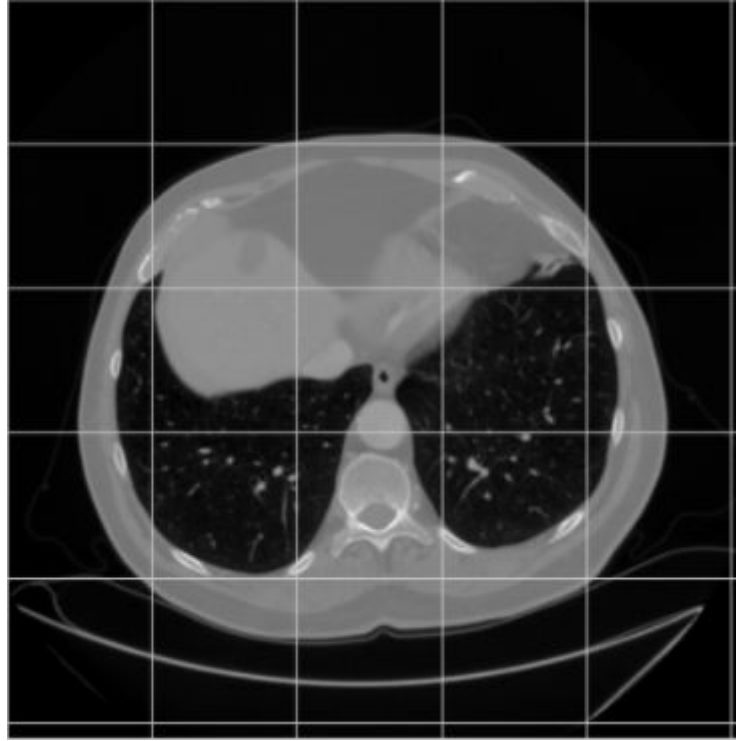
SLIC Algorithm can be summarized as-

Algorithm 1 Efficient superpixel segmentation

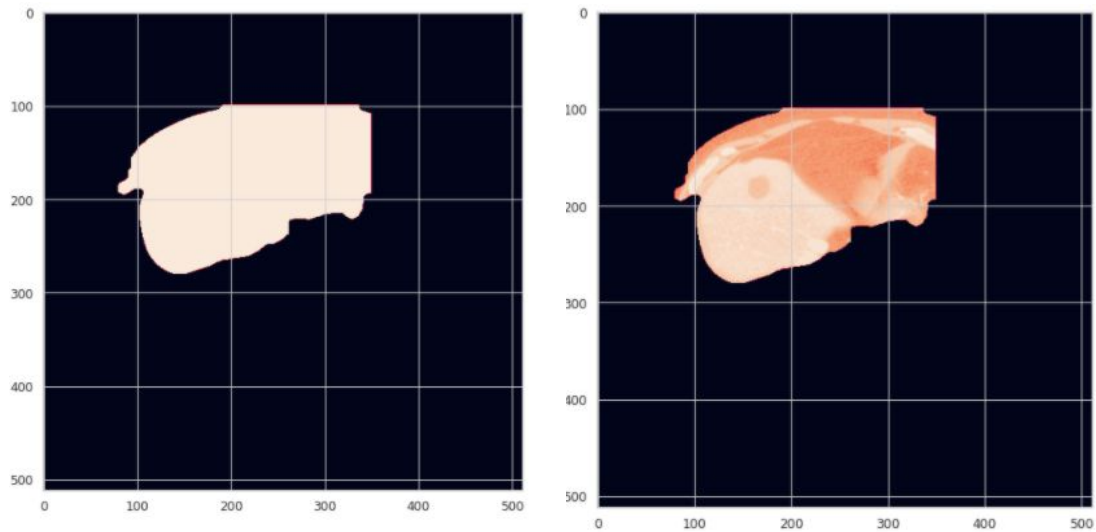
-
- 1: Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S .
 - 2: Perturb cluster centers in an $n \times n$ neighborhood, to the lowest gradient position.
 - 3: **repeat**
 - 4: **for** each cluster center C_k **do**
 - 5: Assign the best matching pixels from a $2S \times 2S$ square neighborhood around the cluster center according to the distance measure (Eq. 1).
 - 6: **end for**
 - 7: Compute new cluster centers and residual error E {L1 distance between previous centers and recomputed centers}
 - 8: **until** $E \leq \text{threshold}$
 - 9: Enforce connectivity.
-

Steps of work

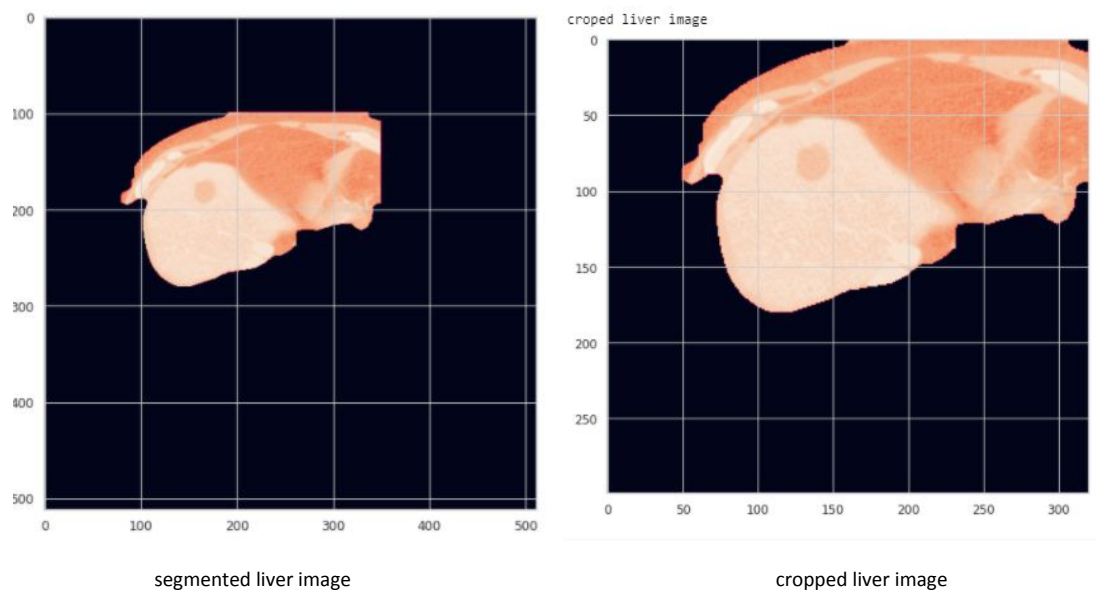
1. Extract the liver and its mask from the patient's abdominal slice with the previous mentioned steps.



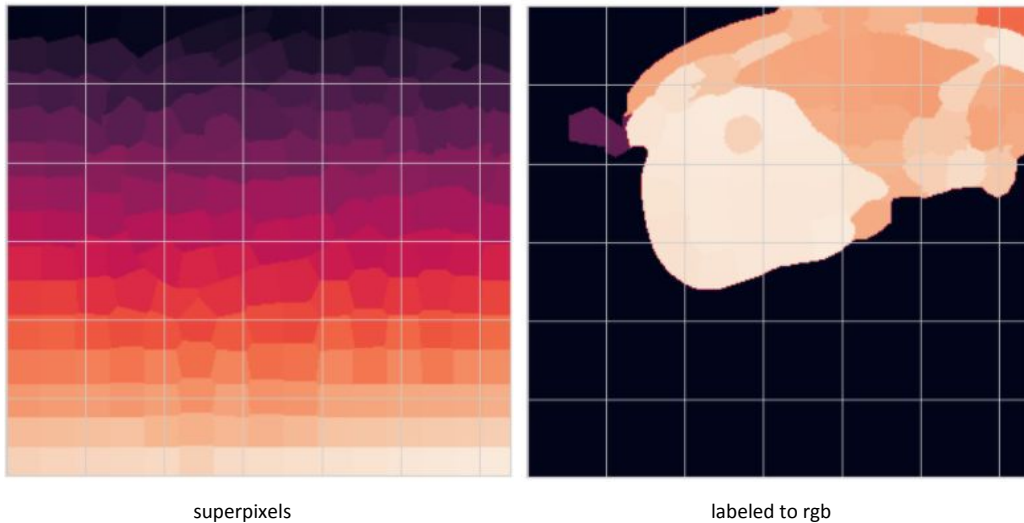
patient 9 abdominal slice 88 in the dataset



2. Each image is cropped before any operation on it. The main reason for this is to focus only on the area in which the liver is concentrated and also reduce the time to perform operations using a smaller image so we cropped the image and removed the last 150 columns and the last 100 rows.

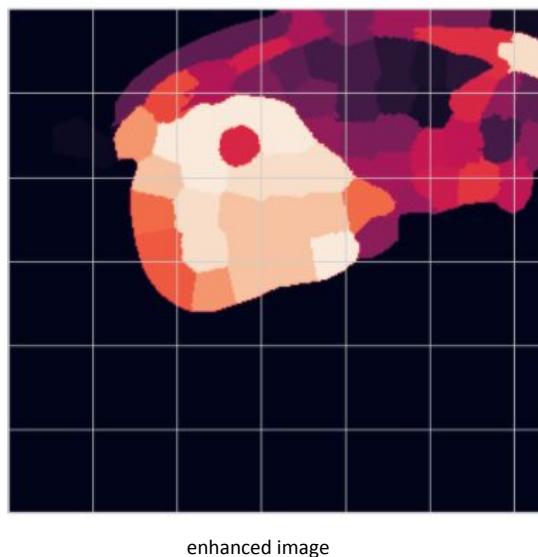


3. Apply the superpixel (slic) algorithm that we described before.



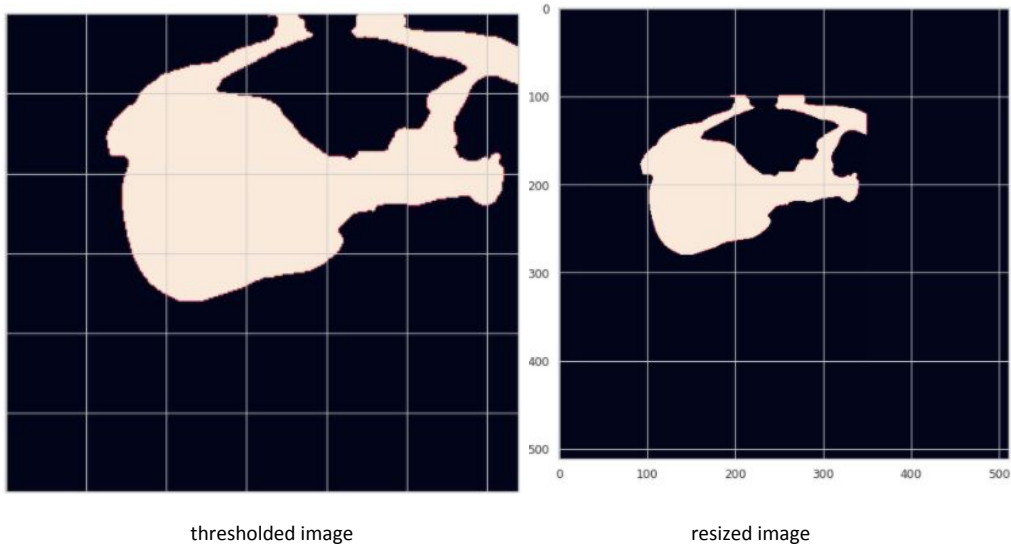
4. Apply Histogram Equalization.

Histogram Equalization works by improving the contrast of images by stretching the intensity distribution of the image. The change is quite noticeable and it can be seen that the image detail is more apparent and that the tumor and blood vessels can be seen much more easily than before.

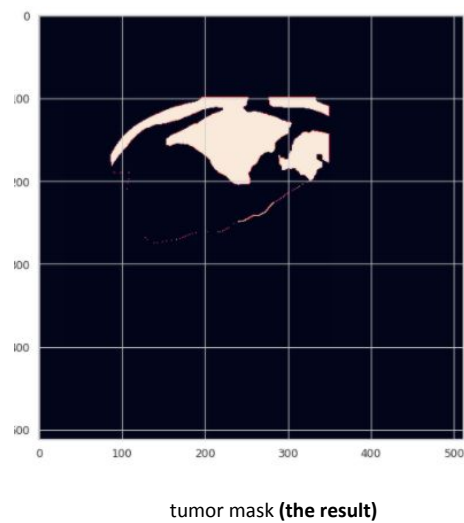
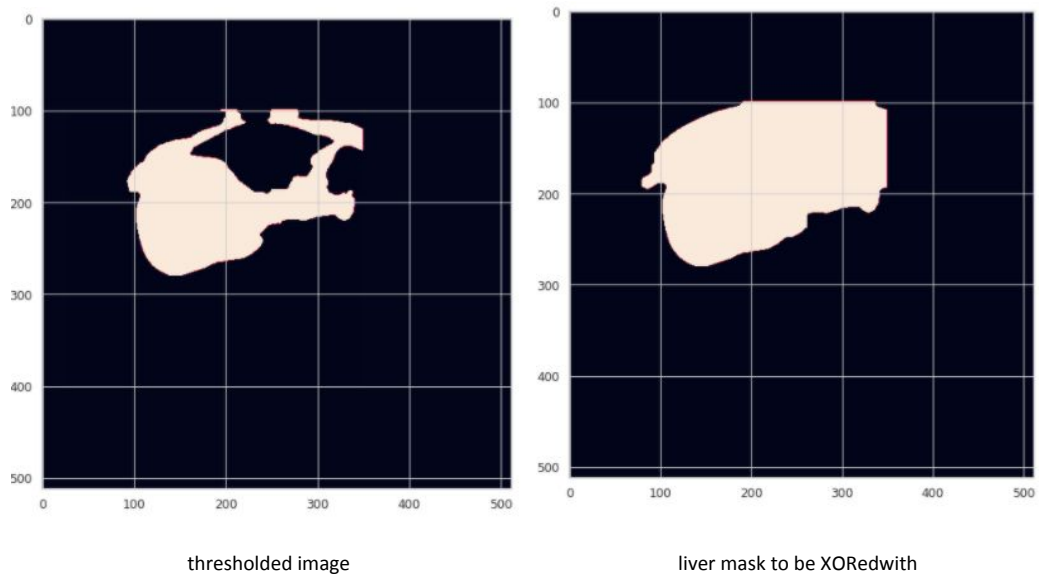


5. Apply thresholding and Mask To Original Size as we did in liver segmentation.

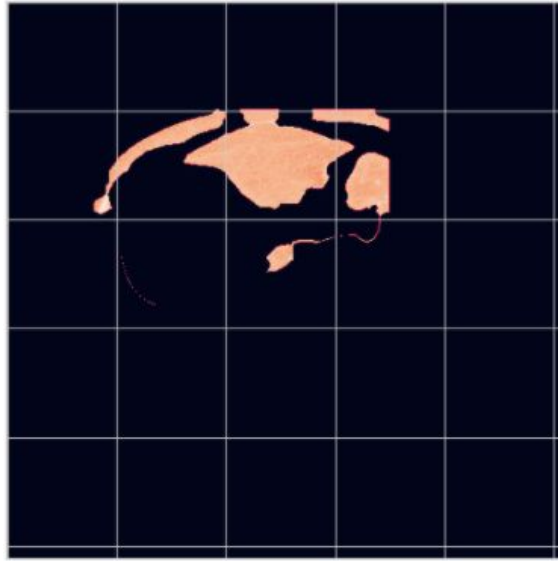
LIVER TUMOR SEGMENTATION IN CT SCANS



6. Applying thresholding will get the liver without the tumor so we XOR it with the liver mask.



7. Finally we apply the mask on the original photo



the Extracted tumor

Results

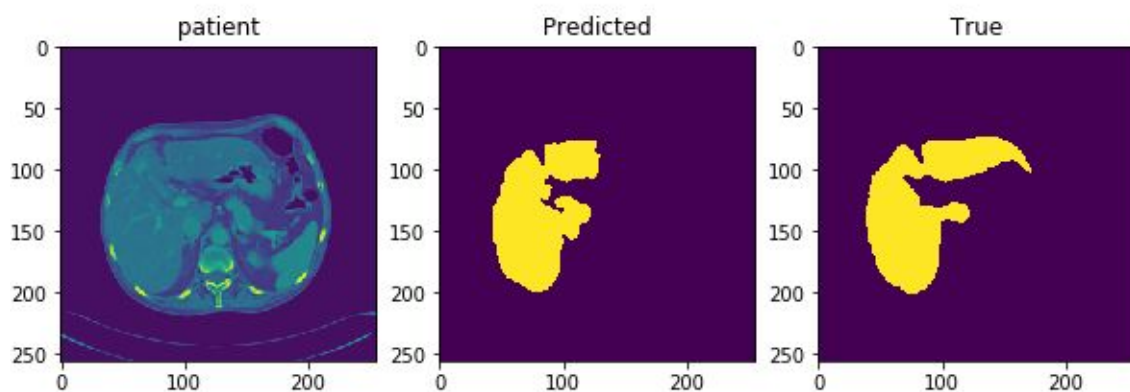
1- Deep Learning

- Cascaded U-nets :

we trained a u-net model to segment the liver first then another u-net model will look at the liver and segment its tumors

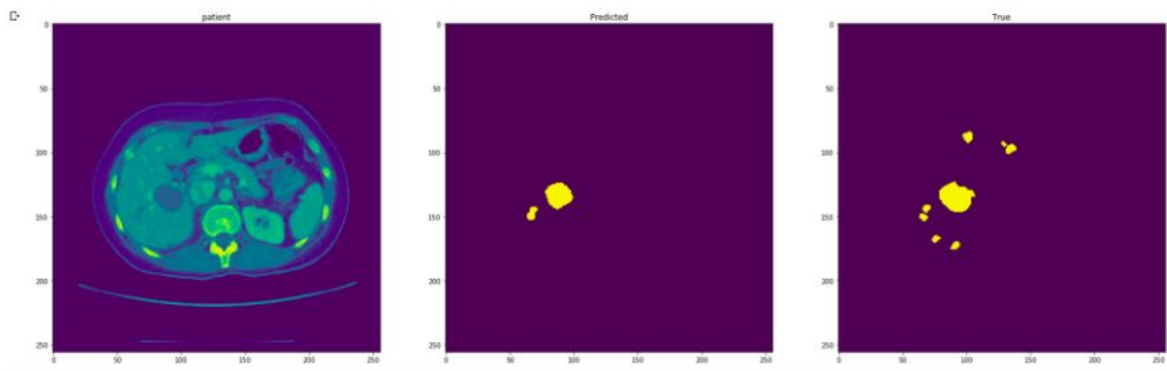
Liver Segmentation:

Liver Training IOU	98.63 %
Liver testing IOU	88.25%



Tumor Segmentation:

Tumor Training IOU	91%
tumor testing IOU	66%

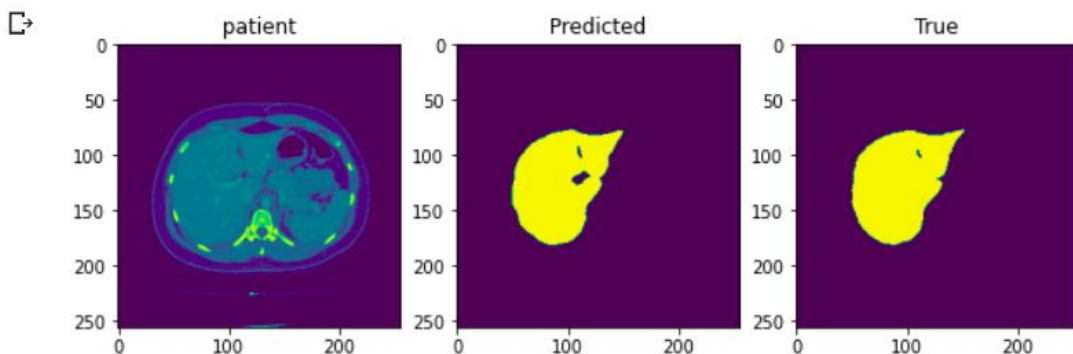


- **Cascaded U-nets (with augmentation):**

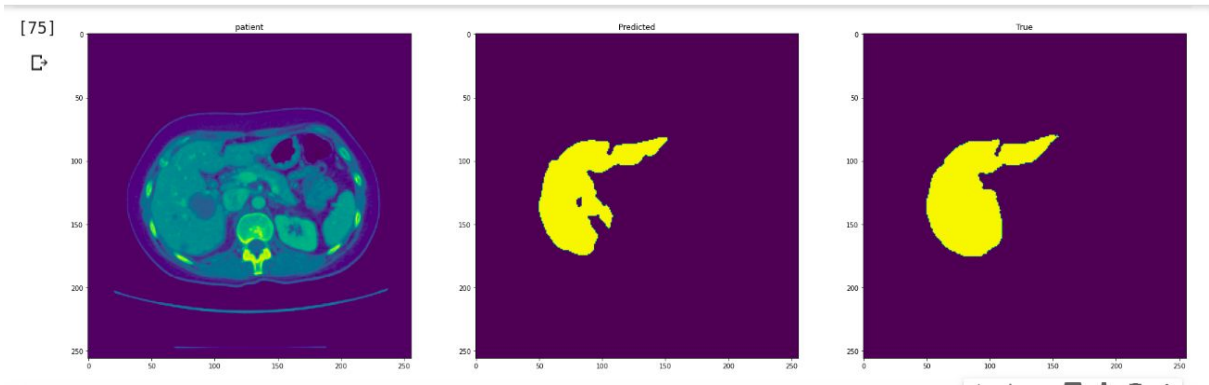
Liver Segmentation:

Liver Training IOU	97.04 %
Liver testing IOU	90.25%

Training Sample:



Testing Sample:

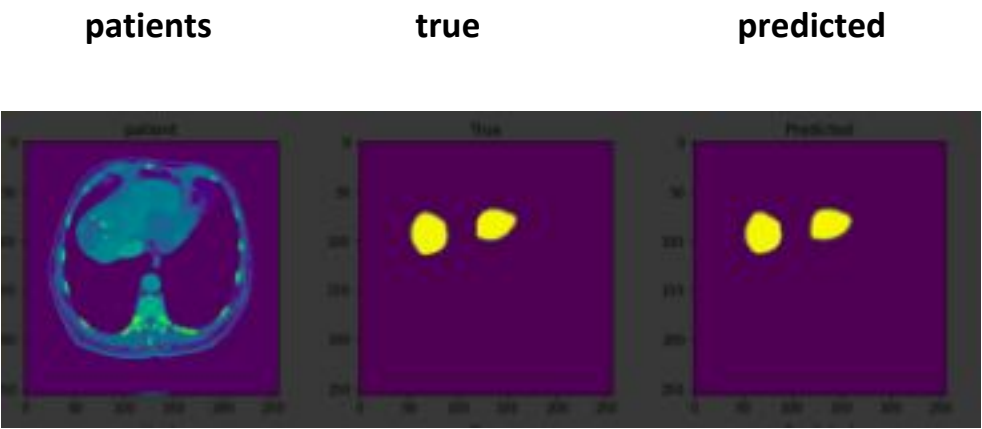


● Direct U-Net:

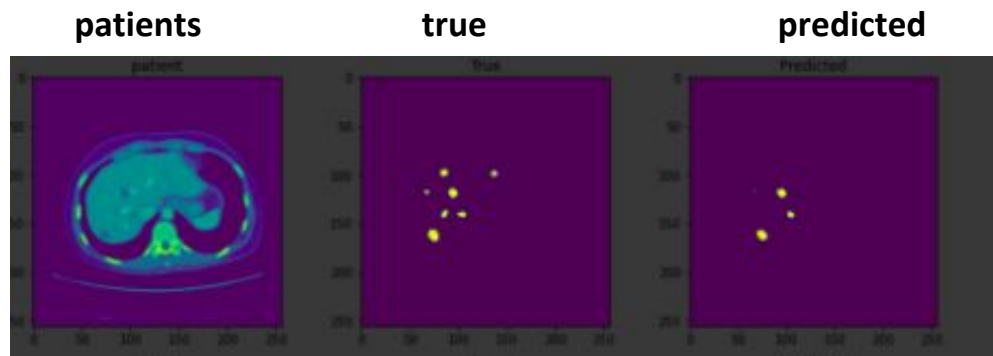
Tumor Segmentation:

Tumor Training IOU	90 %
Tumor testing IOU	71%

Training Sample:



Testing Sample:



2- Image Processing

Liver

- The first path gave a score of 78.2% on 20 patients.
- The second path gave a score of 67% on 20 patients.

Tumor

- 45%.

Conclusion

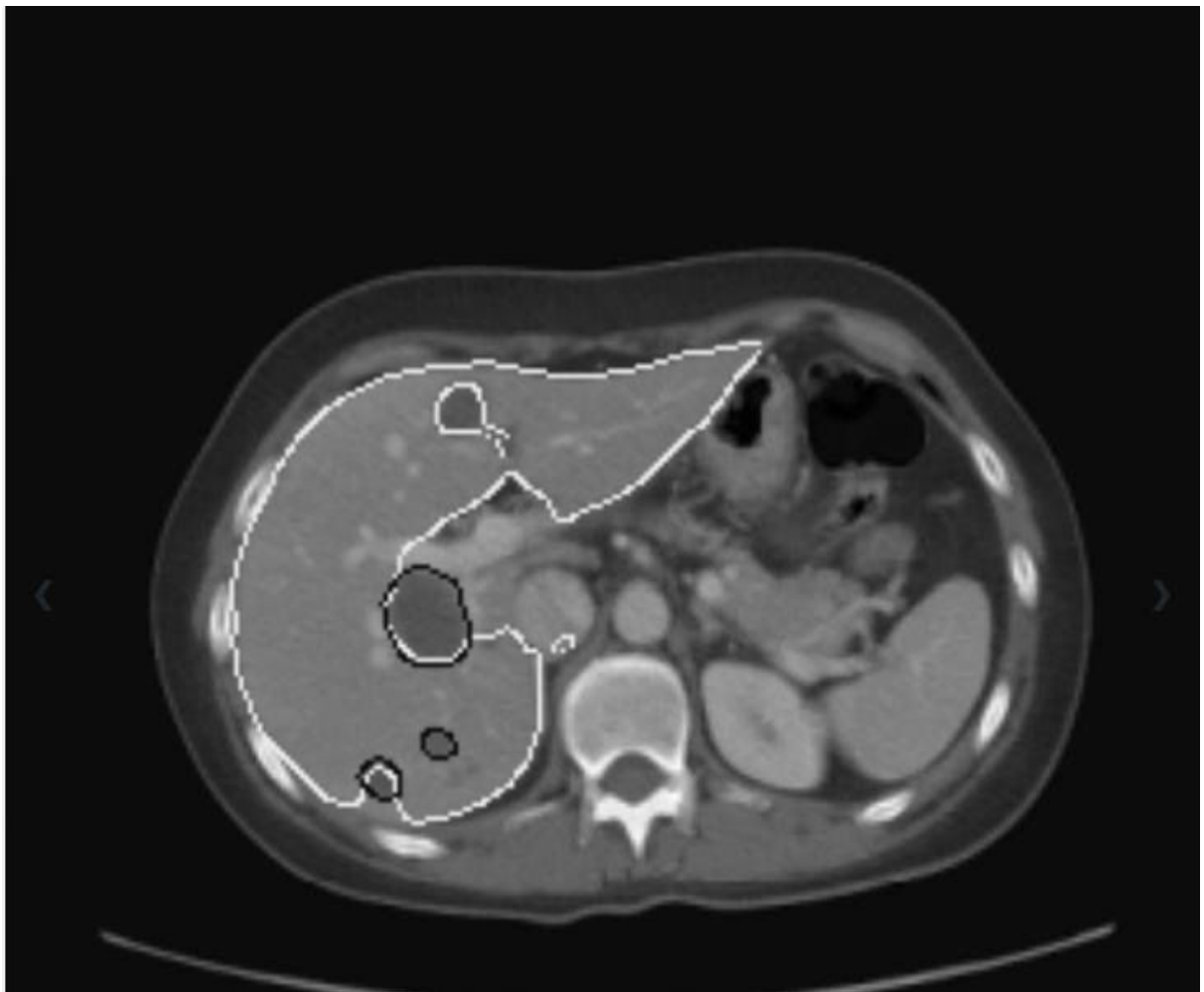
In this work, we describe our methods for liver and tumor segmentation in abdominal CT scans based on a 2D deep neural network and image processing techniques. Our top method achieves segmentation quality for detected tumors comparable to a human expert and can detect above 70% of potentially measurable tumor lesions. We observed that the neural network is capable of detecting bigger lesions overall and in some cases it can get the small ones also our methods suffer when it comes to the top and bottom parts of the liver it can hardly consider them as part of the liver, this happens because 2D methods lose the context in the third dimension and we could have tried to apply 3D methods but we have insufficient data. We presume that the interest in the automatic tumor segmentation will lead to gathering more data which will be crucial for applying 3D techniques. We think that the 3dircadb data collection from multiple patients is a great initiative, but it is clear that more work needs to be done to match the human detection performance. Moreover, an evaluation in a clinical setting will be required to assess the clinical utility of automatic liver tumor segmentation methods.

Pros:

- Dataset was well formatted and clean with no missing folders
- Resample everything to 1mm 1mm 1mm pixels we can use 3D convnets without worrying about learning zoom/slice thickness invariance and obtain certain isotropic resolution.
- We worked with radiodensities ranging from -200 to around 500. Anything above 500 is not interesting to us, as these are simply bones with different radiodensity, and anything below -200 is fat or lung or air.
- Although data is small and imbalanced, deep learning gets better results than image processing, as the automated solution of deep learning.

Cons

- Although the Cascaded U-Net produced really good results on the Liver Segmentation but for our problem some of the tumors were located near the edges of the liver so the liver model considered the edgy parts with tumors not belonging to the liver which lead to making the tumor model not seeing them initially leading to increase in false negatives as shown, Note: we have extracted liver part from cascaded model and tumor part is extracted from direct model.



- dataset is small and it makes task harder and imbalanced
 - 2D models lose context in the third dimension and it leads us not to see liver and tumors which have small size.
 - Augmentation doesn't make a huge difference in results, besides it has more GPU computations and high memory.
 - although we predict that the cascaded model will have better results , but the direct has better results, as the cascaded one had predicted some liver parts wrong.
-

Tools used

We used Python and many tools in our project to be able to segment Liver Tumors, such as:

- Anaconda, a distribution of the Python programming language which eases package management and deployment.
- Numpy, a fundamental package for scientific computation in python, adds support for multidimensional lists and matrices and mathematical functions to work on these matrices.
- Matplotlib, a plotting library for python which helps visualizing data and plotting graphs.
- TensorFlow, an open-source library developed by Google Brain Team for machine learning and deep neural networks research.
- Keras, a High-Level API used in building and training Deep Learning Models. It simplifies the process for non-experts in Machine Learning and Deep Learning and runs on multiple backend engines such as TensorFlow, CNTK and Theano without the need for learning the syntax and how the engines work.

- For training, we used Google Colab as it is a cloud service and now it supports GPU and TPU which is easy to use and available whenever we need.

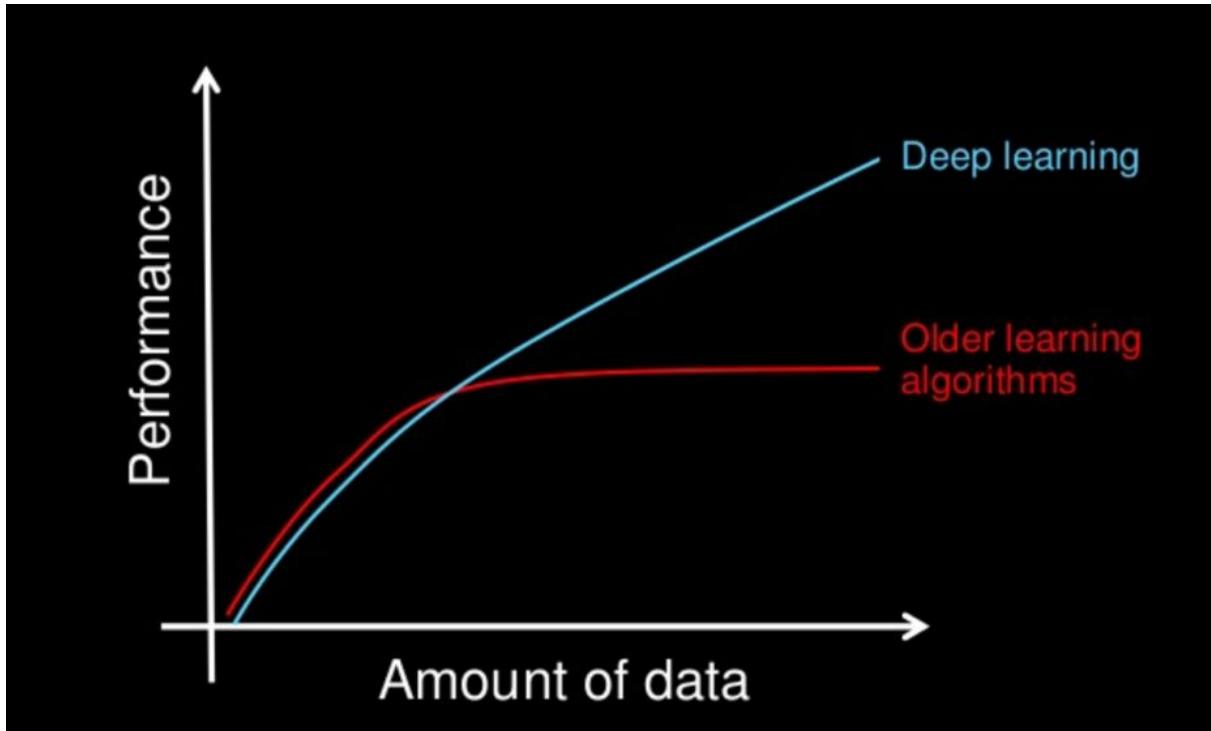
Future Work

- **3D models to capture third dimension context:**

The performance of 2D deep learning far exceeds that of 3D mainly because of the relatively small size of the available 3D datasets [5].

Many works on 3D CNNs (e.g. OctNet, VoxNet) down-sample the 3D space into a grid of voxels of predefined size (voxelization) to overcome the memory challenge when dealing with 3D data. However recent works (e.g. PointNet, PointNet++, SegCloud) have achieved point-level segmentation. I think that 3D deep learning will be more effective than 2D in the future, due to the increased level of information in 3D data, if the efficiency of these algorithms continues to improve at the rate it is [6] .

- **Gathering more data:**



Getting a better accuracy with deep learning algorithms is either due to a better Neural Network, more computational power or huge amounts of data. Eventually you will reach a certain point where you don't have enough data left or where you can't improve the algorithm anymore because it then will take too much time to train. as medical imaging data is not many and not accessible easily especially in cases like cancer CT scans as we saw in our project we had only access to 20 CT scans for patients with only 15 of them having cancer.

Bibliography

- [1] <https://www.kaggle.com/gzuidhof/full-preprocessing-tutorial>
- [2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. CoRR, abs/1505.04597, 2015.
- [3] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. CoRR, abs/1711.10684, 2017.
- [4] Fully automatic liver segmentation combining multi-dimensional graph cut with shape information in 3D CT images
- [5] <https://medium.com/stanford-ai-for-healthcare/dont-just-scan-this-deep-learning-techniques-for-mri-52610e9b7a85>
- [6] Is there a dedicated advantage of 3D deep learning vs 2D for the detection of brain lesions?