मेडी-केप्स विश्वविद्यालय, इन्दौर
Medi-Caps University, Indore

Medical Image Segmentation Using Deep Learning

A Project given by

Bhabha Atomic Research Center(BARC)

(15TH Jan 2020 – 15th June 2020)

for the partial fulfillment of B.Tech degree

Submitted by:

Name – Tanusha Shrivastava

Medicaps University, Indore

Enrollment no.- EN16CS301278

Submitted to:

Mrs. Padmini Sridharan

Designation: Scientific Officer (H+),

Electronics Department

BARC(Mumbai)

Contents

INTRODUCTION

Medical Image Segmentation

- Image segmentation is considered the most essential medical imaging process as it extracts the region of interest (ROI) through a semiautomatic or automatic process.
- It divides an image into areas based on a specified description, such as segmenting body organs/tissues in the medical applications for border detection, tumor detection/segmentation, and mass detection.
- The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images

CONCEPT

Segmentation

In case of medical image segmentation, the aim is to:

- Study anatomical structure
- Identify Region of Interest i.e. locating the body parts, tumor and other abnormalities
- Measure tissue volume to measure growth of tumor (also decrease in size of tumor with treatment)

Segmentation

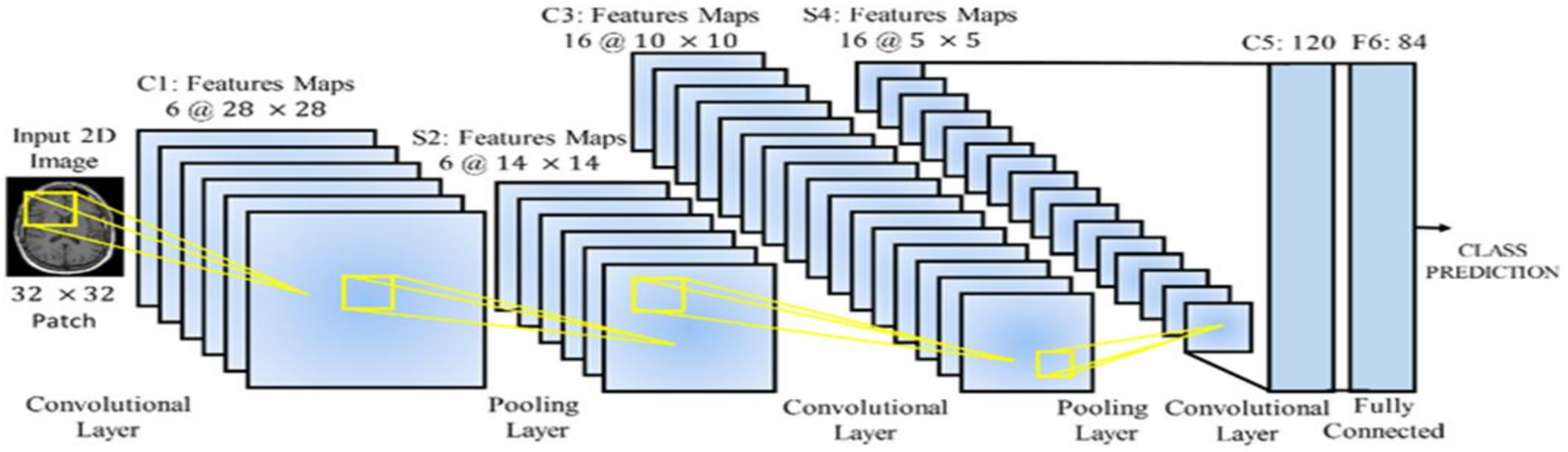
Various types of method can be used to segment medical images:

- Automatic Segmentation
- Edge based Segmentation
- Neural Network based Segmentation
- Region based Segmentation

This project is mainly concerned with Neural Network based Segmentation.

Convolutional Neural network (CNN)

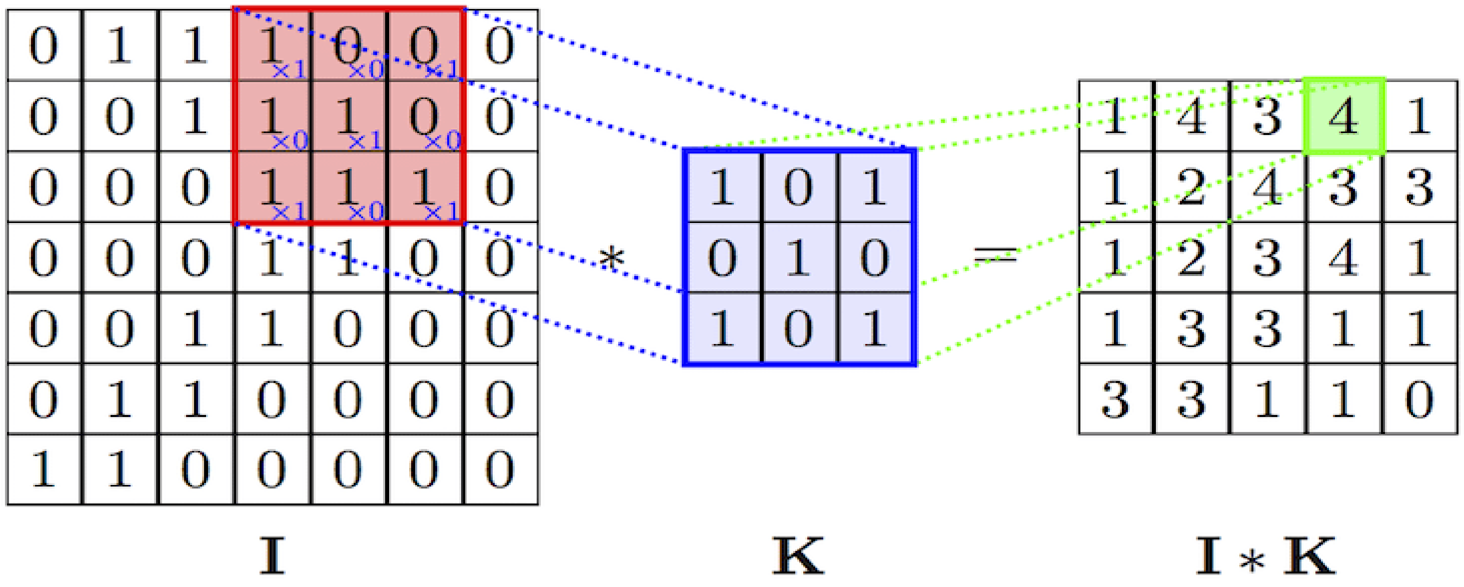
- A convolutional neural network (CNN) is a specific type of artificial neural network that uses perceptrons, a machine learning unit algorithm, for supervised learning, to analyze data. CNNs apply to image processing, natural language processing and other kinds of cognitive tasks.
- A convolutional neural network is also known as a ConvNet.
- It contains 5 layers: Input Layer, Convolutional Layer, Activation function layer, Pooling Layer and Fully Connected layer.



CNN LAYER

1. Convolutional Layer

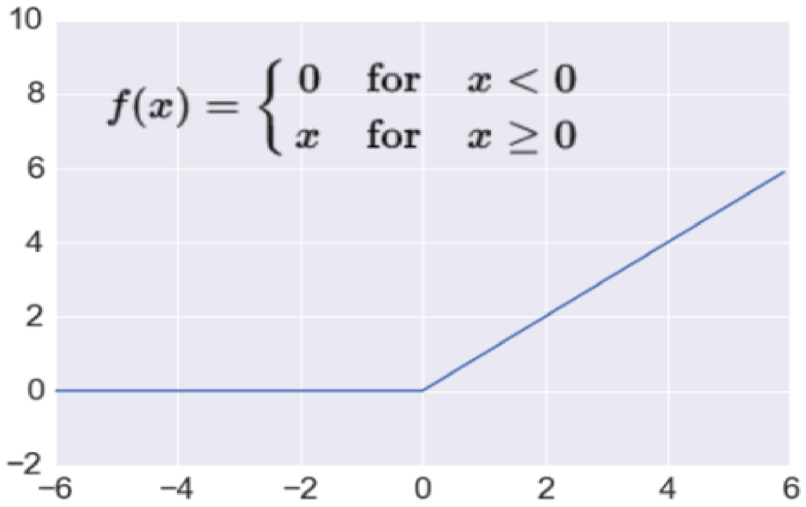
Convolution puts the input images through a set of convolutional filters, each of which activates certain features from the images.



CNN LAYER

2. ReLU Layer

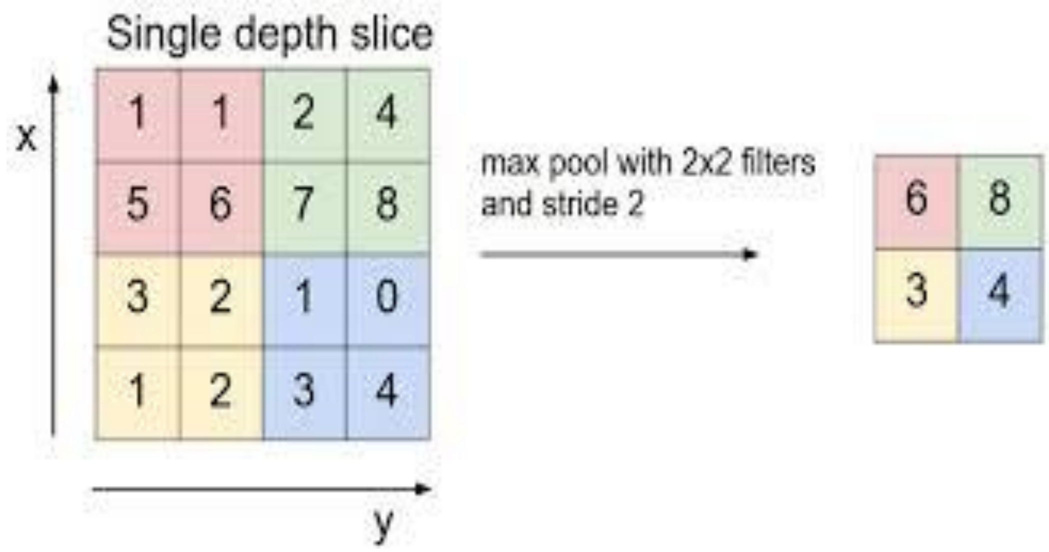
Rectified linear unit (ReLU) allows for faster and more effective training by mapping negative values to zero and maintaining positive values.



CNN LAYER

3.Pooling Layer

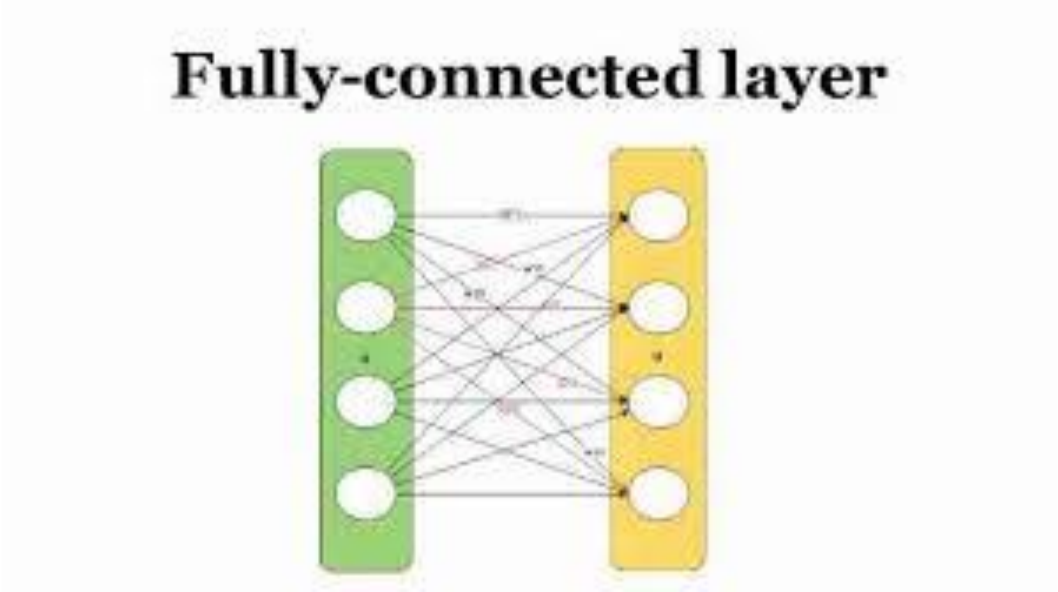
Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently.



CNN LAYER

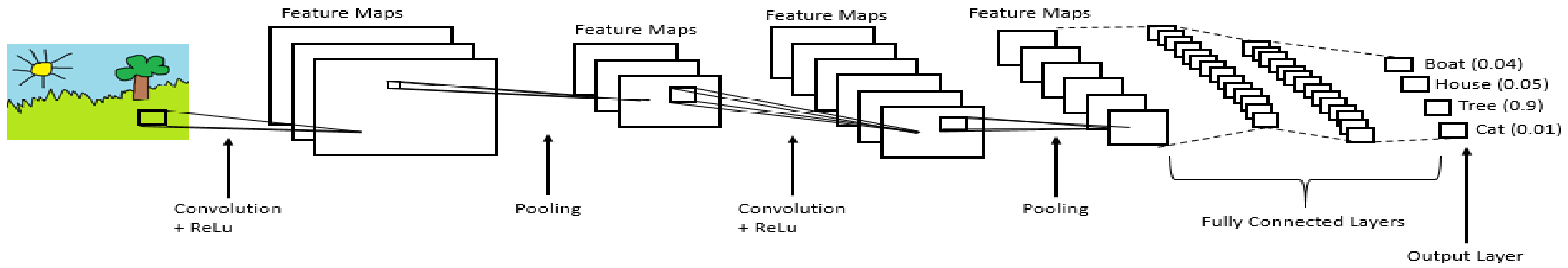
4. Fully Connected Layer

The **FC** is the fully connected layer of neurons at the end of CNN. Neurons in a fully connected layer have full connections to all activations in the previous layer.

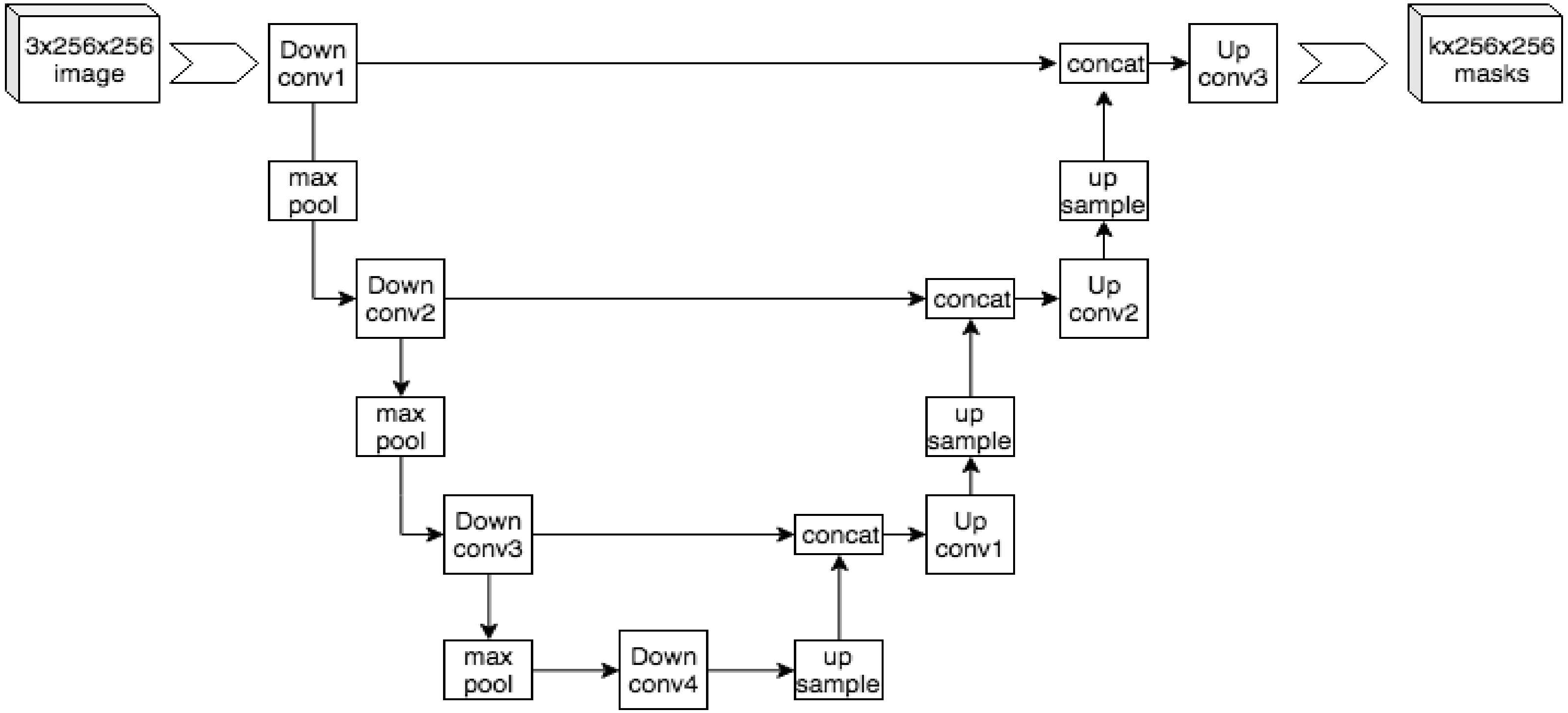


U Net Architecture

- The **U-Net** architecture stems from the so-called “fully convolutional network”.



- The recipe behind U-Net is to make use of the same feature maps that are used for contraction to expand a vector to a segmented image. This would preserve the structural integrity of the image which would reduce distortion enormously.
- U-Net architecture consists of three sections: The contraction, The bottleneck, and the expansion section. The contraction section is made of many contraction blocks.
- The first half of the network by minimizing a cost function related to the operation desired and at the second half it would be able to construct an image.



Layers in U-Net Architecture

DATASET

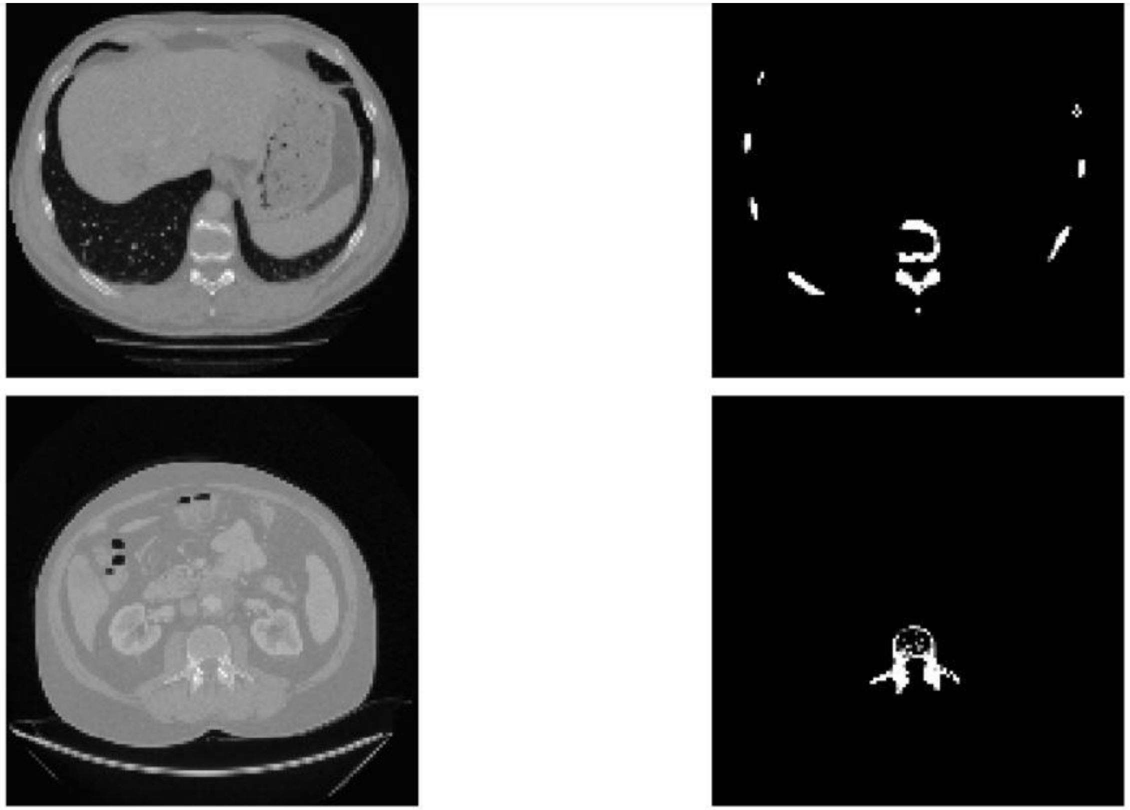
Data Set in the project

- The 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.
- These folders are called “3D-IRCADb-1-number” . Each “**3D-IRCADb-01-*number***” folder contains sub-folders called “**PATIENT_DICOM**” and “**MASKS_DICOM**”.The images are provided by the authors in DICOM format in 512x512 pixels.

Data Set in the project

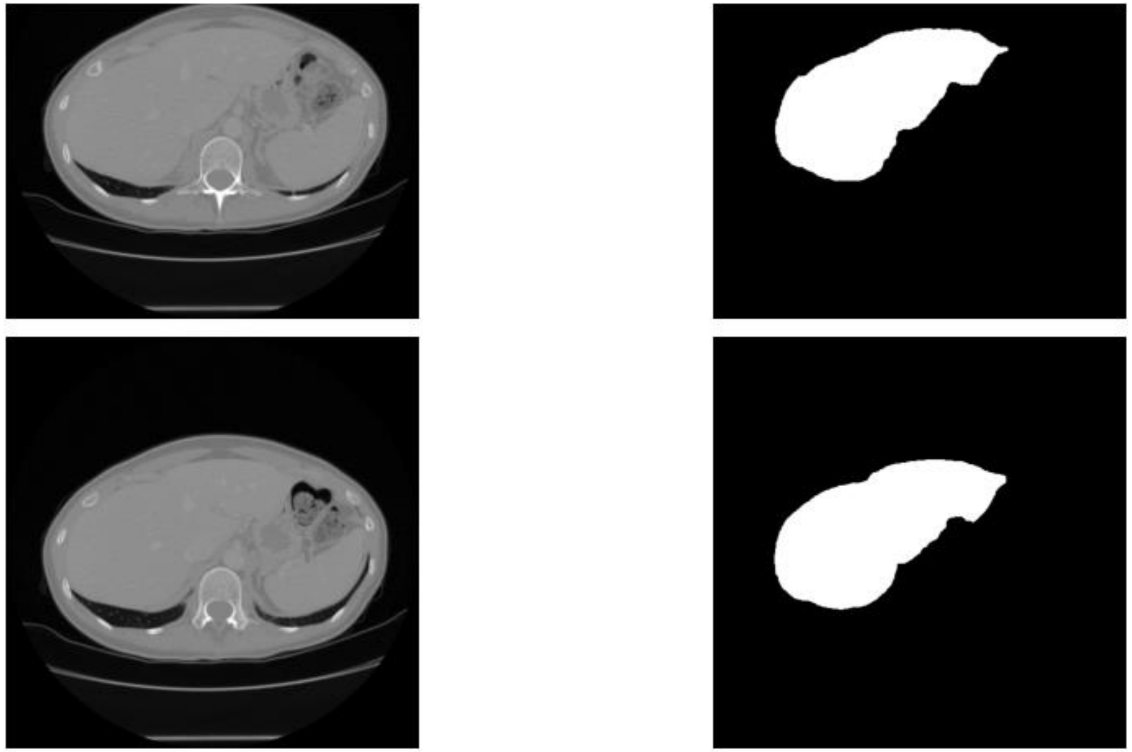
```
.
├── 3Dircadb1.1
    ├── LABELLED_DICOM
    ├── MASKS_DICOM
    │   ├── artery
    │   ├── bone
    │   ├── leftkidney
    │   ├── leftlung
    │   ├── liver
    │   ├── liverkyst
    │   ├── livertumor01
    │   ├── livertumor02
    │   ├── livertumor03
    │   ├── livertumor04
    │   ├── livertumor05
    │   ├── livertumor06
    │   ├── livertumor07
    │   ├── portalvein
    │   ├── rightkidney
    │   ├── rightlung
    │   ├── skin
    │   ├── spleen
    │   └── venoussystem
    ├── MESHES_VTK
    ├── PATIENT_DICOM
```

Snapshot of data of the project



The left column is the original images for training and right column indicate its label

Snapshot of data of the project

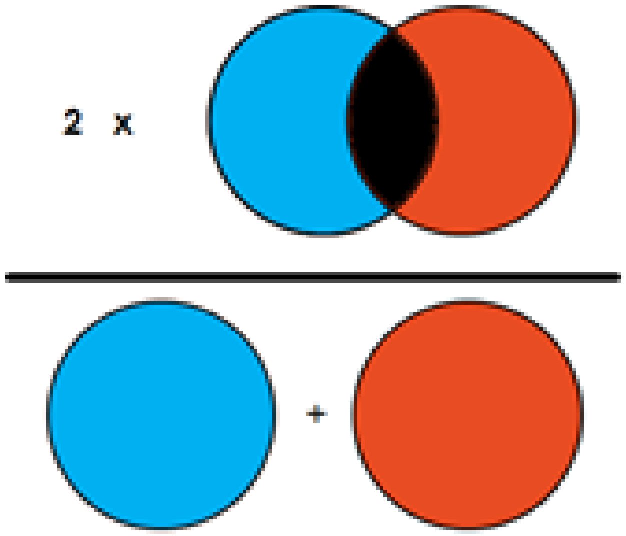


The left column is the original images for training and right column indicate its label

LOSS FUNCTION

Dice Coefficient

- Dice Coefficient is one of the most commonly used metrics in semantic segmentation. It is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth.
- Dice Coefficient is 2 * the Area of Overlap divided by the total number of pixels in both images.



LEARNING NEW APPROACH

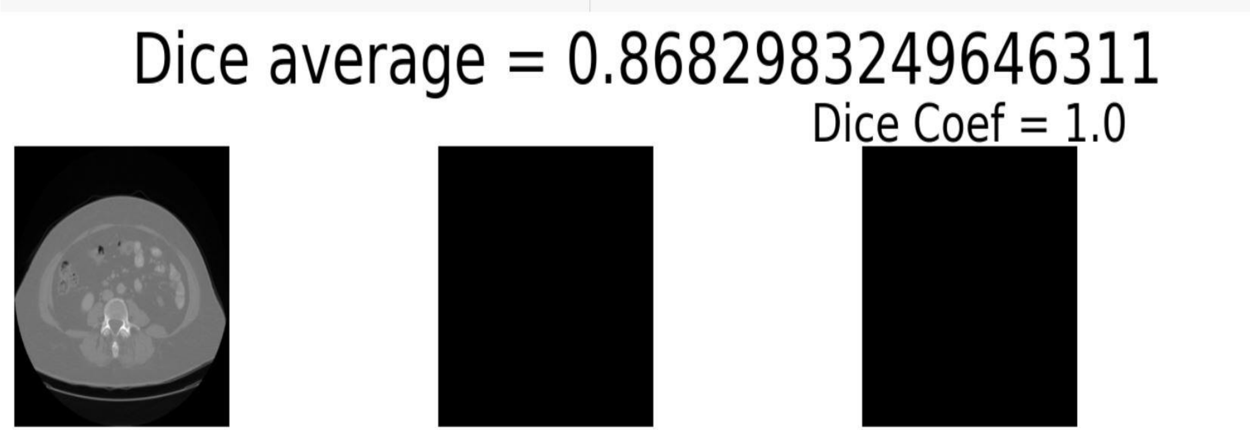
Transfer Learning

- Transfer learning make use of the knowledge gained while solving one problem and applying it to a different but related problem.
- For example, knowledge gained while learning to recognize cars can be used to some extent to recognize trucks.
- It is applied if :
- If the new dataset is very small, it's better to train only the final layers of the network to avoid overfitting, keeping all other layers fixed. So remove the final layers of the pre-trained network. Add new layers. Retrain only the new layers.
- If the new dataset is very much large, retrain the whole network with initial weights from the pretrained model.

In this project, Transfer Learning is being used to analyze the accuracy of model trained from scratch versus accuracy of the model trained using transfer learning. For this I have used the model previously trained on 'bones' data and used it to another model to predict 'liver' data.

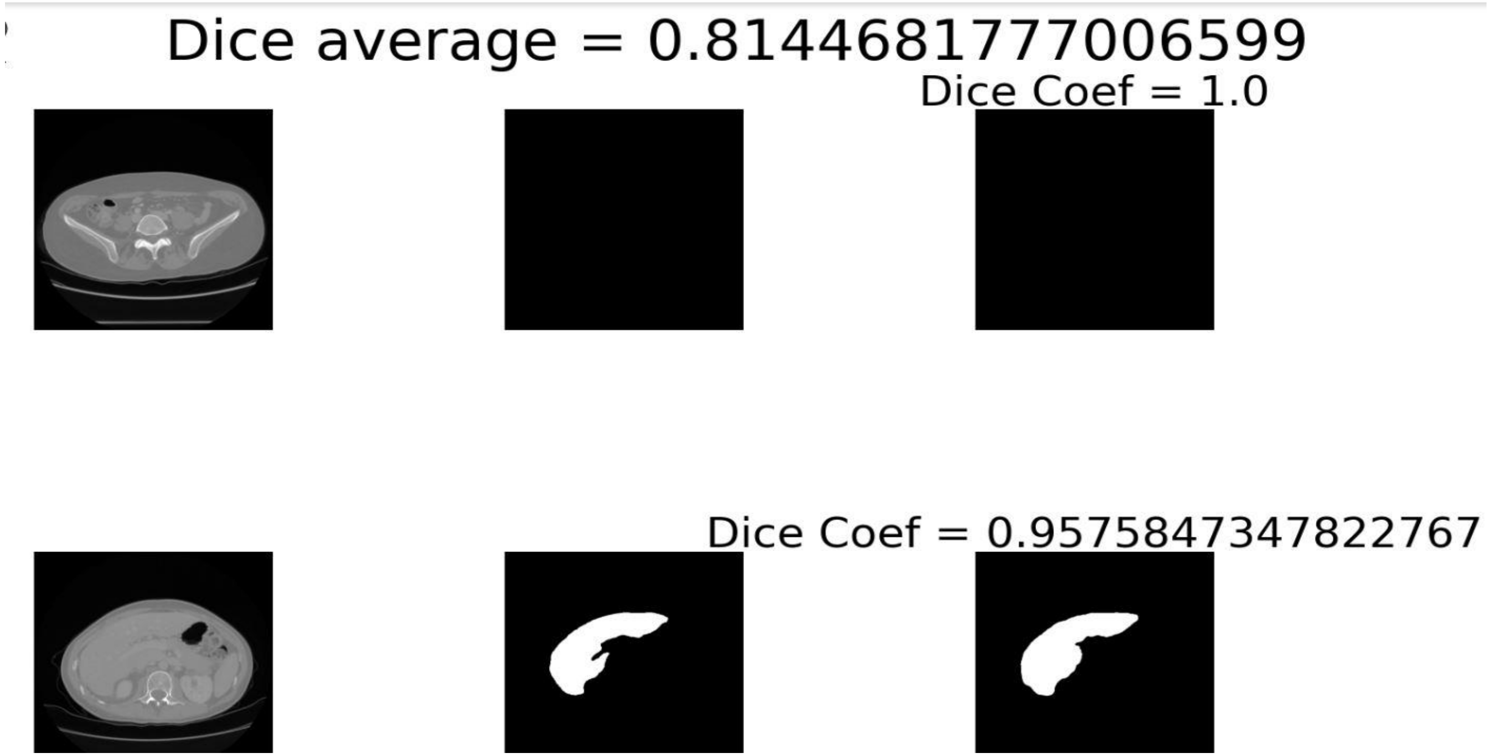
Results of Transfer Learning

- The comparison between these two models can be done based on these results:



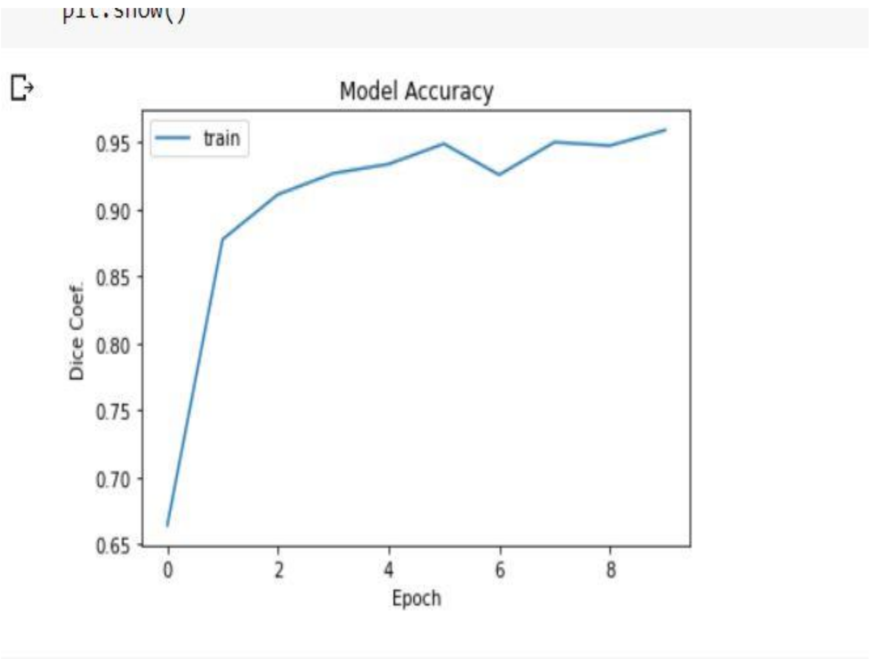
Average Dice Coefficient of Transfer Learning Model

Results of Transfer Learning

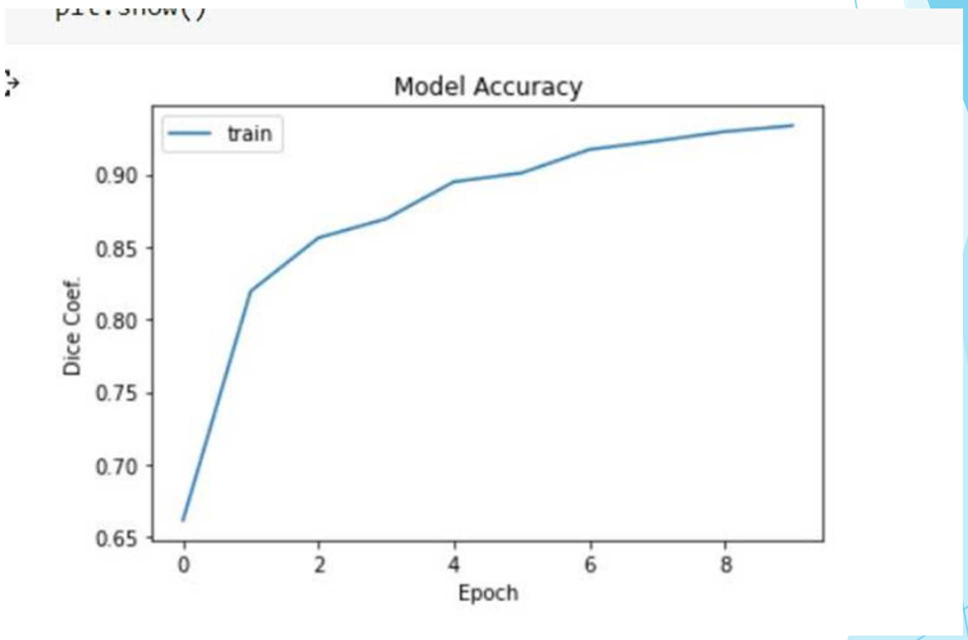


Average Dice Coefficient of Model trained from scratch

Graph of Models



Accuracy graph of Transfer Learning Model



Accuracy graph of Model trained from scratch

Results of Transfer Learning

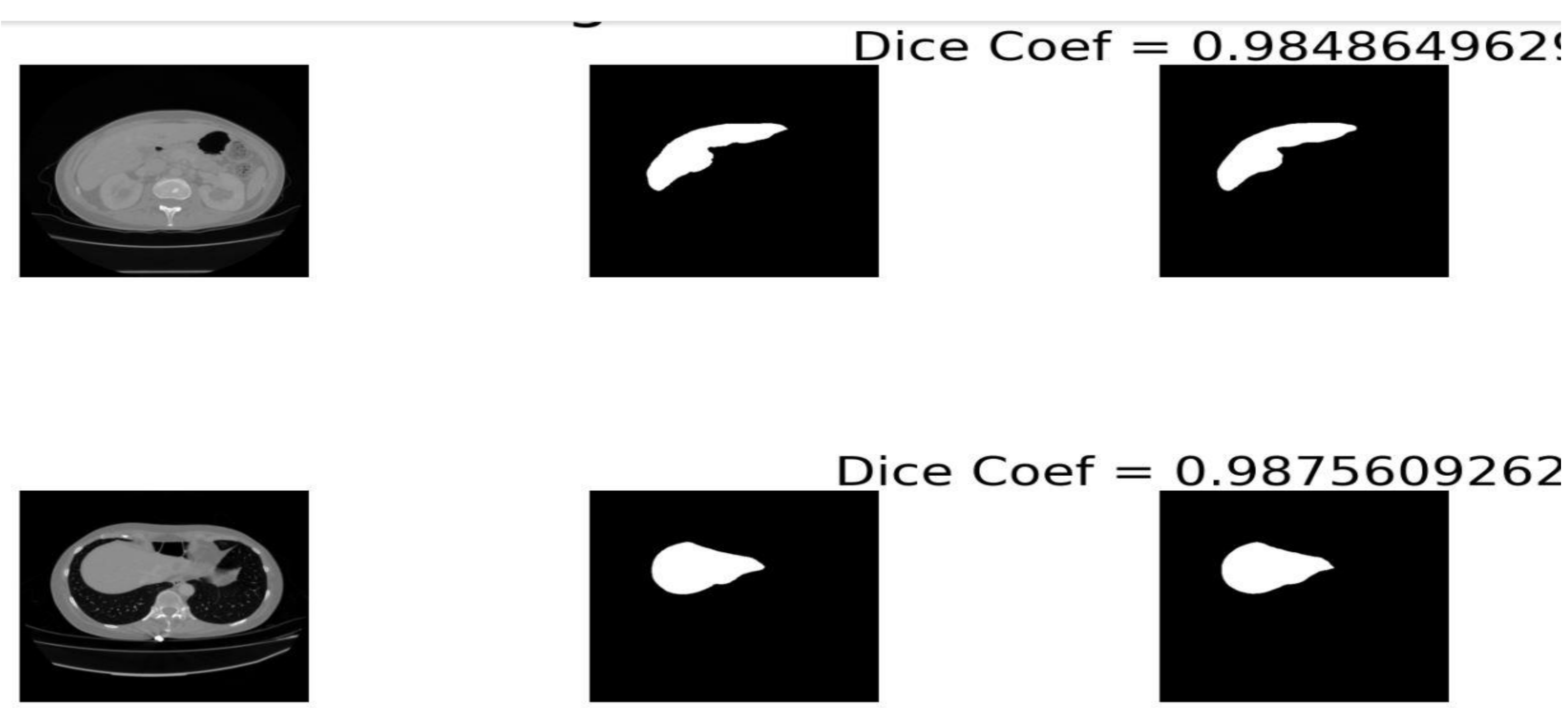
- From the above comparison we can see, model trained from scratch achieved accuracy of 95% in 10 epochs with 1622 samples whereas the 97% accuracy was achieved with transfer learning approach within 10 epochs and with using only 25% of samples.
- The reason behind this can be, as we are using a network which is already trained to detect bones from CT images, this transferred model network has learned to identify basic image components like lines, corners, intensity variations from CT images.
- Thus, transfer learning approach greatly reduced the training time and proved its usefulness in cases where lesser amount of labelled data is available.

RESULTS

Snapshot of Segmented Image

The leftmost column is the CT Scan image. The middle column is the real segmentation and the rightmost column is the segmentation generated by our 10-layer U-Net.

Snapshot of Segmented Image



The leftmost column is the CT Scan image. The middle column is the real segmentation and the rightmost column is the segmentation generated by our 10-layer U-Net.

VARIATION OF SOME IMAGES WITH THE ACTUAL

Dice Coefficient: 0.776348956

Dice Coefficient: 0.8603815214

The left image is the predicted image and the right image is the actual image

Selection of Epoch and Batch size

- Epoch selection is done by analyzing the result of the model. We need to analyze at which point of time the relation between the error and epochs gets constant or in other words when our model shows the saturation point.
- When the batch size is more than one sample and less than the size of the training dataset, the learning algorithm is called mini-batch gradient descent.

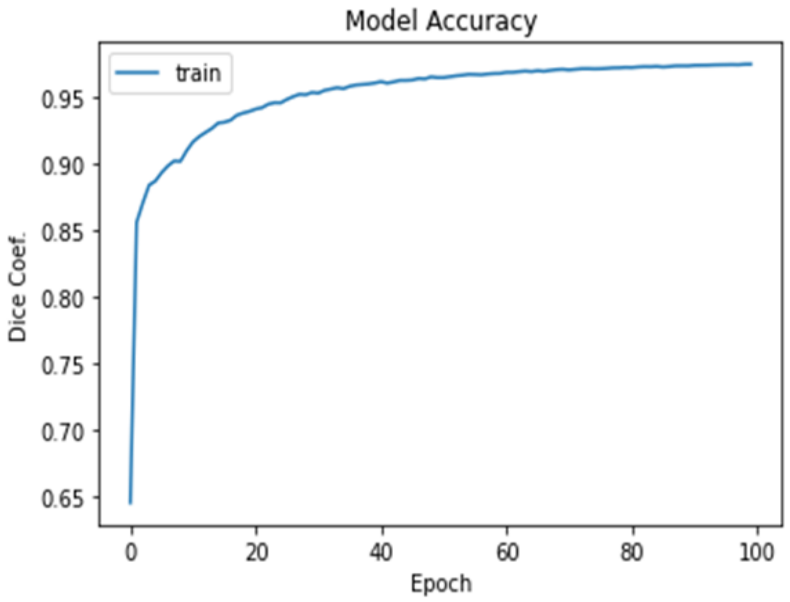
  Batch Gradient Descent. Batch Size = Size of Training Set

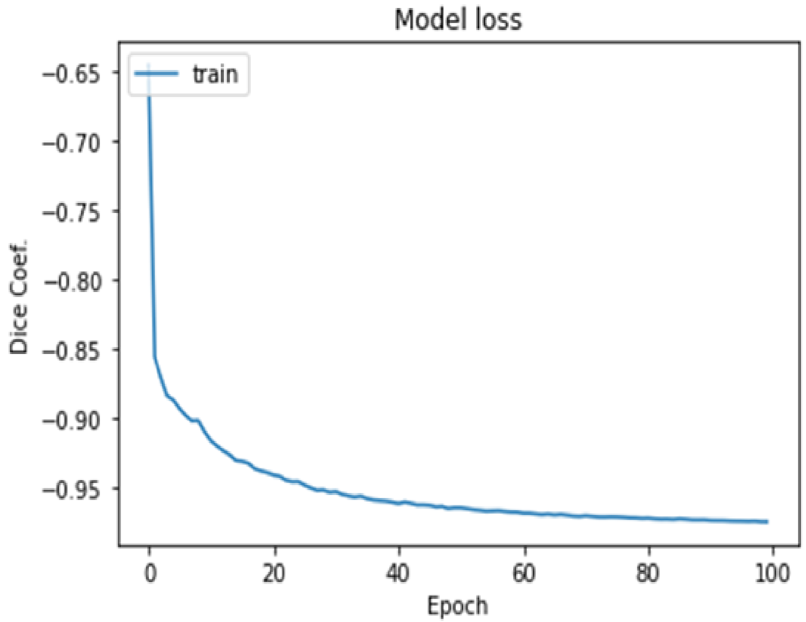
  Stochastic Gradient Descent. Batch Size = 1

  Mini-Batch Gradient Descent. 1 < Batch Size < Size of Training Set
- For this project I have used 100 epoch and 16 as batch size.

Graph of Trained Model

Model is trained with:
Batch size: 16
No. of epoch: 100





FUTURE SCOPE

Future Scope

1. **Adding more datasets**: This code is trained only on particular image i.e. part of abdominal region. So, adding more datasets to it and training on those datasets would help in getting vast results.
2. **Creating more interactive output**: Labelling every part in an image may give clear identification of the output image.
3. **Segmentation of more parts**: Since the model is trained on some specific body parts, the project would be better if it can segment a greater number of parts.
4. **Improving Accuracy**: We have to improve dice coefficients for better results.
5. **Improving algorithm:** We have to improve our algorithm to achieve far better results on a far better dataset.

ACCOMPLISHMENT

My Role

This project is of great importance in the Medical field which helps the medical professionals to analyze any disease in the patient and further cure them.

In this on-going project I am working as a:

- Backend developer using Deep Learning
- Analyzing the data
- Understanding U Net Architecture
- Researching the topics related to the project.

Thank You