Project Report

Entitled

"Automatic Liver and Tumor Segmentation of Computed Tomography Using Fully Convolutional Neural Networks"

Submitted to the Department of Electronics Engineering
in Partial Fulfillment of the Requirement for the Degree of

Bachelor of Technology

(ELECTRONICS & COMMUNICATION ENGINEERING)

: Presented & Submitted By :

Mr. Shrijeet Jain
(Roll No. U17EC078)
Mr. Raghav Bansal
(Roll No. U17EC081)
Mr. Harshwardhan Bhangale
(Roll No. U17EC115)
B. TECH. IV (EC), 8th Semester

: Guided By :

Dr. Jignesh N Sarvaiya
Professor, ECED.

(Year: 2020_21)

DEPARTMENT OF ELECTRONICS ENGINEERING
Sardar Vallabhbhai National Institute of Technology
Surat-395007, Gujarat, INDIA.

Sardar Vallabhbhai National Institute of Technology

Surat-395 007, Gujarat, INDIA.

ELECTRONICS ENGINEERING DEPARTMENT

CERTIFICATE

This is to certify that the **PROJECT REPORT** entitled **"Automatic Liver and Tumor Segmentation of Computed Tomography Using Fully Convolutional Neural Networks"** is presented & submitted by Candidate **Mr Shrijeet Jain, Mr Raghav Bansal, Mr Harshwardhan Bhangale,** bearing **Roll No. Roll No U17EC078, U17EC081, U17EC115** respectively, of **B.Tech. IV, 8th Semester** in the partial fulfillment of the requirement for the award of **B. Tech** Degree in **Electronics & Communication Engineering** for academic year 2020-21.

They have successfully and satisfactorily completed their **Project Examination** in all respect. We, certify that the work is comprehensive, complete and fit for evaluation.

Dr. Jignesh N Sarvaiya
Professor &Project Guide

PROJECT EXAMINERS:

Name of Examiner **Signature with date**

1. Dr J N Sarvaiya ____________________

2. Dr K P Upla ____________________

3. Dr Suman Deb ____________________

4. Dr Raghvendra Pal ____________________

Dr. P. N. Patel
Associate Professor &
Head, ECED, SVNIT.

DEPARTMENT SEAL
(May 2021)

ACKNOWLEDGEMENT

Apart from our efforts, the success of this project till date report depends largely on the encouragement and guidelines of many others. I take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We would like to express the deepest appreciation to our guide, Dr Jignesh N Sarvaiya, Professor, Electronics Engineering Department, S.V. National Institute of Technology (Surat), India, who has helped and encouraged us a lot in selection of project topic and content gathering. Without his guidance and persistent help this seminar would not have been possible.

Finally, we are thankful to everyone who directly or indirectly helped in this seminar.

SJain.

Shrijeet Jain ,U17EC078

Raghav Bansal

Raghav Bansal ,U17EC81

HBhangale

Harshwardhan Bhangale ,U17EC115

ABSTRACT

PROJECT TITLE: Automatic Liver and Tumor Segmentation of Computed Tomography Using Fully Convolutional Neural Networks

Abstract- Liver cancer has been as the second most fatal cancer to cause death for men and sixth for women. Early diagnosis by Computed Tomography (CT) could lead to high recovery rate, however going through all the CT slices for thousands or even millions of patients manually by professionals is hard, tiresome, expensive, time-consuming and prone to errors. Therefore, we needed a reliable, simple and accurate method to automate this process. In this thesis we used CNNs to overcome all the aforementioned obstacles in our research, a hybrid of Multi-feature pyramid-based U-Net, short skip connections and a Feature similarity module are proposed for early detection of the tumor. The proposed algorithm focuses on improving the tumor segmentation performance with fewer training parameters. The performance of the proposed algorithm is evaluated on the basis of the dice score coefficient of tumor segmentation. We have achieved a dice score of 0.753 and 0.950 on tumor and liver, respectively on the Liver Tumor Segmentation (LiTS) dataset. On comparison with earlier models, our model has achieved a higher dice coefficient with less training time and close to 6 million parameters.

Keywords: ***Liver-Tumor Segmentation, Multi-Feature Pyramid Network, Feature Similarity Module, Skip Connections, LiTS Dataset***

Signature of Students: SJain. Raghav Bansal Bhangale

Student Name: Shrijeet Jain, Raghav Bansal, Harshvardhan Bhangale

Roll No. U17EC078, U17EC081, U17EC115

Guide Name: Dr Jignesh N Sarvaiya

Date of Project Examination: 12th May 2020

Time slot of Examination: 10 am to 1 pm

Examiner Name: Dr Kishor P Upla and Dr Suman Deb, Dr Raghvendra Pal

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

The exponential growth of cells in the liver is termed as Liver cancer. Liver cancer is one of the main causes of death among both men and women. It is the fifth most common detected disease in men and ninth most common detected disease in women. It is estimated that 42,430 adults consisting of 29,890 men and 12,340 women will be diagnosed with the liver cancer in 2021. Also, research has shown that 30,320 people will die from this disease bifurcated into 20,300 men and 12,340 women [1]. But it is confirmed that early detection of the disease ensures the long survival of the patient. Computed Tomography has been used for detection of tumor from the liver slices. Segmentation of liver and tumor facilitates early detection of the disease. But, the manual segmentation of liver tumor by radiologists is time-consuming, industrious, and most importantly subjective to individual doctors. Therefore, an efficient and automatic technique for segmentation of tumor is required. In our research, we have proposed lightweight and simple automatic model for liver and tumor segmentation. We have incorporated pyramidal structure at the decoder side, feature similarity module for extracting relationship between pixels along with short skip connection for efficient training. We are using Dice coefficient of tumor segmentation as the parameter for performance evaluation.

1.1 LIVER ANATOMY

The Liver is composed of two lobes Fig. 1.1 and it is one of the most essential organs in the Digestive System. It carries plenty of functions, one of the main function is processing the absorbed nutrients by the small intestine. The Liver also secretes Bile juice into the small intestine which aids in digesting fats. It also takes raw materials absorbed by the intestine and makes the chemicals the body needs to function and detoxify the body from harmful chemicals.

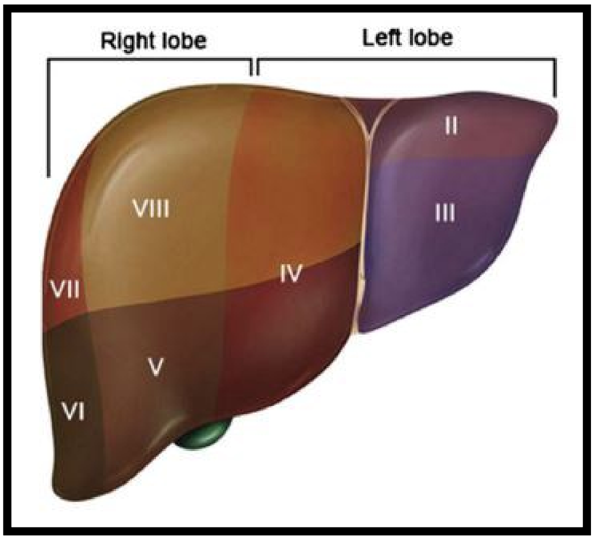

Fig. 1.1 Anatomy of the Liver [2]

1.1.1 LIVER TUMOR AND STAGES

Liver cancer has been reported to be the second most frequent cancers to cause death in men and sixth for women. About 750,000 people got diagnosed with liver cancer 696,000 of which died from the cancer in 2008. Globally the rate of infection of males is twice than that of females. The highest infection rates are found in East and South-East Asia and in Middle and Western Africa. Liver cancer incidence rate is increasing in many parts in the world including United States and Central Europe, which is possibly caused by obesity and the increase in Hepatitis C Virus (HCV).

Liver Tumors has many stages which require different treatment and diagnosis processes Fig. 1.2 - 1.5 [3]. The stage of a cancer tells you its size and whether it has spread. It helps your doctor decide which treatment you need. There are different staging systems that doctors can use to stage cancer that started in the liver (primary liver cancer).

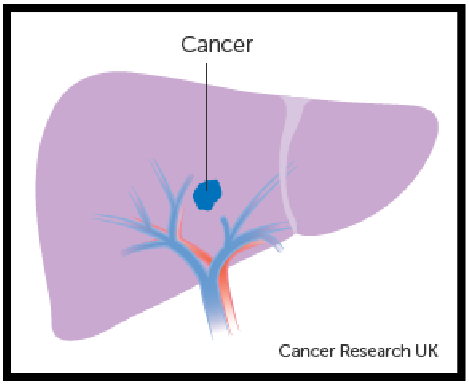

Fig. 1.2 Stage 1A of liver tumors [3]

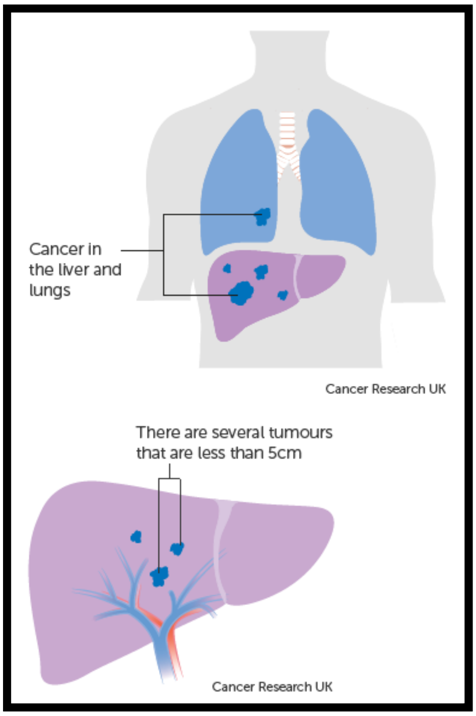
Cancer in the liver and lungs
Cancer Research UK
There are several tumours that are less than 5cm
Cancer Research UK

Fig. 1.3 Stage 2A & 2B of Liver tumors [3]

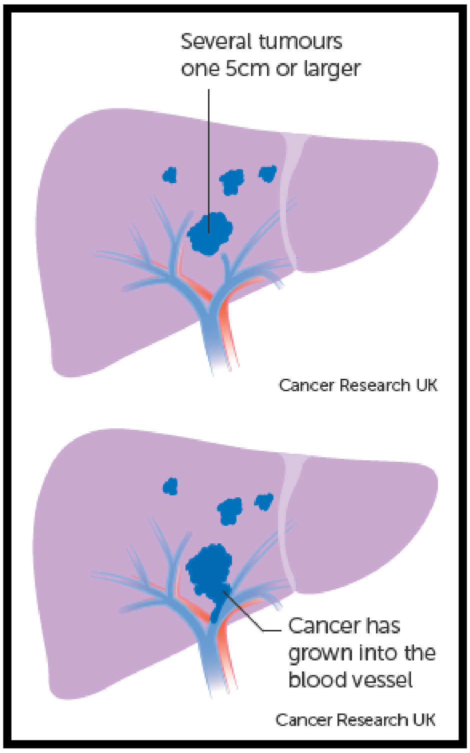
Several tumours one 5cm or larger
Cancer Research UK
Cancer has grown into the blood vessel
Cancer Research UK

Fig. 1.4 Stages 3A & 3B of liver tumors [3]

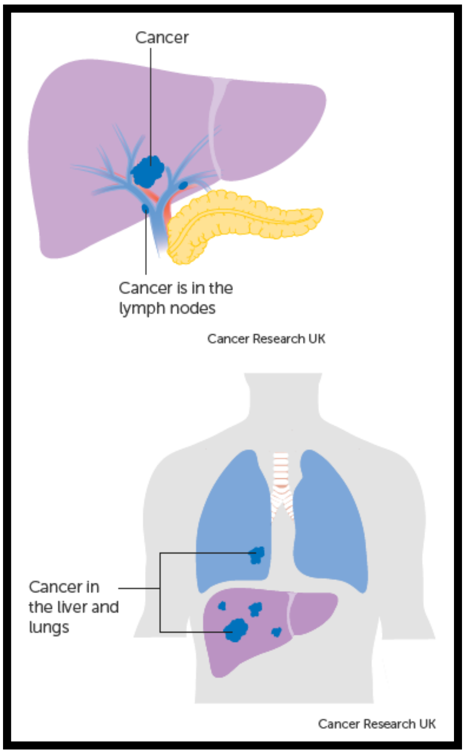

Fig. 1.5 Stages 4A & 4B of liver tumor [3]

The main aim of the study is to automatically segment liver and tumor from the CT slices. For this, we have to build a simple and lightweight model which can perform liver tumor segmentation. We have performed this using the multi class semantic segmentation. From the CT slice, we have first segmented the liver and then from the liver we have segmented tumor, which fulfils our aim.

1.2 Segmentation [4]

The goal of image segmentation is to cluster pixels into salient image regions, i.e., regions corresponding to individual surfaces, objects, or natural parts of objects. A segmentation could be used for object recognition, occlusion boundary estimation within motion or stereo systems, image compression, image editing, or image database look-up.

Semantic Segmentation: Objects classified with the same pixel values are segmented with the same colormaps. The original image and the semantic segmented image is shown in Fig. 1.6 and Fig. 1.7 respectively.

Instance Segmentation: It differs from semantic segmentation because different instances of the same object are segmented with different color maps. The original image and instance segmented image is shown in Fig. 1.6 and Fig 1.8 respectively.

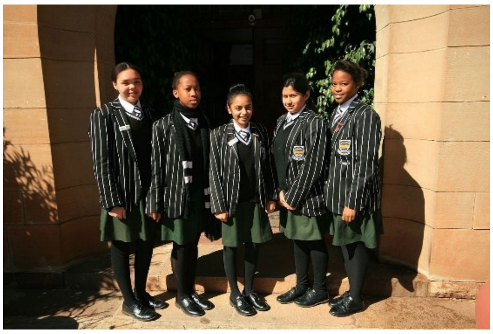

Fig. 1.6 Original Image [4]

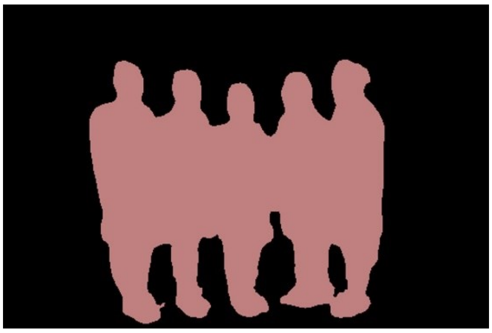

Fig. 1.7 Semantic Segmented image [4]

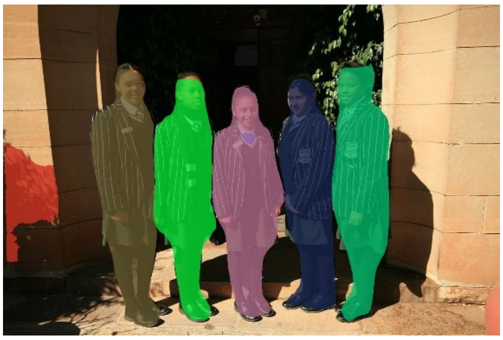

Fig. 1.8 Instance Segmented Image [4]

Segmentation in Image Processing is being used in the medical industry for efficient and faster diagnosis, detecting diseases, tumors, and cell and tissue patterns from various medical imagery generated from radiography, MRI, endoscopy, thermography, ultrasonography, etc.

1.2.1 Classical Segmentation Methods: [5]

Image Segmentation is the process by which a digital image is partitioned into various subgroups (of pixels) called Image Objects, which can reduce the complexity of the image, and thus analyzing the image becomes simpler. Use of various image segmentation algorithms to split and group a certain set of pixels together from the image. By doing so, we are actually assigning labels to pixels and the pixels with the same label fall under a category where they have some or the other thing common in them.

I. Threshold Method [5]

This is perhaps the most basic and yet powerful technique to identify the required objects in an image. Based on the intensity, the pixels in an image get divided by comparing the pixel's intensity with a threshold value. The threshold method proves to be advantageous when the objects in the image in question are assumed to be having more intensity than the background (and unwanted components) of the image. At its simpler level, the threshold value T is considered to be a constant. But that approach may be futile considering the amount of noise (unwanted information) that the image contains. So, we can either keep it constant or change it dynamically based on the image properties and thus obtain better results. Based on that, thresholding is of the following types:

a) Simple Thresholding

This technique replaces the pixels in an image with either black or white. If the intensity of a pixel ($I_{i,j}$) at position (i,j) is less than the threshold (T), then it is replaced with black and if it is more, then it is replaced with white. This is a binary approach to thresholding. The figure for simple thresh holing is shown in Fig. 1.9.

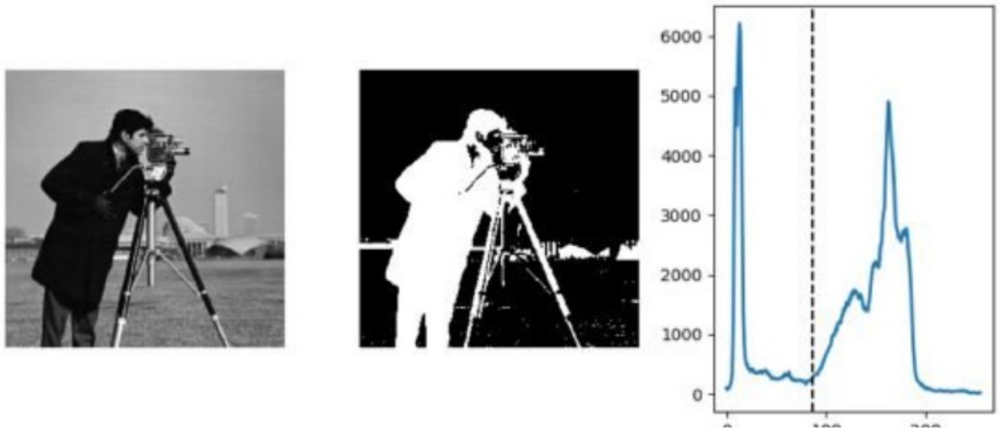

Fig. 1.9 Simple Thresholding [5]

b) Otsu's Binarization

In global thresholding, an arbitrary value for threshold value is used and it remains a constant. The major question here is, how can we define and determine the correctness of the selected threshold? A simpler but rather inept method is to trial and see the error. But, on the contrary, let us take an image whose histogram has two peaks (bimodal image), one for the background and one for the foreground. According to Otsu binarization, for that image, we can approximately take a value in the middle of those peaks as the threshold value. So, in simply put, it automatically calculates a threshold value from image histogram for a bimodal image. The disadvantage here, however, is for images that are not bimodal, the image histogram has multiple peaks, or one of the classes (peaks) present has high variance. However, Otsu's Binarization is widely used in document scans, removing unwanted colors from a document, pattern recognition etc.

c) Adaptive Thresholding [5]

A global value as threshold value may not be good in all the conditions where an image has different background and foreground lighting conditions in different actionable areas. Need for an adaptive approach that can change the threshold for various components of the image. In this, the algorithm divides the image into various smaller portions and calculates the threshold for those portions of the image. Hence, we obtain different thresholds for different regions of the same image. This in turn gives us better results for images with varying illumination. The algorithm can automatically calculate the threshold value. The threshold value can be the mean of neighborhood area or it can be the weighted sum of neighborhood values where weights are a Gaussian window (a window function to define regions).

II. Region Based Segmentation [5]

The region-based segmentation methods involve the algorithm creating segments by dividing the image into various components having similar characteristics. These components, simply put, are nothing but a set of pixels. Region-based image segmentation techniques initially search for some seed points – either smaller parts or considerably bigger chunks in the input image. Next, certain approaches are employed, either to add more pixels to the seed points or further diminish or shrink the seed point to smaller segments and merge with other smaller seed points. Hence, there are two basic techniques based on this method.

a) Region Growing

It's a bottom to up method where we begin with a smaller set of pixel and start accumulating or iteratively merging it based on certain pre-determined similarity constraints. Region growth algorithm starts with choosing an arbitrary seed pixel in the image and compare it with its neighboring pixels. If there is a match or similarity in neighboring pixels, then they are added to the initial seed pixel, thus increasing the size of the region. When we reach the saturation and hereby, the growth of that region cannot proceed further, the algorithm now chooses another seed pixel, which necessarily does not belong to any region(s) that currently exists and start the process again. Region growing methods often achieve effective Segmentation that corresponds well to the observed edges.

b) Region Splitting and Merging

The splitting and merging based segmentation methods use two basic techniques done together in conjunction – region splitting and region merging – for segmenting an image. Splitting involves iteratively dividing an image into regions having similar characteristics and merging employs combining the adjacent regions that are somewhat similar to each other. A region split, unlike the region growth, considers the entire input image as the area of business interest. Then, it would try matching a known set of parameters or pre-defined similarity constraints and picks up all the pixel areas matching the criteria.

III. Watershed Based Methods [6]

Watershed is a ridge approach, also a region-based method, which follows the concept of topological interpretation. The slope and elevation of the said topography are distinctly quantified by the gray values of the respective pixels – called the gradient magnitude. The watershed transform decomposes an image into regions that are called "catchment basins". For each local minimum, a catchment basin comprises all pixels whose path of steepest descent of gray values terminates at this minimum. In a simple way of understanding, the algorithm considers the pixels as a "local topography" (elevation), often initializing itself from user-defined markers. Then, the algorithm defines something called "basins" which are the minima points and hence, basins are flooded from the markers until basins meet on watershed lines. The watersheds that are so formed here, they separate basins from each other. Hence the picture gets decomposed because the pixels are assigned to each such region or watershed.

1.3 Terminology

1.3.1 Dense Block [7]

In DenseNet, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers. Concatenation is used. Each layer is receiving a "collective knowledge" from all preceding layers. Dense block helps to obtain more complex features, which improves the segmentation performance of the model. The Dense connections are shown in Fig. 1.10.

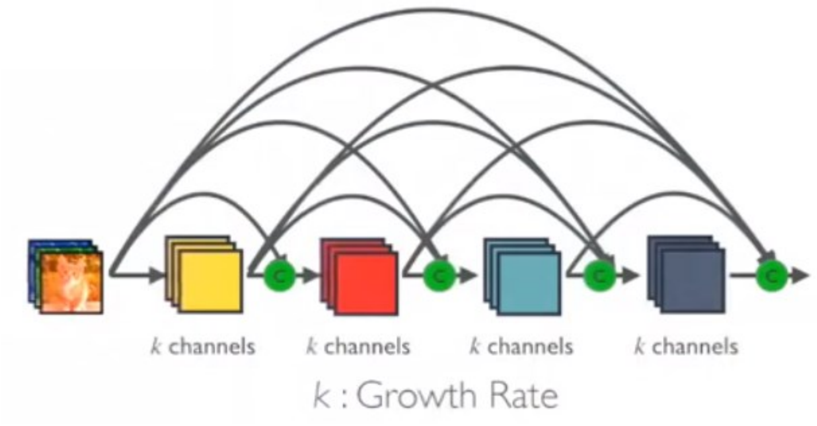

Fig. 1.10 DenseNet Architecture [7]

1.3.2 Skip Connections [8]

The utility of long and short skip connections for training fully convolutional networks for image segmentation is important. The variant with both long and short skip connections is not only the one that performs best but also converges faster than without short skip connections. the combination of both long and short skip connections performed better than having only one type of skip connection, both in terms of performance and convergence speed. In the study [8] we observed that at this depth, a network could not be trained without any skip connections. Finally, short skip connections appear to stabilize updates. It is observed that model performance actually drops when using short skip connections in those models that are shallow enough for all layers to be well updated. Moreover, batch normalization was observed to increase the maximal updatable depth of the network. Networks without batch normalization had diminishing updates toward the centre of the network and with long skip connections were less stable, requiring a lower learning rate. The skip connections are shown in Fig. 1.11.

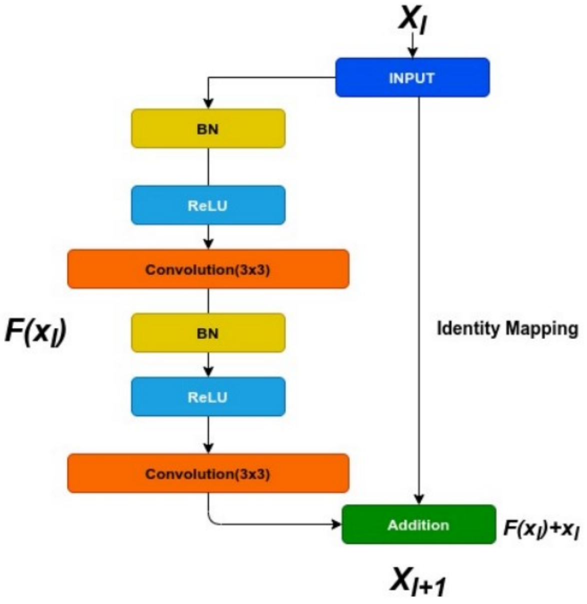

Fig. 1.11 Skip connections in ResNet [8]

1.3.3 Depthwise Separable Convolutions [9]

Unlike spatial separable convolutions, depthwise separable convolutions work with kernels that cannot be “factored” into two smaller kernels. Hence, it is more commonly used. This is the type of separable convolution seen in keras.layers.SeparableConv2D or tf.layers.separable_conv2d. The depthwise separable convolution is so named because it deals not just with the spatial dimensions, but with the depth dimension — the number of channels — as well. An input image may have 3 channels: RGB. After a few convolutions, an image may have multiple channels. You can image each channel as a particular interpretation of that image; in for example, the "red" channel interprets the "redness" of each pixel, the "blue" channel interprets the "blueness" of each pixel, and the "green" channel interprets the "greenness" of each pixel. An image with 64 channels has 64 different interpretations of that image. Similar to the spatial separable convolution, a depthwise separable convolution splits a kernel into 2 separate kernels that do two convolutions: the depthwise convolution and the pointwise convolution. But first of all, let's see how a normal convolution works.

I. Depthwise Convolution:

In the first part, depthwise convolution, the input image is given to a convolution without changing the depth. We do so by using 3 kernels of shape 5×5×1 which is shown in Fig. 1.12.

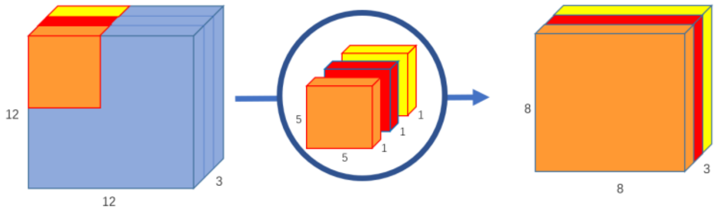

Fig. 1.12 Depthwise convolution, uses 3 kernels to transform a 12×12×3 image to an 8×8×3 image [9]

Each 5×5×1 kernel iterates 1 channel of the image (note: 1 channel, not all channels), getting the scalar products of every 25-pixel group, giving out an 8×8×1 image. Stacking these images together creates an 8×8×3 image.

II. Pointwise Convolution:

Remember, the original convolution transformed a 12×12×3 image to an 8×8×256 image. Currently, the depthwise convolution has transformed the 12×12×3 image to a 8×8×3 image. Now, we need to increase the number of channels of each image.

The pointwise convolution is so named because it uses a 1×1 kernel, or a kernel that iterates through every single point. This kernel has a depth of however many channels the input image has; in our case, 3. Therefore, we iterate a 1×1×3 kernel through our 8×8×3 image, to get a 8×8×1 image. The pointwise convolution is shown in Fig. 1.13.

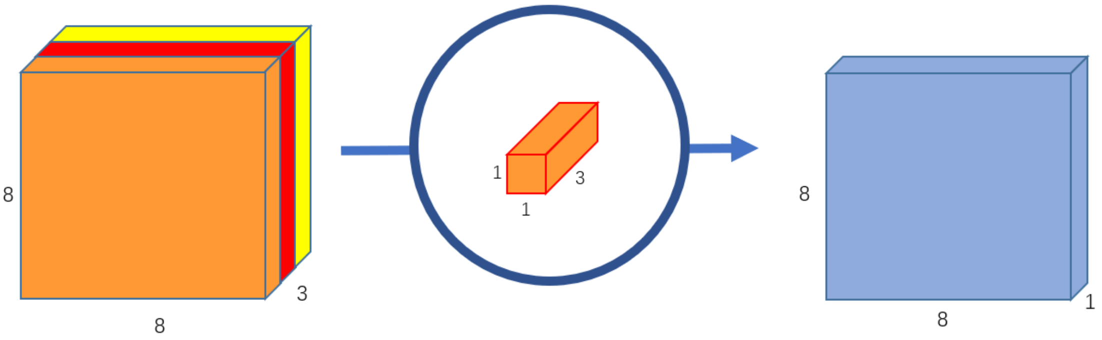

Fig. 1.13 Pointwise convolution, transforms an image of 3 channels to an image of 1 channel [9]

We can create 256 1×1×3 kernels that output a 8×8×1 image each to get a final image of shape 8×8×256. Pointwise convolution with 256 kernels is shown in Fig. 1.14.

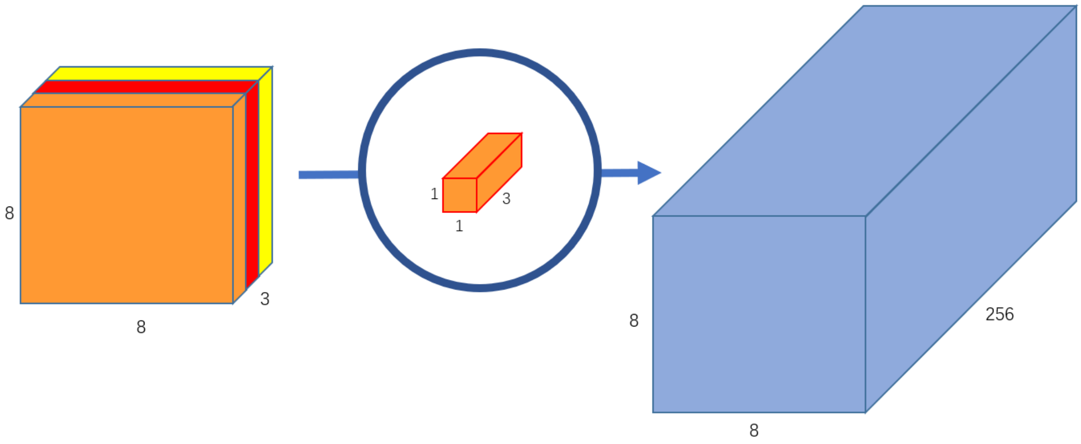

Fig. 1.14 Pointwise convolution with 256 kernels, outputting an image with 256 channels [9]

We've separated the convolution into 2: a depthwise convolution and a pointwise convolution. In a more abstract way, if the original convolution function is 12×12×3 — (5×5×3×256) →12×12×256, we can illustrate this new convolution as 12×12×3 — (5×5×1×1) — > (1×1×3×256) — >12×12×256.

The main difference is this: in the normal convolution, we are transforming the image 256 times. And every transformation uses up 5×5×3×8×8=4800 multiplications. In the separable convolution, we only really transform the image once — in the depthwise convolution. Then, we take the transformed image and simply elongate it to 256 channels. Without having to transform the image over and over again, we can save up on computational power.

1.3.4 Feature Similarity Module [10]

The Feature Similarity Module (FSM) is specially designed to capture long-range dependencies and help in utilizing the contextual information available to facilitate better segmentation. To address the issue of long-range dependencies, we have LSTM based architectures, also we have Atrous convolution-based models [10] to capture the complex spatial & contextual information available among the pixels but, none of the above models works as effectively as expected when the number of parameters to train decreases i.e., in case of a light-weight model. The FSM block used over here can be plugged into any architecture and aims to amass as much positional sensitive information as possible among the pixels and later encodes it into feature maps. The block diagram of the FSM module is shown in Fig. 1.15.

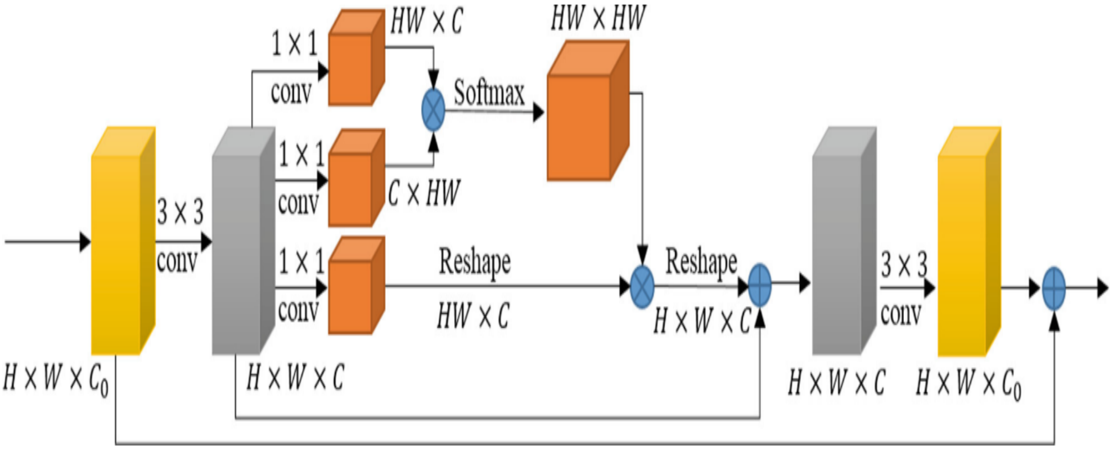

Fig. 1.15 Block diagram of the Feature Similarity Module [10]

The FSM uses a mathematical function as given in Eq. (1.1) where f(x_i,x_j) measures the j th position's impact on i th position, α(x_i) and β(x_j) are embedded layers implemented by 1 ×

1 convolution, and N is the number of positions in the feature map and it can be infered that it tries to capture the relationship between two pixels by measuring the impact of one pixel over the another and at the end we have a matrix which consists of both the original feature map and the captured contextual information.

$$f\left(x_i, x_j\right) = \frac{exp\left(\alpha(x_i)^T \beta(x_j)\right)}{\sum_{j=1}^{N} exp\left(\alpha(x_i)^T \beta(x_j)\right)} \tag{1.1}$$

Dense context features for discrimination are essential in pixel-level visual tasks, which could be obtained by capturing long-range dependencies. In order to model abundant contextual relationships over feature representations, we propose a Feature Similarity Module (FSM). This module extracts a wide range of position sensitive contextual information and encoded it into feature maps. Treating FSM as a network module that can be plugged to other fully convolutional neural networks, it may see wide applications in different situations for different tasks.

1.3.5 Multi-Feature Pyramid Network [11]

Detecting discrete objects of different scale is a difficult task especially for tiny objects and therefore to overcome these issues pyramid of same image at different resolution is used to detect objects. Hence, Feature Pyramids is a fundamental module in image recognition and object detection task. They are effective in combining low-resolution, semantically strong features with high resolution and complex feature maps. Combinations is usually done through top-down pathway and lateral connection. The feature pyramids have details from all levels and its construction is also easy but they are slightly inefficient in terms of memory. Multi-feature pyramid is depicted in Fig 1.16.

I. Bottom-Up Pathway

The bottom-up pathway is the feedforward computation of the backbone ConvNet. It is defined that one pyramid level is for each stage. The output of the last layer of each stage will be used as the reference set of feature maps for enriching the top-down pathway by lateral connection.

II. Top-Down Pathway and Lateral Connection

The higher resolution features are upsampled spatially coarser, but semantically stronger, feature maps from higher pyramid levels. More specifically, the spatial resolution is upsampled by a factor of 2 using the nearest neighbour for simplicity. Each lateral connection merges feature maps of the same spatial size from the bottom-up pathway and the top-down pathway. Specifically, the feature maps from bottom-up pathway undergoes 1×1 convolutions to reduce the channel dimensions. And the feature maps from the bottom-up pathway and the top-down pathway are merged by element-wise addition.

III. Prediction

Finally, a 3×3 convolution is appended on each merged map to generate the final feature map, which is to reduce the aliasing effect of upsampling. This final set of feature maps is called {P2, P3, P4, P5}, corresponding to {C2, C3, C4, C5} that are respectively of the same spatial sizes. Because all levels of the pyramid use shared classifiers/regressors as in a traditional featurized image pyramid, the feature dimension at output d is fixed with d = 256. Thus, all extra convolutional layers have 256-channel outputs. The multi feature pyramid model is shown in Fig. 1.16.

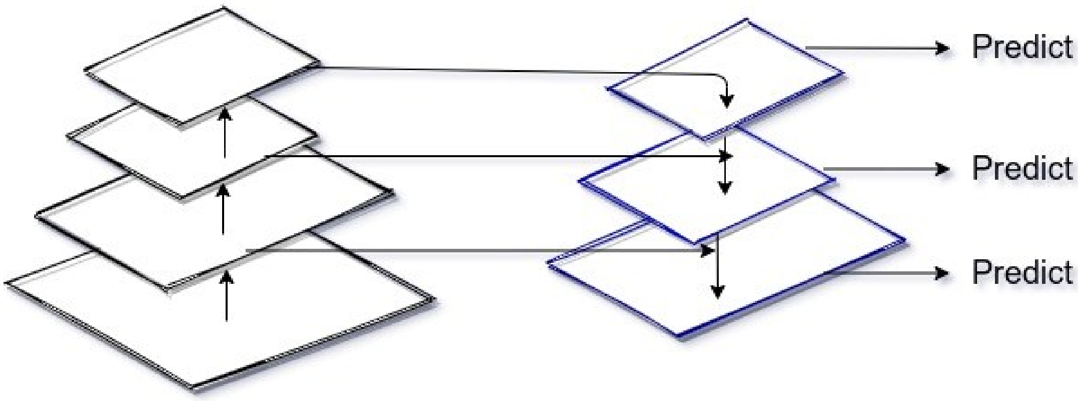

Fig. 1.16 Multi Feature Pyramid [11]

1.4 Architectures for Medical Image Segmentation

The approach of using a "fully convolutional" network trained end-to-end, pixels-to-pixels for the task of image segmentation was introduced by Long *et al.* [12] in late 2014. The network architecture uses the VGG 16- layer net. The network is appended with a 1×1 convolution with channel dimension 2 to predict scores for lesion or liver at each of the coarse output locations, followed by a deconvolution layer to upsample the coarse outputs to pixel-dense outputs. The upsampling is performed in-network for end-to-end learning by backpropagation from the pixelwise loss. The FCN-8s DAG net was used as our initial network, which learned to combine coarse, high layer information with fine, low layer information. d the additional value of adding another lower level linking layer creating an FCN-4s DAG net. This was done by linking the Pool2 layer in a similar way to the linking of the Pool3 and Pool4 layers in Fig. 1.17.

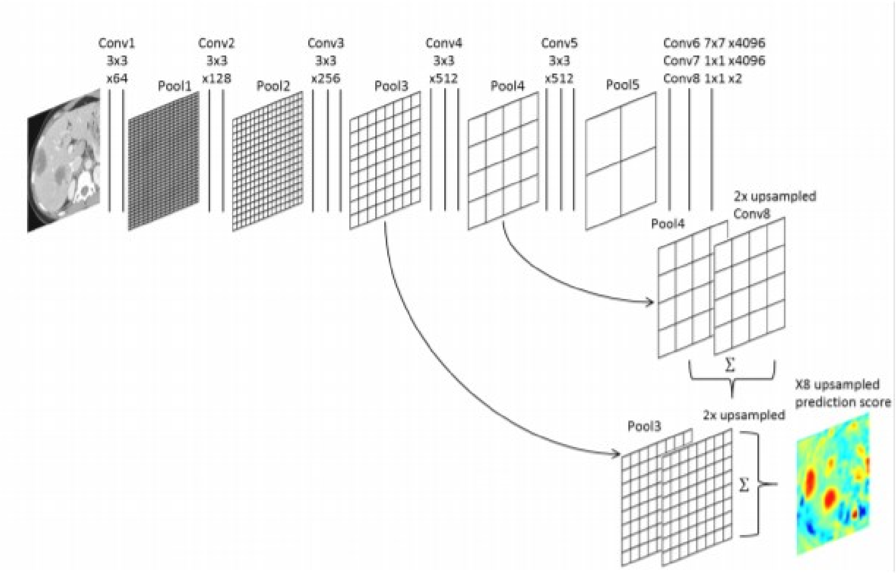

Fig. 1.17 Architecture of FCN [12]

1.4.1 U-Net [13]

U-net was originally invented and first used for biomedical image segmentation. Its architecture can be broadly thought of as an encoder network followed by a decoder network. Unlike classification where the end result of the deep network is the only important thing, semantic segmentation not only requires discrimination at pixel level but also a mechanism to project the discriminative features learnt at different stages of the encoder onto the pixel space.

- The encoder is the first half in the architecture diagram (Fig. 1.18). It usually is a pre-trained classification network like VGG/ResNet where you apply convolution blocks followed by a maxpool downsampling to encode the input image into feature representations at multiple different levels.

- The decoder is the second half of the architecture. The goal is to semantically project the discriminative features (lower resolution) learnt by the encoder onto the pixel space (higher resolution) to get a dense classification. The decoder consists of **upsampling** and **concatenation** followed by regular convolution operations.
- The main contribution of U-Net in this sense is that while upsampling in the network, the author is also concatenating the higher resolution feature maps from the encoder network with the upsampled features in order to better learn representations with following convolutions. Since upsampling is a sparse operation, we need a good prior from earlier stages to better represent the localization.

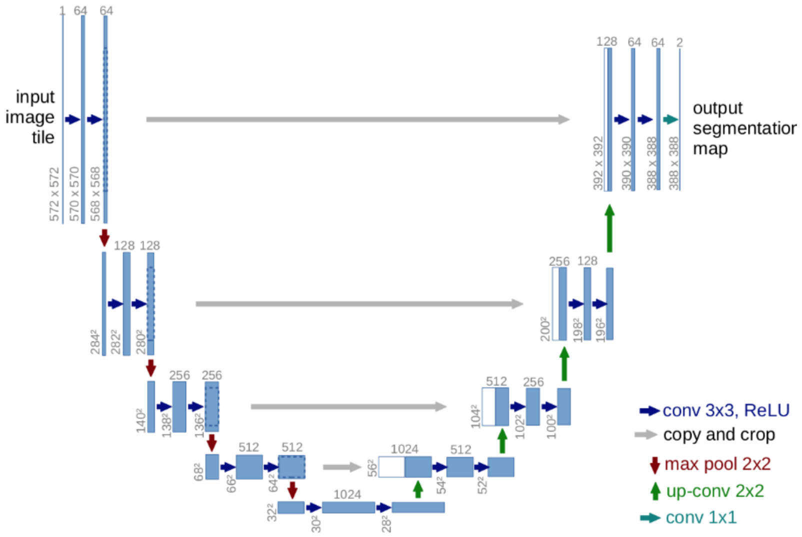

Fig. 1.18 U-Net [13]

1.4.2 U-Net++ [14]

UNet++ aims to improve segmentation accuracy by including Dense block and convolution layers between the encoder and decoder. Segmentation accuracy is critical for medical images because marginal segmentation errors would lead to unreliable results; thus, will be rejected for clinical settings. Algorithms designed for medical imaging must achieve high performance and accuracy despite having fewer data samples. Acquiring these sample images to train a model can be a resource-consuming process as requires high quality uncompressed and precisely annotated images vetted by professionals. The architecture of U-Net++ is shown in Fig. 1.19.

U-Net++ have 3 additions to the original U-Net:

I. Redesigned skip pathways (shown in green)

II. Dense skip connections (shown in blue)

III. Deep supervision (shown in red)

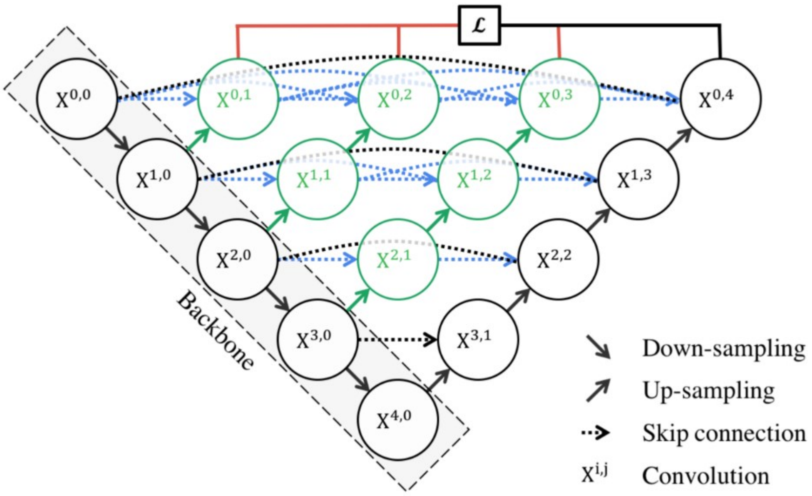

Fig. 1.19 U-Net++ Architecture [14]

I. Redesigned skip pathways

In UNet++, the redesigned skip pathways (shown in green) have been added to bridge the semantic gap between the encoder and decoder subpaths. The purpose of these convolutions layers is aimed at reducing the semantic gap between the feature maps of the encoder and decoder subnetworks. As a result, it is possibly a more straightforward optimisation problem for the optimiser to solve. All convolutional layers on the skip pathway use kernels of size 3×3.

II. Dense Skip connections

In UNet++, Dense skip connections (shown in blue) has implemented skip pathways between the encoder and decoder. These Dense blocks are inspired by DenseNet with the purpose to improve segmentation accuracy and improves gradient flow. Dense skip connections ensure that all prior feature maps are accumulated and arrive at the current node because of the dense convolution block along each skip pathway. This generates full resolution feature maps at multiple semantic levels.

III. Deep Supervision

In UNet++, deep supervision (shown in red) is added, so that model can be pruned to adjust the model complexity, to balance between speed (inference time) and performance. For accurate mode, the output from all segmentation branch is averaged. For fast mode, the final segmentation map is selected from one of the segmentation branches. Zhou *et al.* [14] conducted experiments to determine the best segmentation performance with different levels of pruning. The metrics used are Intersection over Union and inference time.

1.4.3 Bi-Directional ConvLSTM U-Net with Densley Connected Convolutions [15]

I. Encoding Path

The contracting path of BCDU-Net includes four steps. Each step consists of two convolutional 3×3 filters followed by a 2 × 2 max pooling function and ReLU. The number of feature maps are doubled at each step. The contracting path extracts progressively image representations and increases the dimension of these representations layer by layer. Ultimately, the final layer in the encoding path produces a high dimensional image representation with high semantic information.

The idea of densely connected convolutions has some advantages over the regular convolutions. First of all, it helps the network to learn a diverse set of feature maps instead of redundant features. Moreover, this idea improves the network's representational power by allowing information flow through the network and reusing features. Furthermore, dense connected convolutions can benefit from all the produced features before it, which prompt the network to avoid the risk of exploding or vanishing gradients.

II. Decoding Path

Each step in the decoding path starts with performing an up-sampling function over the output of the previous layer. In the standard U-Net, the corresponding feature maps in the contracting path are cropped and copied to the decoding path. These feature maps are then concatenated with the output of the up-sampling function. In BCDU-Net, they employ BConvLSTM to process these two kinds of feature maps in a more complex way. The architecture for the BCDU-Net model is shown in Fig. 1.20.

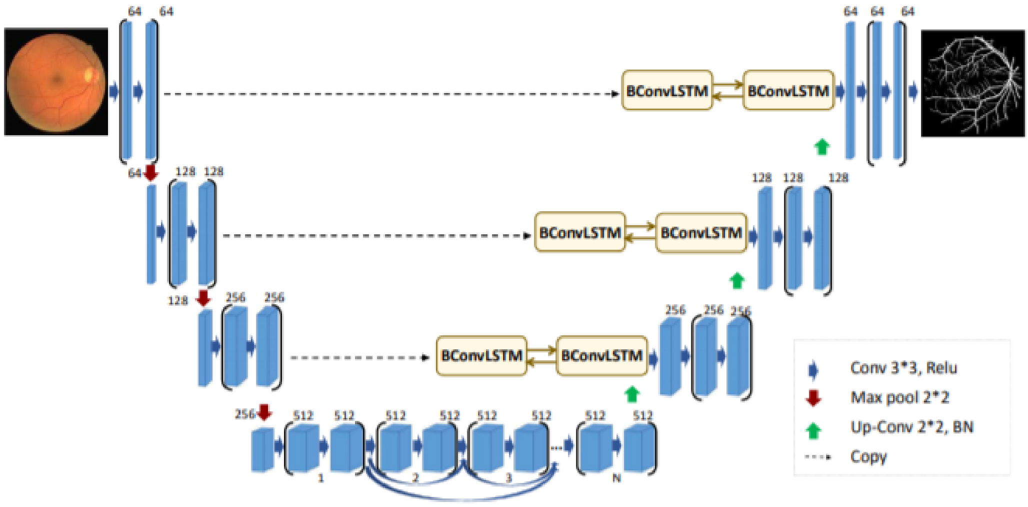

Fig. 1.20 BCDU-NET Architecture [15]

1.4.4 Attention Block [16]:

Attention, in the context of image segmentation, is a way to highlight only the relevant activations during training. This reduces the computational resources wasted on irrelevant activations, providing the network with better generalization power. Essentially, the network can pay "attention" to certain parts of the image. Attention comes in two forms, hard and soft.

I. Hard Attention: -

Hard attention works on the basis of highlighting relevant regions by cropping the image or iterative region proposal. Since hard attention can only choose one region of an image at a time, it has two implications, it is non-differentiable and requires reinforcement learning to train. Since it is non-differentiable, it means that for a given region in an image, the network can either pay "attention" or not, with no in between. As a result, standard backpropagation cannot be done, and Monte Carlo sampling is needed to calculate the accuracy across various stages of backpropagation. Considering the accuracy is subject to how well the sampling is done, there is a need for other techniques.

II. Soft Attention: -

Soft attention works by weighting different parts of the image. Areas of high relevance is multiplied with a larger weight and areas of low relevance is tagged with smaller weights. As the model is trained, more focus is given to the regions with higher weights. Unlike hard attention, these weights can be applied to many patches in the image. Due to the deterministic nature of soft attention, it remains differentiable and can be trained with standard backpropagation. As the model is trained, the weighting is also trained such that the model gets better at deciding which parts to pay attention to. Soft attention implemented at the skip connections will actively suppress activations in irrelevant regions, reducing the number of redundant features brought across. The attention gates introduced by Oktay *et al* [16] uses additive soft attention. The example for soft attention is shown in Fig. 1.21.

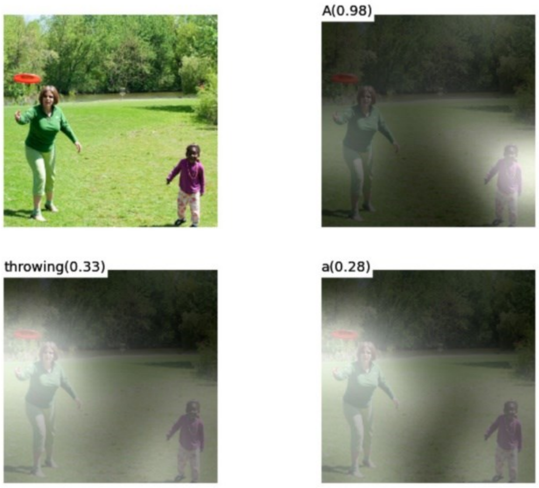

Fig. 1.21 Visualisation of soft attention weights on an image when processing different words in a sentence. Regions that are brighter have higher weights [16]

1.4 Motivation

The earlier models for automatic liver and tumor segmentation utilized a large number of parameters and very complex network. Due to large number of parameters, the time and memory consumption is much higher. For early detection of tumor and to save time of people, we need a model that is lightweight and simple. In our proposed model, we have used Feature pyramid network, feature similarity module and skip connections for improving the performance of tumor segmentation from the CT slices. We have incorporated feature pyramid network to fully utilize the high-resolution features in the decoder by combing the top-down pathway with the lateral connections. Feature similarity module is used to remove the long-range dependencies so as to improve the segmentation performance. The short skip connections are added to solve the problem of vanishing gradients in deep neural networks.

1.5 Outline of the project

The Chapter 1 in the report consists of Introduction and the terminologies that we have used throughout the report. Chapter 2 consists of Literature Survey and research gaps. The chapter 3 consists of the proposed model, the chapter 4 has the results obtained from the proposed model. The Chapter 5 contains the conclusion of the research study.

CHAPTER 2

LITERATURE SURVEY

In this section, we will describe medical image segmentation architectures similar to our research study. Häme *et al*. [17] a two-staged segmentation algorithm for Liver Tumor segmentation in CT scan images. In the first stage, a rough estimated segmentation of tumor is obtained using normal thresholding and utilizing normal morphological operations, the second stage refines the previous stage output using a fuzzy clustering approach and geometric deformable model.

Huang *et al*. [18] suggested a liver tumor segmentation algorithm based on an ensemble of extreme learning machines (ELMs), also used an ELM auto encoder to boost the overall performance. A classifier ensemble comprises of learning algorithms and classifies new voxel by a majority voting approach.

In recent years, deep learning has become a dominant research topic in numerous fields. Specially, Convolutional Neural Networks (CNN) have been used for many challenges in computer vision. CNN obtained outstanding performance on different tasks, such as visual object recognition, image classification, handwritten character recognition and more. Deep CNNs introduced by LeCun *et al*. [19], is a supervised learning model formed by multi-layer neural networks. CNNs are fully data-driven and can retrieve hierarchical features automatically by building high-level features from low-level ones, thus obviating the need to manually customize hand-crafted features.

Since then, CNNs have contributed significantly in the areas of image understanding. CNN-based approaches are placed in the leaderboard of the many image understanding challenges, such as Medical Image Computing and Computer Assisted Intervention (MICCAI) biomedical challenge, Brain Tumor segmentation (BRATS) Multimodal Brain Tumor Segmentation challenge [20] and Ischemic Stroke Lesion Segmentation (ISLES) challenge [21]. CNN has become a powerful choice as a technique for medical image understanding. Researchers have successfully applied CNNs for many medical image understanding applications like detection of tumors and their classification into benign and malignant, detection of skin lesions, detection of optical coherence tomography images, detection of colon cancer, blood cancer, anomalies of the heart, breast, chest, eye etc. Also, CNN-based models like CheXNet [22], used for classifying 14 different ailments of the chest achieved better results compared to the average performance of human experts. New challenges [23] are proposed on regular interval to improve medical image diagnosis and CNN is the primary choice of most of the researchers.

Ben-Cohen *et al*. [24] referred a fully convolutional neural networks for liver tumor segmentation in CT images. This was the first instance when FCNs were utilized over patch-based methods and outperformed previous methods by a significant margin.

Li *et al*. [25] proposed CNNs for Automatic segmentation of region consisting liver & tumor in CT scan. The architecture consists of 7 layers and the network was trained using small patches in order to get diversified feature maps. Christ *et al*. [26] proposed a cascaded FCNs particularly they utilized two cascaded U-Net with 3D dense conditional random fields (CRF) as post-processing step to achieve maximum accuracy as well as maintaining low computational complexity and less memory consumption.

Lu *et al*. [27] combined graph-cut methods along with 3D CNNs for effective localization of liver in CT scan images. Vivanti *et al*. [28] presented a four-staged approach consisting of deformable registration, automatic segmentation of liver, CNN voxel classifier, tumor segmentation in follow up the manner with the previously learned classifier. Also, Nowadays, deep learning methodologies are achieving exceptional results in various domains and by analyzing the recent research trend in liver and tumor segmentation from CT images and different challenges, we can conclude that deep learning methods are outperforming the previous state of the art by a significant margin. Also, novel approaches are being presented periodically.

Sun *et al*. [29] improved the accuracy of existing FCN based methods by introducing a multi-channel fully convolutional network. They exploited the fact that different imaging phase of CT scan images consists of various level of information about tumor. They merged these features at a high level and demonstrated improved results.

Chen *et al* [30] used a multi-plane integrated network (MPNet) to segment the liver from the abnormal CT images. Then, a deep fully convolutional neural network (DC-FCN) is designed as the generator to predict the liver tumor from the liver ROI, which is based on convolutional encoder-decoder backbone accompanied with multi-plane convolution, dilated convolution, dense connection and multi-scale feature fusion to enhance the network performance. The training of ADCN is equivalent to jointly optimize the cross-entropy loss with the adversarial training loss which is used to discriminate the output of DC-FCN and ground truth. The discriminator is a convolutional neural network which predicts its input belong to fake (output of DC-FCN) or real (ground truth).

Chlebus *et al*. [31] proposed modified residual U-Net kind of structures for the same task, while Vorontsov *et al*. [32] presented two FCNs, one on top of the other for segmentation of 2D axial slices. Both of them used U-Net type structure and achieved magnificent results showcasing the power of modifying U-Net and exploring different variants of it.

Bi *et al*. [33] A cascaded ResNet architecture to iteratively refine and constrain the lesion boundaries at both training and testing time. During training, the cascaded ResNet learns from the training data and the estimated results derived from the previous iteration. The ability to learn from the previous iteration optimizes the learning of both the liver and liver lesion boundaries, which are usually difficult to segment.

Meng *et al*. [34] proposed an algorithm that was mainly based on a 3D convolutional neural network with the dual scale from two paths. They Segmented the 3D CT images into several sub-image blocks, which were used as the input of TDP-CNN. There were two paths in the TDP-CNN, and each path was composed of eight blocks, and all the blocks had the same architecture. The feature maps of two paths were fused, and input into the fully connected layer, and then classified in the softmax layer. The trained TDP-CNN was used to segment the liver and liver tumor, and generate probability maps of the segmentation results; Finally, the probability maps were post-processed by a fully connected conditional random field algorithm to obtain the final segmentation results of liver and liver tumors.

Chen *et al* [35] proposed a Feature Fusion Encoder-Decoder Network (FED-Net) based 2D model for image segmentation. First, they design a novel feature fusion method based on the attention mechanism, which can effectively embed more semantic information into low-level features and improve the current feature fusion mode based on the U-Net architecture network. Then, to compensate for the information loss during the up-sampling process, they propose a dense up sampling convolution and also add residual convolutional blocks to refine the rough boundary of the target.

Kulava *et al* [36]. proposed a fully automatic 2-stage cascaded approach for segmentation of liver and its tumors in CT (Computed Tomography) images using densely connected fully convolutional neural network (DenseNet). They independently train liver and tumor segmentation models and cascade them for a combined segmentation of the liver and its tumor. Roth el al [37] implemented standard U-Net and Tiramisu architecture with Tversky loss function for effective segmentation.

Xu *et al*. [38] proposed a framework for liver segmentation in CT sequences using context information, much deeper network and quicker convergence strategy. They utilized U-Net with residual unit along with modified loss function which is combination of both Dice loss and Cross entropy loss. Lipkova *et al*.[39] used the Cahn-Hilliard equation to remove the noise and separate the mixture into two distinct phases with well-defined interfaces. This simplifies the lesion detection and segmentation task drastically and enables to segment liver lesions by thresholding the Cahn-Hilliard solution.

2.1 Research Gap

For obtaining optimal results in semantic segmentation, it is advisable to use low order features while also gaining information from higher order semantics and U-Net [13] is one of standard architecture when it comes to problem of sematic segmentation. But there are certain criteria where U-Net and its various versions are not perfect and still needs improvement. First one is, U-Net uses normal convolution while in our proposed model we have used a combination of normal and residual network this helps us in alleviate the training issues such as vanishing gradient. Secondly we have employed a FSM block for extracting high level features and to capture relationships between various pixels. Lastly, U-Net and most of its variants uses only features from last layer for defining output but our proposed model is based on the concept of Feature Pyramid networks on the decoder side, eventually we utilize all the feature maps for final segmentation purpose.

2.2 Objectives

I. To understand the nuances of Medical Image segmentation

II. To be proficient in implementing Deep CNN architectures in python

III. To build a simple CNN model for liver tumor segmentation

IV. To maximize the dice score coefficient for tumor segmentation

CHAPTER 3

PROPOSED MODEL FOR MULTI-FEATURE SIMILARITY FOR LIVER- TUMOR SEGMENTATION

3.1 Dataset description [40]

We have used Liver Tumor Segmentation Benchmark (LiTS) dataset which was the part of the IEEE ISBI 2017 and MICCAI 2017 conference. The dataset contains 201 abdomen computed tomography images. Out of the total dataset, 194 CT scans contains liver lesions. The data was collected from seven clinical institutions and academic institutes throughout the world. Manually, the data has been annotated with liver and lesion labels by trained radiologists. The dataset has been divided into 131 training instances and 70 testing instances.

3.2 Evaluation Metric

In accordance with the 2017 LiTS challenge, we have used Dice global score as the evaluation parameter to compute the performance of liver and tumor segmentation. Dice global is the score obtained for segmentation when all the datasets are combined into one. Dice score is the similarity index between two slices, more the dice score better will be the segmentation performance. The aim of our proposed study is to maximize the dice score to improve the liver tumor segmentation process. For the two instances A and B, the dice score (DSC) is given in Eq. (3.1).

$$\mathrm{DICE}(A, B) = \frac{2|A \cap B|}{|A| + |B|} \tag{3.1}$$

3.3 Proposed Model

For obtaining optimal results in semantic segmentation it is advisable to use low order features while also gaining information from higher order semantics and U-Net [13] is one of standard architecture when it comes to problem of sematic segmentation. But there are certain criteria where U-Net is not perfect and still needs improvement. First one is, U-Net uses normal convolution while in our proposed model we have used a combination of normal and residual network this helps us in alleviate the training issues such as vanishing gradient. Secondly we have employed a FSM block for extracting high level features and to capture relationships between various pixels. Lastly, U-Net and most of its variants uses only features from last layer for defining output but our proposed model is based on the concept of Feature Pyramid networks on the decoder side. The configuration for our proposed algorithm is shown in Fig. 3.1. The network structure of the model is given in Table 3.1.

Table 3.1. NETWORK STRUCTURE OF OUR MODEL

	Unit Level	**Layer**	**Output Size**		**Unit Level**	**Layer**	**Output Size**
Input			512×512×3	**Decoder**	Layer 5	DConv1	32×32×256
Encoder	Layer 1	Conv × 2	512×512×16			Concat1	32×32×512
		Pool	256×256×16			Conv × 2	32×32×256
	Layer 2	Conv × 2	256×256×32		Layer 4	DConv1	64×64×128
		Pool	128×128×32			Concat1	64×64×256
	Layer 3	Conv × 2	128×128×64			Conv × 2	64×64×128
		Pool	64×64×64		Layer 3	DConv1	128×128×64
	Layer 4	Conv × 2	64×64×128			Concat1	128×128×128
		Pool	32×32×128			Conv × 2	128×128×64
	Layer 5	Conv × 2	32 ×32×256		Layer 2	DConv1	256×256×32
		Pool	16×16×256			Concat1	256×256×64
FSM (Input)			16×16×256			Conv × 2	256×256×32
FSM (Output)	16×16×256				Layer 1	DConv1	512×512×16
				Output			512×512×1

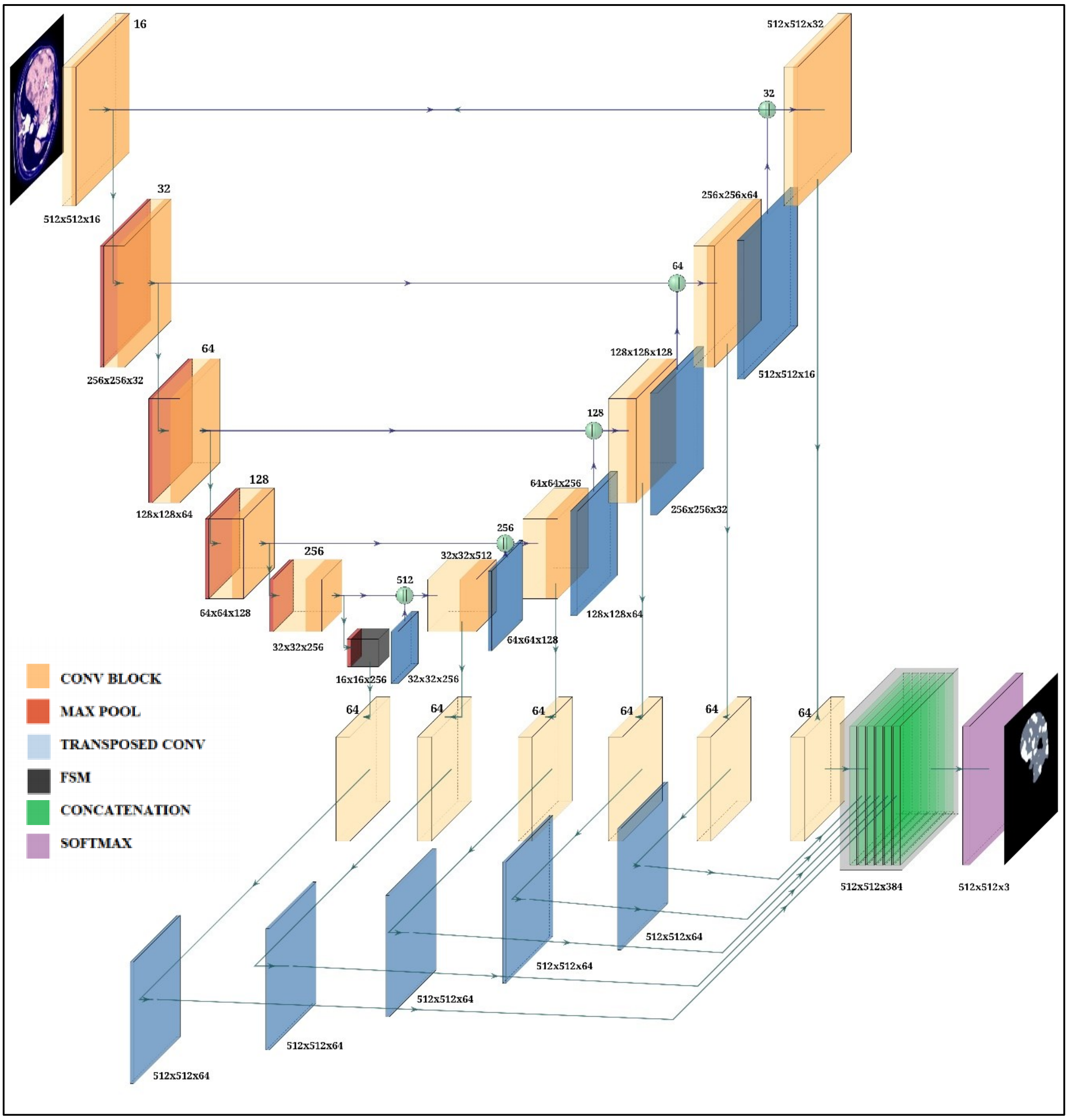

Fig. 3.1 Proposed Model

In our model, encoder and decoder both contain feature extraction blocks having structure same as shown in Fig. 3.2. Each block sequence of consecutive CONV-BN-ReLU with later one replicating residual connection. The filter size in doubled at each stage of encoding and resolution is down sampled by a factor of 2 at every layer. The bridge contains FSM Block as for obtaining high level semantics and describing pixel level relationship in the network. FSM is used to extract the long-range dependencies as to improve the segmentation performance. For decoder transposed convolution is used instead of up sampling and total trainable parameters are slightly less than 6 million. For designing feature pyramid, output from each layer is passed through feature extraction block described in Fig. 3.2, these outputs are again up sampled using transposed convolution and concatenated channel-wise and eventually generating final outcome. The skip short connections of ResNet are added to solve the problem of vanishing gradients in deep neural networks.

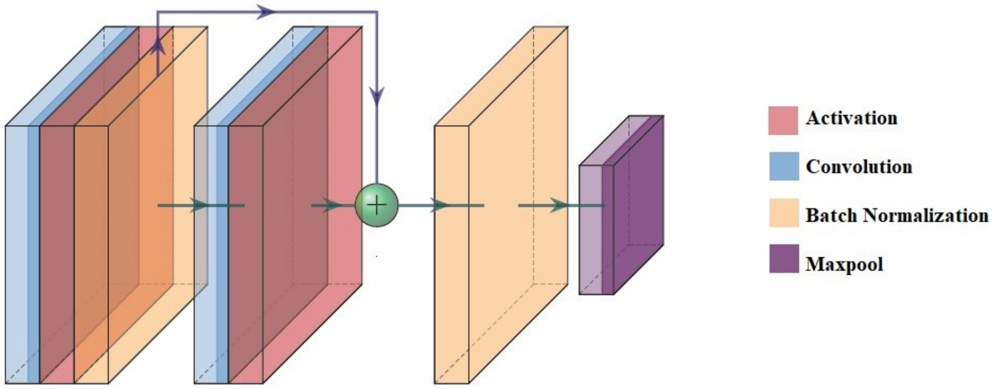

Fig. 3.2 CONV Block

3.4 Data Pre-processing

3.4.1 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 have 212 values which is 4096 ranging from -1024 HU to 3071 HU. 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.

$$HU = \frac{1000 \times (\mu tissue - \mu H_2O)}{\mu H_2O} \tag{3.1}$$

The HU values assigned to each pixel are computed by assigning a gray-scale intensity to each value, higher value mean brighter pixels. After slides are read in DICOM format, Hounsfield Windowing was applied to ranges [-100, 400]. The original image and the processed image after Hounsfield Windowing is shown in Fig. 3.3. (a) and (b) respectively.

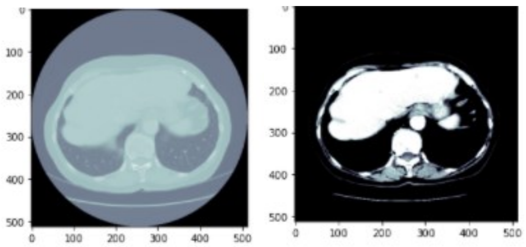

Fig. 3.3. (a) Original Image (b)Image after HU windowing

3.4.2 Histogram Equalization

Histogram equalization (HE) enhances the contrast in the images and especially when the contrast is contained in a narrow window. Histogram Equalization is applied to the images obtained after Hounsfield Windowing to increase the contrast between the liver and its neighbouring organs.

3.4.3 Conversion to JPEG Format

One of the many challenges we faced during the project implementation was handling the NII files. The LITS dataset contains 131 NII files in total. On direct utilization of NII files as input and convert them into npy arrays lead to memory issues as this process consumed more than 50 GB of disk space. So we converted the NII images to JPG reaching a total count of 58k images but occupying only a tenth of space i.e., approx. 5GB. So, to reduce memory consumption, we have converted the NII files to jpeg files. The original image is depicted in Fig. 3.4 (a) and the image after converting to JPEG format is shown in Fig. 3.4(b).

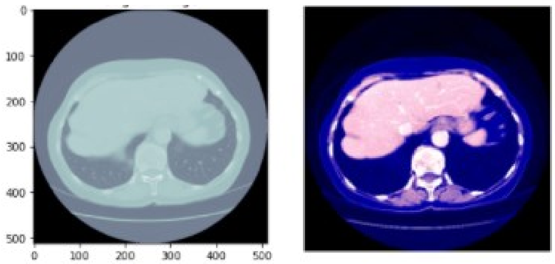

Fig. 3.4. (a) Original Image (b) Image after conversion to JPEG

3.5 Implementation Details

The model was implemented using TensorFlow and its high-level API Keras. There were total 58638 CT scan images, which were randomly shuffled in training and validation sets, 85% of the images were incorporated for training while rest were used for validation. The resolution of input image was 512×512×3. The network was optimized using Adam optimizer with learning rate 10-4 the learning rate was reduced by factor of 0.1 keeping the patience value as 3. Initially we chose cross-entropy and weighted cross entropy as loss function but later shifted to dice loss Eq. (3.2).

$$\text{Dice Loss}(A, B) \ = 1 \ - \frac{2|A \cap B|}{|A| + |B|} \tag{3.2}$$

We also employed standard data augmentation techniques such flipping and scaling apart from that certain complex augmentation techniques such as Elastic Transform and Grid Distortion were employed. The model was trained with batch size is equal to 4 for 7 epochs on Tesla P100 GPU with 16GB RAM. The total number of learnable parameters are close to 6 million.

3.6 Challenges faced

1. Dataset handling: The LITS dataset contains 131 NII files. While utilizing NII files directly as input and converting them into npy array, we faced memory issues as it would require 50GB plus disk space to store the npy array. To overcome the challenge, we converted the NII files into JPEG reaching a count of a total of 58K images but occupying only one-tenth of space i.e., 4.7GB.

2. RAM Overloading: after converting the NII files to JPEG images, we faced RAM out of memory error as appending this amount of images in a list of any other data structure lead to the total consumption of available RAM(19 GB). So, to deal with it, instead of loading the image, we provided the path to images as input and the code accessed the images using this path during training and for data visualization task.

3. Data visualization: Matplotlib internal support is different in both Google Colab and Kaggle and therefore there were issues regarding visualization of masks. Also, on both platforms while converting a nii file to Numpy array order of axes in different which forces us to stick to single platform.

CHAPTER 4

SIMULATION RESULTS

We have compared our performance of tumor segmentation for the LiTS dataset with various algorithms. We computed dice coefficient for lesion segmentation using U-Net, a hybrid of U-Net and Multi-Feature Pyramid (MPF), a hybrid of U-Net and Feature Similarity Module (FSM) and a hybrid of U-Net, MFP, FSM and skip connections of res block. Our proposed algorithm is a hybrid of these algorithms. From the Table 4.1 it is evident that our proposed algorithm has achieved better global dice coefficient than individual components for tumor segmentation.

In the Table 4.2, we have compared results of our algorithm with the previous research studies. X Chen *et al.* [35] calculated the lesion segmentation dice score as 0.745 using U-Net and ResNet, L Bi at al. [33] computed tumor segmentation dice score of 0.5001 for the cascaded ResNet model and J Lipková [39] obtained 0.53 dice score by implementing Cahn-Hiliard separation. K Roth [37] using 3D CNN with dual scale calculated dice score of 0.689 for tumor segmentation, KC Kaluva *et al.* [36] calculated tumor segmentation of 0.625 for 2D- Dense CNN and K Roth [37] computed a tumor segmentation dice score of 0.660. Our tumor segmentation score is 0.753 which is better than the previous algorithms and our model effectively uses the decoder layer parameters. Our model has close to 6M parameters which is comparatively less than the parameters consumed by earlier models. As validation set is not available in the LiTS dataset, but we have shown experimental results on our own validation data in Fig. 4.1. The input image displayed here is obtained after pre-processing with the above-mentioned techniques.

Table 4.1 COMPARISON OF OUR MODEL WITH INDIVIDUAL ALGORITHMS

Method	**Dice Global**
U-Net	0.680
U-Net + MFP	0.688
U-Net + FSM	0.705
U-Net + MFP + FSM	0.719
U-net + MFP + FSM + Res Block	**0.753**

Table 4.2 COMPARISON OF OUR MODEL WITH PREVIOUS STUDIES

Method	**Dice Global**
U-Net	0.680
U-Net + ResNet [35]	0.745
Cascaded ResNet (w Multi-scale Fusion) [33]	0.5001
Cahn-Hilliard separation (CHS) [39]	0.53 ± 0.27
3D CNN with dual scale [34]	0.689
2D- Dense CNN [36]	0.625
U-Net + Tiramisu (TL) LeS [37]	0.66
U-net + MFP + FSM + Res Block	**0.753**

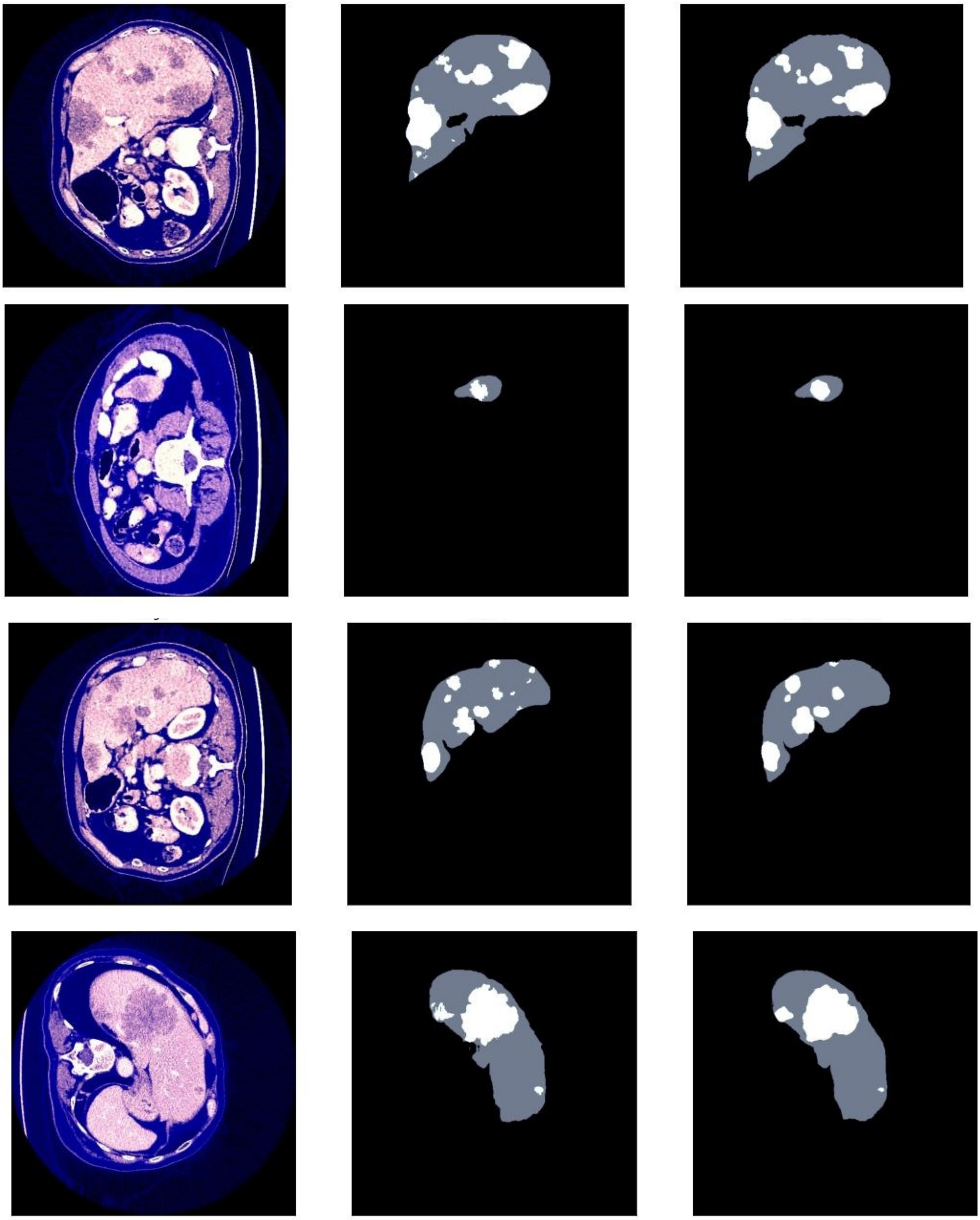

Fig. 4.1 (a) Input Image (b) Ground Truth Mask (c) Generated Mask

We have compared our results with some notable entries in LiTS challenge and outperformed some of the previous works. Our network incorporates feature pyramid structure which helps in final segmentation output and also while training through implicit deep supervision. FSM block in the bottleneck part is responsible of various pixel relationships, our network is underperforming when there is sharp edge especially and tumor and some techniques to refine edges can be incorporated to improve the overall results. Most of the submission in LiTS challenge is based on 2.5D or 3D approach while our method is completely developed upon 2D structure hence it can utilize for other tasks also. Furthermore, our model is trained with very less epochs in comparison to other architectures which require a lot of computation for training of model.

CHAPTER 5

CONCLUSION

In this report, we have proposed a novel segmentation Model for Liver-Tumor Segmentation. The proposed method employs advantages of both long and short skip connections also FSM module and Feature pyramid networks to gain maximum accuracy by increasing complexity a bit but also keeping very limited learnable parameters. The proposed model has performed better than well-known algorithms and by introducing certain complex modules we have improved representation of existing network up to certain extent. Still the prediction of our model is not perfect and there is scope for improvement, certain post-processing algorithms like 3D Conditional Random Field (3D CRF) and 3D connected components analyzing (3D CCA), can boost the final score. To further improve results, we suggest to use 3D CNN along with efficient convolution techniques like depth-wise separable convolution so as to increase the score and also keeping the number of parameters minimum, apart from that LSTM based techniques can also be incorporated as this task has spatio-temporal dimensions. As number of features at depth is substantially greater than at coarser resolution certain techniques must be include to integrate low level features. Techniques like attention mechanism or Edge based methods to preserve boundaries of object in interest, such works are recent inclusion in literature and attaining better results in certain task when compared to previous state of the art.

REFERENCES

1: "Liver Cancer - Statistics," Cancer.net, 25-Jun-2012. [Online]. Available: https://www.cancer.net/cancer-types/liver-cancer/statistics. [Accessed: 28-Apr-2021].

2: Sherif R Z Abdel-Misih and Mark Bloomston. Liver anatomy. The Surgical clinics of North America, vol 90, no 4, pages 643-653, 2010.

3: Sung, H, Ferlay, J, Siegel, RL, Laversanne, M, Soerjomataram, I, Jemal, A, Bray, F. "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries". CA Cancer J Clin .: vol 71, no 3, Feb., pages 209- 249, 2021.

4: G. Seif, "Semantic segmentation with deep learning - towards data science," Towards Data Science, 19-Sep-2018. [Online]. Available: https://towardsdatascience.com/semantic-segmentation-with-deep-learning-a-guide-and-code. [Accessed: 28-April-2021].

5: Gonzalez, R. C. and Woods, R. E. [2018]. Digital Image Processing, 4th ed., Pearson/Prentice Hall, NY.

6: Bieniek A, Moga A. "An efficient watershed algorithm based on connected components. Pattern recognition.", vol 33, no 6 pages 907-916,2000

7: Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. "Densely connected convolutional networks". In Proc of the IEEE conference on computer vision and pattern recognition, pages. 4700-4708, 2017.

8: M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal. "The importance of skip connections in biomedical image segmentation". In Deep Learning and Data Labeling for Medical Applications, Springer, pages 179–187 , 2016.

9: C.-F. Wang, "A basic introduction to separable convolutions - towards data science," Towards Data Science, 14-Aug-2018. [Online]. Available: https://towardsdatascience.com/a-basic-introduction-to-separable-convolutions. [Accessed: 21-April-2021].

10: K. Qi. "X-net: Brain stroke lesion segmentation based on depthwise separable convolution and long-range dependencies". In Lecture Notes in Computer Science,. Springer, 2019.

11: Moradi S, Ghelich-Oghli M, Alizadehasl A, Shiri I, Oveisi N, Oveisi M, Maleki M, Dhooge J. “A Novel Deep Learning Based Approach for Left Ventricle Segmentation in Echocardiography: MFP-Unet”. arXiv preprint arXiv:1906.10486. 2019.

12: Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. “M2det: A single-shot object detector based on multi-level feature pyramid network”. In AAAI, 2019.

13: O. Ronneberger, P. Fischer, and T. Brox. “U-net: Convolutional networks for biomedical image segmentation”. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer,2015.

14: Zhou Z, Siddiquee MM, Tajbakhsh N, Liang J. Unet++: “A nested u-net architecture for medical image segmentation”. In Deep learning in medical image analysis and bi- learning for clinical decision support, pages. 3-11, 2018

15: R. Azad, M. Asadi-Aghbolaghi, M. Fathy, and S. Escalera, “Bi-directional ConvLSTM U-net with densley connected convolutions”. In 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 2019.

16: O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz. “Attention u-net: Learning where to look for the pancreas”. arXiv preprint arXiv:1804.03999, 2018.

17: Häme, Y. “Liver tumor segmentation using implicit surface evolution”. The Midas Journal, pages.1-10, 2008.

18: Huang W, Yang Y, Lin Z, Huang GB, Zhou J, Duan Y, Xiong W. “Random feature subspace ensemble based extreme learning machine for liver tumor detection and segmentation”. In Proc. IEEE 36th annual international conference of the engineering in medicine and biology society pages. 4675-4678, 2014.

19: LeCun Y, Bottou L, Bengio Y, Haffner P. “Gradient-based learning applied to document recognition”. In Proc IEEE., vol 86 no 11 pages:2278-2324, 1998.

20: B. H. Menze *et al*., “The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS),”In Proc IEEE Trans. Med. Imaging, vol. 34, no. 10, pages. 1993–2024, 2015.

21: “ISLES: Ischemic stroke lesion segmentation challenge 2018,” Org. [Online]. Available: http://www.isles-challenge.org/. [Accessed: 29-April-2021].

22: Phillips NA, Rajpurkar P, Sabini M, Krishnan R, Zhou S, Pareek A, Phu NM, Wang C, Ng AY, Lungren MP. “Chexphoto: 10,000+ smartphone photos and synthetic photographic transformations of chest x-rays for benchmarking deep learning robustness”. arXiv preprint arXiv:2007.06199. 2020.

23: “Challenges,” Grand-challenge.org. [Online]. Available: https://grand-challenge.org/challenges/. [Accessed: 04-March-2021].

24: A. Ben-Cohen, I. Diamant, E. Klang, M. Amitai, and H. Greenspan, “Fully convolutional network for liver segmentation and lesions detection,” in Deep Learning and Data Labeling for Medical Applications, Springer, pages 77–85, 2016.

25: X. Li, H. Chen, X. Qi, Q. Dou, C. -W. Fu and P. -A. Heng, "H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation From CT Volumes". In IEEE Transactions on Medical Imaging, vol. 37, no. 12, pages. 2663-2674, 2018.

26: Christ PF, Ettlinger F, Grün F, Elshaera ME, Lipkova J, Schlecht S, Ahmaddy F, Tatavarty S, Bickel M, Bilic P, Rempfler M. “Automatic liver and tumor segmentation of CT and MRI volumes using cascaded fully convolutional neural networks”. arXiv preprint arXiv:1702.05970., 2017.

27: Lu F, Wu F, Hu P, Peng Z, Kong D. “Automatic 3D liver location and segmentation via convolutional neural network and graph cut”. International journal of computer assisted radiology and surgery, vol 12, no 2 pages 171-182, 2017.

28: Vivanti R, Ephrat A, Joskowicz L, Karaaslan O, Lev-Cohain N, Sosna J. “Automatic liver tumor segmentation in follow-up CT studies using convolutional neural networks”. In Proc. Patch-Based Methods in Medical Image Processing Workshop 2015.

29: Sun C, Guo S, Zhang H, Li J, Chen M, Ma S, Jin L, Liu X, Li X, Qian X. “Automatic segmentation of liver tumors from multiphase contrast-enhanced CT images based on FCNs”. Artificial intelligence in medicine., vol 83, pages 58-66,2017.

30: Chen L, Song H, Wang C, Cui Y, Yang J, Hu X, Zhang L.” Liver tumor segmentation in CT volumes using an adversarial densely connected network”, BMC bioinformatics., vol 20, pages 587-590, 2019.

31: 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. Scientific reports.", vol 8, no 1 pages:1-7, 2018.

32: Vorontsov E, Tang A, Pal C, Kadoury S. "Liver lesion segmentation informed by joint liver segmentation". In Proc 15th IEEE International Symposium on Biomedical Imaging (ISBI 2018) pages. 1332-1335, 2018.

33: Bi L, Kim J, Kumar A, Feng D. "Automatic liver lesion detection using cascaded deep residual networks". arXiv preprint arXiv:1704.02703.,2017.

34: Meng L, Tian Y, Bu S. "Liver tumor segmentation based on 3D convolutional neural network with dual scale". Journal of applied clinical medical physics.", vol. 21, no 1 pages 144-157, 2020.

35: Chen X, Zhang R, Yan P. "Feature fusion encoder decoder network for automatic liver lesion segmentation". In Proc IEEE 16th international symposium on biomedical imaging (ISBI 2019) pages. 430-433, 2019.

36: Kaluva KC, Khened M, Kori A, Krishnamurthi G. "2D-densely connected convolution neural networks for automatic liver and tumor segmentation". arXiv preprint arXiv:1802.02182., 2018.

37: Roth K, Konopczyński T, Hesser J. "Liver lesion segmentation with slice-wise 2d tiramisu and tversky loss function". arXiv preprint arXiv:1905.03639. ,2019.

38: Xu W, Liu H, Wang X, Qian Y. "Liver segmentation in CT based on ResUNet with 3D probabilistic and geometric post process". In Proc IEEE 4th International Conference on Signal and Image Processing (ICSIP) pages. 685-689, 2019.

39: Lipková J, Rempfler M, Christ P, Lowengrub J, Menze BH. Automated unsupervised segmentation of liver lesions in ct scans via cahn-hilliard phase separation. arXiv preprint arXiv:1704.02348. 2017.

40: "Competition," Codalab.org. [Online]. Available: https://competitions.codalab.org/competitions/17094. [Accessed: 01-May-2021].

PUBLICATIONS

We have prepared Manuscript for submission in IEEE International Conference. The title of our paper is “Multi-feature Similarity Based Deep Learning Framework for Semantic Segmentation” and the authors are Harshwardhan Bhangale, Raghav Bansal, Shrijeet Jain, and Jignesh Sarvaiya.

Multi-feature Similarity Based Deep Learning Framework for Semantic Segmentation

Harshwardhan Bhangale, Raghav Bansal, Shrijeet Jain, Jignesh Sarvaiya
Department of Electronics and Communication Engineering
Sardar Vallabhbhai National Institute of Technology
Surat, India
hbhangale3@gmail.com, raghav12bansal@gmail.com, sj1161199@gmail.com, jns@eced.svnit.ac.in

Abstract— **Liver tumor is one of the major causes of death among men and women, but it is confirmed that early detection of the disease ensures the long survival of the patient. In our research, a hybrid of Multi-feature pyramid based U-Net, short skip connections and a Feature similarity module are proposed for early detection of the tumor. The proposed algorithm focuses on improving the tumor segmentation performance with fewer training parameters. The robustness of the proposed algorithm is evaluated on the basis of the dice score coefficient of tumor segmentation. We have achieved a dice score of 0.753 and 0.950 on tumor and liver, respectively on the Liver Tumor Segmentation (LiTS) dataset. On comparison with earlier models, our model has achieved a higher dice coefficient with less training time and close to 6 million parameters.**

Keywords— liver-tumor segmentation, multi-feature pyramid network, feature similarity module, skip connections, LiTS dataset

I. Introduction

The exponential growth of cells in the liver is termed Liver cancer. Liver cancer is one of the main causes of death among both men and women. It is the fifth most commonly detected disease in men and the ninth most commonly detected disease in women. It is estimated that 42,430 adults consisting of 29,890 men and 12,340 women will be diagnosed with liver cancer in 2021. Also, research has shown that 30,320 people will die from this disease bifurcated into 20,300 men and 12,340 women [1]. But, it is confirmed that early detection of the disease ensures the long survival of the patient.

Computed Tomography has been used for the detection of tumor from the liver slices. Segmentation of liver and tumor facilitates early detection of the disease. But, the manual segmentation of liver tumor by radiologists is time-consuming, industrious, and most importantly, subjective to individual doctors. Therefore, an efficient and automatic technique for the segmentation of tumor is required.

The earlier models for automatic liver and tumor segmentation utilized a large number of parameters and were very complex network. Due to a large number of parameters, the time and memory consumption is much higher. So, we need a model that is lightweight and simple.

In our research, we have proposed a lightweight and simple automatic model for liver and tumor segmentation. We have incorporated Feature Pyramid U-net [26] with feature similarity module [3] and have added short skip connections in the encoder part. We are using the Dice coefficient of tumor segmentation as the parameter for performance evaluation.

The paper is described below. Section II is for related works. The methodology are discussed in Section III. In Section IV, the proposed algorithm is described. Results are discussed in Section V.

II. Related Work

In this section, we will briefly discuss CNN based segmentation architecture similar to our study. It's been a while ever since Computer-aided Diagnosis has assisted clinicians in their practice and decrease their effort in diagnosing various diseases. Häme et al. [4] a two-staged segmentation algorithm for Liver Tumor segmentation in CT scan images. In the first stage, a rough estimated segmentation of tumor is obtained using normal thresholding and utilizing normal morphological operations, the second stage refines the previous stage output using a fuzzy clustering approach and geometric deformable model.

Huang et al. [5] suggested a liver tumor segmentation algorithm build on an ensemble of extreme learning machines (ELMs), also used an ELM auto encoder to boost the overall performance. Ben-Cohen et al. [6] referred a fully convolutional neural networks for liver tumor segmentation in CT images. This was the first instance when FCNs were utilized over patch-based methods and outperformed previous methods by a significant margin.

Li et al. [7] proposed CNNs for automatic segmentation of region consisting liver & tumor in CT scan. The architecture consists of 7 layers and the network was trained using small patches in order to get diversified feature maps. In 2017, Christ et al. [8] proposed a cascaded FCNs particularly they utilized two cascaded U-Net with 3D dense conditional random fields (CRF) as a post-processing step to achieve maximum accuracy as well as maintaining low computational complexity and less memory consumption.

Lu et al. [9] combined graph-cut methods along with 3D CNNs for effective localization of liver in CT scan images. Li et al. [10] proposed a hybrid dense U-Net model which incorporates hybrid features, i.e. both intra-slice and inter slice features for optimizing the outcome of liver-tumor segmentation.