

Generating 3D print models from biomedical images via volumetric segmentation

CREATIVE & INNOVATIVE PROJECT (CS6611)

Reported by,

Team 25

S Sowmiya Sree - 2018103598

J Juanita - 2018103544

B. E. COMPUTER SCIENCE AND ENGINEERING



COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2021

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certificate that this report titled **GENERATING 3D PRINT MODELS FROM BIOMEDICAL IMAGES VIA VOLUMETRIC SEGMENTATION** is the bonafide work of **S Sowmiya Sree (2018103598)** and **J Juanita (2018103544)** who carried out the project work under my supervision, for the fulfillment of the requirements as part of the CS6611 Creative and Innovative Project Laboratory. Certified further that to the best of my knowledge, the work reported herein does not form part of any other thesis on the basis of which a degree or an award was conferred on an earlier occasion on these or any other candidates.

Place: Chennai

Angelin Gladston

Date: 28.05.2021

Associate Professor,

Department of Computer Science and Engineering,
Anna University, Chennai-25.

COUNTERSIGNED

Dr. Valli S,

Head of the Department,

Department of Computer Science and Engineering,
Anna University Chennai,
Chennai - 600 025.

ACKNOWLEDGEMENT

Foremost, we would like to express our sincere gratitude to our project guide, Ms. Angelin Glaston, Associate Professor, Department of Computer Science and Engineering, College of Engineering Guindy, Chennai for her constant source of inspiration. We thank her for the continuous support and guidance which was instrumental in taking the project to successful completion.

We are grateful to Dr. S. Valli, Professor and Head, Department of Computer Science and Engineering, College of Engineering Guindy, Chennai for her support and for providing the necessary facilities to carry out our project.

We would also like to thank our friends and family for their encouragement and continued support. We would also like to thank the Almighty for giving us the moral strength to accomplish our task.



S Sowmiya Sree



J Juanita

ABSTRACT

The prime idea is to generate a workflow that can be used to convert volumetric medical imaging data (MRI and CT images) to 3D models. Our work focuses on implementing advanced segmentation algorithms which would be capable of detecting smaller structures and separate boundary regions precisely followed by producing 3D models capable of 3D printing. The project is widely categorized into 3 modules namely, volumetric image segmentation, mesh refinement and 3D print modeling. The first module focuses on precise segmentation of tumors and the results are displayed in the results section. The next two modules focus on cleaning and generating 3D models.

MOTIVATION

Anatomical models are important training and teaching tools in the clinical environment and are routinely used in medical imaging research. These models can be produced from 3D printers. Volumetric scans are great source for extracting regions of medical interest via segmentation. Although direct measurement and analysis of 3D images is possible in some scenarios, segmented images are the basis for most 3D image analysis. Extraction of the geometry of a region of interest through 3D image segmentation allows for conversion into 3D models, which permits visualization and quantification of the scanned subject. Further virtual analysis of these models, for example through computer simulation, or even obtaining a physical representation of the subject through 3D printing, all require segmentation to be completed on the 3D images.

TABLE OF CONTENTS

1. <u>CHAPTER 1 (Introduction)</u>	4
2. <u>CHAPTER 2 (Related work)</u>	7
3. <u>CHAPTER 3 (System design)</u>	9
4. <u>CHAPTER 4 (Experimental results)</u>	18
5. <u>CHAPTER 5 (Result analysis)</u>	32
6. <u>CHAPTER 6 (Conclusion)</u>	37

CHAPTER 1 - Introduction

Anatomical models are important training and teaching tools in the clinical environment and are routinely used in medical imaging research. Medical image segmentation plays a pivotal role in computer-aided diagnosis systems which are helpful in making clinical decisions. In the clinic, the physical interaction with models facilitates learning anatomy and how different structures interact spatially in the body.

Simulation-based training with anatomical models reduces the risks of surgical interventions, which are directly linked to patient experience and healthcare costs. For example, improvement of central venous catheter insertions has been achieved by the use of anatomically and ultrasonically accurate teaching phantoms. In addition, the phantoms can be used for pre-operative surgical planning, which has been shown to be beneficial in craniofacial surgery and is being explored in a number of other surgical fields . Lastly, anatomical phantoms can be designed to mimic tissue when imaged with the modality of interest; most commonly ultrasound, Computed Tomography (CT), or Magnetic Resonance Imaging (MRI). Imaging phantoms are also important for the development of novel imaging modalities such as photoacoustics, or for

validation of image-based biomarkers such as pore size estimation using nuclear magnetic resonance, where they provide controlled experimental environments.

Segmenting a region of interest like an organ or lesion from a medical image or a scan is critical as it contains details like the volume, shape and location of the region of interest. Automatic methods proposed for medical image segmentation help in aiding radiologists for making fast and labor-less annotations. Early methods were based on traditional pattern recognition techniques like statistical modeling and edge detection filters. Later, machine learning approaches using hand-crafted features based on the modality and type of segmentation task were developed.

Recent advances in segmentation software have made it increasingly easy to automatically or semi-automatically extract the surface of structures of interest from three-dimensional (3D) medical imaging data. This has made it possible to generate anatomical models using a standard personal computer with little prior anatomical knowledge. At the same time 3D printers, traditionally used in industrial applications, are now available for home use thanks to low-cost desktop alternatives. This technology enables fast creation of 3D models without the need for classical manufacturing expertise.

The state of the art methods for medical image segmentation for most modalities like magnetic resonance imaging (MRI), computed tomography (CT) and ultrasound (US) are based on deep learning. As CNNs extract data-specific features which are rich in quality and effective in representing the image and the region of interest, deep learning reduces the hassle of extracting manual features from the image. Advances in segmentation algorithms and increased availability of 3D printers have made it possible to create cost-efficient patient-specific models without expert knowledge.

The prime idea is to generate a workflow that can be used to convert volumetric medical imaging data (MRI and CT images) to 3D models. Our work focuses on implementing advanced segmentation algorithms which would be capable of detecting smaller structures and separate boundary regions precisely (patient specific). The volumetric segmentation is followed by mesh refinement which focuses repairing, smoothing and appending with other related segments. This post-processing of the segments is expected to generate clean deliverables for 3D model generation. Software like Autodesk Meshmixer can be used for refining and visualizing the 3D models. The core intention is to build efficient image segmentation architecture which serves as the backbone for the entire workflow.

CHAPTER 2 - Related work

The segmentation of biomedical images have been a topic of great interest since the past. Parallelly, there also have been works for modelling 3D objects. Seg-Net [1] was the first such type of network that was widely recognized. In the encoder block of Seg-Net, every convolutional layer is followed by a max-pooling layer which causes the input image to be projected onto a lower dimension similar to an undercomplete auto-encoder. The receptive field size of the filters increases with the depth of the network thereby enabling it to extract high-level features in the deeper layers. The initial layers of the encoder extract low-level information like edges and small anatomical structures while the deeper layers extract high-level information like objects (in the case of vision datasets) and organs/lesions (in the case of medical imaging datasets). A major breakthrough in medical image segmentation was brought by U-Net [2] where skip connections were introduced between the encoder and decoder to improve the training and quality of the features used in predicting the segmentation. U-Net has became the backbone of almost all the leading methods for medical image segmentation in recent years. Subsequently, many more networks were proposed which built on top of U-Net architectures. The U-Net++ proposed using nested

and dense skip connection for further reducing the semantic gap between the feature maps of the encoder and decoder. The UNet3+ proposed using full-scale skip connections where skip connections are made between different scales. 3D U-Net [3] and V-Net [4] were proposed as extensions of U-Net for volumetric segmentation in 3D medical scans. The main problem with the above family of networks is that they lack focus in extracting features for segmentation of small structures. As the networks are built to be more deeper, more high-level features get extracted. Even though the skip connections facilitate transmission of local features to the decoder, from our experiments we observed that they still fail at segmenting small anatomical landmarks with blurred boundaries. Although U-Net and its variants are good at segmenting large structures, they fail when the segmentation masks are small or have noisy boundaries. The U-Net 3D failed to capture the low level features in the 3D image. To the contrary, Kite-Net 3D was good at capturing the low level features but couldn't retain the overall shapes of the provided image. Hence, both of these architectures were combined to produce over-complete convolutional architecture KiU-Net 3D [5]. Delving into the 3D modeling part, the publication by group of radiologists in America [6] talked about the emerging future of 3D printing in medical/surgical field. Yet another publication by a group of medical

researchers [8] talks about replacement of reconstructing the 3D vessel geometry from computed tomography patient scans with the 3D printing to produce phantoms. The detailed procedures involved in transforming the volumetric images to efficient 3D print models is thoroughly investigated in [9].

CHAPTER 3 - System design

The overall architecture of the project is as follows

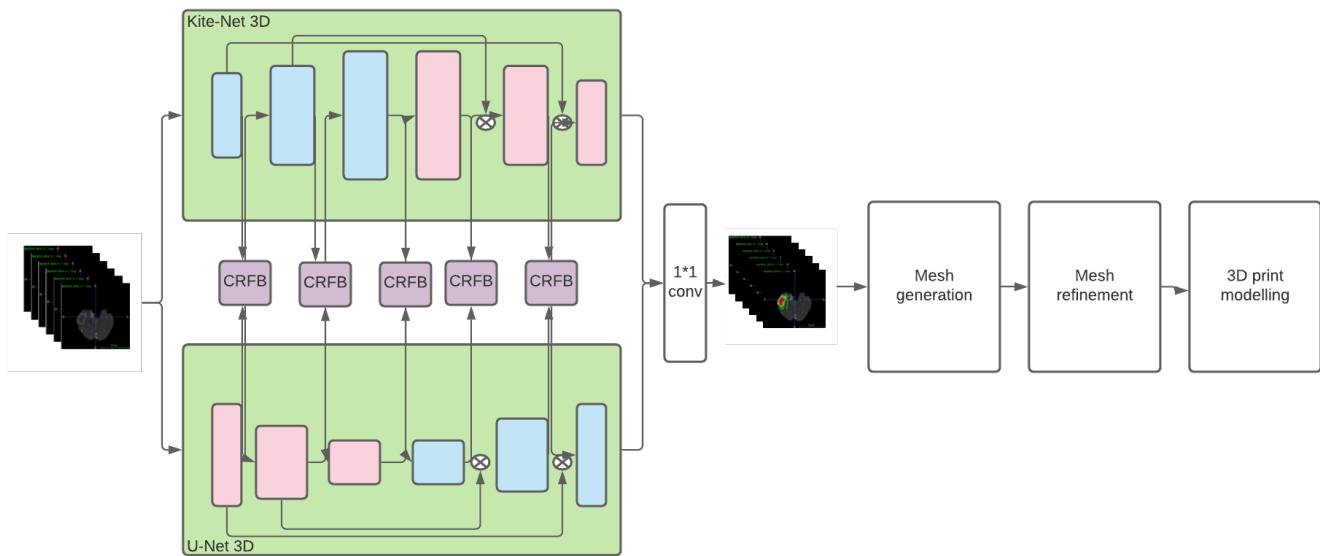


Figure 1 : The overall architecture diagram of the proposed system

The overall project as shown in Figure 3 is split into 3 big modules whose aim, input, process involved & outputs are mentioned below.

In this section, we describe the the process of going from medical imaging data (CT scan) to a finished 3D printed model from a general point of view. We have implemented a streamlined pipeline on regions of interest, which will be described in the sections below.

MODULE 1 - Volumetric segmentation

Segmentation is a process that partitions an image into regions. It is an image processing approach that allows us to separate objects and textures in images.

This module focuses on segmenting 3D biomedical images accurately. Prior works lack detecting small structures and finer edges. Our prime goal is to identify sharp edges and minute structures in the scan.

Input - 3D MRI (Brain) and CT (Liver) scans

Process - The input scans in voxel space are initially preprocessed and fed into the devised CNN model. The model is split into 4 major components namely,

- *Kite-Net 3D* - an over complete convolutional network which learns to capture fine details and accurate edges of the input.
- *U-Net 3D* - for learning high level features.
- *Cross residual fusion block* - residual features of Kite-Net 3D are learned and added to the features of U-Net 3D to forward

the complementary features to U-Net and vice-versa.

- *1*1 3D Convolution* - The feature maps from last layer of both the branches (Kite-Net 3D and U-Net 3D) are added and passed into this 1*1 3D convolutional layer to get the prediction.

Output - Raw 3D segments

MODULE 2 - Mesh refinement

This module focusses on mesh manipulation and refinement of the predicted segment by provided by the deep learning model.

Input - Extracted segments (from input scans)

Process - The raw segments are fine tuned and mesh refined so that the segments are error-free in the 3D printing software's (Cura) environment. Apart from resolving errors, additional preprocessing are essential for rendering accurate 3D models. The prime tasks involves are,

- *Repairing* - Errors and discontinuities that arise in the image segmentation and exporting process are repaired.
- *Smoothing* - Smoothing the surface of the mesh model to further mitigate the stair-casing error resulting from the resolution of original medical image.
- *Appending* - Combine all the necessary segments and removing unneeded parts from segmentation.

Output - Mesh refined segments capable of 3D model generation

MODULE 3 - 3D model generation

This module focusses on processing the mesh refined models from the previous module and generating 3D models capable of 3D printing.

Input - Mesh refined models

Process - The mesh refined models are processed thoroughly and rendered to produce models for 2 main purposes as follows,

Segment analyzation - The segmented portion from the scan i.e., Tumor from brain scans and lesion from liver scans are rendered with intrinsic features capable of examination by surgeons.

Segment localisation - The identified segment is projected with along with its organ to precisely locate the defect.

Output - Final 3D models ready to be deployed for 3D printing, intrinsic evaluation (surgeons & anatomy students)

There are many different 3D printing technologies available, each with their own characteristics. Here we provide an overview of the 3D printing methods, which are suitable for the creation of anatomical models and highlight their respective advantages. The

relevant 3D printing technologies can be classified into three groups: extrusion printing, photopolymerisation and powder-based printing. The most common example of extrusion printing is known as Fused Deposition Modelling (FDM), which is based on melting and depositing a material via a nozzle, building the desired shape layer by layer. In photopolymerisation, liquid polymers are selectively cured, typically using UV light. Important examples are Stereolithography (SLA) and Digital Light Processing (DLP), which selectively cure a plastic in a bath. Moreover, the photopolymer can be sprayed onto the print in thin layers, where it is subsequently cured. This technique is known as Material Jetting (MJ).

SECTION 3. 1 - FUNCTIONAL REQUIREMENTS

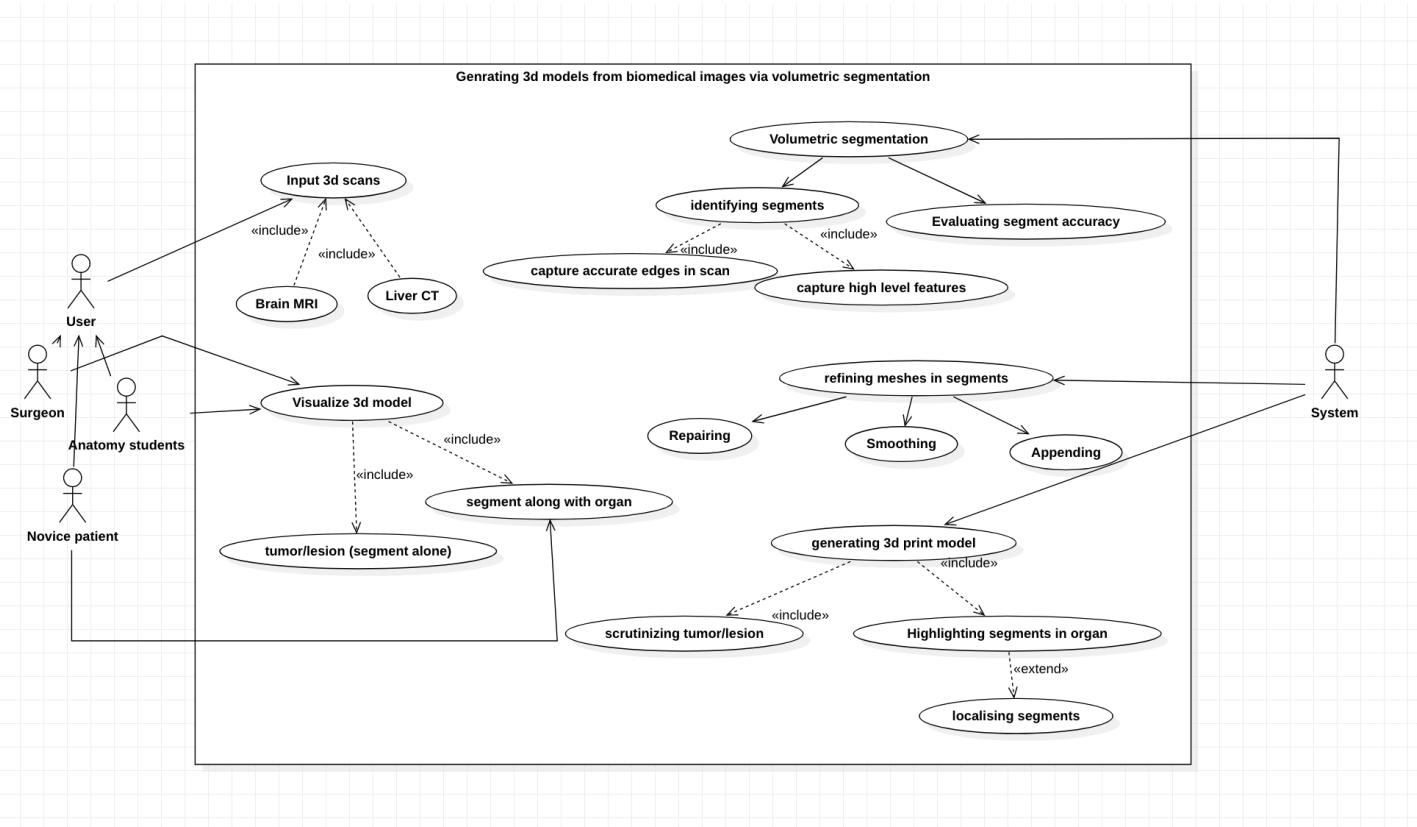


Figure 2 : the functional requirements depicted in the form of use case diagram

USE CASES	EXPLANATION
Input 3d scans	We take the MRI brain scans as provided for the MICCAI BraTS 2020 challenge
Visualize 3d models	The surgeons and anatomy students who need to study the segmented portions get access to scrutinize the tumors as well as localizing the tumors in the brain. Whereas novice patients who are unaware of the intrinsic details, get to visualize the where the defect is located in their organ and how it looks like.
Volumetric segmentation	The input scans in 3d voxel space is passed into the over complete CNN model to segment the defect and is followed by evaluating the accuracy of segmentation. The metrics used for the evaluation are specific for 3d image segmentation tasks (DICE and JAC)
Refining meshes in segments	To transform the raw segments obtained from segmentation process, into usable form in 3D printing software's environment, mesh refinement is done. The process is inclusive of Repairing, Smoothing and Appending.
Generating 3d print model	The mesh refined models are transformed into the final 3D models capable of 3D printing used by the anatomical students for practical studies

SECTION 3. 2 - TECHNICAL REQUIREMENTS

MODULE 1 - VOLUMETRIC SEGMENTATION

Pycharm CE / Jupyter (Anaconda)

1. GPU compatible with OpenGL 3.2
2. RAM: 8 GB
3. Graphics card compatible with CUDA

MODULE 2 - MESH REFINEMENT

Autodesk Meshmixer 3.5 (state-of-the-art software for working with triangle meshes. Dedicated for tasks like cleaning up 3d scans)

MODULE 3 - 3D PRINT MODEL GENERATION

Cura 4.8.0 (an open source slicing application for 3D printers.)

SECTION 3.3 - COMPARISON OF THE EXISTING MODEL & PROPOSED MODEL

The base paper as submitted [5] focuses only on segmenting the 3D scans. Whereas, we have proposed a system that would use this segmentation to devise 3D models for surgical and academic purposes

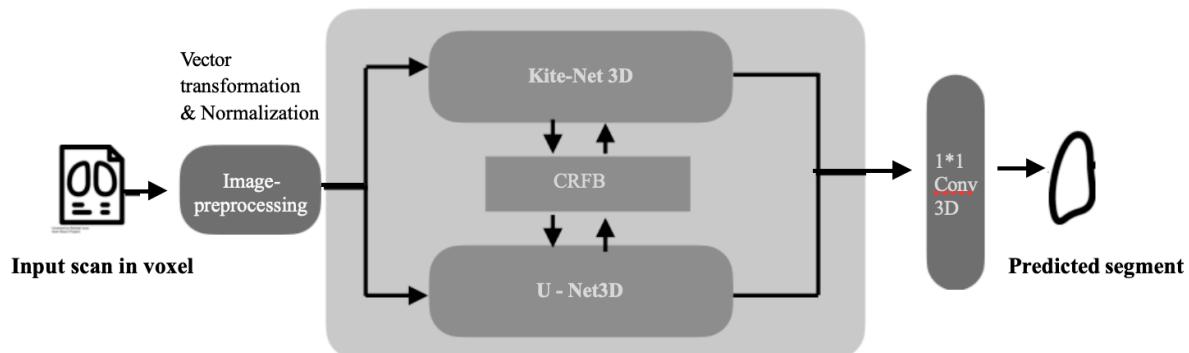


Figure 3: The existing system which generates segment from input 3D scan

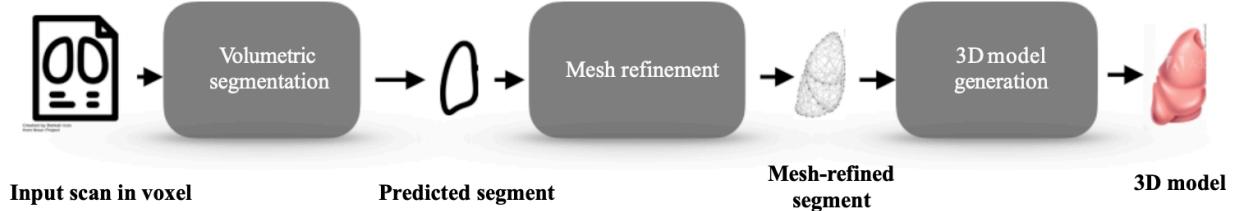


Figure 4 : The proposed system which generates 3D model from the segments

Figure 3 and 4 : The existing system uses image preprocessing by vector transforming ans Normalization for predicting segments. We propose a system using over- complete convolutional architectures for segmentation. We call our overcomplete architecture Kite-Net (Ki-Net) which transforms the input to higher dimensions (in the spatial sense). Note that Kite-Net does not follow the traditional encoder-decoder style of architecture, where the inputs are mapped to lower dimensional embeddings (in the spatial sense). Compared to the use of max-pooling layers in the traditional encoder and upsampling layers in the traditional decoder, Kite-Net has upsampling layers in the encoder and max-pooling layers in the decoder. This ensures that the receptive field size of filters in the deep layers of the network does not increase like in U-Net. This ensures that the Kite-Net is able to extract fine details of boundaries as well as small structures even in the deeper layers. Although Kite-Net extracts high quality low-level features, the lack of filters extracting high-level features makes Kite-Net not perform on par with U-Net when the dataset consists of both small and large structure annotations.

CHAPTER 4 - Experimental results

SECTION 4.1 - Dataset details

The 3D scans in MRI format provided as a part of the MICCAI BraTS challenge was chosen for the project. BraTS utilizes multi-institutional pre-operative MRI scans and focuses on the segmentation of intrinsically heterogeneous (in appearance, shape, and histology) brain tumors, namely gliomas. Furthermore, to pinpoint the clinical relevance of this segmentation task, BraTS also focuses on the prediction of patient overall survival, via integrative analyses of radiomic features and machine learning algorithms.

The Dataset provides images of 4 contrasts namely,

1. ***Flair*** - Fluid-attenuated inversion recovery (FLAIR) is a magnetic resonance imaging (MRI) sequence that produces strong T2 weighting, suppresses the CSF signal, and minimizes contrast between gray matter and white matter.

2. ***T1*** - The list of entities associated with a high signal intensity on T1-weighted images is extensive and classically

includes fat, proteins, hemorrhage, melanin and gadolinium.

3. **T1ce** - The T1 scan with enhanced contrast

4. **T2** - T2 reflects the length of time it takes for the MR signal to decay in the transverse plane. A short T2 means that the signal decays very rapidly. So substances with short T2's have smaller signals and appear darker than substances with longer T2 values.

Acquiring the dataset involves following certain procedures and it is provided as 2 folders namely, Training and Validation. Each folder consists of several sub-folders each of which corresponds to the input scans of a single sample. Inside these folders are several contrasts of the sample considered. The training set additionally consists of the tumor segment for training the model whereas the validation folder doesn't contain any of those.

As in [figure 5](#), the input scan is visualized in Mango software, Mango (Multi-Image Analysis GUI) is a non-commercial software for viewing, editing and analyzing volumetric medical images. Mango is written in Java, and distributed freely in precompiled versions for Linux, Mac OS and Microsoft Windows.

As in [figure 6](#), the input scan for all 4 contrasts: Flair,T1,T2 and T1 contrast has been shown as multimedia scans

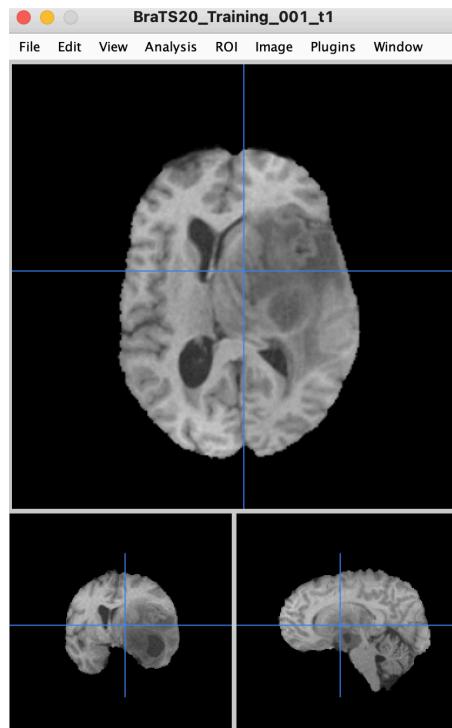


Figure 5: Input scan visualized in Mango software

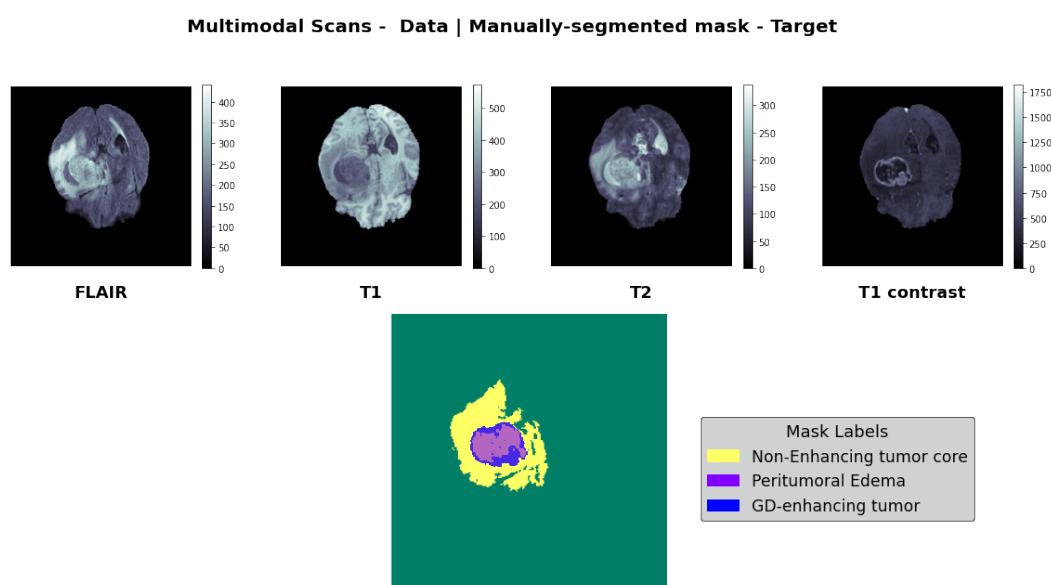


Figure 6: The 4 contrasts of inputs visualized via python code

SECTION 4.2 - Experimental setup

MODULE 1 - Volumetric segmentation

The main algorithm involved is the ***KiU-Net 3D*** along with the preprocessing procedures are explained below

- Images are loaded with pixel representation features
- Extraction of feature map
- The input image is forwarded to both Kite-Net and U-Net in parallel
- Over-complete network enables higher flexibility for capturing structure in the data
- Kite-Net, the encoder projects the input image into a spatially higher dimension by incorporating bilinear upsampling layers in the encoder.
- We assume that the pooling coefficient and stride are both set as 2 in both overcomplete and under- complete network.
- In each conv block in the decoder of Kite-Net, we have a conv 2D layer followed by a max- pooling layer with kernel size of two and ReLU activation
- To combine the features of the two networks at multiple scales through a novel cross residual feature block (CRFB)
- In the encoder of Kite-Net 3D branch, every conv block has a conv 3D layer followed by a trilinear upsampling layer with coefficient of two and ReLU activation
- The output of both the branches are then added and forwarded to $1 \times 1 \times 1$ conv 3D layer to get the prediction voxel

DEEP INVESTIGATION INTO THE SEGMENTATION ARCHITECTURES

U - Net 3D

The network consists of a contracting path and an expansive path, which gives it the u-shaped architecture. The contracting path is a typical convolutional network that consists of repeated application of convolutions, each followed by a rectified linear unit (ReLU) and a max pooling operation. During the contraction, the spatial information is reduced while feature information is increased. The expansive pathway combines the feature and spatial information through sequence of up-convolutions and concatenations with high-resolution features from the contracting path. The visual representation of the image is given in Figure 3.

Drawbacks - Experiments suggest that these methods fail to detect tiny structures in most of the cases. This does not cause much decrement in terms of the overall dice accuracy for the prediction since the datasets predominantly contain images with large structures. However, it is crucial to detect tiny structures with a high precision since it plays an important role in diagnosis. Furthermore, even for the large structures, U-Net based methods result in erroneous boundaries especially when the boundaries are blurry.

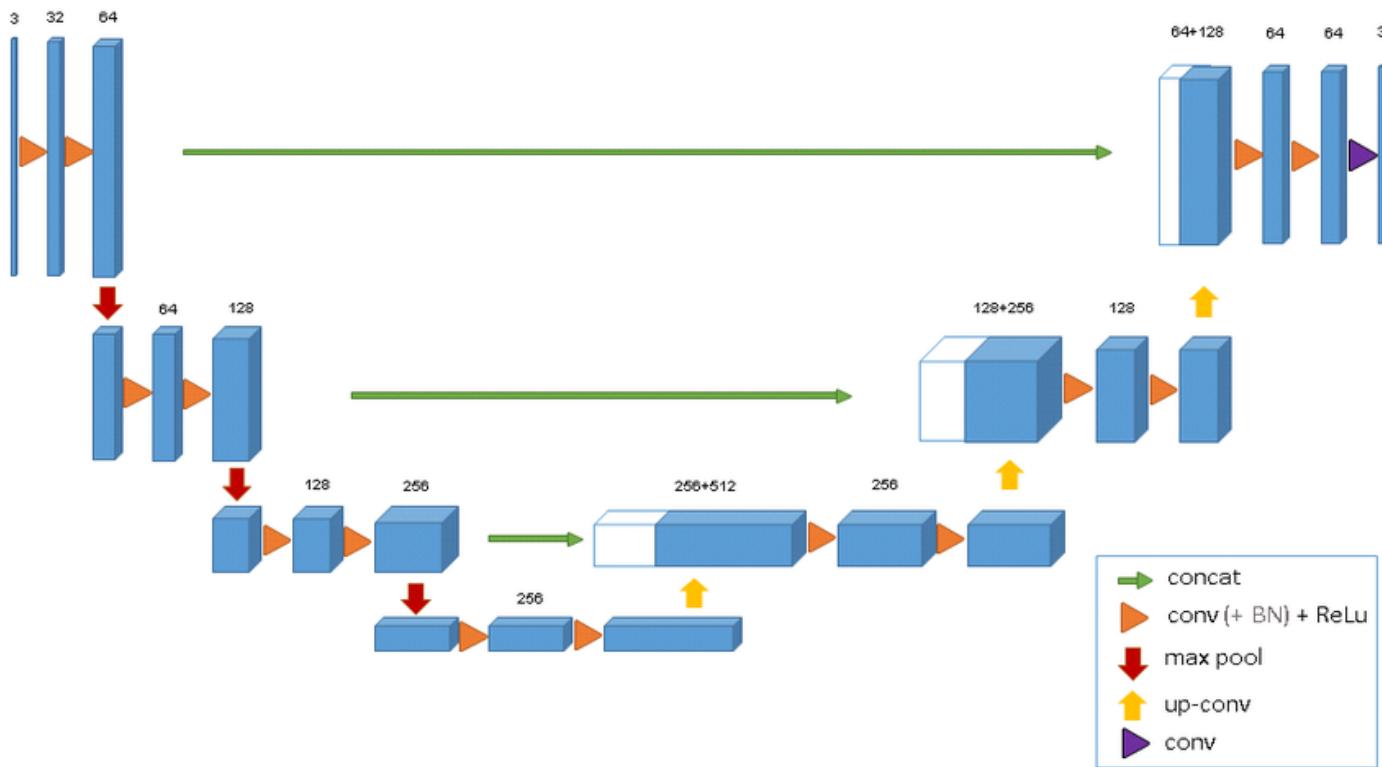


Figure 7 : The basic U-Net 3D architecture

Figure 7 clearly depicts the first sight, as it has a “U” shape. The architecture is symmetric and consists of two major parts — the left part is called contracting path, which is constituted by the general convolutional process; the right part is expansive path, which is constituted by transposed 2d convolutional layers.

Kite-Net 3D

Kite-Net learns low-level features better than U-Net, it does not learn any high-level features. Due to this, Kite-Net is unable to

segment out any large masks present in the input image. To overcome this, we propose KiU-Net 3D, which efficiently combines both Kite-Net and U-Net 3D while achieving the best of both networks.

KiU-Net 3D : U-Net 3D and Kite-Net combined

KiU-Net is a two-branch network where one branch is Kite-Net and the other is U-Net 3D. This network exploits the low-level fine edge capturing power of Kite-Net 3D as well the high-level shape capturing power of U-Net 3D. The input image is forwarded to both Kite-Net 3D and U-Net 3D in parallel. In the encoder of Kite-Net 3D branch, every convolution block has a convolution 3D layer followed by a trilinear upsampling layer with coefficient of two and ReLU activation. In de-coder, every convolution block has a convolution 3D layer followed by a 3D max-pooling layer with coefficient of two and ReLU activation. Similarly, in the encoder of U-Net 3D branch, every convolution block has a convolution 3D layer followed by a 3D max-pooling layer with coefficient of two and ReLU activation. In decoder, every convolution block has a convolution 3D layer followed by a trilinear upsampling layer with coefficient of two and ReLU activation. The visual representation of the architecture is shown in figure 4.

CRFB - In order to further exploit the capacity of the two networks, we propose to combine the features of the two networks at multiple scales through a novel cross residual feature block (CRFB). That is, at each level in the encoder and decoder of KiU-Net 3D, we combine the respective features using a CRFB. As we know that the features learned by U-Net 3D and Kite-Net 3D are different from each other, this characteristic can be used to further improve the training of the individual networks. So, we try to learn the complementary features from both the networks which will further improve the quality of features learned by the individual networks.

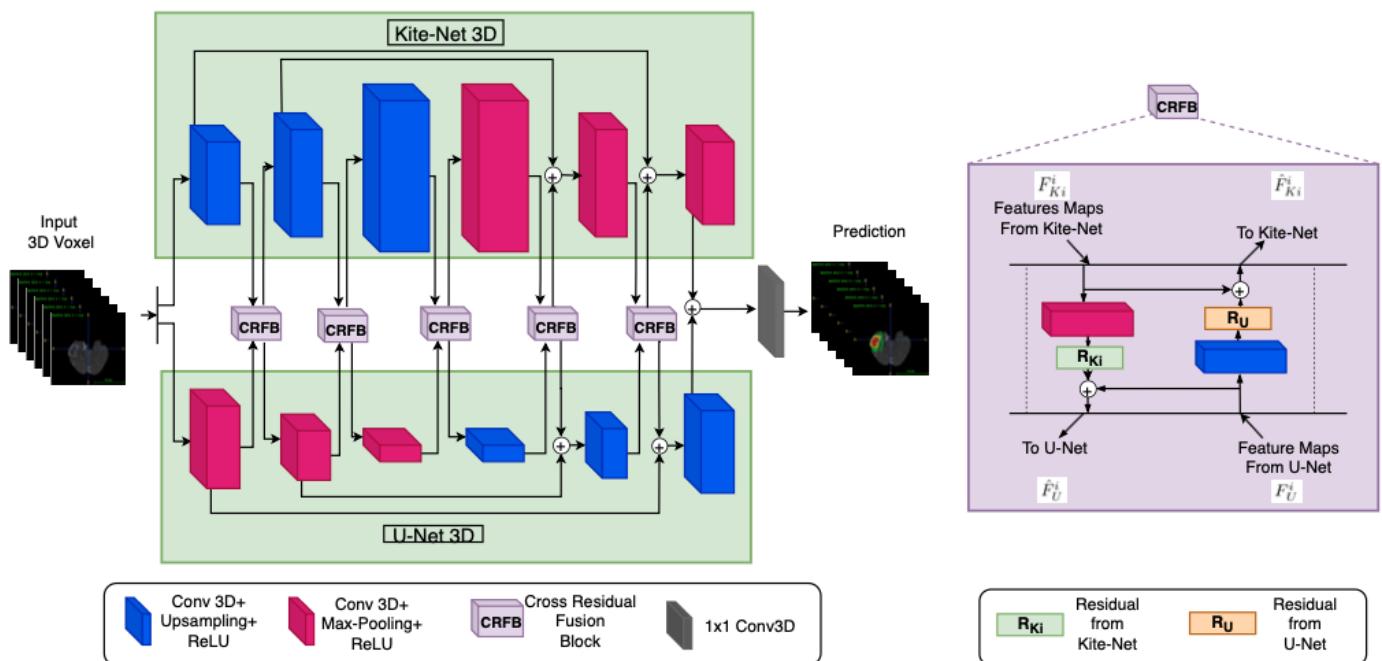


Figure 8: The proposed architecture which combines U-Net 3D and Kite-Net 3D via CRFB:Cross Residual Fusion Block

Figure 8 : In figure 8(a), we have Architecture details of KiU-Net 3D for 3D volumetric segmentation. In figure 8 (b), we have the Details of Cross Residual Fusion Block (CRFB) for KiU-Net 3D. In KiU-Net 3D, the input 3D voxel is forwarded to the two branches of KiU-Net 3D: Kite-Net 3D and U-Net 3D which have 3D CRFB blocks connecting them at each level. The feature maps from the last layer of both the branches are added and passed through 1×1 3D conv to get the prediction. In CRFB, the residual features of Kite-Net 3D are learned and added to the features of U-Net 3D to forward the complementary features to U-Net and vice-versa.

MODULE 2 - Mesh refinement

There is no specific algorithm for this module. The series of procedures involved in creating and refining the mesh from the generated raw segment is given below

- Mesh generation
 1. The raw segmented tumor in the form of a torch tensor is initially converted into 3D numpy array
 2. The vertices and faces of the 3D image in numb array is extracted with the help of “meshplot” library
 3. These vertices and faces are then used to create a mesh object with the help of “dataToMesh” function from the

same library and the 3D object is saved with “.stl” extension

- Mesh refinement
 - 4. Making the surface of the mesh smoother - the degree of smoothness is customizable for different portions of the mesh object
 - 5. Further techniques for smoothing the surface like, “global smoothing” and “Robust smooth” removes step artifacts due to finite voxel size
 - 6. The unphysiological holes in the mesh object was removed with the help of “Flatten” and “Inflate” brushes

MODULE 3 - 3D print model generation

There is no specific algorithm for this module. The series of procedures involved in defining and customizing the imported mesh which is then transformed into 3D model is given below

- Making small corrections in the cleaned mesh file
- Customizing the layer height
- Adjusting the thickness (top, bottom, left, right)

- Providing the infill percentage for portions of the 3D model
- Deciding the required print material

SECTION 4.3 - Results

The results of the volumetric segmentation is measured in terms of two specific metrics namely, dice and Jaccard metrics and the conventional accuracy as well.

The Dice score and Jaccard index are commonly used metrics for the evaluation of segmentation tasks in medical imaging.

Convolutional neural networks trained for image segmentation tasks are usually optimized for (weighted) cross-entropy. This introduces an adverse discrepancy between the learning optimization objective (the loss) and the end target metric

1. **Dice coefficient (DICE)** - $2 * \text{Area of Overlap} / (\text{total number of pixels in both images})$. Also called the overlap index, is the most used metric in validating medical volume segmentations.

$$DICE = \frac{2 |(S_g)^1 \cup (S_t)^1|}{|(S_g)^1| + |(S_t)^1|} = \frac{2TP}{2TP + FP + FN} \dots\dots\dots (1)$$

2. **Jaccard (JAC)** - The Intersection over Union (IoU) metric, is essentially a method to quantify the percent overlap between the target mask and our prediction output. This metric is closely related to the Dice coefficient which is often used as a loss function during training.

$$JAC = \frac{|(S_g)^1 \cap (S_t)^1|}{|(S_g)^1 \cup (S_t)^1|} = \frac{TP}{TP + FP + FN} \dots\dots\dots (2)$$

Further, there are 3 kinds of classes of tumors (WT, TC, ET), which are tested for segmentation accuracy with the given model.

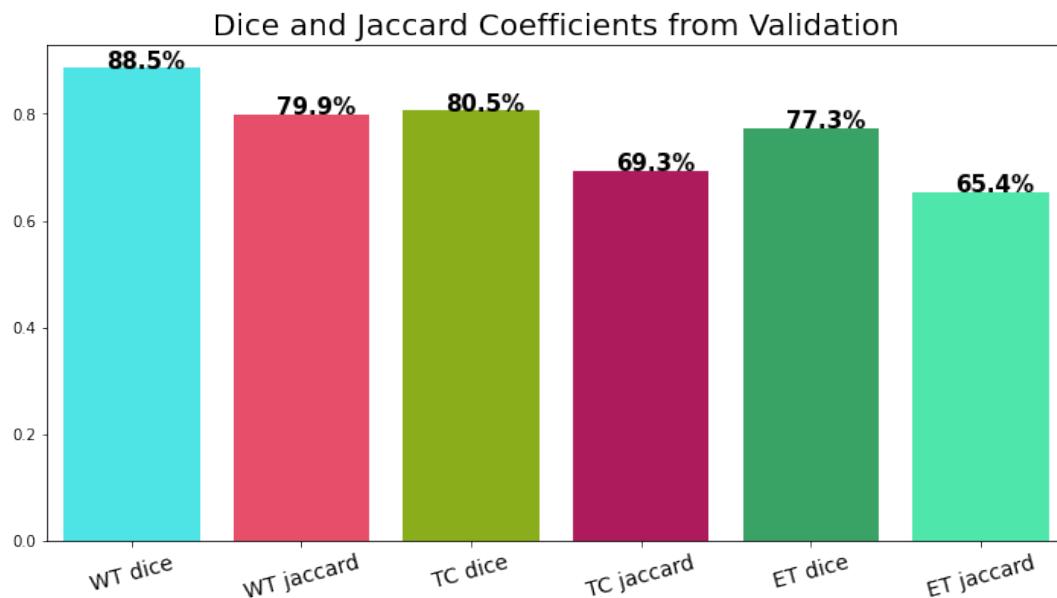


Figure 9 : The DICE and JAC scores of each class of tumor

The overall accuracy plots inclusive of loss, DICE and JAC are plotted for train and test set as follows

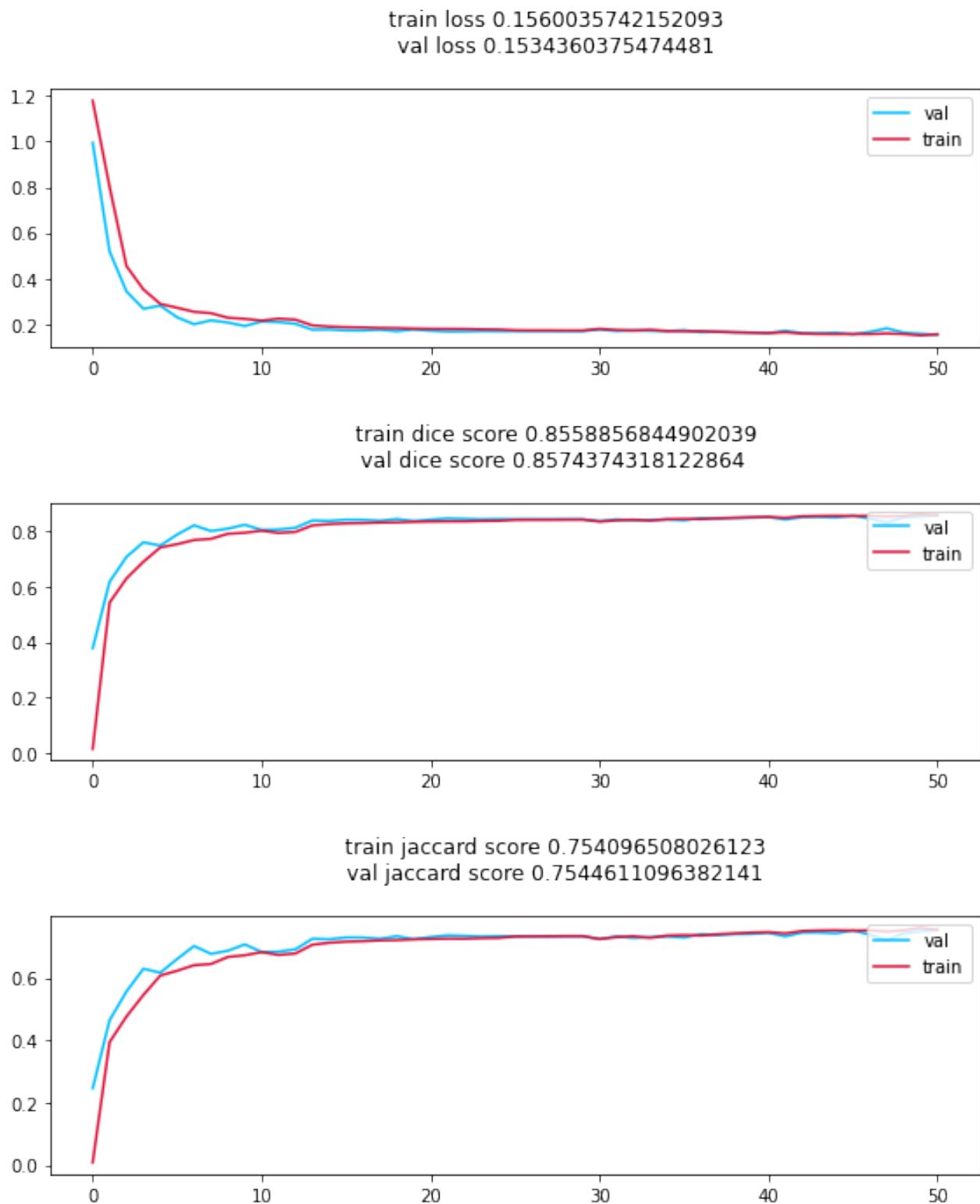


Figure 10 : The accuracy curves of DICE and JAC metrics and the loss curves of the overall segmentation system

Having numerically analyzed the performance of the model, we have also visualized the segmented tumors via a gif file as follows.

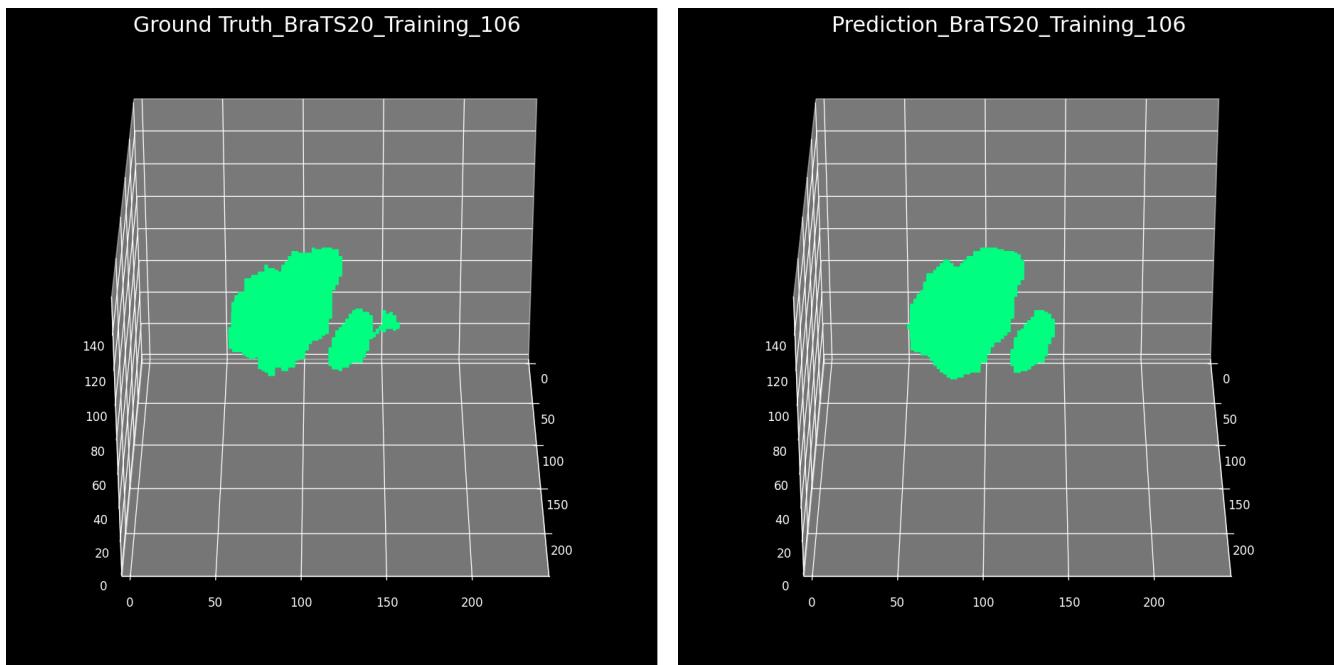


Figure 11 : The gif generated from the raw segments from module 1

The segment is transformed into a mesh file and the generated mesh is refined and cleaned in the Autodesk Meshmixer software (bottom left). Sample results from the “Mesh refinement” module is shown below. below (right)

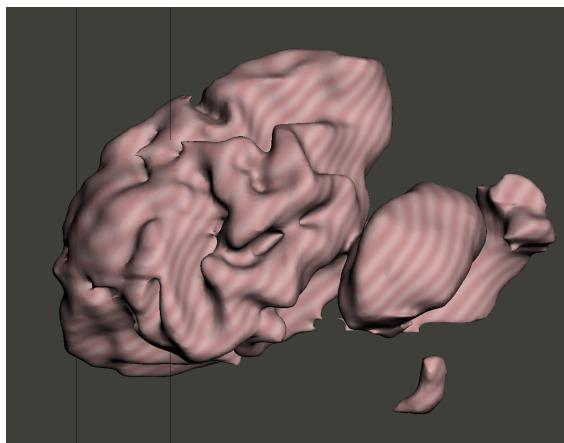


Figure 12 : The tumor segmented in mesh file format

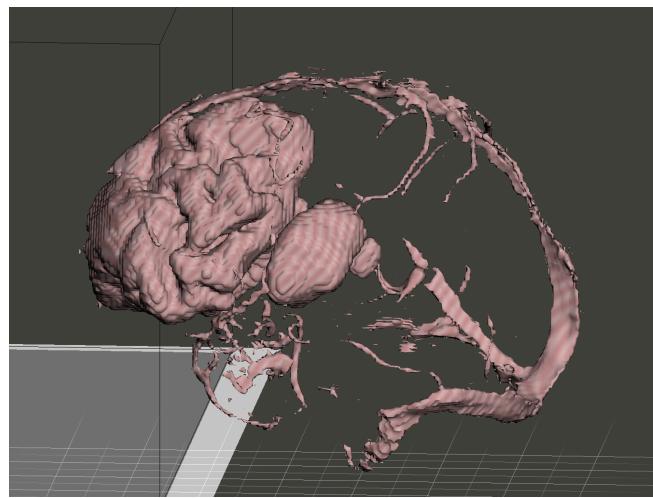


Figure 13 : The visualization of location of tumor in the brain

CHAPTER 5 - Result analysis

SECTION 5.1 - Analysis of computational complexity and convergence trends

As we propose a generic solution to image and volumetric segmentation, we believe it is important to study the computational complexity and convergence trends of the proposed network. In this section, we provide these details and compare the proposed method with other approaches in terms of number of parameters and rate of convergence.

Number of parameters

Seg-Net and U-Net have a 5 layer deep encoder and decoder. The number of filters in each block of these networks increase gradually as we go deeper in the network. For example, U-Net uses this sequence of filters for its 5 layers - 64, 128, 256, 512 and 1024. Although KiU-Net is a multi-branch network, we limit the complexity of our network by using fewer layers and filters. Specifically, we use a 3 layer deep network with 32, 64 and 128 respectively as the number of filters. The comparison is given below in [figure 10](#).

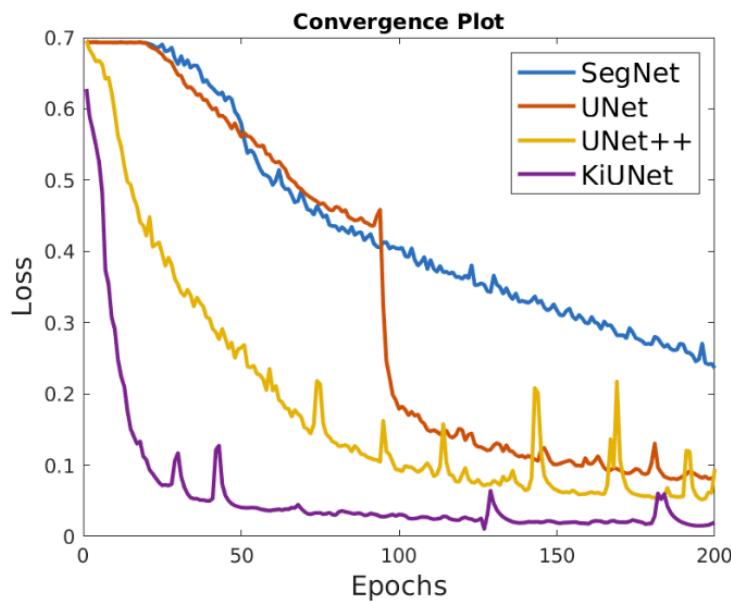


Figure 14 : The convergence speed of all the basic segmentation architectures along with proposed model

Convergence :

The convergence of loss function is an important characteristic associated with a network. A faster convergence means is always beneficial as it results in significantly lower training complexity.

Figure 14 below compares the convergence trends for Seg- Net, U- Net, and KiU-Net when trained on GLAS dataset. It can be observed that KiU-Net converges faster when compared to other networks.

Ablation study :

We conduct an ablation study to analyze the effectiveness of different blocks in the proposed method (KiU-Net). For these experiments, we use the brain anatomy segmentation US dataset. We start with the basic undercomplete (traditional) encoder-decoder convolutional architecture (UC) and overcomplete convolutional architecture (OC). These networks do not contain any skip connections (SK). Next, we add skip connections to the UC and OC baselines. These networks are basically U-Net (UC+skip connections) and Kite-Net (OC+skip connections). We then fuse both these networks by adding the feature map output of both the networks at the end. This is in fact KiU-Net without the CRFB block. Finally, we show the performance of our proposed architecture - KiU-Net.

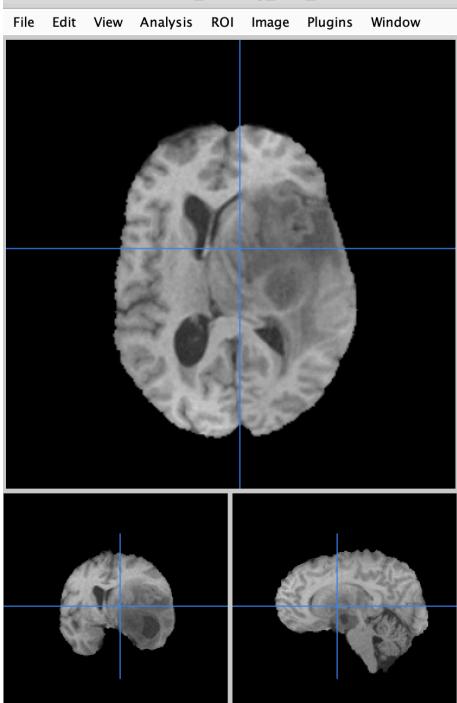
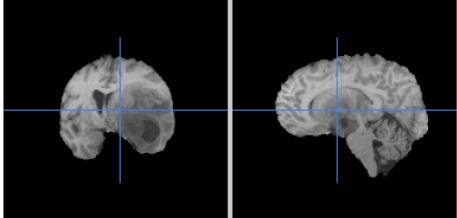
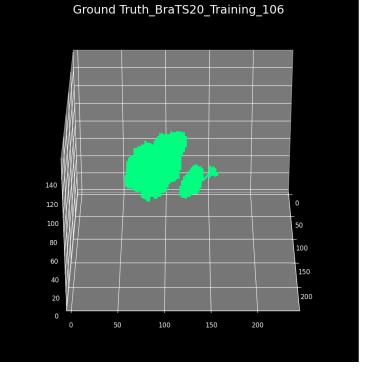
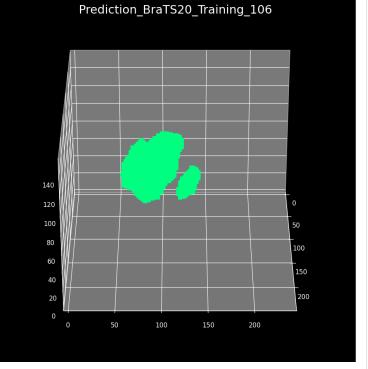
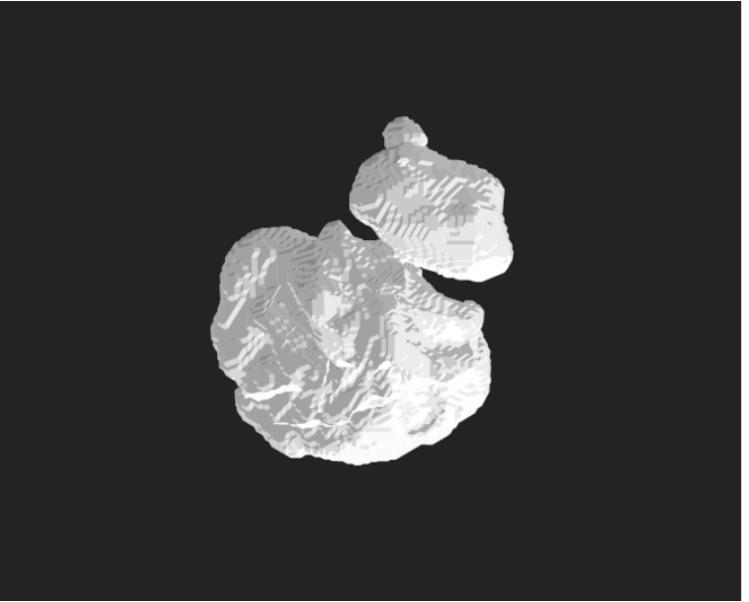
Network	Seg-Net [1]	U-Net [2]	U-Net++ [3]	KiU-Net
No. of Parameters	12.5M	3.1M	9.0M	0.29M

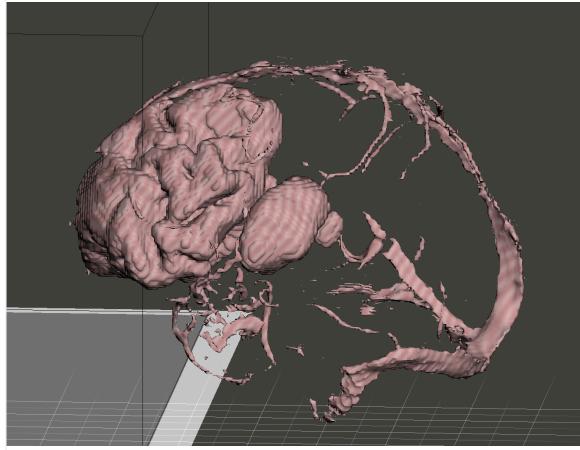
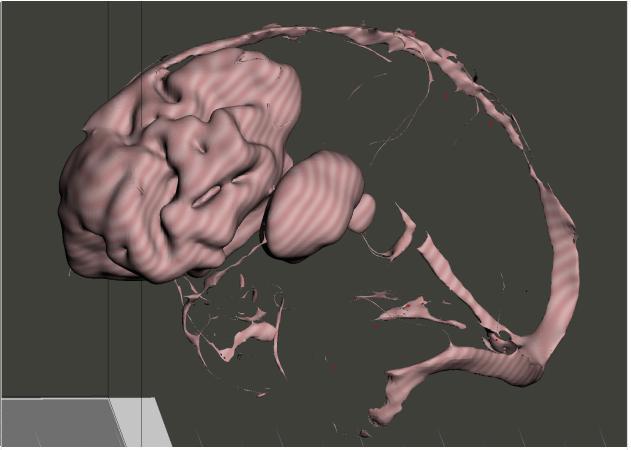
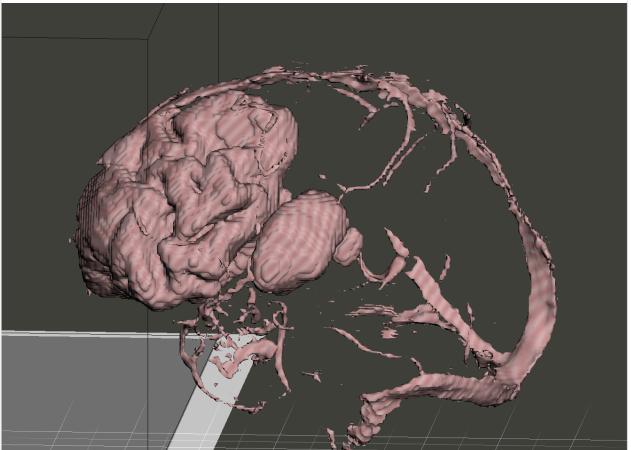
Figure 14 shows the results of all these ablation experiments. It can be observed that the performance improves with addition of each block to the network. The performances of OC and Kite-Net are lower because these predictions contain only the edges of the masks and do not contain any high-level information.

Metrics	UC	OC	UC+SK	OC+SK	UC+OC+SK	KiU-Net
Dice	0.82	0.56	0.85	0.60	0.86	0.89
Jaccard	0.75	0.43	0.79	0.47	0.78	0.83

Figure 15 : The ablation study conducted by varying the number of parameters

SECTION 5.2 - Tabulating the test use cases & their outputs

Test case	Input MRI Scan	Expected output vs Predicted output
Tumor segmentation	 	<p>Ground Truth_BraTS20_Training_106</p>  <p>Prediction_BraTS20_Training_106</p> 
Mesh Generation	<p>The torch tensor which is to be converted into 3D numpy array which is then converted into “stl” file I.e., mesh object.</p>	

Test case	Input (mesh in Autodesk)	output (mesh object refined)
Mesh Refinement	 A 3D rendering of a human brain mesh within a 3D coordinate system. The mesh is a coarse wireframe representation of the brain's internal structures.	 The same brain mesh as the input, but with significantly more detail and refinement added to the internal structures, making the brain appear more complex and accurate.
Locating the position of tumor	3D Numpy array converted from the tensor and the predicted segment & input file combined	 A 3D rendering of the brain mesh with a specific area highlighted in red, indicating the location of the tumor.

CHAPTER 6 - Conclusion

The task of Generating 3D models from the biomedical images via volumetric segmentation was split into three main modules out of which the first module ie., “The brain tumor segmentation” was the backbone. The Brain scans of all the 4 contrasts were stacked for accurate segmentation employing the over complete “KiU-Net 3D” which best utilizes the peculiar aspects of “Kite-Net 3D” and “U-Net 3D”. The main motive of the project was to generate 3D models from precisely segmented brain tumors for doctors to scrutinize and for the study purposes of anatomical students. The results have come out really well with decent DICE and JAC scores as given in the “Experimental results” section. Further, the model’s capability was displayed with the scores of segmenting all 3 classes of tumors which is quite impressive. This was then followed by Mesh generation and refinement of the generated meshes. Repairing, smoothing and making the mesh error-free were the prime motive of the module. It’s very crucial in transforming a simple mesh file into proper error-free and usable 3D mesh which could be imported in the 3D software environment for further use. As a component of innovation, we incorporated additional deliverable which is locating the tumor in the input brain which is coded patient-specific. The details of this innovation was well explained in the above sections and also visualized as a mesh file in Autodesk Meshmixer. This feature though helpful for doctors, it’s extremely good for promoting patient education. Finally, these models were incorporated into the Cura software and mesh files are transformed into 3D models.

References

- [1] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [2] 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*, pp. 234–241, Springer, 2015.
- [3] O. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, “3d u-net: learning dense volumetric segmentation from sparse annotation,” in *International conference on medical image computing and computer-assisted intervention*, pp. 424–432, Springer, 2016.
- [4] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in *2016 fourth international conference on 3D vision (3DV)*, pp. 565–571, IEEE, 2016.
- [5] Jeya Maria Jose Valanarasu, Vishwanath A. Sindagi, Ilker Hacihaliloglu , Vishal M. Patel, “KiU-Net: Overcomplete Convolutional Architectures for Biomedical Image and Volumetric Segmentation” in *2020 MICCAI BraTS challenge, MICCAI, 2020*
- [6] Trace AP, Ortiz D, Deal A, Retrouvey M, Elzie C, Goodmurphy C, “Radiology’s Emerging Role in 3- D Printing Applications in Health Care” *J Am Coll Radiol. Elsevier Inc*; 2016; 13: 856–862.e4.
- [7] Rengier F, Mehndiratta A, Von Tengg-Kobligk H, Zechmann CM, Unterhinninghofen R, Kauczor HU, “ 3D printing based on imaging data: Review of medical applications” *Int J Comput Assist Radiol Surg. 2010; 5: 335–341.*
- [8] Aidan J. Cloonan, Danial Shahmirzadi, Danial Shahmirzadi, Barry J. Doyle, Elisa E. Konofagou, Tim M. McGlooughlin, “3D-Printed Tissue-Mimicking Phantoms for Medical Imaging and Computational Validation Applications”
- [9] Thore M. Bücking, Emma R. Hill, James L. Robertson, Efthymios Maneas, Andrew A. Plumb, Daniil I. Nikitiche, “From medical imaging data to 3D printed anatomical models”
- [10] Autodesk Meshmixer [Internet]. [cited 1 Jan 2016]. Available: <http://www.meshmixer.com/>