

Fetal Brain Component Segmentation Using 2-Way Ensemble U-Net



Shinjini Halder, Tuhinangshu Gangopadhyay, Paramik Dasgupta, Kingshuk Chatterjee, Debayan Ganguly, Surjadeep Sarkar, and Sudipta Roy

Abstract Fetal brain segmentation has been a field of interest since a long time. However, it is a challenging task as well for reasons, like blurred images due to fetal motion. Recently, deep learning has been successful in performing this task with good accuracy. In this paper, we developed 2-way Ensemble U-Net model, a convolutional neural network architecture for performing segmentation on the fetal brain image to divide it into its seven components: intracranial space and extra-axial cerebrospinal fluid spaces, gray matter, white matter, ventricles, cerebellum, deep gray matter, and brainstem and spinal cord. The fetal brain image can be obtained by segmenting it from the fetal magnetic resonance images using any of the previous works on fetal brain segmentation, which presents our work as an extension of the already existing segmentation works. The Jaccard similarity and Dice score for this task are 83% and 88%, respectively. This is higher than that returned by any of the previous models, when trained for the same task, thus showing the potential of our model in segmentation related tasks.

Keywords Convolutional neural networks · Deep learning · Ensemble model · Fetal brain segmentation · Medical image processing

S. Halder · T. Gangopadhyay · D. Ganguly · S. Sarkar
Government College of Engineering and Leather Technology, Kolkata 700106, India
e-mail: debayan@gcelt.org

S. Sarkar
e-mail: surjadeep@gcelt.org

P. Dasgupta
Asian Institute of Technology, Khlong Nueng, Thailand

K. Chatterjee
Government College of Engineering and Ceramic Technology, Kolkata 700010, India
e-mail: kingshukchatterjee@gcet.ac.in

S. Roy (✉)
Artificial Intelligence and Data Science, Jio Institute, Navi Mumbai 410206, India
e-mail: sudipta1.roy@jioinstitute.edu.in

1 Introduction

Ultrasonography (USG) as the name suggests uses high-intensity sound waves to produce images inside the body. It has been the primary method for prenatal diagnosis for a long time now. But USG produces 2D images, so we lose some information. To get more explicit images, we use magnetic resonance imaging (MRI). MRI is a non-invasive medical imaging technique that uses strong magnetic fields and radio waves. It provides a larger view and a high range of echogenicity according to medium contrast and 3D reconstruction hence producing detailed images of the organs. Artificial intelligence has been quite a stir after its foundation in the twentieth century [1]. Artificial intelligence can be defined as a computer's way of mimicking human behaviors. Advancements in AI have caused a worldwide revolution in industries, health care, and academia. After the introduction of artificial intelligence in health care [2], it has learned to predict medical data statistics, diagnose various diseases like breast cancer [3], assist in human-machine interactions, etc.

Machine learning and deep learning, developed as a subsection of artificial intelligence, have the ability to learn from instances. These methods have been incorporated in the medical field for performing several operations. One of the most common use cases of AI in the medical field is image segmentation, which is involved in classifying image pixels in various objects and boundaries. This process reduces complexity in images to simply further operations. However, the lack of adequate amounts of data and additional noises due to fetal movements or arbitrary orientation of fetuses make it challenging to process fetal MRI scans. In image recognition, deep learning models, predominantly the convolutional neural network (CNN), can discover intricate structures in image datasets and are proven to be producing near-accurate results in computer vision [4, 5].

In this paper, we have proposed 2-way Ensemble U-Net, a CNN architecture, to perform the task of segmentation. This model specializes in segmenting the fetal brain into its 7 major components, which are: intracranial space and extra-axial cerebrospinal fluid spaces (CSF), gray matter (GM), white matter (WM), ventricles (LV), cerebellum (CBM), deep gray matter (SGM), and brainstem and spinal cord (BS). This model is inspired by the U-Net architecture [6]. The motivation behind our work is:

1. Motivation behind using U-Net as the base model is its ability to efficiently encode the necessary spatial and orientation information, which can then be decoded to return an accurate segmentation. U-Net, along with its modifications like Deep U-Net [7], has already been proven to be successful in many medical segmentation use cases.
2. Previous works on fetal MRI using machine learning and deep learning include works like segmenting the fetal brain from the fetal MRI [8], noise removal from MRI to return a clean image [9], etc. Thus, our work can be seen as an extension of the existing works on segmenting the fetal MRI, in the sense that our model is using this output as its input to perform its operation.

Our work made the following contributions:

1. We have proposed a 2-way Ensemble U-Net model for the purpose of fetal brain segmentation. It is an ensemble of two modified versions of the U-Net architecture.
2. We have moved one step forward in using the segmented fetal brain as input to segment it into its constituent parts, to help focus on any of the parts at a time, which can help in deeply analyzing it.
3. We have compared our model with five previously proposed brain segmentation architectures: FCN [10], RFBSNet [8], SegNet [11], U-Net [6], and Deep U-Net [7] and showed that our model gives the best performance.

2 Related Works

Over a timeline, new methods are being developed, advanced, or rejected based on their performances. In documentation by Sahiner et al. [12] about the history of medical image analysis, they observed the progress and popularity of these methods from time to time. In a summary, they noted the resurgence of ANNs in deep learning in 2010 after it was replaced by SVMs in the late 1990s, the evolution of feature extraction techniques [12]. In health care, recent advances in AI can better interpret, identify, classify, and discover patterns in images [13].

One of the advantages of automation is it can significantly reduce user biasness; hence, it has become a preferred method of use. A conjunction between fast k-means, morphology, and level set is utilized by S. Roy [14] to localize and segment tumors in MRI data. The issue of limited data and the option of an expert rater to pixel-wise label a training set is often not feasible and was resolved by weak forms of annotations [15]. Rajchl et al. [10] employed crowd annotations on T2-weighted MRI thereafter feeding it to a fully connected CNN. For MRI segmentation, Bijen Khagi et al. [11] used deep neural networks in heterogeneously distributed pixels and assigned a label to each pixel. This supervised learning method is extortionate because labeling is expensive and uses fewer training images. The fetal brain images are often distorted by noises and need reconstruction of MR images. Kuklisova-Murgasova et al. made an attempt to reconstruct volumetric fetal MRI from 2D slices incorporated with motion correction. This method applied on motion-corrupted MRI of a preterm neonate produces high-quality reconstruction results irrespective of their thickness, motion corruption, and intensity artifacts of MRIs [9].

Ramesh et al. in their paper have assembled various methods developed for image segmentation. These methods can be broadly classified into the thresholding approach, region-based methods, clustering approach, edge detection, and model-based algorithms [16]. We have mainly concentrated on U-Net developed by Olaf Ronneberger, Philipp Fischer, and Thomas Brox [6] which is a fast and automatic segmentation technique that uses CNN and relies on data augmentation. This model comprises two steps: first, a contracting path that is principally a convolutional

network, followed by an expansive path, which is an effective segmentation model. U-net and its variations like U-Net++ [17], residual U-Net, residual recurrent U-Net, attention U-Net, and attention residual recurrent U-Net [18], etc. are extensively used architectures because of their efficiency, real-time motion detection, tracking, and also for 3D reconstruction of MRI [19]. For 2D feature extraction from fetal brain MRI, Andrik Rampun [7] used a modified U-Net architecture which used 7 convolution blocks, a combination of balanced cross-entropy in conjunction with Dice coefficient loss function, and exponential linear unit and RMSprop in contrast to the original U-Net that uses 4 convolutional blocks, a combination of the pixel-wise softmax with cross-entropy loss function, and ‘Relu’ and ‘Adam’ for activation and optimizer functions. The modified U-Net architecture reduces spatial information loss and is proven to be appropriate under several circumstances. Seyed Sadegh Mohseni Salehi et al. [20] developed another variation of U-Net that segments 2D fetal MRI slices in real-time based on 2D U-Net and auto-context, and when compared to a voxel-wise fully convolutional network and a method based on SIFT features, random forest, and conditional random, this model outperformed the other methods. Razieh Faghihpirayesh et al. introduced RFBSNet [8], a small CNN that combines spatial details at high resolution with context features extracted at lower resolutions.

3 Materials and Methods

The first step of our work involves applying some preprocessing steps to the input, making it suitable for our model to work on. Then, several modified versions of the U-Net model, along with the original one, are experimented with for the segmentation purpose, outputs are checked and compared, some explanatory analyses are performed, and the best model is selected. We have used Google Colaboratory for developing and testing our work.

3.1 Data Availability

FeTA 2.1 Dataset (Fetal Tissue Annotation Dataset) has been used to train and test our model [21]. This dataset is owned by the University Children’s Hospital Zurich. It contains 3D fetal brain volumes of 80 fetuses of gestational age ranging from 20 to 35 weeks. The shape of each volume is $(256 \times 256 \times 256)$. Each volume is accompanied by its segmented counterpart.

3.2 Data Preprocessing

The essential preprocessing steps we used are discussed below:

1. Each volume is divided into 2D images, such that it contains images along all the axes (sagittal, coronal, and axial) from all the volumes. A total of $(80 * 3 * 256) = 61,440$ 2D images can be returned by all 80 volumes. However, for speeding up the training process and to prevent the model from overfitting, around 10,450 images were included in the final dataset (9900 for training and validation, and 550 for testing), which is exclusive of any ‘completely black’ images, i.e., the images full of zeroes, containing no brain part. Also, the 10,450 images are not chosen randomly, rather each volume (along each axis) is divided into a certain number of equally spaced parts, and one slice is chosen from each part, after eliminating the images/slices completely filled with zeroes. This will ensure that slices from each portion of the brain, along all the axes, are included in the dataset.
2. Some of the images are flipped vertically, horizontally, or in both ways to introduce some rotational and translational independence at the time of training the model.
3. Finally, slightly normal, horizontal, and vertical blurring is applied to some of the images to replicate the possibility that the fetal MRI may get slightly blurred or stretched out due to the movement of the fetus at the time of the MRI scan. This will enable the model to identify some motion-correcting features at the time of training. However, this blurring effect is not applied to the images in the dataset which were already blurred (i.e., the blurred volumes in the dataset).

The preprocessing steps just discussed are all applied to both the input images and the output segmented images (except for the blurring effect, which is applied only to the input image). Thus, the train and test set contains the following type of images: normal brain images, blurred images, flipped images, and images of different sizes (depending on the gestational age of the fetus at the time of scan). Lastly, the input of the models is the preprocessed images, and the output for each image is the corresponding segmented image.

3.3 Network Architecture

An U-Net architecture [6] consists of two main parts, namely the contracting path (encoder) and the expanding path (decoder). The contracting path learns to down-sample/encode the necessary information from the image given as input, and the decoder path learns to upsample/decode the encoded information to return the required segmentation. All of these operations are performed using a set of convolutional layers and pooling layers in a stepwise manner. Several architectures were implemented as a part of this work, of which the selected one is an ensemble model. This model has the capabilities of two of our best-performing models, rendering the

best performance, and at the same time, keeping the number of parameters much less than some of the state-of-the-art models. Each of our models has the same following specifications:

1. The model consists of several ‘encoder blocks’ and ‘decoder blocks.’ Each encoder block has two convolutional layers and one max-pool layer. Each decoder block has a deconvolutional layer for upsampling from the previous layer, a concatenation layer, and followed by two convolutional layers. The encoder blocks form the encoding path, and the decoder blocks form the decoding path.
2. These blocks are arranged in a stepwise manner, as in the original U-Net model. The output from the second convolutional layer in the encoder block in each layer is concatenated with the output of the deconvolutional layer in the corresponding decoder block by the concatenation layer, either directly (as in the original U-Net) or via a convolutional layer. This modification is used for the model to store some necessary connection information in that layer (especially spatial or orientational information), which might not be captured by the lowest layer due to the bottleneck or excessive compression; thus, the performance is expected to improve. The lowest layer consists of two convolutional layers, which take inputs from the lowest encoder block and the output goes to the lowest decoder block.
3. The number of filters in the convolutional layers increases while going ‘down’ in the encoding path, with the image size also reducing continuously after each layer, whereas the number of filters decreases while going ‘up’ in the decoding step, with the image slowly returning to its original shape. The convolutional layers in the lowest layer have the maximum number of filters. However, the number of filters used in our models is much less than that in the original U-Net model. The number of filters in any of our models starts with as low as 16 filters in the first layer, up to a maximum of 300 filters in the lowest layer (whereas that in the original U-Net starts with 64 filters in the first layer up to 1024 filters in the lowest layer.)
4. The input image shape is 256×256 , and the output given by the model is of the shape $256 \times 256 \times 8$, since for each pixel, the model is performing a classification task, and there are 8 classes in total: the 7 components segmented by the model and the background (for the pixels where there is no brain part present). The class with the highest probability for each pixel is chosen as the output for that pixel.
5. As our task can be interpreted as a classification task with 8 classes, we have used sparse categorical cross-entropy as the loss function and accuracy as the metrics while training the model. We have used Adam’s optimizer because they are faster, require less number of parameters, and are quite efficient. We have used 30 epochs to train the model. We have also used the callback ‘early stopping’ with patience = 3, to prevent the model from overfitting.

Naming pattern. The name of a model reflects its architecture. The first part of the model’s name is based on the number of encoder–decoder combination layers used in the model, which excludes the lowest layer since the two convolutional layers in the lowest layer serve as a bottleneck for the model. The second part of the name is

dependent on whether a convolutional layer is present in between the encoder and decoder block connection. Thus, a name like ‘3 layer mod’ means the model contains 3 encoder–decoder combination layers, with a convolutional layer in between each encoder and decoder blocks and finally the lowest layer; and a named ‘4 layer no-mod’ means there are 4 encoder–decoder combination layers, with no convolutional layer in between the blocks, followed by the lowest layer.

We have trained five models namely ‘3 layer no-mod,’ ‘4 layer no-mod,’ ‘5 layer no-mod,’ ‘3 layer mod,’ and ‘4 layer mod,’ respectively. Training multiple models with similar configurations, except for the number of layers, is a part of our **ablation study**, where the aim is to examine the effects of increasing the number of layers, filters, and encoder–decoder blocks in the model on the quality of the segmentation output.

Table 1 shows that the best-performing models are: 4 layer mod, followed by 4 layer no-mod and 3 layer mod. Therefore, two ensemble models have been designed from these models to combine their predictions, using the concept of transfer learning. The first one, named ‘3 mod 4 no-mod ensemble,’ is the ensemble of 3 layer mod and 4 layer no-mod, and the second one, named ‘2-way Ensemble U-Net’, which is our selected model, is the ensemble of 4 layer no-mod and 4 layer mod (as per the naming pattern, it should be named ‘4 mod 4 no-mod ensemble’). For training these ensemble models, their corresponding pre-trained models have been used, followed by the removal of their last layers, concatenation of their outputs, and then the addition of some supplementary layers at the end, which will learn to combine the results from these models.

Figure 1 shows the architecture of the 2-way Ensemble U-Net. Since it is dependent on the models 4 layer mod and 4 layers no-mod, their architectures are also represented in the same figure. It also shows the architecture of an encoder block and a decoder block, along with all the colors representing the layers and the blocks.

4 Evaluation Metrics

The following evaluation metrics have been used to evaluate the segmentation results: precision (P), sensitivity (S), Jaccard similarity (J), Dice score (D), and accuracy (A), in a way defined in [14] and [7]. Precision and sensitivity give the number of correct predictions with respect to all the predictions for a particular class. Jaccard similarity and Dice score tell us about the degree of overlapping of the prediction with the ground truth segmentation for each class. These four metrics can be individually calculated for each class in every image (using the true positives, false positives, and false negatives for each class). However, accuracy is the total number of correctly predicted labels (pixels) in an image, irrespective of the class, divided by the total number of pixels in the image; i.e., accuracy is calculated only for the overall image.

However, in some evaluation processes, we have also used the average precision, sensitivity, Jaccard similarity, and Dice score, which are the average of those values over all the classes for any particular image, as shown below:

Table 1 Each cell showing four evaluation metrics (precision, sensitivity, Jaccard similarity, Dice score) for each model and for each brain part for the whole test set

	CSF	GM	WM	LV	CBM	SGM	BS
FCN [10]	00.81 ± 0.35	00.81 ± 0.35	00.85 ± 0.22	00.87 ± 0.25	00.92 ± 0.22	00.83 ± 0.31	00.89 ± 0.26
	00.78 ± 0.27	00.75 ± 0.21	00.90 ± 0.14	00.87 ± 0.22	00.89 ± 0.26	00.90 ± 0.24	00.83 ± 0.33
	00.64 ± 0.31	00.59 ± 0.28	00.79 ± 0.24	00.79 ± 0.29	00.83 ± 0.32	00.77 ± 0.35	00.77 ± 0.37
	00.72 ± 0.31	00.70 ± 0.25	00.85 ± 0.23	00.85 ± 0.27	00.86 ± 0.30	00.80 ± 0.33	00.80 ± 0.35
RFBSNet [8]	00.78 ± 0.27	00.77 ± 0.22	00.90 ± 0.16	00.87 ± 0.22	00.96 ± 0.13	00.90 ± 0.23	00.89 ± 0.25
	00.80 ± 0.26	00.77 ± 0.20	00.90 ± 0.16	00.92 ± 0.17	00.92 ± 0.21	00.92 ± 0.20	00.89 ± 0.26
	00.67 ± 0.30	00.64 ± 0.26	00.82 ± 0.21	00.81 ± 0.26	00.90 ± 0.24	00.84 ± 0.28	00.82 ± 0.32
	00.75 ± 0.30	00.75 ± 0.22	00.88 ± 0.19	00.87 ± 0.23	00.92 ± 0.22	00.88 ± 0.25	00.84 ± 0.31
SegNet [11]	00.50 ± 0.34	00.51 ± 0.32	00.63 ± 0.38	00.57 ± 0.41	00.32 ± 0.43	00.60 ± 0.45	00.55 ± 0.46
	00.77 ± 0.24	00.67 ± 0.27	00.85 ± 0.18	00.75 ± 0.32	00.85 ± 0.30	00.74 ± 0.39	00.76 ± 0.38
	00.43 ± 0.30	00.37 ± 0.28	00.55 ± 0.34	00.45 ± 0.40	00.27 ± 0.40	00.46 ± 0.46	00.43 ± 0.46
	00.53 ± 0.35	00.48 ± 0.31	00.63 ± 0.37	00.51 ± 0.40	00.30 ± 0.41	00.49 ± 0.45	00.46 ± 0.46
Deep U-Net [7]	00.81 ± 0.25	00.78 ± 0.21	00.88 ± 0.17	00.89 ± 0.22	00.87 ± 0.29	00.80 ± 0.32	00.82 ± 0.34
	00.74 ± 0.30	00.72 ± 0.24	00.89 ± 0.17	00.86 ± 0.22	00.93 ± 0.22	00.94 ± 0.18	00.87 ± 0.30
	00.65 ± 0.30	00.62 ± 0.27	00.81 ± 0.22	00.79 ± 0.28	00.81 ± 0.34	00.77 ± 0.34	00.74 ± 0.39
	00.74 ± 0.31	00.73 ± 0.23	00.87 ± 0.19	00.84 ± 0.25	00.84 ± 0.32	00.81 ± 0.32	00.77 ± 0.37
U-Net [6]	00.82 ± 0.23	00.78 ± 0.21	00.89 ± 0.17	00.90 ± 0.17	00.88 ± 0.27	00.90 ± 0.23	00.91 ± 0.23
	00.77 ± 0.29	00.77 ± 0.20	00.91 ± 0.14	00.90 ± 0.18	00.96 ± 0.15	00.92 ± 0.19	00.89 ± 0.26
	00.68 ± 0.30	00.65 ± 0.25	00.82 ± 0.21	00.83 ± 0.23	00.85 ± 0.29	00.84 ± 0.28	00.83 ± 0.32
	00.75 ± 0.30	00.76 ± 0.21	00.88 ± 0.19	00.88 ± 0.20	00.88 ± 0.28	00.87 ± 0.26	00.85 ± 0.30
3 layer no-mod	00.76 ± 0.29	00.77 ± 0.21	00.89 ± 0.19	00.84 ± 0.26	00.89 ± 0.27	00.85 ± 0.29	00.88 ± 0.28
	00.82 ± 0.24	00.75 ± 0.21	00.89 ± 0.15	00.93 ± 0.15	00.93 ± 0.20	00.92 ± 0.19	00.87 ± 0.28
	00.67 ± 0.29	00.63 ± 0.26	00.81 ± 0.22	00.79 ± 0.28	00.84 ± 0.32	00.79 ± 0.33	00.79 ± 0.35
	00.75 ± 0.30	00.74 ± 0.23	00.87 ± 0.20	00.85 ± 0.25	00.86 ± 0.31	00.83 ± 0.31	00.81 ± 0.34

(continued)

Table 1 (continued)

	CSF	GM	WM	LV	CBM	SGM	BS
4 layer no-mod	00.82 ± 0.22	00.79 ± 0.20	00.88 ± 0.17	00.91 ± 0.17	00.95 ± 0.17	00.91 ± 0.22	00.89 ± 0.24
	00.80 ± 0.27	00.74 ± 0.21	00.93 ± 0.13	00.88 ± 0.20	00.94 ± 0.19	00.93 ± 0.18	00.90 ± 0.25
	00.69 ± 0.28	00.64 ± 0.26	00.83 ± 0.20	00.82 ± 0.24	00.89 ± 0.25	00.85 ± 0.26	00.83 ± 0.31
	00.77 ± 0.28	00.75 ± 0.21	00.89 ± 0.18	00.88 ± 0.21	00.91 ± 0.23	00.89 ± 0.24	00.86 ± 0.29
5 layer no-mod	00.81 ± 0.24	00.79 ± 0.21	00.86 ± 0.20	00.92 ± 0.18	00.92 ± 0.21	00.91 ± 0.22	00.90 ± 0.24
	00.77 ± 0.28	00.71 ± 0.23	00.93 ± 0.14	00.86 ± 0.21	00.94 ± 0.18	00.91 ± 0.21	00.89 ± 0.26
	00.67 ± 0.29	00.62 ± 0.27	00.81 ± 0.22	00.80 ± 0.26	00.87 ± 0.27	00.83 ± 0.29	00.82 ± 0.32
	00.76 ± 0.29	00.73 ± 0.23	00.87 ± 0.21	00.86 ± 0.23	00.90 ± 0.25	00.87 ± 0.26	00.85 ± 0.30
3 layer mod	00.80 ± 0.27	00.76 ± 0.22	00.90 ± 0.17	00.89 ± 0.21	00.94 ± 0.19	00.85 ± 0.29	00.86 ± 0.29
	00.77 ± 0.27	00.79 ± 0.18	00.90 ± 0.15	00.90 ± 0.18	00.94 ± 0.19	00.93 ± 0.18	00.90 ± 0.26
	00.66 ± 0.30	00.66 ± 0.26	00.83 ± 0.21	00.81 ± 0.26	00.89 ± 0.25	00.81 ± 0.31	00.80 ± 0.34
	00.74 ± 0.30	00.75 ± 0.22	00.89 ± 0.19	00.87 ± 0.23	00.92 ± 0.23	00.85 ± 0.29	00.83 ± 0.32
4 layer mod	00.84 ± 0.22	00.77 ± 0.20	00.90 ± 0.16	00.90 ± 0.19	00.95 ± 0.17	00.91 ± 0.22	00.94 ± 0.18
	00.78 ± 0.22	00.79 ± 0.18	00.91 ± 0.14	00.90 ± 0.19	00.94 ± 0.19	00.92 ± 0.20	00.88 ± 0.27
	00.68 ± 0.29	00.66 ± 0.25	00.83 ± 0.20	00.83 ± 0.24	00.90 ± 0.24	00.85 ± 0.28	00.84 ± 0.30
	00.76 ± 0.29	00.76 ± 0.21	00.89 ± 0.18	00.88 ± 0.21	00.92 ± 0.22	00.88 ± 0.25	00.87 ± 0.29

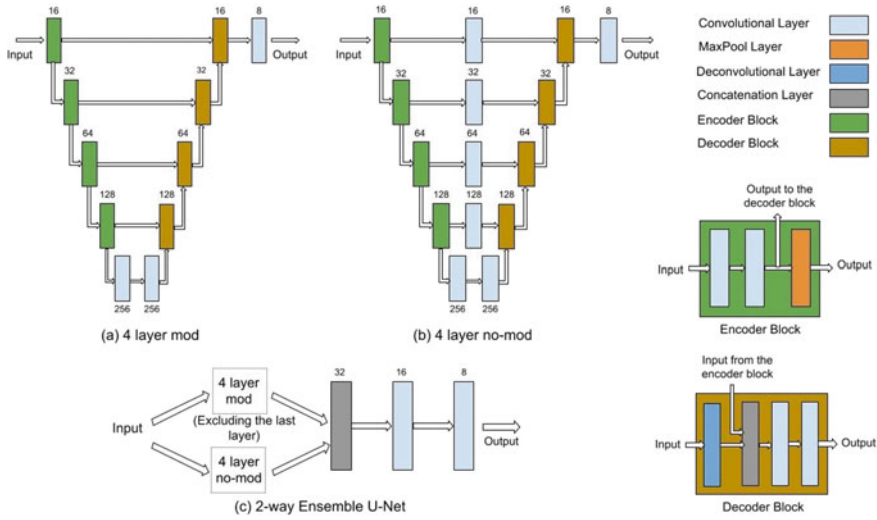


Fig. 1 Architecture of **a** 4 layer mod, **b** 4 layer no-mod, and **c** 2-way Ensemble U-Net. Content of encoder and decoder block is shown in the bottom right corner

$$\text{Average Precision (AP)} = \frac{\sum_{\text{All classes}} P}{\text{Total number of classes}} \quad (1)$$

$$\text{Average Sensitivity (AS)} = \frac{\sum_{\text{All classes}} S}{\text{Total number of classes}} \quad (2)$$

$$\text{Average Jaccard Similarity (AJ)} = \frac{\sum_{\text{All classes}} J}{\text{Total number of classes}} \quad (3)$$

$$\text{Average Dice Score (AD)} = \frac{\sum_{\text{All classes}} D}{\text{Total number of classes}} \quad (4)$$

5 Results

All the models have been inferenced on the test set of 550 images. The performances of our models have also been compared with some of the previously proposed state-of-the-art segmentation architectures: FCN [10], RFBSNet [8], SegNet [11], U-Net [6], and Deep U-Net [7]. Table 1 shows the performance of each model on the following metrics (and in the following order): precision, sensitivity, Jaccard similarity, and Dice score for segmentation of each of the seven brain components. They are expressed in the form: Mean \pm SD, where ‘mean’ and ‘SD,’ respectively, represent the mean and the standard deviation of the performance of a model (indicated

in the row) on all the images in the test set for the segmentation of a particular brain component (indicated in the column). Each cell represents the four metric values for the evaluation of the model and the brain part under consideration. For each brain component and for each metric, the model giving the best result is written in bold.

It can be seen from the above table that the models with the best performance are: 4 layer mod, followed by 4 layer no-mod and 3 layer mod. Moreover, many of the metric values of the 4 layer mod model exceeds the corresponding values of other models by at least 2–3 percent and as much as 6–7 percent. Among the previously proposed architectures, the best results are given by the RFBSNet and U-Net models.

The performance of the ensemble models, created from the three best-performing models on the test set, is given in Table 2, in a format same as that in Table 1. For each brain component and for each metric, the model giving the highest value is written in bold.

It can be seen that the best result is returned by the 2-way Ensemble U-Net model among the two models, even though the metric values for both the models are quite similar to each other. Upon comparing with the values in Table 1, it can be seen that this ensemble model's performance exceeded that of the 4 layer mod model by around 2–3 percent for most of the metric values.

6 Discussion

The detailed performance of each model and for each brain part are shown in Tables 1 and 2 with the help of multiple metrics. This gives us 28 metric values for each model. In Table 3, the average performance of each model for the segmentation of every brain part on all the images in the test set is shown. For each model, there are only 5 metric values, thus facilitating the model comparison task. It is also expressed in the form: Mean \pm SD, where 'mean' and 'SD' are mean and standard deviation of the performance of the model (mentioned in the row) evaluated on the metric (mentioned in the column) for all images in the test set. For each metric, the model giving the highest value is written in bold.

It can be seen that both the ensemble models gave similar performances, having the highest values for all metrics, exceeding the other model values by around 2–3%. They returned an average Jaccard similarity (AJ) and average Dice score (AD) of 83% and 88%, respectively. These values show the average overlapping of the predicted segmentation with the actual segmentation for all the classes. The underlined values in the above table show the best-performing models for those metrics, which have no bold values among the non-ensemble models. It can be seen that the best-performing non-ensemble models are 4 layer mod, followed by 4 layer no-mod, both having AJ of 82% and AD of 87%. Please note that the first paragraph of a section or subsection is not indented.

Thus, the evaluation metrics and the ablation study show that adding a convolutional layer in between the encoding and the decoding blocks is beneficial. They perform better than the direct connection between the blocks, as proposed in the

Table 2 Each cell showing four evaluation metrics for each ensemble model and for each brain part for the whole test set

	CSF	GM	WM	LV	CBM	SGM	BS
3_mod 4_no-mod ensemble	00.84 ± 0.21	00.80 ± 0.18	00.90 ± 0.15	00.92 ± 0.16	00.95 ± 0.14	00.91 ± 0.21	00.92 ± 0.19
	00.79 ± 0.28	00.77 ± 0.20	00.92 ± 0.13	00.89 ± 0.19	00.95 ± 0.17	00.93 ± 0.19	00.90 ± 0.25
	00.70 ± 0.28	00.67 ± 0.24	00.84 ± 0.19	00.84 ± 0.23	00.91 ± 0.21	00.84 ± 0.27	00.85 ± 0.29
	00.78 ± 0.28	00.77 ± 0.20	00.89 ± 0.17	00.89 ± 0.20	00.93 ± 0.19	00.89 ± 0.24	00.88 ± 0.27
2-way ensemble U-Net	00.83 ± 0.20	00.80 ± 0.19	00.91 ± 0.14	00.91 ± 0.16	00.96 ± 0.14	00.93 ± 0.19	00.93 ± 0.18
	00.80 ± 0.27	00.78 ± 0.18	00.91 ± 0.14	00.90 ± 0.19	00.94 ± 0.17	00.91 ± 0.35	00.89 ± 0.26
	00.70 ± 0.28	00.67 ± 0.24	00.84 ± 0.19	00.84 ± 0.23	00.91 ± 0.22	00.85 ± 0.27	00.85 ± 0.29
	00.78 ± 0.27	00.78 ± 0.19	00.89 ± 0.17	00.89 ± 0.20	00.93 ± 0.20	00.89 ± 0.25	00.87 ± 0.28

Table 3 Evaluation of each model with respect to five evaluation metrics

	AP	AS	AJ	AD	A
FCN [10]	0.86 ± 0.12	0.86 ± 0.12	0.77 ± 0.16	0.82 ± 0.14	0.97 ± 0.02
RFBSNet [8]	0.88 ± 0.10	0.89 ± 0.10	0.81 ± 0.14	0.86 ± 0.12	0.98 ± 0.02
SegNet [11]	0.59 ± 0.16	0.80 ± 0.16	0.49 ± 0.17	0.55 ± 0.16	0.95 ± 0.04
Deep U-Net [7]	0.86 ± 0.13	0.87 ± 0.11	0.78 ± 0.16	0.83 ± 0.14	0.97 ± 0.02
U-Net [6]	0.88 ± 0.11	0.89 ± 0.09	0.81 ± 0.14	0.86 ± 0.12	0.98 ± 0.02
3 layer no-mod	0.86 ± 0.13	0.89 ± 0.10	0.79 ± 0.16	0.84 ± 0.14	0.98 ± 0.02
4 layer no-mod	0.89 ± 0.10	0.89 ± 0.10	0.82 ± 0.14	0.87 ± 0.11	0.98 ± 0.02
5 layer no-mod	0.89 ± 0.11	0.88 ± 0.10	0.81 ± 0.15	0.86 ± 0.12	0.98 ± 0.02
3 layer mod	0.87 ± 0.11	0.89 ± 0.09	0.80 ± 0.14	0.85 ± 0.12	0.98 ± 0.02
4 layer mod	0.90 ± 0.09	0.89 ± 0.10	0.82 ± 0.13	0.87 ± 0.11	0.98 ± 0.02
3_mod 4_no-mod ensemble	0.91 ± 0.09	0.89 ± 0.10	0.82 ± 0.13	0.87 ± 0.11	0.98 ± 0.02
2-way ensemble U-Net	0.91 ± 0.09	0.89 ± 0.10	0.83 ± 0.13	0.88 ± 0.11	0.98 ± 0.02

original U-Net paper [6]. It can also be seen that increasing the number of layers does not necessarily improve the performance of the model as in the case of the 5 layer no-mod model.

The FeTA 2.1 Dataset was made available as a part of a challenge conducted in 2021, which in turn was organized as a part of the Medical Image Computing and Computer Assisted Intervention (MICCAI) 2021 conference. The best model (proposed by the team ‘NVAUTO’) had an average Dice score of 0.786 ± 0.161 [22]. It used an encoder–decoder framework (as in [23]), with the encoder using ResNet blocks and the decoder using transposed convolution, followed by the addition of the output from the encoder in the corresponding level. An impressive 10% performance improvement (w.r.t. Dice score) has been achieved by our model!

After checking the results on the test set, Fig. 2 below shows our best model tested on 4 more images of varying types to see the performance of the model on these images. The images are arranged in a row-wise manner, with the actual and predicted segmentation and the evaluation metric values shown beside them.

The results given by the 2-way Ensemble model are quite satisfactory, especially for the fourth image, since it is blurred to some extent, yet the model has been quite accurate with its segmentation. But its performance degraded on the second image, which is also blurred. However, upon observing the segmentation output, it can be seen that the model has tried to smooth out the blurred portion, which is an effect of the motion correction we introduced in the model, whereas the actual segmentation has tried to maintain the blurred/stretched out effect of the input image. This difference can be seen only in those images of the dataset which were blurred by default. This difference might account for the reduction in the metric values to some extent.

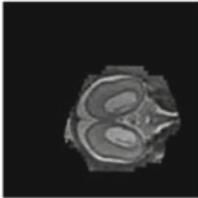
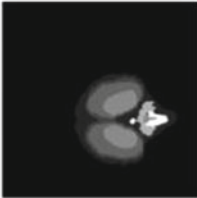
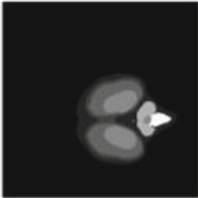
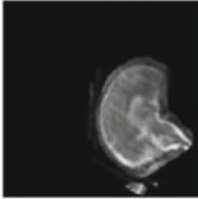
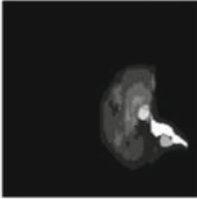
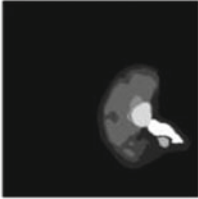
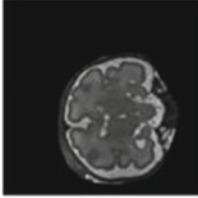
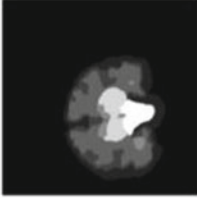
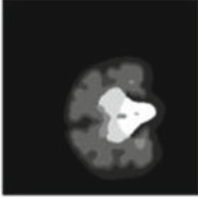
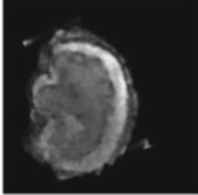
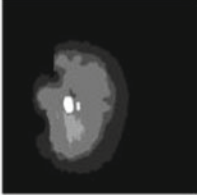
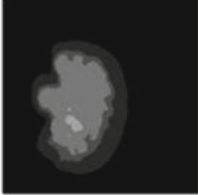
Fetal Brain Image	Actual Segmentation	Segmentation by 2-way Ensemble U-Net	Evaluation Metrics
			AP=0.88 AS=0.89 AJ=0.80 AD=0.89 A=0.98
			AP=0.68 AS=0.76 AJ=0.55 AD=0.68 A=0.94
			AP=0.91 AS=0.90 AJ=0.83 AD=0.90 A=0.97
			AP=0.92 AS=0.74 AJ=0.69 AD=0.76 A=0.96

Fig. 2 Performance of the model on four images, along with their outputs

The advantage of our model over the original U-Net model is that as the number of filters in our model is much less than in the original model, there are much fewer parameters in our model, resulting in lesser time required to train and test the model as well as less space required to store the model than that for the original U-Net. Using fewer filters yielded better results, thus showing the possibility of an overfitting problem in the original U-Net model and proving the efficiency of our model. Upon checking, it is found that the original U-Net has around 34.5 M parameters, whereas our proposed ensemble model has only around 4.5 M parameters. That’s a huge reduction! Thus, our mode has been able to achieve a better performance than some of the state-of-the-art models, while using much less parameters.

As stated earlier, the model was created using 2D slices of the 3D volumes. Thus, a possible future work can be to extend this work to the third dimension. Another

future prospect can be to search for additional features and develop architectures that work on 2D slices and provide a better performance, at the same time making it less complex.

7 Conclusion

In this paper, we proposed 2-way Ensemble U-Net, a CNN architecture for performing the task of segmenting the fetal brain into its 7 major components. This work is an extension of the already existing works on segmentation of the fetal MRI. The input is the 2D fetal brain slices, and the output is the segmentation of the image. Our work is inspired from the U-Net model. We have tested our model on the test set and also on some individual inputs and compared their outputs. The results have been quite satisfactory. Our model returned better results than state-of-the-art models, while using much fewer parameters.

8 Code Availability

The code used in this work is uploaded and publicly available in the following link: <https://github.com/tg2001/2-way-Ensemble-U-Net>.

References

1. Dick S (2019) Artificial intelligence. *Harvard Data Sci Rev* 1(1). <https://doi.org/10.1162/99608f92.92fe150c>
2. Roy S, Meena T, Lim SJ (2022) Demystifying supervised learning in healthcare 4.0: a new reality of transforming diagnostic medicine. *Diagnostics* 12(10):2549. <https://doi.org/10.3390/diagnostics12102549>
3. Roy S, Whitehead TD, Li S et al (2022) Co-clinical FDG-PET radiomic signature in predicting response to neoadjuvant chemotherapy in triple-negative breast cancer. *Eur J Nucl Med Mol Imaging* 49:550–562. <https://doi.org/10.1007/s00259-021-05489-8>
4. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. in *nature*, vol. 521(7553). Springer, Science and Business Media LLC, pp436–444. <https://doi.org/10.1038/nature14539>
5. Hagerly J, Stanley RJ, Stoecker WV (2017) Medical image processing in the age of deep learning. In: *Proceedings of the 12th international joint conference on computer vision, imaging and computer graphics theory and applications (VISIGRAPP)*, pp 306–11
6. Ronneberger O, Fischer P, Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In: *Lecture notes in computer science*. Springer International Publishing, pp 234–241. <https://doi.org/10.1007/978-3-319-24574-428>
7. Rampun A, Jarvis D, Griffiths P, Armitage P (2019) Automated 2d fetal brain segmentation of MR images using a deep U-Net. In: *Asian conference on pattern recognition*. Springer, Cham, pp 373–386

8. Faghihipirayesh R, Karimi D, Erdogmus D, Gholipour A (2022) Deep learning framework for real-time fetal brain segmentation in MRI (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.2205.01675>
9. Kuklisova-Murgasova M, Quaghebeur G, Rutherford MA, Hajnal JV, Schnabel JA (2012) Reconstruction of fetal brain MRI with intensity matching and complete outlier removal. *Med Image Anal* 16(8):1550–1564. Elsevier BV. <https://doi.org/10.1016/j.media.2012.07.004>
10. Rajchl M, Lee MCH, Schrans F, Davidson A, Passerat-Palmbach J, Tarroni G, Alansary A, Oktay O, Kainz B, Rueckert D (2016) Learning under distributed weak supervision (Version 1). arXiv. <https://doi.org/10.48550/ARXIV.1606.01100>
11. Khagi B, Kwon GR (2018) Pixel-label-based segmentation of cross-sectional brain MRI using simplified segnet architecture-based CNN. *J Healthcare Eng* 2018:1–8. Hindawi Limited. <https://doi.org/10.1155/2018/3640705>
12. Sahiner B, Pezeshk A, Hadjiiski LM, Wang X, Drukker K, Cha KH, Giger ML et al (2019) Deep learning in medical imaging and radiation therapy. *Med Phys* 46(1):e1–e36
13. Shen D, Wu G, Suk HI (2017) Deep learning in medical image analysis. *Annu Rev Biomed Eng*. 19:221–248. Epub 2017 Mar 9. PMID: 28301734; PMCID: PMC5479722. <https://doi.org/10.1146/annurev-bioeng-071516-044442>
14. Roy S, Shoghi K (2019) Computer-aided tumor segmentation from T2-weighted MR images of patient-derived tumor xenografts. <https://doi.org/10.1007/978-3-030-27272-214>
15. Papandreou G, Chen LC, Murphy K, Yuille AL (2015) Weakly- and SemiSupervised learning of a DCNN for semantic image segmentation (Version 3). arXiv. <https://doi.org/10.48550/ARXIV.1502.02734>
16. Ramesh K, Kumar GK, Swapna K, Datta D, Rajest SS (2021) A review of medical image segmentation algorithms. *EAI Endorsed Trans Pervasive Health Technol* 7(27):e6. <https://doi.org/10.4108/eai.12-4-2021.169184>
17. Zhou Z, Rahman Siddiquee MM, Tajbakhsh N, Liang J (2018) Unet++: a nested u-net architecture for medical image segmentation. In: *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, Cham, pp 3–11
18. Moustafa MS, Mohamed SA, Ahmed S, Nasr AH (2021) Hyperspectral change detection based on modification of UNet neural networks. *J Appl Remote Sens* 15(2):028505
19. Siddique N, Paheding S, Elkin CP, Devabhaktuni V (2021) U-Net and its variants for medical image segmentation: a review of theory and applications. *IEEE Access* 9:82031–82057. <https://doi.org/10.1109/ACCESS.2021.3086020>
20. Salehi SSM et al (2018) Real-time automatic fetal brain extraction in fetal MRI by deep learning. In: *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pp 720–724. <https://doi.org/10.1109/ISBI.2018.8363675>
21. Payette K, de Dumast P, Kebiri H et al (2021) An automatic multi-tissue human fetal brain segmentation benchmark using the fetal tissue annotation dataset. *Sci Data* 8:167. <https://doi.org/10.1038/s41597-021-00946-3>
22. Payette K et al (2021) Fetal brain tissue annotation and segmentation challenge results. arXiv. <https://doi.org/10.48550/arXiv.2204.09573>
23. Myronenko A (2019) 3D MRI brain tumor segmentation using autoencoder regularization. In: Crimi A, Bakas S, Kuijf H, Keyvan F, Reyes M, van Walsum T (eds) *Brainlesion: Glioma, multiple sclerosis, stroke and traumatic brain injuries*. *BrainLes* 2018. Lecture notes in computer science, vol 11384. Springer, Cham. https://doi.org/10.1007/978-3-030-11726-9_28