# Classification of Common Fetal Anatomical Planes from Ultrasound Imaging using Dempster Shafer Theory and Deep Learning

by

A.M. Tayeful Islam
19101107
Marshia Nujhat
19101100
Atanu Roy
19101267
Ahmed Mayeesha Reza Agomoni
19101181

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
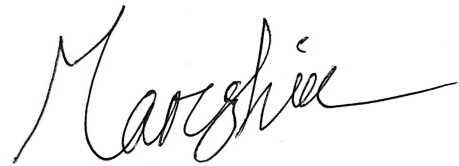September 2022

# Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

**Student's Full Name & Signature:**



_____
A.M. Tayeful Islam
19101107



_____
Marshia Nujhat
19101100



_____
Atanu Roy
19101267



_____
Ahmed Mayeesha Reza Agomoni
19101181

# Approval

The thesis/project titled "Classification of Common Fetal Anatomical Planes from Ultrasound Imaging using Dempster Shafer Theory and Deep Learning" submitted by

1. A.M. Tayeful Islam(19101107)

2. Marshia Nujhat(19101100)

3. Atanu Roy(19101267)

4. Ahmed Mayeesha Reza Agomoni(19101181)

Of Summer, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on September 25, 2022.

**Examining Committee:**

Thesis Supervisor:
(Member)

_____

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

_____

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

_____

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

# Ethics Statement

We have studied different journals, conference publications, websites for our thesis purpose. We have also collected a publicly available dataset consisting of fetal ultrasound images.

# Abstract

Ultrasound (US) examination is a widely used important instrument to monitor mother and fetus health in a cost-effective and non-invasive way. The acquisition of Ultrasound (US) images to determine vital fetal organs for the screening of fetal abnormalities requires identifying the exact plane and region of the desired organs. Even after following guidelines from appropriate committees, a sonologist sometimes may have difficulties in acquiring an excellent fetal plane image or make errors in judgement for several reasons like inexperienced operators, faulty equipment or movement of the fetus. Furthermore, sometimes due to the fetus being in critical positions or due to the increase of adipose tissue inside the mother, it can create various problems in the imaging like artifacts, acoustic shadows or even low signal to noise ratio. Also, in an appropriate institute, a specialist of fetal images reviews the sonographer's analysis and chooses images that contains structures of interest which later gets reviewed by a senior maternal-fetal expert or a specialist doctor. This is a manual process which is expensive, cumbersome and sensitive to mistakes. So we propose a method that combines Convolutional Neural Network (CNN) and Dempster-Shafer theory (DST) to create a DST based evidential classifier or evidential CNN called E-CNN for the classification of common fetal anatomical planes like brain, abdomen, thorax, femur as well as the maternal cervix from its ultrasound images.

**Keywords:** Ultrasound (US) images, Convolutional Neural Network (CNN), Dempster-Shafer theory (DST), evidential classifier, E-CNN, classification, common fetal anatomical planes

# Dedication

We dedicate our work to our parents and our respected teachers whose guidance and support allowed us to accomplish this report. Without their belief and encouragement, this would not have been possible and we are truly humbled.

# Acknowledgement

At the beginning, we would like to thank and praise the Almighty for His blessings that allowed us to accomplish our work without any notable hinderances.

Secondly, we would like to show our appreciation for our Advisor Dr. Md. Golam Rabiul Alam Sir for his help and guidelines in our work. We appreciate and honor his constant support and aid.

And finally we would like to acknowledge the continuous support provided by our parents. It is their prayers and guidance for which we have been able to come close to our graduation.

# Table of Contents

# List of Figures

# List of Tables

# Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document.

$CNN$  Convolutional Neural Networks

$Conv2D$  Convolutional 2D

$ConvNet$  Convolutional Neural Networks

$CT$    Computed Tomography

$DCNN$  Deep Convolutional Neural Networks

$DS$    Dempster-Shafer

$DST$  Dempster-Shafer Theory

$E-CNN$  Evidential-Convolutional Neural Networks

$MRI$  Magnetic Resonance Imaging

$US$    Ultrasonogram

$VGG$  Visual Geometry Group

$XAI$  Explainable Artificial Intelligence

# Chapter 1

# Introduction

## 1.1 Research Motivation

Ultrasound is one of the most common methods for the analysis of fetal abnormalities, weight or doppler blood flow [2]. The US images are obtained after obeying the international guidelines set by the respective committees for each test [7][9]. For the proper diagnosis of the fetal health, the biomarkers must be taken from the appropriate fetal plane which seldom gets mislabeled for various reasons starting from the operator's lack of expertise, faulty equipment, time limitations and fetal movement. Furthermore, due to missing boundaries, low signal to noise ratio and speckle noise in ultrasound imaging, identification of the fetal organs have been proved to be very complicated.

In the underdeveloped regions of the world, these problems are made worse due to unavailability of good quality equipment, and trained sonologist specialists leading to mislabeling of the fetal plane as a result of false evaluations of the biomarkers. As a result of the low cost in comparison to other alternatives such as MRI, CT or 3D ultrasound, 2D US images were used the most. Keeping all these in mind, we have come up with the proposal to use Dempster-Shafer Theory on top of traditional neural networking models so that the resultant classifier can be used for the accurate identification of fetal planes. This will help mitigate the problem of having little to no proper technical assistance that is crucial to properly annotate such imaging, ensuring a safe diagnosis for both the mother and the fetus.

## 1.2 Research Problem

In obstetrics, ultrasound imaging is used for assessing the development of a fetus during pregnancy. The contouring of the areas under study has to be done by a physician, requiring specialized knowledge and can be time-consuming. Additionally, the detection becomes difficult due to speckle noise, low signal-to-noise ratio, varying intensities of acoustic shadows, motion blurring, missing boundaries, and inter-operator errors [18]. Despite being prone to some errors, 2D ultrasound imaging is used to measure biometrics which is likely to hinder early detections of malformations. In contrast, 3D ultrasound might be preferred over 2D ultrasound due to its capacity to analyze volumetric features of the fetus and provide other countermeasures to some of the existing issues with the 2D version. Usually, these

fetal anatomical planes are identified from ultrasound imaging. Ultrasound images are not easy to analyze with the traditional neural network detection methods like perceptron or multilayer perceptron. Also, data availability may vary from place to place. Misclassifications or misinterpretations may occur with precise classification models. Further information is thus required regarding a model's prediction.

Various works has been seen in recent years that use traditional probabilistic classifiers for classification problems. However, little work has been done in using evidential classifiers to determine the results in cases of highly uncertain datasets. Work with evidential deep learning using Dempster-Shafer theory is in particular, comparatively less. The use of belief function classifiers in ConvNets is noteworthy. However, in the situation of conflicting features, this classifier is susceptible to assigning a rejection action [20]. Various classifier fusion models have been used in previous works but little work has been done with an evidential classifier where features extracted from one probabilistic model are broken down into elementary mass functions and combined in a DS layer. We observe that even a high performing model with a very high accuracy can underperform while categorizing images of some classes.

## 1.3 Research Contribution

The purpose of this research was to correctly identify the abdomen, femur, thorax, brain of a fetus as well as the maternal cervix. Thus we opted to design a DST based evidential classifier using an existing CNN model and apply the Dempster-Shafer theory for uncertainty classification. Here we are breaking down our research contributions into short steps for ease of understanding:

- Analyze the dataset based on the anatomical planes from the ultrasound (US) images, namely: abdomen, femur, thorax and brain of the fetus as well as the maternal cervix and propose an evidential classifier using an existing high performing CNN model, i.e. VGG-19 with DST to predict the different classes of common fetal anatomical planes mentioned above as there are high possibilities of uncertainty in fetal ultrasound images.

- We aimed to find a model that will be able to give us better accuracy and also evaluate the parameters, optimizers and prototype number needed to achieve that. A DST based evidential classifier with VGG-19 gave a significantly satisfactory result for this case.

- We noted the changes in accuracy and uncertainty in the data and model when VGG-19 was applied only and when a DS layer was applied over the CNN layer.

- We opted to construct a utility layer after the DS layer which gives us more information about the prediction of the model in cases of high uncertainty and presence of outliers. But in cases when there is precise classification, the evidential model still performed very well while classifying the images belonging to a single class.

- Scenarios where the model underperformed while categorizing images of certain classes were predicted to belong to a few classes with similar probabilities instead of a single class.

# Chapter 2

# Literature Review

While planning our workflow to achieve the desired outcome, we have come across some research workings on DST, CNN, DCNN and hybrid models consisting of DST and CNN. In this section, we will briefly present some background studies on the models and algorithms that we plan on using in our research, along with a brief overview of the related works we have gone through.

## 2.1 Background Study

Here we have included some theoretical knowledge about the models and algorithms that we plan on using, along with what other researchers have also tried in their respective research works. This section first recalls some research studies on the Convoluted Neural Network (Section 2.1.1) model followed by Dempster-Shafer Theory (Section 2.1.2) and Evidential Classifier (Section 2.1.3).

### 2.1.1 Convolutional Neural Network

Convolutional neural networks (CNN) are deep learning algorithms that are frequently used for the segmentation and classification of images to be used in pattern recognition and object identification. CNN is currently regarded as the best algorithm for the automated processing of images and identifying the objects in those images [25]. Various models of CNN are used for different types of applications. It has had remarkable success in the medical sector as it is extensively used to analyze X-Ray images, MRI results, Ultrasound images and in brain tumor segmentation. CNN consists of three layers which are the convolutional layers, pooling layers, and fully connected layers and has been able to extract local features and use them to determine global features. [20]. For example, if the lower layers of CNN can identify edges, the higher layers can identify more abstract objects like human faces and body parts [24]. Another major reason for using CNN in our model is its robustness and automation. While training a model with deep representation, there may be translation or distortion issues. Ultrasound images are especially prone to motion blurring and noise. CNN has a high tolerance to such factors and can also execute data driven object representation in an automated method [20].

### 2.1.2 Dempster-Shafer Theory

Despite using CNN and other deep learning models which gives a precise classification, there still lies possibilities of leaving out potential evidence to the end result. To be cautious of such possibilities, the probabilities of events can be assigned as highly uncertain samples to sets of classes using the Dempster-Shafer theory [24]. Dempster-Shafer (DS) theory is a generic form of probability theory in a finite discrete space where probabilities of events are allocated to sets instead of being mutually exclusive [5]. In DST, we can combine evidence from multiple sources, i.e. multiple possible events. One of the essential aspects of DST is that it can deal with various levels of precision about any particular information. It also directly portrays the uncertainty of system responses where a set or interval can demonstrate an indistinct input and even the output that we get from our results is a set or interval. This is why the Dempster-Shafer theory has been perpetually used over the last two decades, especially in pattern recognition and supervised classification of objects [24]. One such method that integrates the CNN model and the DST theory is to design classifiers that will give a decision output based on evidence of all possible inputs.

### 2.1.3 Evidential Classifier

One form of classifier used for decision making based on DST and deep CNN that has gained popularity in recent works is the evidential classifier, popularly known as Evidential CNN (E-CNN) or Evidential Deep Learning. This type of DST based evidential classifier segments the evidence of our input features into simple mass functions and then clusters them using the Dempster-Shafer rule. Unlike traditional probability approaches which provide outputs based on single evidence, this classifier can generate outcomes that are much more broad and helpful in estimating a result [24]. Thus, the collection of multiple evidence allows the quantification of uncertainty of data and rejects the incorrectly classified data. Despite the emergence of studies regarding DST and deep learning, research works that use evidential classifiers are comparatively scanty. Many of the previous works are based on deep learning classifier fusion. Regardless, no such work has been found that performs the classification of common fetal anatomical planes using evidential classifiers.

## 2.2 Related Works

An approach to segment the skull of a fetus was written in 2018 [14]. Cerrolaza et al. designed a new framework with ultrasound physics by using 66 fetal 3D ultrasound images for accurate segmentation of the whole skull of a fetus. The automatic framework consisted of a two stage CNN and incorporated extra structural and contextual data into the segmentation process. To test the precision of the segmentation process, Shadow Casting MAP (SCM) and Incidence Angle Map (IAM) were used. The complete evaluation was seen in the case of combining both SCM and IAM which gave a DC of $0.83 \pm 0.06$. The results were also compared to that of a single channel which did not show much variation to the error test result obtained using

SCM.

Later in 2019, [18] Sobhaninia et al. used a multitask deep neural network developed on Link-net to segment fetal head and its head circumference. Close to thousand 2D US images of the fetal head were used to analyze the performance difference between a multitask and an already existing single task network by Heuvel et al. [15]. The performance was analyzed using a Dice Similarity Coefficient (DSC), Difference (DF), Hausdorff Distance (HD) and Absolute Difference (AD). After training the dataset with a multitask learning network (MTLN), it was found that the DSC score of the proposed model by Heuvel et al. was slightly higher than the multitask network model by Sobhaninia et al. But the ADF score was 2.12 + 1.87 while the HD score was 1.72 + 1.39 which was marginally higher than the ADF and HD score of the single task network model.

In 2020, Qu et al. [17] made another approach where they proposed two methods DCNN and CNN-based transfer learning to recognize the six standard planes that fetal brains have automatically. The CNN model used down-sampling for a fast and efficient training process, whereas the transfer learning, accompanied by data augmentation, served as a countermeasure to the overfitting problem. All the layers in Dataset 2 were trained using the proposed DCNN model that had been previously trained on Dataset 1, even though the images were alike in both datasets. The proposed transfer learning-based method proved to have an accuracy of 89.1%. However, network out-fitting is likely to occur on a small dataset due to misuse of DCNN which might lead to a degradation in performance, despite it resulting in the best performance out of all the other methods.

In this [22] paper, Skeika and the other authors have tried to amalgamate a variety of strategies and come up with a new version of the existing V-net models where the networks were modified to receive 2D inputs named VNet-c. Additionally, the network was deepened to improve its learning capacity and inferred analysis, resulting in overfitting due to the increasing number of trainable parameters.To avoid that, the Data Augmentation along with Dropout techniques were introduced where a collection of neurons were deactivated from the interconnected layer at each traversal of the adjusting stage and newly generated artificial images were produced to increase the training dataset. Furthermore, Batch-normalization was also brought into use which sped up the learning and training processes. For research, the dataset was acquired by Heuvel and collaborators from the HC18 challenge with the proposed method coming up with an accuracy of 97.92%.

Then in 2018, Yu et al. [13] have tried to automatically detect the fetal facial standard plane using deep convolutional neural networks (DCNN). Their proposed structure has 16 convolutional layers with small kernels and three fully connected layers. They have also used a Global Average Pooling (GAP) with the final layer, improving the result and fixing the overfitting problems. Here they have named their model CNN-19-GAP. They have also used batch normalization (BN) to deal with any convergence issues. For their research, they have collected in house data of 20 to 36 weeks old fetuses. According to their results, their models have achieved 96.32% accuracy which is 7-8% higher than the non-GAP models. The CNN-19-GAP

is also more memory efficient than its non-GAP counterparts.

In this [10] paper published in 2017, the researchers have considered the error factor of the 2D imaging due to the operator and the process of finding and marking from that 2D image in general. Therefore they have proposed a segmentation method based on Random Forest and using 3D ultrasound images of the fetus to incorporate volumetric data which helps with the plane selection and provides a better understanding of fetal cranial structure. They have used a new model, SGeo-RF and compared it with the more traditional CNN and plain Random Forest (RF). The proposed SGeo-RF model achieved an accuracy of 98% whereas CNN got 94% and RF got 93%. According to the authors, this accuracy that 3DUS can achieve can also transfer the pressure off the manual identification of the fetal planes.

Since 2D ultrasound images are hard to identify visually, sometimes in many cases, resized smaller images can misclassify specific shapes like kidney. So to prevent this, here in this [19] research, Sridar et al. used a method of mixing the prediction from resized or reshaped regions of fetal structures to get a more accurate value and identification of the fetal organs by using CNN. This method ensures regions of organs that have more bones do not get misclassified by using contextual information.

In 2018, Chen et al. [3] gave a deep learning model, U-Net, for the automated segmentation and measurement of the fetal lungs. The model was trained by over three thousand datasets augmented from 250 US images and the manual annotations were done by an ultrasound physician, that represented the ground truth for assessing the performance of the automated segmentation method proposed by this paper. A max-pooling in the down-sampling layer halves a feature map, accompanied by two 3 x 3 convolution layers with padding which aids for a more accurate depiction of the images. ReLU function and a batch normalization layer were also used with each convolution layer to attain a good convergence, reaching an accuracy of 98%.

Another approach can be seen in this [6] paper where the authors have proposed the usage of DS theory to combine values of R (red), G (green), and B (blue) components of the same cell from an image. The main goal of using Dempster Shafer's theory is to partition the image into homogeneous regions by fusing the pixels coming from the three images. Initially, with the help of the DS combination rule, the mass functions for all pixels of each of the three images are combined using the orthogonal sum after the mass function values have been determined. Next, the DS decision strategy aids in acquiring the final image segmentation. This decision strategy is selecting the hypothesis deemed the best fit after considering the maximum belief value calculated from the previously fused mass functions from the three images. Even though this model works well with using only some pieces of information such as details concerning the grey levels covering each of the three component images, it still requires a priori knowledge.

This research paper [12] proposed an architecture to detect the heartbeat from a linear ultrasound video. They used a dense feature extraction then they encoded SIFT, SURF and rootSIFT features using BoVW, VLAD, and FV encoding. In their case, rather than using CNN they used SVM to classify the regions of fe-

tal heart since their data set was small and gave a mean accuracy of 93.1%. In 2019, Tong et al. [20] used a classifier that is based on Convolutional Neural Network (CNN) and Dempster-Shafer theory to detect object with inconclusive pattern recognition. By combining ConvNet and a belief function classifier known as ConvNet-BF classifier, ConvNet was used as a feature producer, the BF classifier as a mass function generator and a decision rule to detect objects like birds, cats or trucks in three different datasets. One of the datasets, the CIFAR-10, consisted of 10 classes while the CIFAR-100 had a similar size and formatting as the CIFAR-10 but with 100 classes. The results from the ConvNet-BF classifier was compared to a NIN classifier and it was found that the ConvNet classifier performed on the CIFAR-10 dataset has a lower test set error rate than a NIN classifier when the rejection rate of erroneous classified patterns was higher than 7.5%. A similar result was obtained using the CIFAR-100 dataset where the test set error rate was slightly higher while using ConvNet classifier without rejection (40.62%) than using a NIN classifier without rejection (39.42%). However, again the test set error rate significantly decreased when ConvNet classifier was used by rejecting some incorrect classifications.

In 2019, Denœux [16] talks about how the high level features can be converted into DS (Dempster Shefer) mass functions and then adding them up by the combination rule of Dempster. The high degrees of freedom of the mass function carries a lot more information which helps to identify the lack of evidence and conflicting evidence separately. This also allows for the implementation of decision rules like the interval dominance rule, which selects a collection of classes when the available evidence does not unequivocally lead to a single class, lowering the error rate. According to their findings, DS theory can be used to design new classifiers, including deep neural networks as opposed to using belief functions in everything.

In 2021, Tong et al. [24] proposed a classifier consisting of CNN and DS(Dempster-Shefer) theory, called the evidential classifier for set based classification. After getting the high dimensional features from the input data they convert those into mass functions using Dempster's Rule in the DS layer. Then they have trained evidential deep-learning classifiers with a stochastic gradient descent algorithm. According to their findings, implementing DS theory with deep CNN and evidential classifiers improves the overall accuracy by assigning ambiguous patterns to the sets.

Shoyaib et al. in this [8] research paper aimed to solve the inaccuracy issue seen while working on skin detection due to fewer data, more extended training period and often the tedious process of fine tuning which is sometimes not even possible due to the state of the dataset. To solve these issues they are proposing a hybrid model using Dempster Shafer theory. They are using this theory in particular due to its powerful and flexible nature with ambiguous datasets, making it more suitable for the other object detection methods. Their final result shows that their proposed hybrid model works well when the training data is deficient, taking the accuracy to 87.47%from 68.81%.

Here in this paper, [26] Yin et al. proposed a blackboard-oriented system that will use the Dempster Shefer theory and particularly the compatible frames and

multivariate belief functions. The suggested Medical Image Understanding System (MIUS) comprises three phases of which the acceptance of the hypothesis brought about in phase two will use the guidance of the proposed system into creating anatomic structures in the said image after extracting the entities as a form of regions or curves. The multivariate belief function model has the evidence parameter that is evaluated to obtain the belief of the hypothesis. [26] The beliefs of the internal hypothesis, which are based on the evidential space, are assessed by estimating the beliefs associated with the multivariate belief functions to the respective margin. Belief intervals evaluate the probability of the hypothesis and strengths of evidence as opposed to the point values.

When discussing medical image segmentation it is of utmost concern to create trust between sonologist and deep learning models. To do that, [23] this research uses AlbuNet which diagnoses pneumothorax in x-ray images. After that they used a three block trial where in the first block the expert's prediction of the AI diagnoses and in the second block participants evaluated the explanations created through XAI by certifying the AI for different cases. In the research, the radiologists accurately assumed the AI's judgement on average 6 out of 8 trials. Despite the limitations of small datasets and few participants this research demonstrated that explanations generated by Bayesian Teaching help medical experts inform certification decisions, thus creating trust between AI and radiologists.

# Chapter 3

# Methodology

The motive of this research is to introduce a Dempster-Shafer based evidential classifier for classification of common fetal anatomical planes, mainly the abdomen, femur, thorax, brain and the maternal cervix. To do that, we gathered a proper dataset containing ultrasound images of the mentioned fetal body parts for our model as depicted in Figure 3.1. The data obtained from the dataset was given as an input and necessary preprocessing was performed on the data. After that, the appropriate features to detect the fetal anatomical planes were selected. Necessary feature encoding was done, and the dataset was split into train and test data. Then the model was trained with the training data fitting it to the E-CNN model and the resultant model was evaluated with test data.

The architecture of an E-CNN model typically consists of three main stages as stated below:

1. The input data at first goes through a multi-stage deep CNN consisting of convolution and pooling layers to represent the necessary features [24].

2. The data then passes through a Dempster-Shafer (DS) layer in order to aggregate the input evidence into mass functions.

3. The final stage is the utility layer which makes a decision on the basis of the outcomes of the previous layers to classify the fetal anatomical planes by partially assigning multi class acts to a set or interval.

The accuracy was evaluated after implementing the CNN layer and then with the DS layer to observe the percentage of matches with the labels. If the accuracy was satisfactory enough, the results were stored and obtained, else necessary changes in parameters were applied and the model was re-trained. Upon achieving the desired results, a complete evidential classifier was then designed by implementing the utility layer.
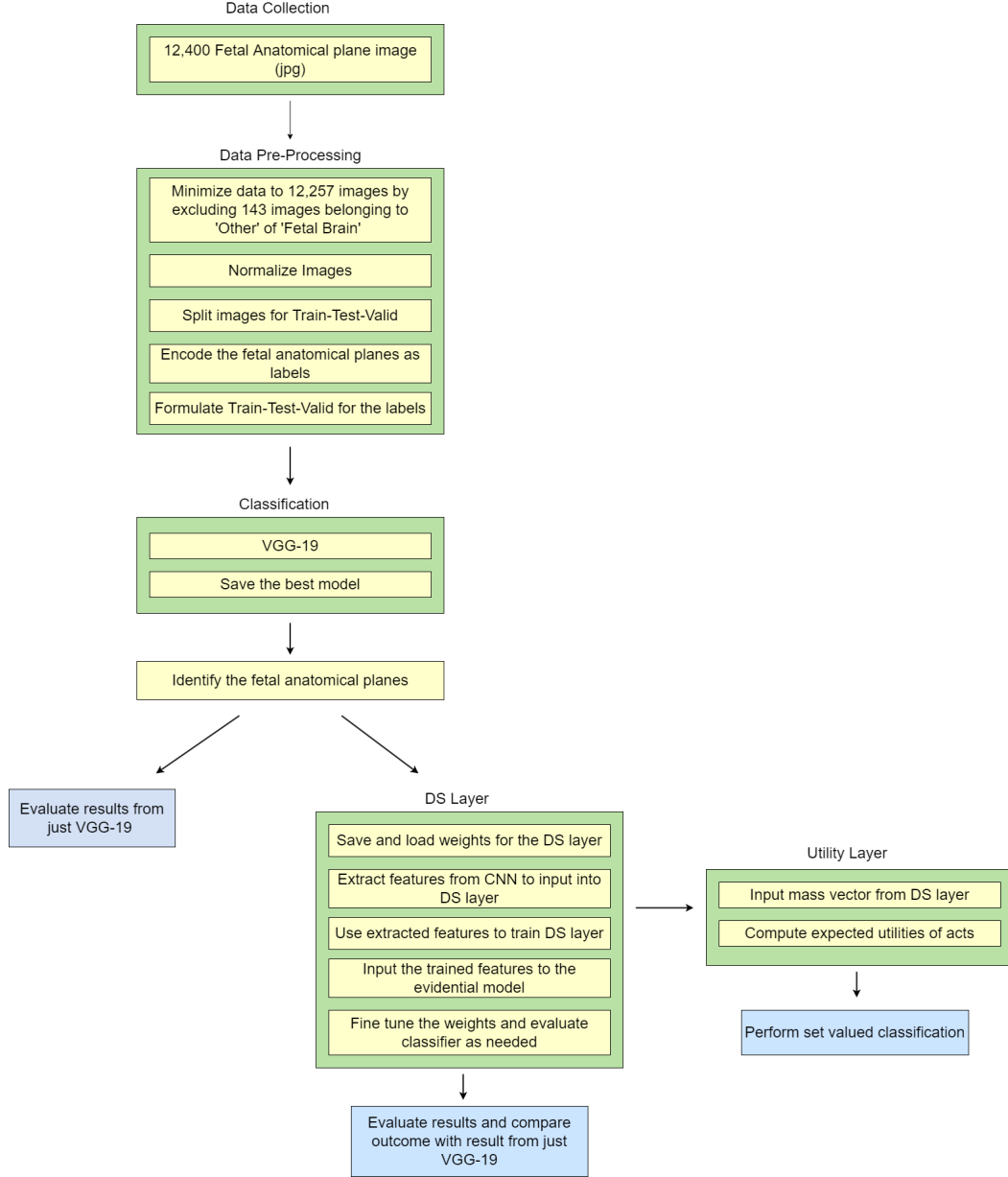
Figure 3.1: Top level overview of the proposed methodology

# 3.1 Data Collection

We have found a relatively large dataset consisting of maternal and fetal screening US images. This dataset has been collected by numerous operators of two different hospitals using different machines. Later, the US images were divided into six classes: Maternal Cervix, Fetal Abdomen, Fetal Brain, Fetal Femur and Fetal Thorax and a general category named 'Other' for the less common fetal planes. The Fetal Brain was also divided into three classes: Trans-thalamic, Trans-cerebellum and Trans-ventricular. So in total, taking the three brain classes into account, there are 8 classes that we used for our analysis from this dataset. This dataset was also declared public by the authors so we were able to use it for our research [21]. The images in the dataset were collected by the following machines: Voluson E6, Voluson S10, Aloka and a group of other machines.

| Anatomical Planes used for detection | Number of Images |
|---|---|
| Fetal abdomen | 711 |
| Trans-thalamic | 1,638 |
| Trans-cerebellum | 714 |
| Trans-ventricular | 597 |
| Fetal femur | 1,040 |
| Fetal thorax | 1,718 |
| Maternal cervix | 1626 |
| Other | 4,213 |
| Total | 12,257 |

Table 3.1: Classes of fetal anatomical planes used for our analysis

## 3.2 Pre-Processing

### 3.2.1 Image Pre-Processing

For the analysis of images, we have converted our png images to jpg file format. While converting to jpg, we have also resized the images into 224x224 pixels and kept the ratio equal to the original image. After that, we read the image data using imshow() function. Then, we converted those images to numpy arrays with data type of float32 and divided them by 255 in order to normalize them. Then, we have stored those images in an array named imgdata.

### 3.2.2 Data Pre-Processing

In our csv file, we listed all the labels for each of our fetal anatomical planes. At first, the labels were distributed between two columns namely 'Plane' and 'Brain Plane'. 'Plane' column had the following values: Other, Fetal abdomen, Fetal brain, Fetal femur, Fetal thorax and Maternal cervix. 'Brain Plane' had the values: Not a brain, Trans-thalamic, Trans-cerebellum and Trans-ventricular. For the ease of our work, we have merged these two columns into one single column named 'Merged_plane'. The new column had the following values: Fetal abdomen, Fetal femur, Fetal thorax, Maternal cervix, Other, Trans-cerebellum, Trans-thalamic and Trans-ventricular. We checked if the 'Plane' value is 'Fetal brain', if it is a brain then we stored the specific class of the brain (Trans-cerebellum, Trans-thalamic and Trans-ventricular), else we stored the 'Plane' class of that image like Other, Fetal abdomen, Fetal femur, Fetal thorax and Maternal cervix. In this way, we had to deal with lesser number of classes for our classification work and the model also performed better with lesser classes. In addition, out of 12,400 images, some images belonged to the 'Fetal brain' class but were being classified as 'Other'. There were 143 images of such sort. Those were excluded before being given as input to the model as those images were being misclassified to a huge extent. So, for better performance, we proceeded to work with 12,257 images.

## 3.3  Model Selection

In our case, we used a DST based evidential convolutional neural network (E-CNN) classifier following the work of Tong et al [24]. But, we decided to use VGG-19 to import the features obtained from the input data and convert those into elementary mass functions and further aggregate them using Dempster's rule. Since this is a distance based classifier, the closeness of an input vector to the prototypes in the model is taken as the evidence for class assignment of test samples in the E-CNN classifier [24]. The model can be visualized in the following figure:
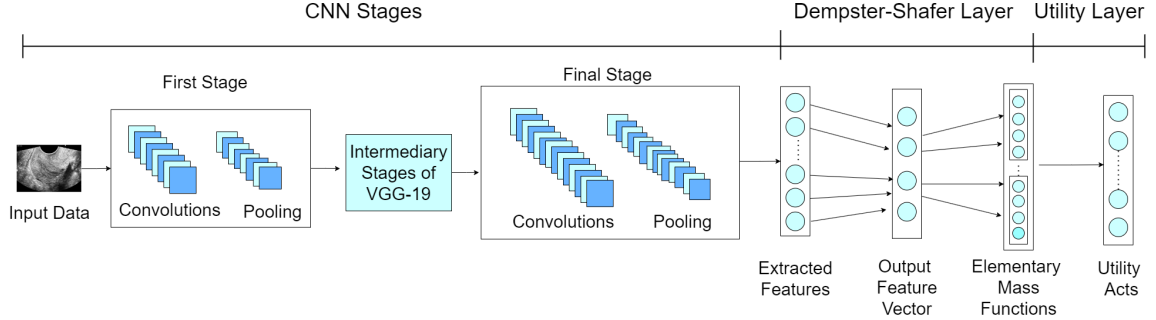


Figure 3.2: Dempster-Shafer Based Evidential Deep Learning Classifier

### 3.3.1  VGG-19

VGG-19 model was mainly used as it is said to have improved accuracy from its predecessors. VGG-19 is one of the variants of the VGG model where VGG-19 consists of 16 Convolution layers, along with 5 MaxPool layers. In addition, it has 3 Fully Connected layers, and 1 Softmax layer, where the softmax activation function is applied.

A Convolution layer runs a filter through the input image in order to retrieve information to it, resulting in a reduction of the input image's dimensionality. Usually, the kernel, better known as the filter, has a size smaller than the supposed input image. Rectified Linear Unit (ReLU) is applied to the output of the convolution layer where it compensates for any sort of signal parsing errors, guiding the signal back to where it is supposed to proceed. ReLU is usually used for the hidden layers. To summarize, the Convolution layer takes in input, having a volume of size $W_1$ x $H_1$ x $D_1$, and needs four variables :

- Amount of filters/kernels, K

- Area or size of the filters, F

- Stride, S

- How much zero padding, P, is applied

With these parameters and input, we get an output of volume size $W_2$ x $H_2$ x $D_2$, where [11] :

$$W_2 = \left( \frac{W_1 - F + 2P}{S} \right) + 1 \qquad (3.1)$$

$$H_2 = \left( \frac{H_1 - F + 2P}{S} \right) + 1 \qquad (3.2)$$

Since the Convolution networks are known to possess the parameter sharing property, $(F.F.D_1)$ weights are introduced per filter where the total number stands at $(F.F.D_1)$x$K$ weights and $K$ biases.

MaxPooling takes into account the largest information through the help of the filter that is examining the image in a given stride. This discards the smaller features and only keeps the largest representative of a given square space, reducing the dimension of the image even further. For instance, in order to reduce the dimension of the given image's height and width by 2, a pooling layer of size 2x2 is used with a stride, S=2. To further reiterate, given the pooling layer accepts input of volume size $W_1$ x $H_1$ x $D_1$ and needs two parameters :

- Area or size of the filters, F

- Stride, S

With the above variables and input, an output of volume size $W_2$ x $H_2$ x $D_2$ is produced as follows [11]:

$$W_2 = \left( \frac{W_1 - F}{S} \right) + 1 \qquad (3.3)$$

$$H_2 = \left( \frac{H_1 - F}{S} \right) + 1 \qquad (3.4)$$

$$D_2 = D_1 \qquad (3.5)$$

Furthermore, this pooling layer does not introduce any new parameters other than the spatial extent and stride and it is rare to see any pooling layer applying zero padding to its input.

A Fully Connected layer transforms the two dimensional matrix into a one dimensional one, and this is then fed into a Softmax layer, an activation function, in the output layer which is responsible for multiclass object classification.

The execution of all of the stages of VGG-19 gives a final output which is the feature representation of our input data.

### 3.3.2 Dempster-Shafer Model

After extracting the high dimensional features using the convolutional layer, a Dempster-Shafer (DS) layer was then applied to our model to be able to predict probability distributions and not just purely deterministic point outputs. This is possible when the extracted features are converted to mass functions and clustered in the Dempster-Shafer layer. Here, the mass functions are considered as independent components of evidence [24]. Let, two such mass functions be m1 and m2. Using Dempster's rule $\oplus$ [1], they can be combined as:

$$(m_1 \oplus m_2)(A) = \frac{(m_1 \cap m_2)(A)}{1 - (m_1 \cap m_2)(\phi)} \tag{3.6}$$

Here, $\Omega = \{\omega_1, ...., \omega_M\}$ is a set of classes representing the data. And, A is a focal component of the mass function, $m$, if $m(A) > 0$ where A belongs to $\Omega$ [24].

The output of this layer will give an (M+1) mass vector [24] as the following :

$$m = (m(\{\omega_1\}), ...., m(\{\omega_M\}), m(\Omega))^T \tag{3.7}$$

The mass $m(\{\omega_1\})$ is a measure of belief of this sample belonging to the $\omega_i$ class [24]. This helps to give a better measure of the uncertainty in the model and how well it is trained.

On the basis of these mass functions, a utility layer is then to be applied to implement set-valued classification on the mass functions [24]. The output obtained from the DS layer, i.e. the mass vector, m, is the input to the utility layer which then computes the expected utilities of the acts. Here, act is considered as the allocation of a test sample to a non empty subset A of $\Omega$ where $\Omega$ is the set of classes [24]. In this way, a new sample can be assigned to a set of classes instead of a single class to help predict the uncertainty in the model better. Using the DS and utility layer not only helps to improve the accuracy of data and model, but also improves the detection capabilities of the model itself especially when it comes to outliers or samples of data that are extremely uncertain to predict.

One of the main disadvantages of using a distance based classifier is the computational complexity and to mitigate this problem, we arrange the learning set in a way so as to cap the representative features or prototypes. Every prototype, i, has been assumed to possess a degree of membership to a class, $w_q$ which is represented by $u_q^i$ and complete membership to a class is constricted. The distance $d_i$ between a sample, x and each prototype $p_i$ is computed by:

$$d_i = ||x - p_i|| \tag{3.8}$$

# Chapter 4

# Implementation

## 4.1   Environment Setup

We used Anaconda Distribution for our virtual environment setup. Then, we created a python version 3.9 based virtual environment and installed the necessary libraries and softwares.
These libraries include:

- cuda toolkit (version 11.2)

- cudnn (version 8.1.0)

- Tensorflow

- Matplotlib

- Pandas

- Opencv-python

- Sckit learn

- Sckit image

- Seaborn

As we are working with an image dataset, we opted to use the tensorflow-gpu for importing our deep learning libraries. For that reason, we have used the nvidia cuda libraries along with the tensorflow installation. We have also used the seaborn library to graphically present the results found in our research model.
Here is the system configuration of our test bench:

- CPU: AMD Ryzen 9 5950X

- GPU: NVDIA RTX 3080 Super

- RAM: 64 GB

- OS: Windows 10 Pro

## 4.2 Implementation of layers

### 4.2.1 Convolutional 2D Layer

To create our VGG-19 network, a fixed size of 224x224 jpg images were fed into the VGG network as input and the size was 224x224x3 where '3' represents the number of channels. A filter of size 3x3 was used with stride equal to 3 so that the entire image is covered. Padding was used as a means to keep the essence of the image intact and to make sure information is not lost when a pooling layer is applied to it. MaxPooling of size 2x2 and stride equal to 2 was used which halves the convolution layer output image.

Since it is not possible to identify non-linear functions with just a single line, ReLU activation function was introduced to detect non-linearity in the network. ReLU is commonly used in the hidden layers given that it is a light-weight function in comparison to the other activation functions like sigmoid or tanh.

ReLU formula is given by:
$$f(z) = max(0, z) \tag{4.1}$$

where, $z$ is the input.

The ReLU activation and its derivative are both monotonic, that is, it is neither increasing or decreasing. If the ReLU function receives any negative input value, it transforms the output to 0, otherwise, it returns the input value meaning that the range of this activation function is from 0 to the input itself.

The first two Fully Connected layers had size 4096 with the last Fully Connected layer consisting of 8 units with a Softmax activation function being used for the classification of 8 classes. Unlike other activation functions, Softmax calculates the relative probabilities meaning that the probabilities of each class are not independent of each other. The formula is as follows [11]:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \tag{4.2}$$

where,
$\sigma(z_i) =$ softmax
$z =$ input vector
$e^{z_i} =$ exponential function for input vector
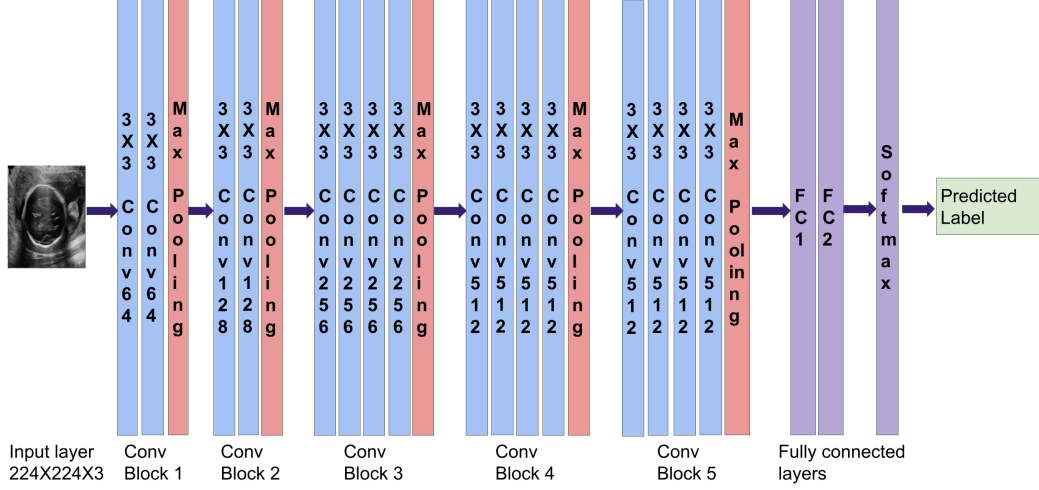$e^{z_j} =$ exponential function for output vector
$n =$ number of classes

Figure 4.1: VGG-19 Architecture

## 4.2.2 Dempster-Shafer Layer

To implement this step, the classes required for the DS layer was imported from a custom made library by Tong et al [24]. Here, the output we get after passing data into the Flatten layer of our convolutional network is passed along with the number of prototypes and another parameter- the shape of the Flatten layer output to the DS1 class of the DS layer. After plotting the model, the shape of the output from the Flatten layer gave us (None, 25088), hence 25088 was given as a parameter and the number of prototypes was taken as 30. The model was also run using 20, 50 and 100 prototypes to evaluate and compare the performance of the model in each case. The distance based support between a test sample, x and each prototype vector in the prototype set, p was calculated [24]. The magnitude of membership of each prototype to a class was also found. After this step, the mass function related to each prototype vector was constructed by combining the magnitude of membership for each prototype and the distance based support. The mass functions computed in the previous step were then clustered using Dempster's rule. The output vector was then finally obtained by the end of the execution of the DS layer to be put as an input to the utility layer.

The DS layer was then trained using 10 epochs. Categorical crossentropy was used to formulate the loss function and 'acc' as the metrics. Previously, the weights obtained from VGG-19 were saved and those were used to train the parameters of the evidential model. 20 epochs were run in this case to train the evidential model with the same metrics and formula for loss function as it was used for training the DS layer.

The DS layer had the following trainable parameters for different prototypes. In all cases, there were no non-trainable parameters.

| Prototype Number | Total Parameters | Trainable |
|:---:|:---:|:---:|
| 20 | 501,960 | 501,960 |
| 30 | 752,940 | 752,940 |
| 50 | 1,254,900 | 1,254,900 |
| 100 | 2,509,800 | 2,509,800 |

Table 4.1: Trainable parameters of the DS Layer

Using different prototypes, we got the following trainable parameters for the evidential model. With 50 and 100 prototypes, even though the trainable parameters are more, it does not necessarily give the best result as it has been seen to depend on the size of the dataset and processing of the model as well.

| Prototype Number | Total Parameters | Trainable |
|:---:|:---:|:---:|
| 20 | 20,525,192 | 20,525,192 |
| 30 | 20,776,172 | 20,776,172 |
| 50 | 21,278,132 | 21,278,132 |
| 100 | 22,533,032 | 22,533,032 |

Table 4.2: Trainable parameters of the Evidential Model

The output obtained from the DS layer is a mass vector which is actually a measure of belief for the model that a certain sample image belongs to a certain class. If there is an uncertainty, that is also categorized in this mass vector by allocating masses consistently across all available classes [24]. The ultimate reality of the result is better depicted after implementing the utility layer.

### 4.2.3 Utility Layer

To implement the utility layer, the necessary classes were imported from a custom made library by Tong et al [24]. The mass vector [24] obtained from the Dempster-Shafer layer was passed to the utility layer to compute the expected utilities. Based on the computed expected utilities of acts, i.e. assignment of a sample to a class, partially a sample was assigned to multiple classes in case the sample is imprecise or confusing. In this case, classes with no evidence pertaining to the sample were rejected using the rejection option. But classes containing evidence found from extracted feature vectors of previous layers, possibly being assignable for the sample image, were assigned to a set for the sample.

# Chapter 5

# Result and Analysis

## 5.1 Performance Metrics

The performance of the model was assessed with parameters like accuracy, recall, f1 score, precision, macro avg and weighted avg. The formulas for some of the metrics are given as:

$$precision = \frac{TP}{TP + FP} \tag{5.1}$$

$$recall = \frac{TP}{TP + FN} \tag{5.2}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{5.3}$$

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{5.4}$$

Here,
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

- Precision indicates the fraction of identifications that were correct for a particular class. If all images of a class were identified as True Positives, then the precision value will give 1.0. In case of any false positives, the value will be less than 1.0.

- Recall denotes the fraction of correctly identified samples by the model [27]. When there are no incorrectly identified samples, recall will give a value of 1.0 for a particular class. If less than 1.0, it is an indicator for the samples that were mislabeled.

- F1 score combines both the precision and recall to give a new evaluation metric. Achieving an f1 score of 1.0 would mean there were no misclassifications while labeling the particular class and that it is giving a perfect result.

- Accuracy is the accuracy percentage of the model. If the model is accurately predicting for all test cases at all times, accuracy will be 1.0. Accuracy graphs have also thus been shown as a performance indicator.

- Macro avg or macro average is the average precision, recall and f1 score between the classes. It is an indicator for which class disparities are present.

- Weighted avg or weighted average is the weighted average precision, recall and f1 score between the classes. This is calculated with respect to the number of samples in each class and hence, class imbalance is a factor for differences between the macro and weighted average.

## 5.2 Performance Study of VGG-19

Since the very first layer of our proposed model is the Convolutional 2D Layer for which we have used VGG-19 architecture to identify the fetal planes into 8 classes, we at first obtained the accuracy for using this architecture only. After training the model with 8579 train images, 1839 test images and 1839 validation images for 20 epochs, we see a comparison between training and validation accuracy. From here, we can interpret that the training accuracy is higher than the validation accuracy.
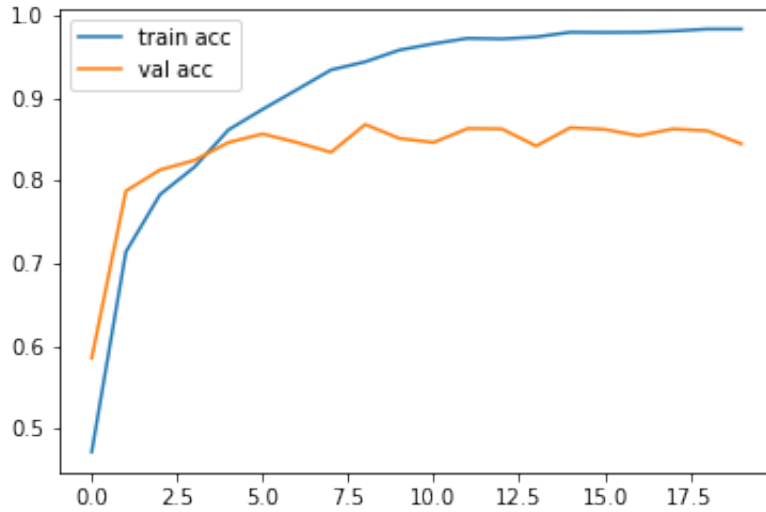


Figure 5.1: Accuracy Graph obtained from applying only VGG-19

|          | precision | recall | f1-score | support |
|----------|-----------|--------|----------|---------|
| 0        | 0.85      | 0.81   | 0.83     | 108     |
| 1        | 0.85      | 0.80   | 0.83     | 158     |
| 2        | 0.89      | 0.91   | 0.90     | 229     |
| 3        | 0.99      | 0.99   | 0.99     | 267     |
| 4        | 0.90      | 0.90   | 0.90     | 661     |
| 5        | 0.74      | 0.77   | 0.76     | 96      |
| 6        | 0.84      | 0.72   | 0.78     | 271     |
| 7        | 0.42      | 0.78   | 0.54     | 49      |
| accuracy |           |        | 0.87     | 1839    |
| macro avg | 0.81     | 0.84   | 0.82     | 1839    |
| weighted avg | 0.88  | 0.87   | 0.87     | 1839    |

Table 5.1: Classification Report obtained from applying only VGG-19

In the above classification report, we get; 'Fetal abdomen' : class 0, 'Fetal femur' : class 1, 'Fetal thorax' : class 2, 'Maternal cervix' : class 3, 'Other' : class 4, 'Trans-cerebellum' : class 5, 'Trans-thalamic' : class 6, 'Trans-ventricular' : class 7.

These accuracies point out that the model with no pre-training most accurately labels maternal cervix class. But then, the model struggled while identifying the three brain plane classes. There were also some class imbalances as fewer samples were assigned to the Trans-cerebellum and Trans-ventricular classes. Hence, there is a difference in the macro average and weighted average values. This report also shows that the model has an accuracy of 87 percent.
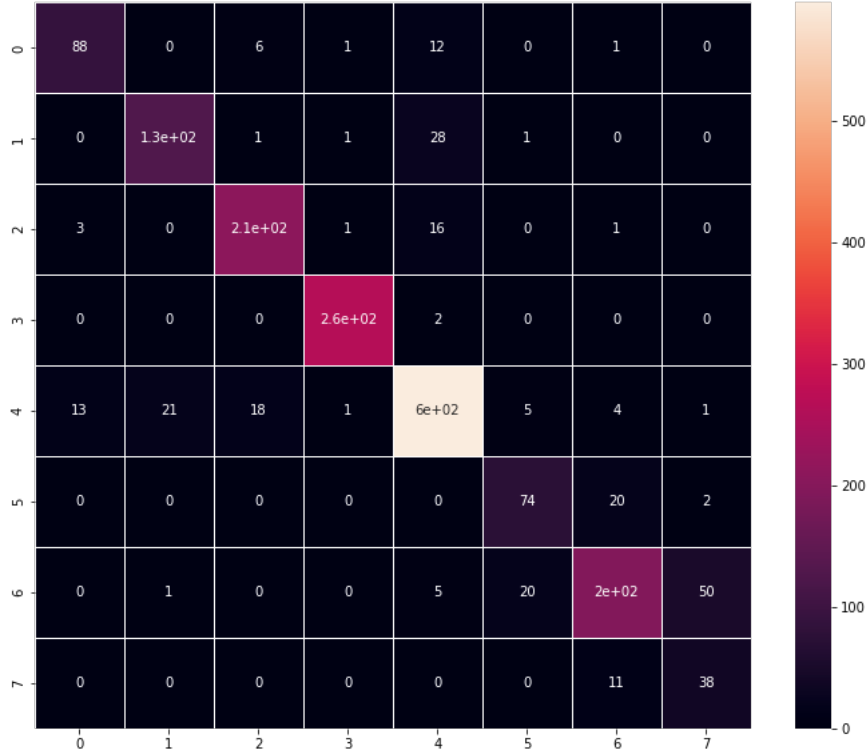


Figure 5.2: Confusion Matrix obtained from applying only VGG-19

From the above confusion matrix, we observe the amount of classes the model classified correctly and which class it mislabeled as the true positive class. We can reach a conclusion from here that, while identifying the Fetal abdomen, Fetal femur, Fetal thorax, Maternal cervix, the model often mislabeled those classes as 'Other' or an unidentified class. Even though it correctly identified the brain class but suffered to distinguish between the three common brain planes. If we take a look at the matrix, from the first row, among 108 images, 88 images were correctly labeled as fetal abdomen, 0 images were incorrectly labeled as fetal femur, 6 images were mislabeled as fetal thorax while 1 image was mislabeled as maternal cervix.

## 5.3   Performance Study Using DS Layer

**Comparison of Obtained Results for Different Prototypes**

After getting the output from the conv2D layer, the DS layer was applied and the obtained results were analyzed using 100 prototypes at first. The model used by Tong et al [24] used 200 prototypes for 60,000 images of the Cifar-10 dataset to find the distance based support from each test sample to the prototypes and construct the mass functions. But since we worked with 12,257 images, the prototype number was then reduced to observe if we could achieve better results. So the DS layer was then run using 20, 30 and 50 prototypes respectively.
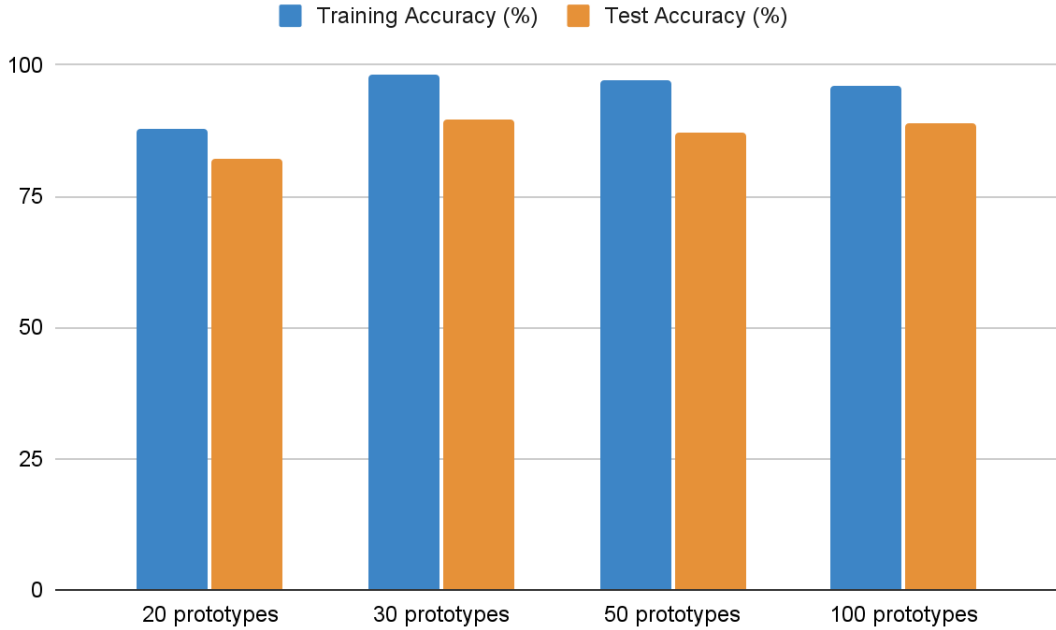


Figure 5.3: Training Vs Test Accuracy of DST based evidential model across different prototypes

With 12,257 images used from the dataset, we saw that the DS layer was trained the best when we used 30 prototypes. It gave a better test accuracy of 89.67% and 98% training accuracy. On the other hand, using 20 prototypes gave a training accuracy of 88% and test accuracy of 82% which underperformed compared to the use of 30 prototypes. Moreover, without the DS layer, using only the Conv2D model gave us

a test accuracy of 87% so the purpose of using the DS layer to get a better accuracy was not served in this case. Using 50 and 100 prototypes respectively gave better accuracy than using 20 prototypes but no better than when 30 prototypes were used. By using 50 prototypes, we got a training accuracy of 97% and test accuracy of 87% which is the same accuracy we got from using VGG-19 only. By using 100 prototypes, we got a training accuracy of 96% and test accuracy of 88.88% which is a little less than when we used 30 prototypes.

So we can say that the model gave better results when 30 prototypes were used with a total of 20,776,172 trainable parameters. If there were more images in the dataset, 30 prototypes may not have given the best accuracy. Then, a higher number of prototypes might have to be selected and the total number of trainable parameters would be increased as well. Besides, using lesser number of prototypes has been said to classify faster and cost less storage as well [4].

The obtained accuracy graphs for each of those prototypes are thus shown below:
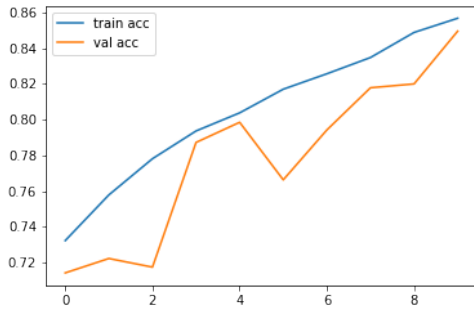


(a) Accuracy graph obtained from using 20 prototypes
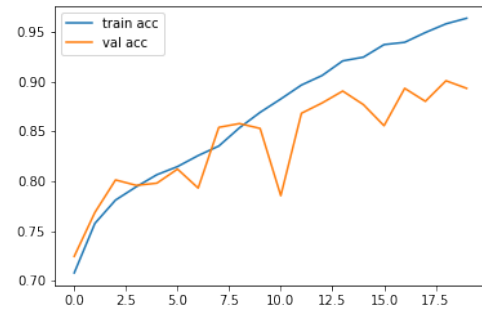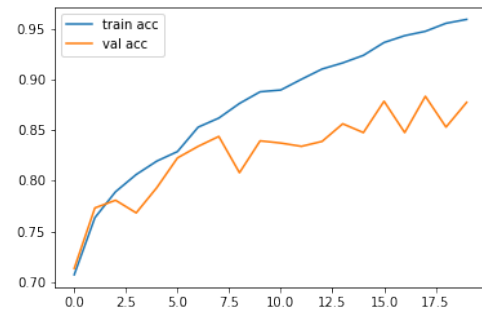
(b) Accuracy graph obtained from using 30 prototypes
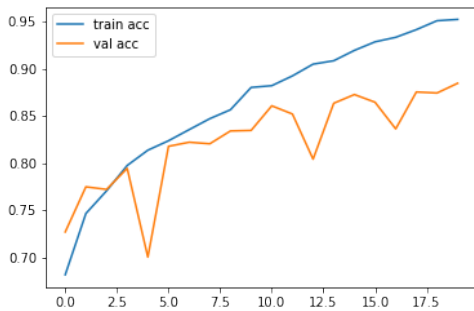


(a) Accuracy graph obtained from using 50 prototypes

(b) Accuracy graph obtained from using 100 prototypes

Figure 5.5: Accuracy graphs for different prototypes in the DS layer

We can also see the differences in the obtained classification reports for 20, 30, 50 and 100 prototypes respectively.

|         | precision | recall | f1-score | support |
|---------|-----------|--------|----------|---------|
| 0       | 0.57      | 0.88   | 0.69     | 67      |
| 1       | 0.74      | 0.86   | 0.79     | 138     |
| 2       | 0.94      | 0.80   | 0.87     | 319     |
| 3       | 0.99      | 0.98   | 0.99     | 240     |
| 4       | 0.90      | 0.89   | 0.89     | 632     |
| 5       | 0.05      | 0.60   | 0.10     | 10      |
| 6       | 0.89      | 0.61   | 0.72     | 363     |
| 7       | 0.60      | 0.71   | 0.65     | 70      |
| accuracy |          |        | 0.82     | 1839    |
| macro avg | 0.71    | 0.79   | 0.71     | 1839    |
| weighted avg | 0.88 | 0.82   | 0.84     | 1839    |

Table 5.2: Classification Report obtained from using 20 prototypes in the DS layer

|         | precision | recall | f1-score | support |
|---------|-----------|--------|----------|---------|
| 0       | 0.90      | 0.87   | 0.88     | 106     |
| 1       | 0.83      | 0.91   | 0.87     | 172     |
| 2       | 0.96      | 0.91   | 0.94     | 266     |
| 3       | 1.00      | 1.00   | 1.00     | 263     |
| 4       | 0.92      | 0.92   | 0.92     | 597     |
| 5       | 0.81      | 0.80   | 0.80     | 98      |
| 6       | 0.87      | 0.76   | 0.81     | 271     |
| 7       | 0.52      | 0.77   | 0.62     | 66      |
| accuracy |          |        | 0.89     | 1839    |
| macro avg | 0.85    | 0.87   | 0.86     | 1839    |
| weighted avg | 0.90 | 0.89   | 0.89     | 1839    |

Table 5.3: Classification Report obtained from using 30 prototypes in the DS layer

|         | precision | recall | f1-score | support |
|---------|-----------|--------|----------|---------|
| 0       | 0.92      | 0.82   | 0.87     | 119     |
| 1       | 0.75      | 0.92   | 0.83     | 134     |
| 2       | 0.94      | 0.94   | 0.94     | 247     |
| 3       | 1.00      | 0.98   | 0.99     | 241     |
| 4       | 0.92      | 0.90   | 0.91     | 636     |
| 5       | 0.59      | 0.84   | 0.69     | 80      |
| 6       | 0.86      | 0.72   | 0.78     | 299     |
| 7       | 0.61      | 0.72   | 0.66     | 83      |
| accuracy |          |        | 0.87     | 1839    |
| macro avg | 0.83    | 0.86   | 0.84     | 1839    |
| weighted avg | 0.88 | 0.87   | 0.88     | 1839    |

Table 5.4: Classification Report obtained from using 50 prototypes in the DS layer

|          | precision | recall | f1-score | support |
|----------|-----------|--------|----------|---------|
| 0        | 0.92      | 0.89   | 0.90     | 101     |
| 1        | 0.83      | 0.83   | 0.83     | 152     |
| 2        | 0.96      | 0.92   | 0.94     | 273     |
| 3        | 0.99      | 1.00   | 0.99     | 251     |
| 4        | 0.92      | 0.94   | 0.93     | 614     |
| 5        | 0.90      | 0.77   | 0.83     | 131     |
| 6        | 0.84      | 0.76   | 0.80     | 270     |
| 7        | 0.38      | 0.72   | 0.50     | 47      |
| accuracy |           |        | 0.89     | 1839    |
| macro avg | 0.84     | 0.85   | 0.84     | 1839    |
| weighted avg | 0.90  | 0.89   | 0.89     | 1839    |

Table 5.5: Classification Report obtained from using 100 prototypes in the DS layer

The classification report in table 5.2 shows that by using 20 prototypes, the model struggled to identify label 5 that is, 'Trans-cerebellum' of a Fetal Brain, resulting in an f1-score of 0.1 only. Using 30 prototypes (Table: 5.3), the f1-score was 0.8, 0.69 using 50 prototypes (Table: 5.4) and 0.83 using 100 prototypes (Table: 5.5). But in case of class 7, that is the 'Trans-ventricular' of a Fetal Brain, using 100 prototypes gave an f1-score of 0.5 (Table: 5.5) only while 30 prototypes gave 0.62 (Table: 5.3). Overall, using 30 prototypes gave a better classification report than using other prototypes.

Then, we analyzed the classification report obtained from using 30 prototypes in the DS layer with respect to the report obtained from using only VGG-19. We see that for all classes, the f1 score is more in case of using DST with VGG-19 than using just VGG-19. By using 30 prototypes, for class 3, i.e. the 'Maternal Cervix' class, we got a perfect score that is, 1.00 (Table: 5.3) which is the best result amongst all the classes. By using only VGG-19, the f1 score for class 3 was 0.99 (Table: 5.1) so we saw a slight improvement.

A comparative analysis of the confusion matrices by using 20, 30, 50 and 100 prototypes are also given below.
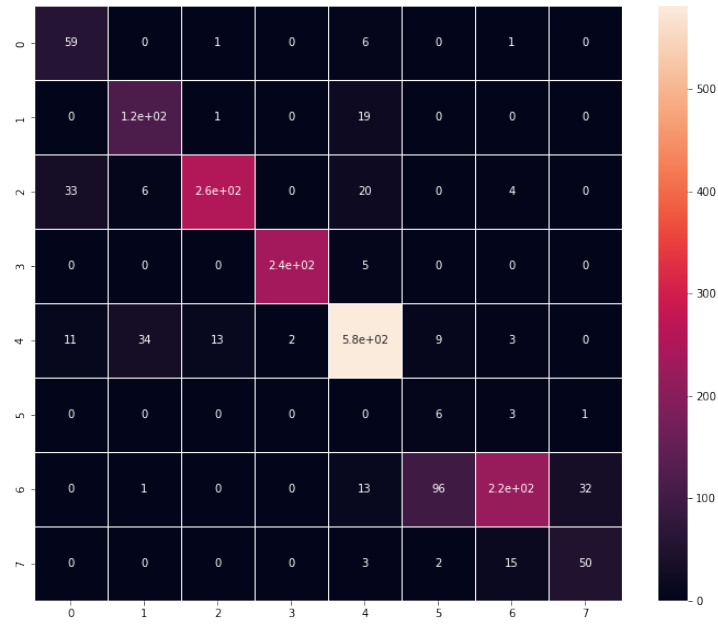
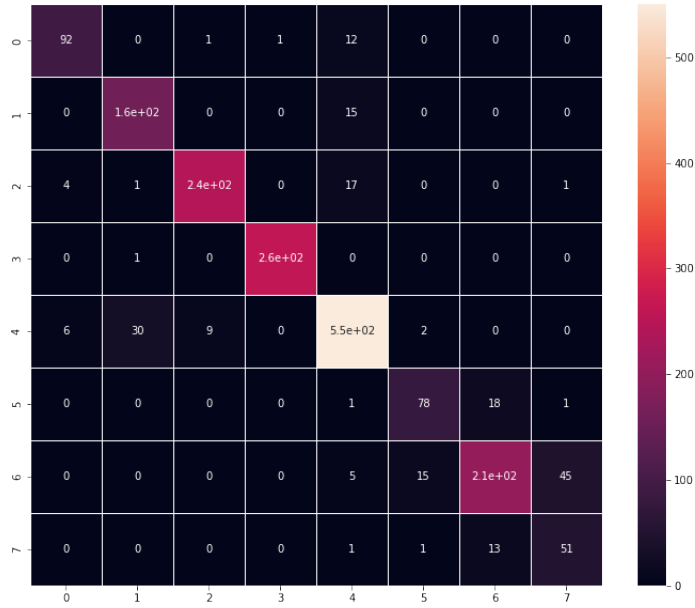Figure 5.6: Confusion matrix obtained from using 20 prototypes in the DS layer



Figure 5.7: Confusion matrix obtained from using 30 prototypes in the DS layer
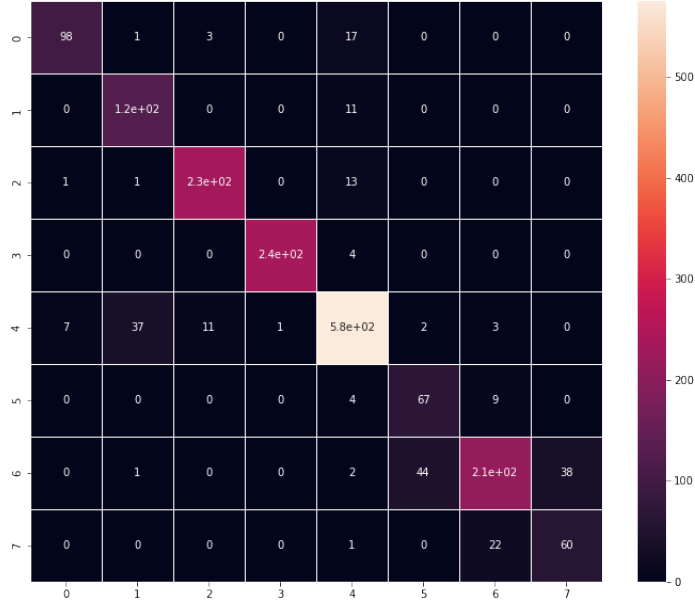
27

Figure 5.8: Confusion matrix obtained from using 50 prototypes in the DS layer
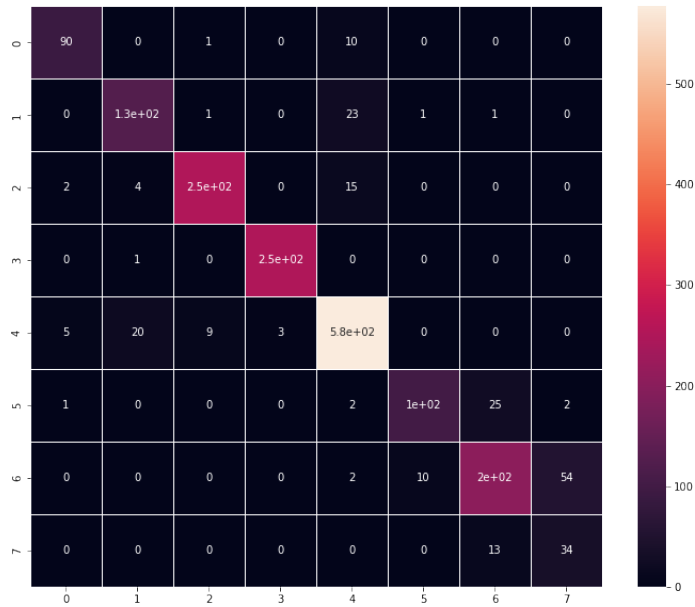


Figure 5.9: Confusion matrix obtained from using 100 prototypes in the DS layer

From these confusion matrices, we were able to identify the number of samples that were being mislabeled as the True Positive class. In all cases of using different

prototypes, the fetal brain planes were most likely to be misclassified. But for 30 prototypes, the confusion matrix (Figure 5.7) showed comparatively better results for identifying the fetal brain planes. We can see that out of 98 images belonging to the 'Trans-cerebellum' class of fetal brain, 78 images were correctly identified as 'Trans-cerebellum' whereas 18 images were mislabeled as 'Trans-thalamic' and 1 image was incorrectly labelled as 'Trans-ventricular' class. After analyzing the obtained matrices, classification reports and accuracy graphs, we were able to conclude that using 30 prototypes gave the comparatively better result for our model. So we implemented the utility layer using 30 prototypes.

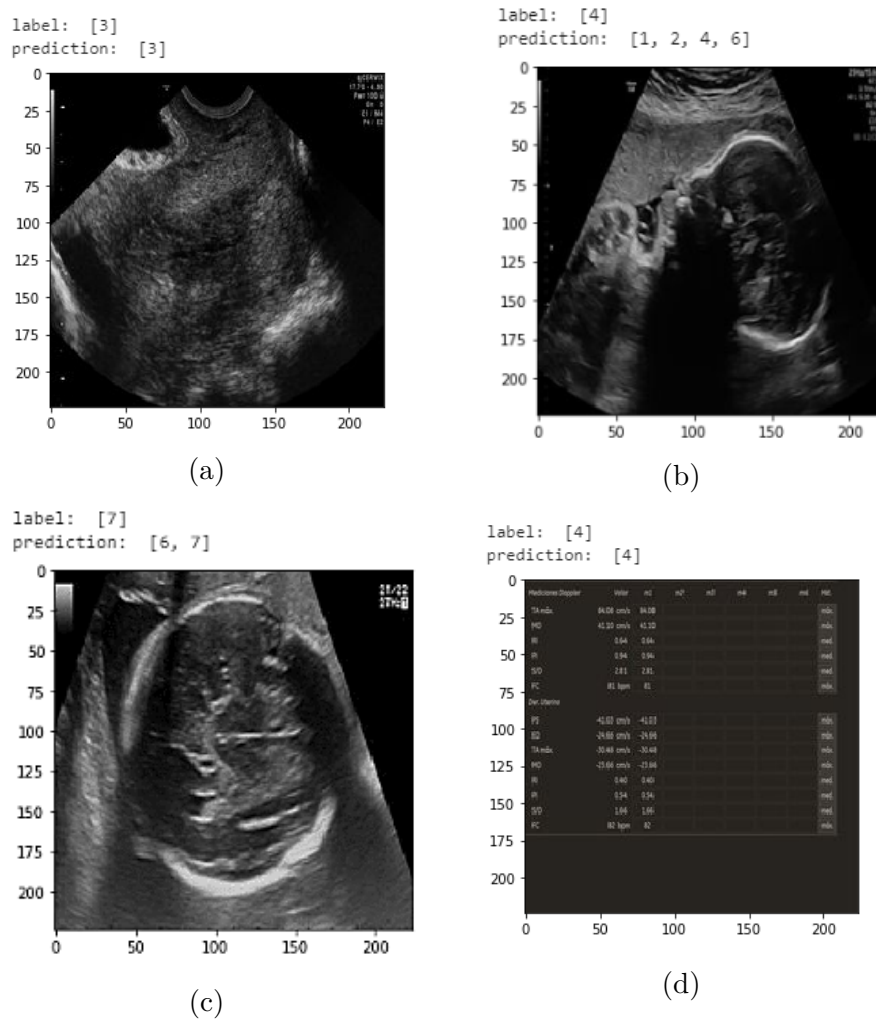## 5.4 Classification Results from Utility Layer



(a)

(b)

(c)

(d)

Figure 5.10: Original label and DST based prediction

We get a set of predictions for classifying an image after constructing the utility layer. For cases with precise classification, the sample image has been assigned to a single set and not multi class sets. This is because there is a certainty in the model that this image belongs to one particular class, as it has been depicted in Figure 5.10a where the fetal anatomical plane belongs to that of the 'Maternal cervix' class.

Complete outliers have been labeled and predicted as label 4 which is the class 'Other' of our dataset as such in Figure 5.10d. So any images that have no resemblance to the other seven classes i.e. the Fetal abdomen (Label 0), Fetal femur (Label 1), Fetal thorax (Label 2), Maternal cervix (Label 3), the three classes of the fetal brain i.e. Trans-cerebellum (Label 5), Trans-thalamic (Label 6) and Trans-ventricular (Label 7) are being classified into the label, 'Other' (Label 4). In case of previous classifiers, complete outliers may be assigned an empty set. In the case of our dataset, since there is a class named 'Other', outliers may be classified in this instead of being assigned an empty set.

For cases where there is uncertainty, set valued classification was performed where the image has been classified to a multi class set. That means, partially, multi class acts have been assigned to that image. For example, for the Figure 5.10b, the sample image creates a confusion about the label it may belong to. In such cases, a normal probabilistic model would not have given us enough information about the prediction of the model. But in this case, we get a set of predicted labels, [1 ,2 ,4 ,6]. This indicates that the image belongs to the class or label, 'Other' which is label 4, but there is a possibility of the model predicting the image as label 1, 2 or 6 as well. In case of Figure 5.10c, the fetal plane belongs to that of 'Trans-ventricular' part of a fetal brain class i.e. label 7 but there is a confusion faced by the model with the 'Trans-thalamic' part of a fetal brain, i.e. class 6. Thus, the utility layer gives a set of possible predictions by the model that contains both label 6 and 7. This is an indicator of the uncertainty that a model could face while identifying the fetal anatomical planes and thus, we are able to obtain more information using an evidential classifier.

## 5.5   Comparative Study

After computing the DS layer with 30 prototypes, we see a significant difference between the accuracy from using a conventional CNN classifier and a DST based Evidential Classifier. Previously, the accuracy obtained from VGG-19 was 87%. And after applying the Dempster-Shafer layer, the accuracy significantly increased to 89% due to the conversion of each evidence obtained from the high dimensional features of the CNN layer into mass functions and combining those evidences in the DS layer.
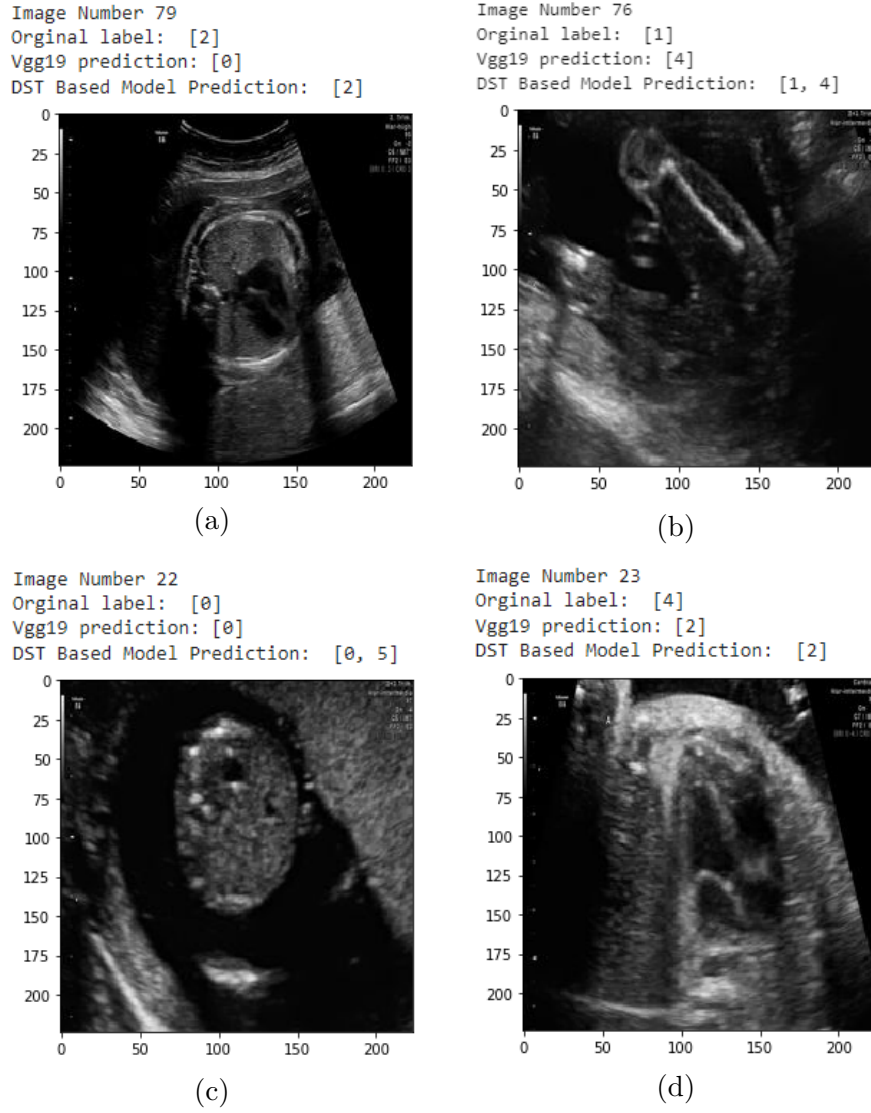
Figure 5.11: Original Label, VGG-19 and DST based prediction

Although in most cases, VGG-19 and the E-CNN model gave us the accurate classification, we have also observed cases in our result where the DST based evidential model was able to predict a sample correctly but VGG-19 did not give the accurate prediction. In Figure 5.11a, we see that VGG-19 incorrectly misclassified the image as label 0 i.e. 'Fetal Abdomen' but DST based model was able to classify it into label 2 i.e. 'Fetal Thorax'. In Figure 5.11b, the correct label for the image is 1 which is the 'Fetal Femur' class. But VGG-19 placed the image in the 'Other' class which is label 4 whereas our DST based model gave a set consisting of both label 1 and 4 as it found evidence of the image belonging to 'Fetal Femur' as well as 'Other'. Thus, it did not reject the probability of the image belonging to the 'Other' class.

There were also cases where VGG-19 gave an accurate prediction but the E-CNN model gave a set of classes that contained classes other than the actual label as the model found similarities of the sample image with the features of that class as well. In the case of Figure 5.11c, this occurred as the DST based E-CNN model did not reject the probability that the sample might belong to label 5 and also kept the

actual label, i.e. label 0 in the set of predicted classes.

There are also a few rare cases where both VGG-19 and DST incorrectly classified the image as such in the case of Figure 5.11d or cases where VGG-19 might have accurately predicted the class but the DST based model failed to provide sufficient information. This may have occurred due to the DS layer clustering evidence from features found more relevant to other classes than the original class label.

# Chapter 6

# Future Work and Conclusion

## 6.1 Future Work

In the future, we would also like to incorporate the results of some other models like ResNet, and Inception to incorporate those as evidence to the Dempster-Shafer layer and create an evidential fusion model. Additionally, we would also like to implement Explainable AI so that it helps us understand the predictions and results obtained from the classifier better and also help interpret those results and the causes behind them. We plan to incorporate any existing XAI architecture like LIME (Local Interpretable Model-agnostic Explanations) or create our own XAI architecture from the existing utility layer of our current model. Using these techniques, we want to be able to explain the reason behind the reason for getting better accuracy while using a DS layer after a CNN layer than a conventional CNN model. We also plan on collecting an ultrasound fetal image dataset from Bangladeshi hospitals and work with that on our model.

## 6.2 Conclusion

Major malformations of the fetal head, abdominal wall and of the placenta and umbilical cord can be detected as early as during the first ten weeks of pregnancy and so it is absolutely crucial to have dependable resources to do so. Be it the ultrasound images having to go through several specialists for the fetal planes to be somewhat correctly identified or having varying intensities of speckle-noise or acoustic shadows, ultrasound still possesses many uncertainties albeit being widely used and lucrative. With our research, we propose a model that will plug DS with CNN to improve the overall performance by taking into account multiple pieces of evidence and parameters to multi-class sets. Additionally, the DS layer gives us a resulting output on the level of uncertainty that we may get on our obtained results. We have already obtained a better result using the DS layer along with VGG-19 than just using VGG-19. We have analyzed cases which might give us the best results and were thus able to obtain a better result than that of previously used classifiers. We also performed set valued classification which allowed us to give more information on the models predictions. In this way, not only the data uncertainty but also the uncertainty of the model was demonstrated. This is especially significant for US images as often, fetal planes can be confusing and may be misclassified. Thus, more information regarding the prediction may help prevent mislabeling.

# Bibliography

[1]  G. Shafer, *A mathematical theory of evidence.* Princeton university press, 1976, vol. 42.

[2]  F. P. Hadlock, R. Harrist, R. S. Sharman, R. L. Deter, and S. K. Park, "Estimation of fetal weight with the use of head, body, and femur measurements—a prospective study," *American journal of obstetrics and gynecology*, vol. 151, no. 3, pp. 333–337, 1985.

[3]  S.-Y. J. Chen, W.-C. Lin, and C.-T. Chen, "Medical image understanding system based on dempster-shafer reasoning," in *Medical Imaging V: Image Processing*, SPIE, vol. 1445, 1991, pp. 386–397.

[4]  T. Denœux, "A neural network classifier based on dempster-shafer theory," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 30, pp. 131–150, Apr. 2000. DOI: 10.1109/3468.833094.

[5]  K. Sentz and S. Ferson, "Combination of evidence in dempster-shafer theory," 2002.

[6]  S. B. Chaabane, M. Sayadi, F. Fnaiech, and E. Brassart, "Color image segmentation based on dempster-shafer evidence theory," in *MELECON 2008-The 14th IEEE Mediterranean Electrotechnical Conference*, IEEE, 2008, pp. 862–866.

[7]  L. J. Salomon, Z. Alfirevic, V. Berghella, C. Bilardo, E. Hernandez-Andrade, S. Johnsen, K. Kalache, K.-Y. Leung, G. Malinger, H. Munoz, *et al.*, "Practice guidelines for performance of the routine mid-trimester fetal ultrasound scan," *Ultrasound in Obstetrics & Gynecology*, vol. 37, no. 1, pp. 116–126, 2011.

[8]  M. Shoyaib, M. Abdullah-Al-Wadud, and O. Chae, "A skin detection approach based on the dempster–shafer theory of evidence," *International Journal of Approximate Reasoning*, vol. 53, no. 4, pp. 636–659, 2012.

[9]  U. Sovio, I. R. White, A. Dacey, D. Pasupathy, and G. C. S. Smith, "Screening for fetal growth restriction with universal third trimester ultrasonography in nulliparous women in the pregnancy outcome prediction (POP) study: A prospective cohort study," *The Lancet*, vol. 386, no. 10008, pp. 2089–2097, Nov. 2015. DOI: 10.1016/s0140-6736(15)00131-2. [Online]. Available: https://doi.org/10.1016/s0140-6736(15)00131-2.

[10]  J. J. Cerrolaza, O. Oktay, A. Gomez, J. Matthew, C. Knight, B. Kainz, and D. Rueckert, "Fetal skull segmentation in 3d ultrasound via structured geodesic random forest," in *Fetal, Infant and Ophthalmic Medical Image Analysis*, Springer, 2017, pp. 25–32.

[11]   B. Gao and L. Pavel, "On the properties of the softmax function with application in game theory and reinforcement learning," *arXiv preprint arXiv:1704.00805*, 2017.

[12]   M. A. Maraci, C. P. Bridge, R. Napolitano, A. Papageorghiou, and J. A. Noble, "A framework for analysis of linear ultrasound videos to detect fetal presentation and heartbeat," *Medical image analysis*, vol. 37, pp. 22–36, 2017.

[13]   Z. Yu, E.-L. Tan, D. Ni, J. Qin, S. Chen, S. Li, B. Lei, and T. Wang, "A deep convolutional neural network-based framework for automatic fetal facial standard plane recognition," *IEEE journal of biomedical and health informatics*, vol. 22, no. 3, pp. 874–885, 2017.

[14]   J. J. Cerrolaza, M. Sinclair, Y. Li, A. Gomez, E. Ferrante, J. Matthew, C. Gupta, C. L. Knight, and D. Rueckert, "Deep learning with ultrasound physics for fetal skull segmentation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 564–567.

[15]   T. L. van den Heuvel, D. de Bruijn, C. L. de Korte, and B. v. Ginneken, "Automated measurement of fetal head circumference using 2d ultrasound images," *PloS one*, vol. 13, no. 8, e0200412, 2018.

[16]   T. Denœux, "Logistic regression, neural networks and dempster–shafer theory: A new perspective," *Knowledge-Based Systems*, vol. 176, pp. 54–67, 2019.

[17]   R. Qu, G. Xu, C. Ding, W. Jia, and M. Sun, "Deep learning-based methodology for recognition of fetal brain standard scan planes in 2d ultrasound images," *Ieee Access*, vol. 8, pp. 44 443–44 451, 2019.

[18]   Z. Sobhaninia, S. Rafiei, A. Emami, N. Karimi, K. Najarian, S. Samavi, and S. R. Soroushmehr, "Fetal ultrasound image segmentation for measuring biometric parameters using multi-task deep learning," in *2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, IEEE, 2019, pp. 6545–6548.

[19]   P. Sridar, A. Kumar, A. Quinton, R. Nanan, J. Kim, and R. Krishnakumar, "Decision fusion-based fetal ultrasound image plane classification using convolutional neural networks," *Ultrasound in medicine & biology*, vol. 45, no. 5, pp. 1259–1273, 2019.

[20]   Z. Tong, P. Xu, and T. Denoeux, "Convnet and dempster-shafer theory for object recognition," in *International Conference on Scalable Uncertainty Management*, Springer, 2019, pp. 368–381.

[21]   X. P. Burgos-Artizzu, D. Coronado-Gutiérrez, B. Valenzuela-Alcaraz, E. Bonet-Carne, E. Eixarch, F. Crispi, and E. Gratacós, "Evaluation of deep convolutional neural networks for automatic classification of common maternal fetal ultrasound planes," *Scientific Reports*, vol. 10, no. 1, pp. 1–12, 2020.

[22]   E. L. Skeika, M. R. Da Luz, B. J. T. Fernandes, H. V. Siqueira, and M. L. S. C. De Andrade, "Convolutional neural network to detect and measure fetal skull circumference in ultrasound imaging," *IEEE Access*, vol. 8, pp. 191 519–191 529, 2020.

[23] T. Folke, S. C.-H. Yang, S. Anderson, and P. Shafto, "Explainable ai for medical imaging: Explaining pneumothorax diagnoses with bayesian teaching," in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III*, SPIE, vol. 11746, 2021, pp. 644–664.

[24] Z. Tong, P. Xu, and T. Denoeux, "An evidential classifier based on dempster-shafer theory and deep learning," *Neurocomputing*, vol. 450, pp. 275–293, 2021.

[25] M. Tripathi, *Image processing using cnn: Beginner's guide to image processing*, Jun. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/image-processing-using-cnn-a-beginners-guide/.

[26] J. Yin, J. Li, Q. Huang, Y. Cao, X. Duan, B. Lu, X. Deng, Q. Li, and J. Chen, "Ultrasonographic segmentation of fetal lung with deep learning," *Journal of Biosciences and Medicines*, vol. 9, no. 1, pp. 146–153, 2021.

[27] *Precision and recall in machine learning - javatpoint*. [Online]. Available: https://www.javatpoint.com/precision-and-recall-in-machine-learning.