MOTIVATION

Ultrasound imaging is a cornerstone in prenatal care, offering patients a way of developing a safe fetus. Its pivotal role in detecting fetal anomalies cannot be overstated. However, the accurate interpretation and diagnosis of fetal anomalies in these images is heavily reliant on the skill and experience of the sonographer, which can lead to variability in diagnoses. The challenge intensifies when identifying specific anatomical structures like the abdomen, thorax, brain, and femur, which are crucial in the early detection and management of fetal abnormalities. The integration of deep learning in the classification of these structures in 2D fetal ultrasound images promises to enhance the accuracy, consistency, and efficiency of prenatal diagnostics, ultimately improving fetal health outcomes.

ABSTRACT

This report presents the development of a deep learning-based model designed to classify key anatomical structures in 2D fetal ultrasound images: the abdomen, thorax, brain, and femur. Ultrasound imaging, being the most prevalent method for prenatal anomaly detection, demands high accuracy in identifying specific fetal planes for correct diagnosis. My model aims to mitigate this by providing a reliable and consistent means of classification. I leveraged advanced deep learning techniques to train our classifier, ensuring robust performance across diverse datasets. The model's effectiveness is evaluated using the given dataset of fetal ultrasound images, demonstrating its potential in enhancing the diagnostic process in prenatal care.

INTRODUCTION

Ultrasound imaging stands as a fundamental tool in prenatal care, providing critical insights into fetal development. Its unparalleled safety profile and widespread accessibility make it the preferred choice for monitoring fetal health and detecting potential anomalies. Despite its advantages, the interpretation of ultrasound images largely depends on the expertise of the sonographer, which can lead to variations in diagnostic accuracy. This is particularly true when identifying specific anatomical structures, such as the abdomen, thorax, brain, and femur, that are vital for assessing fetal well-being and detecting abnormalities. In recent years, deep learning has emerged as a transformative force in medical imaging, offering new avenues for enhancing image analysis and interpretation. Its ability to learn from large datasets and identify complex patterns makes it an ideal candidate for application in fetal ultrasound imaging. By training a deep learning model to classify these key anatomical structures, we can standardise the interpretation process, reduce variability in diagnoses, and support clinicians in making more informed decisions.

DATASET

- **Image_name**: Image of fetus.
- **Plane**: label class.

DATA PREPROCESSING

The primary goal of data preprocessing was to prepare a robust and well-structured dataset for training a machine learning model. This involved several steps to ensure data integrity, address class imbalances, and enhance the dataset's diversity and representativeness.

The following steps were done to perform data handling and processing

1. DataFrame Creation for Image File Paths: Creating a DataFrame that contains the paths to various image files streamlines the management and analysis of our extensive dataset, making it more accessible and organised.

2. Dataset Exploration with .describe()**:** The .describe() function was employed to gain statistical insights into the dataset. This helped understanding key aspects, such as the count, unique values, and frequency of labels, which was crucial for informed decision-making in later stages.

3. Integrity Checks - Null Value Analysis in Label Column: Data integrity is paramount. A check was conducted for null values in the label column, confirming the completeness and reliability of the label information provided with our images.

4. Class-wise Data Segregation: Recognizing the importance of targeted analysis, images are separated based on four different classes.

5. Label Distribution and Imbalance Identification: An in-depth exploration of label distribution was carried out. This step was crucial to identify potential class imbalances, which could bias our model if unaddressed.

6. Addressing Class Imbalance through Oversampling: To mitigate the effects of class imbalance, oversampling techniques were employed. This ensured a more equitable representation of each class in the training process, crucial for the model’s unbiased performance.

7. Mask Analysis in Images: A thorough analysis of image masks was conducted. This provided valuable insights into the presence and characteristics of masks within the dataset, which could be a significant factor in the model's performance.

8. Image Resizing and Normalisation Standardisation of image data by resizing all images to a uniform size and normalising the pixel values to a [0, 1] range.

9. Data Augmentation with ImageDataGenerator: Utilising the ImageDataGenerator, I augmented the dataset by introducing various transformations like rotations, shifts, shears, zooms, and horizontal flips. This augmentation is key to enhancing the model's generalisation capability.

10. Label Encoding for Class Labels: Applied label encoding to transform the class labels into a numeric format. This conversion is necessary for compatibility with most machine learning models.

METHODS

The project employs deep learning modelling techniques to build models capable of classifying anatomical structure in 2D ultrasound images. CNN and advanced CNN models like Resnet101 are employed to build models providing such classifying capabilities.

Method 1 : Simple CNN Model

In the first methodological approach, A convolutional neural network (CNN) named simple_model is used specifically designed for image classification tasks.

Architecture Description and Reason for selection:

The simple_model begins with a Convolutional 2D layer, featuring 32 filters of size 3x3. This layer, activated by the rectified linear unit (ReLU) function, serves as the entry point for feature extraction. It is designed to handle input images of 224x224 pixels in grayscale, indicated by the input shape (224, 224, 1). Following the initial convolutional layer, the model integrates a MaxPooling 2D layer with a 2x2 pooling size. The role of this layer is to downsample the spatial dimensions of the feature maps, which not only aids in reducing computational load but also focuses on the most significant features extracted by the convolutional layer. The architecture then deepens with the addition of another Convolutional 2D layer. This layer increases the filter count to 64 while retaining the 3x3 kernel size, and continues to use the ReLU activation function. This layer builds upon the feature extraction process, further enhancing the model's ability to recognize more complex patterns in the images. Another MaxPooling 2D layer follows suit, performing a similar function to its predecessor, further reducing the dimensions and emphasising crucial features. The network transitions from feature extraction to classification through the inclusion of a Flatten layer. This layer converts the 2D feature maps into a 1D array, a necessary step before feeding the data into fully connected layers. Finally, a Dense layer with 128 neurons and 4 neurons is introduced, maintaining the use of the ReLU activation function. This layer adds to the model's ability to learn non-linear relationships in the data.

This model is selected for its ability to effectively process image data through a series of convolutional and pooling layers, coupled with its capability to classify images into multiple categories through dense layers. This architecture is optimised for performance, balancing the need for depth to capture complex patterns with the necessity for computational efficiency, making it highly suitable for image classification tasks.

LOSS FUNCTION AND OPTIMIZER

In the development of these, careful consideration was given to the choice of loss function and optimizer, two critical components influencing the model's training process.

1. Spatial Cross Entropy Loss:

- Spatial Cross Entropy Loss was chosen as the loss function for this task. This loss is particularly well-suited for image classification problems. Unlike traditional cross-entropy loss, spatial cross entropy takes into account the spatial dimensions of the predicted and ground truth tensors. It accounts for pixel-wise classification errors, making it especially effective when dealing with tasks such as semantic segmentation or dense prediction.

2. Adam Optimizer:

- The optimization algorithm used during training was Adam. Adam (short for Adaptive Moment Estimation) is an advanced stochastic optimization algorithm that combines the benefits of both momentum and RMSprop. Adam dynamically adjusts the learning rates for each parameter, allowing for adaptive and efficient learning. It maintains separate learning rates for each parameter and adaptively updates them based on the first and second moments of the gradients.

OTHER METHOD

Method 2 : Complex CNN Model

The advanced approach in image classification is embodied in a complex convolutional neural network, designed to meticulously capture and analyse intricate patterns within grayscale images for a multi-class classification task. This model, an amalgamation of various layers and techniques, is tailored to maximise feature extraction and minimise overfitting, thereby enhancing the overall efficacy of the classification process.

Architecture Description:

The model commences with an input layer that features a Convolutional 2D (Conv2D) layer. This layer, equipped with 32 filters of a 3x3 kernel size, is the primary gateway for the model to process the incoming grayscale images of 224x224 pixels. Activation within this layer is handled by the Rectified Linear Unit (ReLU), a function chosen for its effectiveness in introducing non-linearity and aiding in learning complex patterns. Immediately succeeding the convolutional layer, the architecture incorporates batch normalisation, following is the MaxPooling2D layer with a 2x2 window, which serves to reduce the spatial dimensions of the derived feature maps, consequently emphasising the most significant features while maintaining computational efficiency. To further refine the model and guard against overfitting, a Dropout layer with a rate of 0.25 is strategically positioned after each MaxPooling2D layer. This layer randomly disables a proportion of neurons during training, compelling the network to learn more diverse and robust features, and not overly rely on any specific neuron set. The architecture deepens with the inclusion of stacked convolutional blocks. These blocks, composed of successive Conv2D layers (first with 64 filters and then 128 filters, both maintaining a 3x3 kernel size), are designed to progressively extract more complex features from the images. Each block is a comprehensive unit, equipped with ReLU activation, batch normalisation, MaxPooling2D, and Dropout, ensuring a thorough and nuanced feature extraction process.

Method 3 : Resnet101 Model

The methods also include leveraging a pretrained ResNet, a renowned deep learning architecture known for its efficiency and accuracy in image classification tasks. Recognizing the potential of this advanced model, we adapted and fine-tuned it to suit our specific requirement of classifying images into four distinct categories. This process involved modifying the ResNet to align with our dataset and training objectives, optimising it to capture the patterns of unique image data.

Architecture Description:

The fine-tuning process was meticulously strategized. I began by modifying ResNet's final layers to align with our four-class classification schema. This was crucial since the original model was tailored for a different class set. We then selectively fine-tuned the model, freezing the initial layers to preserve the general features learned previously while allowing the later layers to adjust to our dataset. Additional layers are added to capture features for our image data.

RESULT AND PERFORMANCE ANALYSIS

This section presents the results of an image classification task, where I evaluated three distinct models: a Simple CNN, a Complex CNN, and a fine-tuned ResNet-101. The performance of each model was assessed based on their accuracy on the test dataset extracted using train_test_split.

1. Simple Convolutional Neural Network (CNN): The Simple CNN model, characterised by its straightforward yet effective architecture, achieved a notable accuracy of 0.78 on the test dataset. This performance underscores the model's capability to efficiently process and classify images, even with a relatively less complex structure. The result is particularly impressive, considering the model's balance between depth and computational efficiency.

2. Complex Convolutional Neural Network (CNN): The Complex CNN, designed to capture more intricate patterns and features within the image data, registered an accuracy of 0.73 on the test dataset. While slightly lower than the Simple CNN, this result is indicative of the model's deeper and more nuanced approach to image classification.

3. Fine-tuned ResNet-101: The standout performer in our tests was the fine-tuned ResNet-101, which achieved an impressive accuracy of 0.79 on the test dataset. This superior performance can be attributed to the model's depth, sophisticated architecture, and the fine-tuning process, which tailored the pre-trained ResNet-101 to our specific classification task.

FUTURE WORKS

The implementation of a deep learning-based model for classifying anatomical structures in 2D fetal ultrasound images lays a solid foundation for accurate detection and diagnosis. The successful implementation of a deep learning-based model for classifying anatomical structures in 2D fetal ultrasound images opens the door to several avenues for future enhancements and research. The following are key areas to explore for the continuous improvement and extension of the model:

- **Multi-Stage Classification:** Consider expanding the model to perform multi-stage classification. Instead of classifying entire images, break down the task into hierarchical stages, where the model first identifies regions of interest and then classifies anatomical structures within those ROIs.
- **Integration of Clinical Context:** Incorporate clinical context into the classification model. Explore ways to integrate patient-specific information, gestational age, or other clinical metadata to enhance the model's understanding of individual cases and improve classification accuracy.
- **Transfer Learning Capabilities:** Assess the model's performance across different ultrasound machines and settings. Investigate transfer learning techniques to adapt the model to variations in image.

- **Handling Variability in Fetal Development:** Explore strategies to handle variability in fetal development across gestational ages. Fine-tune the model to adapt to different stages of pregnancy.
- **Data Augmentation:** Implement advanced data augmentation techniques, especially in scenarios with limited annotated data. Techniques like generative adversarial networks (GANs) or unsupervised domain adaptation may help augment the dataset.