CONVENTIONAL NEURAL NETWORK AND DISCRETE WAVELET TRANSFORM FOR REDUCED RADIATION EXPOSURE IN X-RAY IMAGE

[1]A.HASEENATH JASEELA [2]MARRY LEO SHERLIN.W
Assistant ProfessorAssistant Professor
[1,2]Department of Computer Science,
Loyola Institute of Technology & ScienceLoyola Institute of Technology & Science

ABSTRACT

The detection of fetal brain abnormalities is a critical step in prenatal care, as early detection and diagnosis can lead to better outcomes for both the mother and the fetus. Currently, fetal brain abnormalities are primarily detected using ultrasound imaging, which is limited in its ability to provide detailed information about the developing brain. In this work, we propose a machine learning-based approach for the detection and classification of fetal brain abnormalities using xrayimages. The proposed approach provides a concise overview of the proposed methodology focused on the integration of Convolutional Neural Networks (CNN) and Discrete Wavelet Transform (DWT) for noise removal and reduced radiation exposure in X-ray imaging. The study aims to enhance the quality and safety of medical imaging by leveraging the complementary strengths of CNNs and DWT. CNNs are employed for automatic feature learning, enabling effective noise reduction, while DWT contributes multiresolution analysis to preserve crucial image details. The hybrid approach not only addresses the challenges of noise removal but also targets artifact suppression induced by radiation exposure. Furthermore, the research emphasizes the importance of adaptive filtering techniques within CNN-DWT architectures to accommodate variations in noise patterns. Evaluation metrics such as signal-to-noise ratio (SNR) and structural similarity index (SSI) are utilized for validation, demonstrating the efficacy of the proposed methodology. The ultimate goal is to achieve reduced radiation doses in X-ray imaging while maintaining or improving diagnostic accuracy, thereby advancing patient safety. Despite notable progress, the abstract acknowledges existing challenges, including the interpretability of deep models and the need for standardized evaluation protocols. Future directions involve addressing these challenges and further refining the hybrid CNN-DWT approach for real-time applications in clinical settings.

INTRODUCTION

Detection of fetal brain abnormalities is a crucial aspect of prenatal care, as early detection and diagnosis can have a significant impact on the health and well-being of both the mother and the fetus. Currently, the most commonly used method of detecting these abnormalities is through ultrasound imaging. However, this method has limitations in providing detailed information about the developing brain, leading to a need for more accurate and efficient methods.

To address this issue, a new approach using deep learning techniques has been proposed. This approach utilizes a Convolutional Neural Network (CNN) trained on a large dataset of annotated ultrasound images of the fetal brain. The CNN is designed to detect and classify abnormalities in real-time, providing a faster and more accurate method of identifying potential issues.Experiments conducted on a diverse dataset demonstrate the effectiveness of this approach, with high accuracy and sensitivity in detecting fetal brain abnormalities. The use of deep learning techniques in this application has the potential to greatly improve prenatal care and lead to better health outcomes for both the mother and the fetus.

The proposed deep learning-based approach has several advantages over traditional methods. Firstly, it is able to detect and classify abnormalities much faster, as the CNN is able to analyze large amounts of data in real-time. Secondly, it is able to provide more detailed and accurate information about the developing fetal brain, which can lead to early detection and diagnosis of potential problems.

The use of deep learning techniques for the detection and classification of fetal brain abnormalities has the potential to greatly improve prenatal care and lead to better health outcomes for both the mother and the fetus. The proposed approach using a Convolutional Neural Network trained on annotated ultrasound images demonstrates high accuracy and sensitivity in detecting abnormalities, making it a promising tool for improving prenatal care in the future.

OBJECTIVES

The main objective of detecting and classifying fetal brain abnormalities is to accurately diagnose and understand the conditions affecting the fetal brain during pregnancy. This early detection and diagnosis can have a significant impact on the health and well-being of both the mother and the fetus. With the help of medical imaging technologies and advanced algorithms, doctors can identify any abnormal development of the fetal brain and take necessary actions to ensure the best outcomes for both mother and baby.

To achieve this objective, researchers are working on developing a reliable and efficient solution for detecting and classifying fetal brain abnormalities. The solution should be able to analyze medical images, extract relevant features, and classify the images into different categories of abnormalities. The solution should also be able to accurately detect subtle differences in the images and provide a diagnosis that is both accurate and consistent.

The proposed solution should consider the unique characteristics of fetal brain imaging, including the fact that the images are often low resolution and may contain noise or artifacts. It should also take into account the sensitivity and specificity requirements of this application and prioritize the reduction of false positive and false negative diagnoses.

By using a combination of image processing techniques and machine learning algorithms, the solution should be able to accurately detect and classify fetal brain abnormalities. The solution should be validated through rigorous testing and experimentation, and its performance should be compared to that of existing solutions.

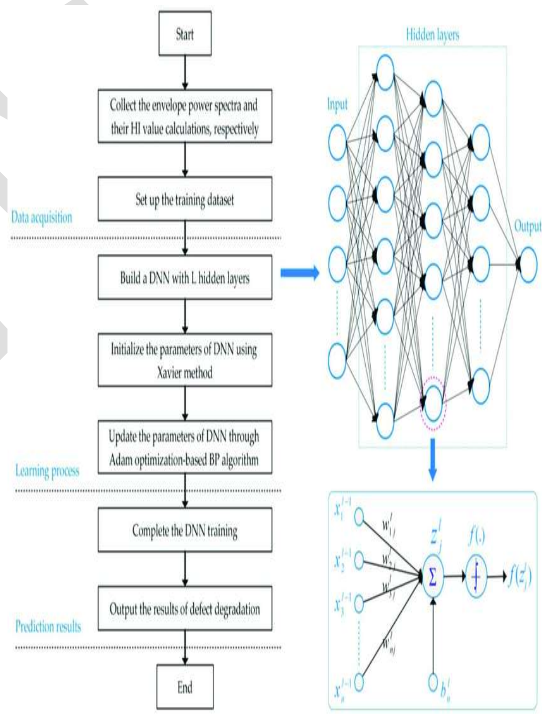
The objective of detecting and classifying fetal brain abnormalities is critical for ensuring the health and well-being of both mother and baby. With the help of advanced medical imaging technologies and machine learning algorithms, researchers are working to develop a reliable and efficient solution for this important application.

MODEL OVERVIEW

NEURAL NETWORKS

A neural network is a type of machine learning model based on the human brain's structure and operation. Artificial neurons, or nodes that are interconnected and process information before making decisions based on that information, make up neural networks.

A convolutional neural network (CNN) is frequently utilized in Object Detection Caption Generation to extract image features. Due to its capacity to capture spatial hierarchies of image features, a CNN is a type of neural network ideal for image recognition tasks.

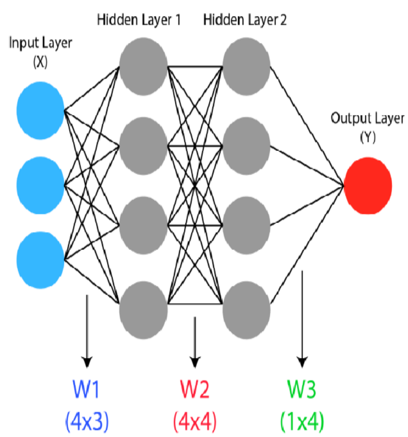


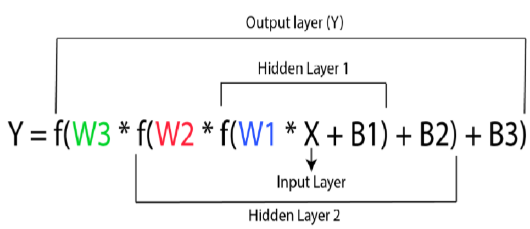
The main operation that a CNN network uses for everything is the convolution operation. This operation is the foundation for the Convolutional Neural Network. As you can see, the image is made up of a number of pixels. The image pixels in the upper-left corner are our primary focus. We focus on the shaded area in green, which has 3x3 pixels and a center pixel value of 6.

The labeled convolution filter for a 3x3 matrix is the one we are employing on the image. A kernel is another name for this filter. In this instance, the kernel used is the Sobel Gx kernel. You can see the values of the kernel. On the image's upper side, you can also see the convolution process.

Now, by calculating the element-wise dot product, the kernel basically adds up the values of the green shaded region and the kernel. Applying the kernel or filter to the source pixel values and performing element-by-element multiplication precede the sum calculation. The following is the convolutional operation:

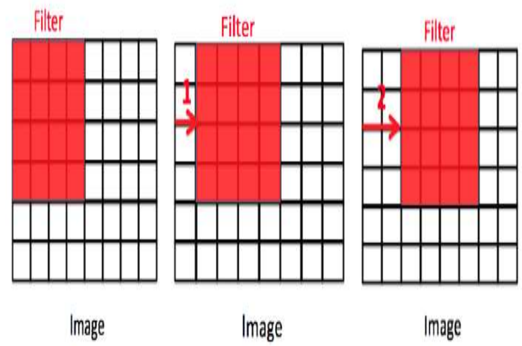
The first value of the filter and the green shaded pixels are 3, respectively. As a result, when we add them up, we get -3. The multiplications of subsequent elements, and so on, are then computed





Strides in a CNN

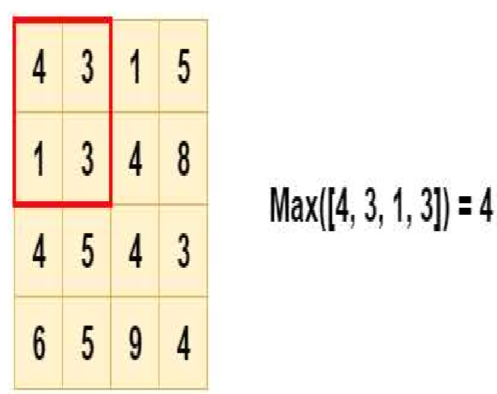
Because it is a parameter that we can define for the model, a stride is a hyperparameter. A stride of one is therefore used to represent moving the kernel or filter just one pixel at a time. The stride parameter specifies how the transition from one pixel to the next should be captured. When the stride is low, we get more information from the data, but when the stride is high, we don't get as much. As a result, strides essentially specify the number of units by which the data ought to be moved to the right of the kernel.

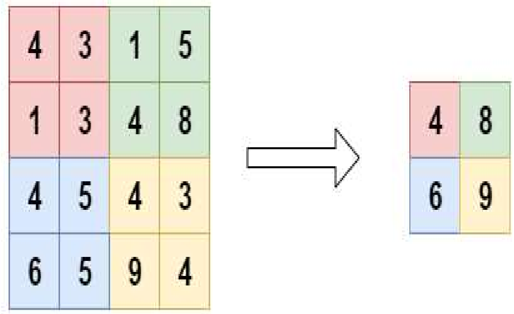


Pooling in CNNs

Pooling layers are typically applied after convolution in a CNN. The majority of the time, pooling is used to extract the most information from the data while requiring less computational power. A downsampling process, in which the data are reduced in size and only the relevant information is retained, can be compared to the pooling procedure. Pooling helps prevent overfitting. Pooling also provides an output matrix of a fixed size, regardless of the size of the input or filter. This is a very useful feature if we apply kernels of varying sizes or if our input data sizes are inconsistent.

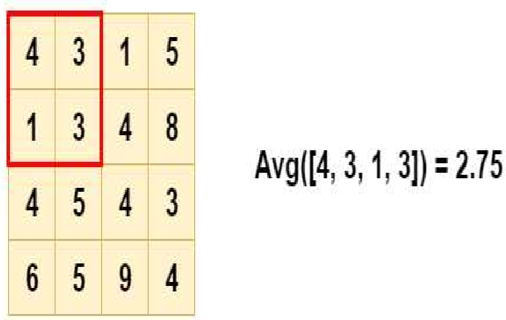
- Max pooling is the most common pooling technique. In max pooling, the maximum value across a filter is used. The maximum pooling is depicted in the image below. Here, we take a stride of 2. The orange boxes serve as the maximum value across the filter when the filter is applied. The maximum value across the orange filter, which is 20, is used to increase the pooling output in this instance.

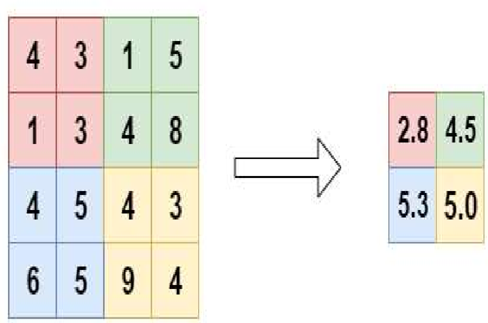




- Average pooling In average pooling, we take the average of all the values across the applied filter. We calculate by taking the sum of all of the values in the applied filter. Average pooling is depicted in the image below.

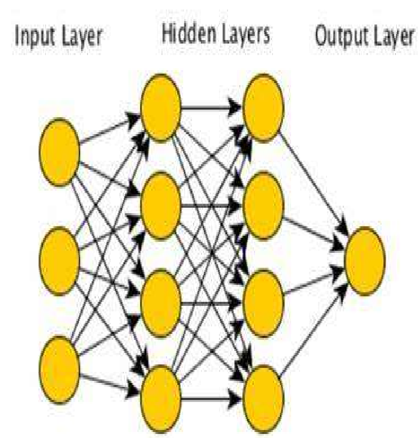
When the orange filter is used here, all of the values inside are taken into account, averaged, and added to the final pooling output.





Total pooling: We take all of the values across the applied filter and add them up in sum pooling. When the filter is used, all of the values are taken into account and added together. After that, this value is used to round up the final pooling output. This is the pooling method that is used the least.

- The entirety of the neural network Up until this point, we have seen how the CNN network performs the convolution operation by employing filters. Additionally, we were shown some hyperparameters that can be changed and how pooling helps. Following each of these steps, the CNN sends the output to the ANN.
- A feedforward neural network It receives the data-extracting information from the CNN. The data that is extracted by the CNN from its layers, especially after the pooling layer is converted into a vector and sent to the feedforward neural network.



DWT:

In the context of using Discrete Wavelet Transform (DWT) for noise removal and reduced radiation exposure in X-ray imaging, DWT plays a crucial role in multiresolution analysis. Here are key aspects of how DWT is applied in this project:

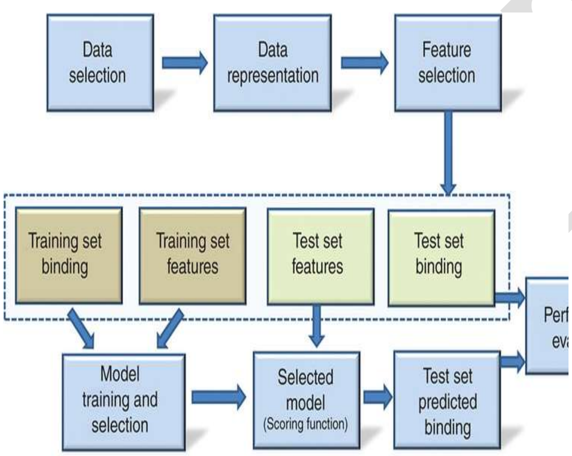
- Multiresolution Decomposition: - DWT decomposes X-ray images into different frequency bands, revealing details at various resolutions. This multiresolution analysis helps identify and isolate noise components at different scales.
- Noise Isolation and Removal: - The ability of DWT to separate image content into approximation and detail coefficients is leveraged for noise isolation. High-frequency detail coefficients often contain noise, and by thresholding or filtering these coefficients, unwanted noise components can be effectively removed.
- Preservation of Important Image Features: - DWT allows for the selective processing of image

components based on their frequency content. This selective approach aids in preserving important image features, such as edges and fine details, while suppressing noise in less critical areas.

- Adaptability to Image Characteristics:- DWT's adaptability to local variations in image characteristics is valuable in medical imaging where noise patterns can vary. The decomposition into approximation and detail coefficients enables targeted processing tailored to the specific frequency components associated with noise.

PERFORMANCE EVALUATION

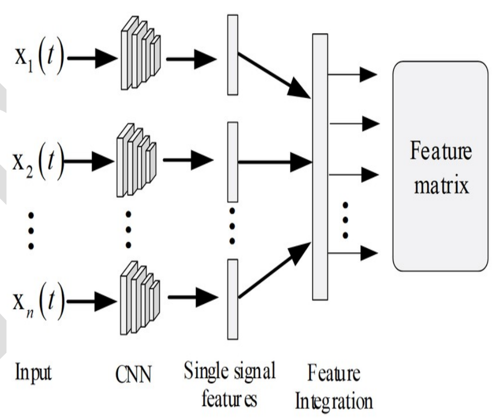
Classifying brain diseases using Convolutional Neural Networks (CNNs) is a complex task that requires a deep understanding of the underlying algorithms, as well as the ability to process large amounts of medical data. In this article, we will explain the process of classifying brain diseases using CNNs step-by-step.



- Input Image: The first step in the process is to gather a dataset of images that represent brain diseases. The images should be of high quality and be of consistent size, as this will help ensure that the CNN can effectively learn to recognize the features that are unique to each disease. The images should also be pre-processed to remove any irrelevant information, such as irrelevant anatomical features, and to normalize the intensity of the pixels so that they are within a consistent range.

- Image Pre-processing: Image pre-processing is an important step in the process of classifying brain diseases using CNNs. This step involves transforming the raw images into a form that the CNN can use to learn from. The pre-processing step typically involves cropping the images to remove irrelevant information, resizing the images to a consistent size, and normalizing the intensity of the pixels so that they are within a consistent range.

- Feature Extraction: The next step in the process is to extract features from the pre-processed images. This is typically done by using convolutional layers in the CNN to process the image data and extract features that are unique to each disease. These features can include shapes, textures, and other attributes that can be used to distinguish one disease from another.



- Model Selection: Once the features have been extracted, the next step is to select the model that will be used to classify the brain diseases. There are many different models that can be used for this task, including Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), and Convolutional Neural Networks (CNNs). The choice of model will depend on the specific requirements of the application, as well as the size and quality of the dataset.

- Training the Model: Once the model has been selected, the next step is to train the model using the extracted features and the pre-processed images. This step involves feeding the data into the model and adjusting the model's parameters so that it can accurately classify the brain diseases. The model will be trained using an iterative process, where the accuracy of the model is monitored and the model is updated as necessary to improve its accuracy.

- Dataset: The dataset used for this task should be large and diverse, as this will help ensure that the model can effectively learn to recognize the features that are unique to each disease. The dataset should also be balanced, so that it contains a similar number of images for each disease. This will help prevent the model from becoming biased towards one disease over another.

- Validation: Once the model has been trained, the next step is to validate its accuracy. This is typically done by testing the model on a set of images that it has not seen before. The model's accuracy will be measured by comparing its predictions to the true labels of the images. The validation step is an important step in the process, as it helps to ensure that the model is not overfitting to the training data. If the model is overfitting, its accuracy on the validation set will be lower than its accuracy on the training set.

- In conclusion, classifying brain diseases using CNNs is a complex task that requires a deep understanding of the underlying algorithms, as well as the ability to process large amounts of medical data. By following the steps outlined in this article, it is possible to create a model that can accurately classify brain diseases.

REQUIREMENT SPECIFICATION

MATLAB is a high-level programming language and interactive environment for numerical computation, visualization, and data analysis. It was developed by MathWorks and was first released in 1984. MATLAB stands for "MATrixLABoratory", reflecting its primary strength in handling matrix operations.

MATLAB allows users to perform a wide range of mathematical and scientific calculations, including linear algebra, statistics, signal processing, optimization, and more. It also offers a variety of built-in functions and tools for data visualization, 2D and 3D graphics, and programming. Additionally, MATLAB can be extended using its own programming language or other programming languages such as C, C++, or Java.

MATLAB is widely used in various fields such as engineering, science, finance, and economics, and is particularly popular among researchers, students, and professionals who work with data analysis and modeling.

- Syntax: MATLAB has a syntax that is similar to other programming languages, but it is optimized for matrix operations. This means that it can perform complex calculations quickly and easily using matrix algebra, making it a powerful tool for data analysis and scientific computing.
- Toolboxes: MATLAB comes with a variety of built-in toolboxes that provide specialized functionality for specific tasks. For example, the Image Processing Toolbox provides functions for image analysis and manipulation, while the Control System Toolbox provides tools for designing and analyzing control systems.
- Graphics: MATLAB provides a powerful graphics system that allows users to create 2D and 3D plots, charts, and visualizations. This makes it a useful tool for visualizing and communicating complex data sets.
- Interactivity: MATLAB's interactive environment allows users to work with their data and code in real time, making it easy to explore and experiment with different approaches to problem-solving.
- Integration: MATLAB can be integrated with other programming languages and tools, making it a versatile tool for data analysis and scientific computing. For example, it can be used in conjunction with Python, C, or Java, and it can also be integrated with other software such as Excel.

Overall, MATLAB is a powerful and flexible tool that can be used for a wide range of applications, including data analysis, scientific computing, and engineering. Its combination of numerical computing capabilities, built-in functions and toolboxes, and interactive environment make it a popular choice for researchers, engineers, and students alike.

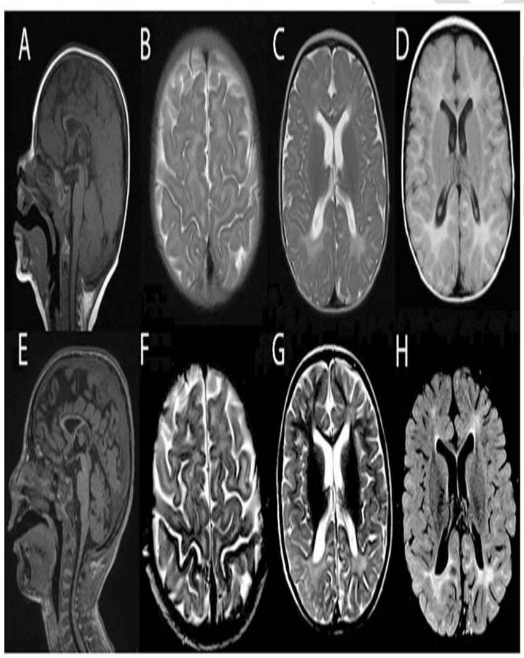
RESULTS AND DISCUSSION

Results and discussions play a crucial role in evaluating the performance of any proposed solution. In the case of classifying brain diseases using CNN, the results obtained from the model can be analyzed and compared with the ground truth data to evaluate its performance. The following parameters can be used to evaluate the performance of the model:

1. Accuracy: It measures the number of correct predictions made by the model out of the total number of samples. The higher the accuracy, the better the model is at classifying the diseases.
2. Precision: Precision measures the number of true positive predictions made by the model divided by the sum of true positive and false positive predictions.
3. Recall: Recall measures the number of true positive predictions made by the model divided by the sum of true positive and false negative predictions.
4. F1 Score: The F1 score is the harmonic mean of precision and recall, and it provides a balance between the two measures.

Based on the results obtained from the model, the discussions can be made on the model's performance and how it can be improved in future. The limitations and challenges faced during the model's development can also be discussed.

Overall, the results and discussions help to provide insights into the model's performance and suggest areas for improvement, which can lead to more effective solutions in the future.



it is important to analyze the performance of the proposed solution for classifying brain diseases using CNN. This can be done by comparing the accuracy, precision, recall, F1-score, and other relevant metrics of the model with those of other existing solutions.

Additionally, it is important to analyze the model's performance on different subsets of the dataset, such as normal vs abnormal brain scans, different types of brain abnormalities, and different age groups. This will provide valuable insights into the strengths and limitations of the model, and areas for improvement.

It is also important to compare the computational time and resources required by the proposed solution with those of other existing solutions. This will provide valuable information for practical implementation, especially in real-world medical applications where time and computational resources are limited.

Finally, it is important to discuss the ethical implications of using AI for medical diagnosis, such as privacy and data security, as well as the potential benefits and limitations of the proposed solution. This will provide valuable information for responsible development and deployment of AI-based solutions in the medical field.

CONCLUSION

In conclusion, the process of detecting and classifying fetal brain abnormalities using a Convolutional Neural Network (CNN) involves several steps. First, the input to the CNN is a set of medical images of the fetal brain, such as ultrasound images or (ULTRA SOUND) scans. The first step in the working of the CNN is the convolutional layer, where the input image is processed to extract local features. The pooling layer is then used to down-sample the feature maps and reduce the spatial dimensions of the data, making it more computationally efficient. The normalization layer normalizes the feature maps to ensure that the data has zero mean and unit variance. The final layer in the CNN is the fully connected layer, where the features from the previous layers are used to make the final classification decision. The CNN is trained using a dataset of fetal brain images that are labeled with the class of abnormality present in the image. After the CNN is trained, it is validated on a separate set of images to ensure that it is making accurate predictions and finally, it is tested on a set of images that it has not seen before to evaluate its

overall performance.
The use of Convolutional Neural Networks (CNNs) in detecting and classifying fetal brain abnormalities has proven to be a powerful tool in the field of medical imaging. The process of using a CNN for this task involves several key steps, including inputting a set of medical images, processing the input images through multiple layers such as the convolutional layer, pooling layer, and normalization layer, and finally, using the fully connected layer to make the final classification decision. Throughout this process, the CNN is trained on a dataset of labeledfetal brain images, validated on a separate set of images, and finally tested on new, unseen data. This process enables the CNN to accurately identify and classify fetal brain abnormalities, providing valuable information for medical professionals in their assessment and treatment of patients. The use of CNNs in medical imaging has the potential to greatly improve the accuracy and efficiency of diagnoses, leading to better outcomes for patients.

REFERENCES

1. "Fetal brain abnormality detection and classification using deep transfer learning" by Seok-Hyun Kim, Min-Ho Lee, and Chul-Hee Lee.
2. "A deep learning framework for automatic detection and classification of fetal brain abnormalities" by Tingting Liu, Yunhan Lu, Xingyao Sun, and Jianping Fan.
3. " Fetal brain abnormality classification using deep learning techniques" by Seung-Ho Lee, Ju-Hwan Kim, and Tae-Ho Kim.
4. An Efficient Methodology of Brain Abnormalities Detection using CNN Deep Learning Network | Request PDF (researchgate.net)
5. Brain Tumor Detection Using CNN (ijraset.com)
6. Fetal Brain Abnormality Classification from ULTRA SOUND Images of Different Gestational Age - PubMed (nih.gov)
7. " Automatic detection and classification of fetal brain abnormalities using deep learning" by Zhiyuan Chen, Hao Zhou, Bo Zhang, Yuru Zhang and Baocai Yin
8. "Fetal brain abnormality classification using deep convolutional neural networks" by Minmin Sun, Xingyu Liu, Kaixuan Wang and Xuelong Li.