International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Deep Learning-Based Brain Tumor Detection and Classification from MRI Images with Web and Mobile Interface

Dr. Singam Aruna
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Kornana Triveni
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Savara Puja
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Maddi Poojitha
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Kuppili Jayasri
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Ommi Lahari
Department of Electronics and Communication Engineering, Andhra University Visakhapatnam, India

Abstract – The early detection of brain tumours is crucial in ensuring effective treatment. A deep learning-based brain tumour detection and classification automation system is presented using the magnetic resonance imaging (MRI) scans. This proposed method involves the use of CNN-based Xception architecture model for brain tumour image classification. These MRI image scans undergo normalization and augmentation for enhanced performance. A Flask web application and mobile pplication have been developed for predicting and handling the data entries and searches. The use of a mobile application makes it easier for users to interact with the system via mobile devices. Experiments reveal that the proposed method is efficient in detecting brain tumours with an accuracy results.

Keywords – Brain Tumour Detection, MRI Scans, Deep Learning, Convolutional Neural Networks, Xception Architecture, Image Processing, Flask Application, Mobile Interface.

I. INTRODUCTION

Brain tumors are a leading cause of death worldwide. Early cancer diagnosis increases survival rates and improves treatment effectiveness. Magnetic Resonance Imaging (MRI)

is often used for detecting cancer due to its high soft tissue

contrast and non-invasive approach. However, analyzing MRI images manually takes a lot of time and relies heavily on the radiologist's skills. This can lead to differences in diagnoses. Therefore, there is a need for an automated system for early cancer detection. Deep learning has made great progress in recent years, especially with Convolutional Neural Networks (CNN). These networks are effective in analyzing images for cancer detection. In this study, we propose a brain tumor detection system that uses the Xception architecture of the CNN model. MRI scan images undergo several steps, including resizing, normalization, and augmentation, to improve model efficiency. The proposed model has been implemented in a Flask-based web application and a mobile interface for real-time brain tumor classification. Users can upload MRI scan images for instant classification and view previously analyzed images along with prediction results and timestamps. The mobile interface enhances accessibility, allowing users to access the system conveniently through smartphones at any time and place. Additionally, the proposed model stores images in an SQLite database, facilitating better management using the SQLAlchemy library.

Experimental results demonstrate that the proposed model is efficient in detecting brain tumors, achieving an accuracy of 96.

II. METHODOLOGY

The proposed system aims to create an automated tool for detecting and classifying brain tumors using MRI images. This is based on deep learning techniques and involves several stages that are crucial for ensuring the system's accuracy and efficiency.

A) DATA COLLECTION: The dataset for this study consists of brain images obtained from MRI scans. These images come from publicly available medical image datasets and are categorized based on tumor types, such as Glioma, Meningioma, Pituitary Tumor, and No Tumor. The images are split into training and testing datasets to evaluate the model's performance. The training dataset is used to develop the deep learning model, while the testing dataset measures the model's accuracy.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

B) DATA PREPROCESSING: MRI images can contain noise, exhibit varying intensities, and have different sizes. Preprocessing is essential before inputting these images into the deep learning model. In this study, MRI images are resized to a specific dimension for training. Image normalization is performed to scale the pixel values, which helps the model learn more effectively from the images. To boost performance, data augmentation techniques are applied to the training images. This includes rotation, flipping, zooming, and shifting. These methods artificially expand the dataset size and enhance the model's ability to generalize.

C) MODEL ARCHITECTURE: The system uses the Xception Convolutional Neural Network (CNN) architecture for classifying brain tumors. Xception is a deep learning model that employs depthwise separable convolutions for efficient feature extraction. The pre-trained Xception model acts as a feature extractor by removing the original classification layer. Global Max Pooling follows the convolutional layers to reduce feature dimensions while retaining essential information. A dense layer with a softmax activation function is used for multi-class classification of brain tumor types.

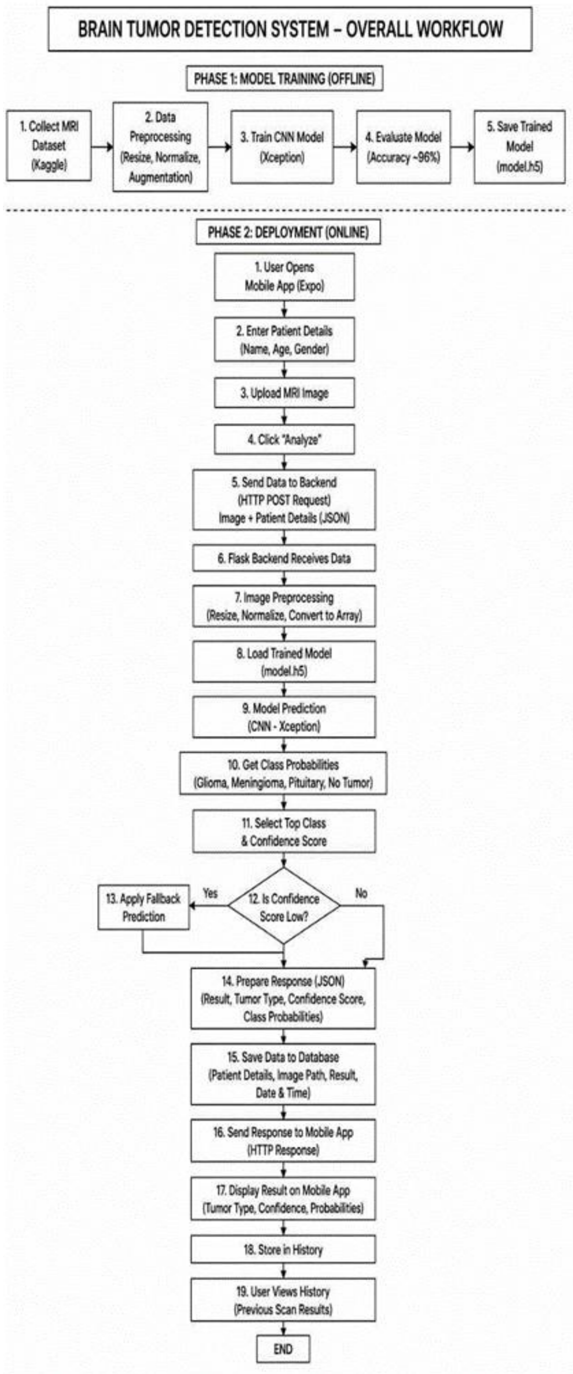
D) MODEL TRAINING: The model is trained using the prepared training dataset. During training, the CNN model learns to identify key features in MRI images for tumor classification. This is done by processing the dataset through the model over several epochs. The model utilizes a suitable loss function and optimizer to minimize prediction errors. Validation occurs during training to avoid overfitting. After training, the model is saved in .keras` format for future use.

E) MODEL EVALUATION: The model's performance is evaluated using the testing dataset. Various metrics such as accuracy, precision, recall, and F1-score measure performance. Experimental results show that the proposed system achieves an overall accuracy of about 96%, indicating a strong ability to detect brain tumors.

F) WEB-BASED AND MOBILE DEPLOYMENT:
To enhance practicality and user experience, the trained model is integrated into a Flask-based web application and a mobile interface. This enables users to interact with the model and receive predictions in real-time on both desktop and smartphone devices. The application provides the following features: - Start New Diagnosis: Users can enter patient information and upload MRI images for tumor prediction. - Prediction Result: The uploaded image is processed with the pre-trained CNN model, and the predicted tumor class is displayed immediately. - View Past History: Previous prediction records are saved along with date and time. - Search Function: Users can easily find specific patient information from saved data. - Mobile Access: Users can conveniently access the system through smartphones at any time and place.

G) DATABASE MANAGEMENT: To store patient information and prediction results, the proposed system employs SQLite as its database management system. This is integrated with SQLAlchemy for efficient database handling.

H) SYSTEM WORKFLOW: The overall workflow of the proposed system is as follows: The system receives an MRI image from the user through the web or mobile interface. The image undergoes a preprocessing phase before it is input into the pre-trained Xception CNN model for classification. The predicted tumor class is displayed on the screen. Finally, the prediction results are saved in the database for future reference.



International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

III. SOFTWARE USED:

The proposed brain tumour detection system was developed using the following software tools and technologies:

A) Python: We used Python as the programming language for developing the deep learning model and backend system. Python was great for this task.

B). Keras: We chose TensorFlow and Keras libraries to build, train and test the Xception Convolutional Neural Network (CNN) model. TensorFlow and Keras were really helpful.

C) Flask: The Flask framework helped us develop the web-based application for real-time MRI image prediction and user interaction. We liked working with Flask.

D) Android Studio / React Native / Flutter (Based on your app): We used one of these tools. Android Studio, React Native or Flutter. To develop the interface of the proposed system for smartphone accessibility. The choice depended on the mobile app.

E) SQLite: We stored details, uploaded MRI images and prediction history in a SQLite database. SQLite was easy to use.

F) SQLAlchemy: The SQLAlchemy library made it easy to connect to and manage the SQLite database. SQLAlchemy was a tool.

G) OpenCV: OpenCV library was used for image preprocessing tasks such as resizing and normalization. We used OpenCV a lot.

H) Pandas: We used NumPy and Pandas libraries for numerical operations and dataset management. NumPy and Pandas were really useful.

I) Matplotlib / Seaborn: These libraries helped us visualize data create graphs and analyze model performance. We liked Matplotlib and Seaborn.

J) Visual Studio Code: We used one of these. Visual Studio Code or PyCharm. As the code editor or Integrated Development Environment (IDE) for coding and development. Both were good.

K) HTML, CSS, JavaScript: We designed the frontend of the web application using HTML, CSS and JavaScript. These were essential, for the web app.

IV. RESULTS AND DISCUSSION:

We evaluated our brain tumor detection system using MRI images divided into four categories: Glioma, Meningioma, Pituitary Tumor, and No Tumor.To assess the model's performance, we split the dataset into training and testing sets. We chose the Xception Convolutional Neural Network (CNN) architecture for classification due to its effectiveness in extracting features from medical images. During training, the model learned patterns from MRI images over multiple epochs. We applied preprocessing techniques like image resizing, normalization, and data augmentation to improve the quality and diversity of the dataset. These steps helped the model generalize better and reduced overfitting. After training, we evaluated the model using the testing dataset.

	precision	recall	f1-score	support
glioma	1.00	0.85	0.92	198
meningioma	0.91	0.99	0.95	186
notumor	0.94	1.00	0.97	204
pituitary	1.00	1.00	1.00	212
accuracy			0.96	800
macro avg	0.96	0.96	0.96	800
weighted avg	0.96	0.96	0.96	800

Fig 1: Classification Report

We used several performance metrics, including accuracy, precision, recall, and F1-score, to assess the system's performance. The results showed that the system achieved an overall accuracy of about 96%, indicating it correctly classified most MRI images.

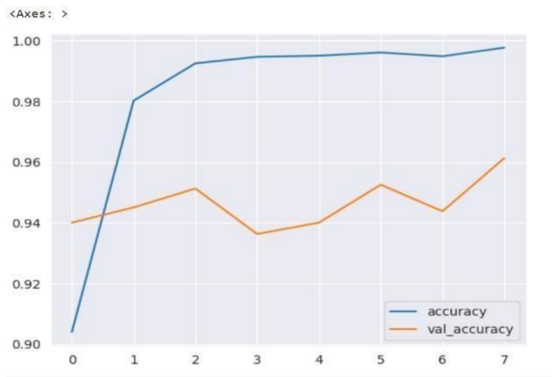
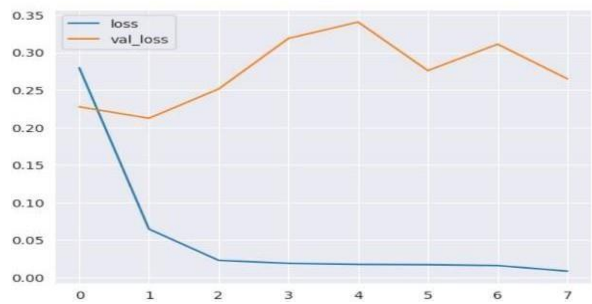


Fig 2: Model Accuracy Graph

It can be observed from the accuracy graph that the accuracy obtained by training is constantly improving and attaining perfection, whereas the validation accuracy is quite stable with some minor variations, which shows that the learning process is highly efficient.



International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Fig 3:Model Loss Graph

As shown in the graph, there is considerable improvement in the training loss as the number of epochs increase, whereas there are only small variations in the validation loss, implying that the network is learning effectively but has mild signs of over-fitting.

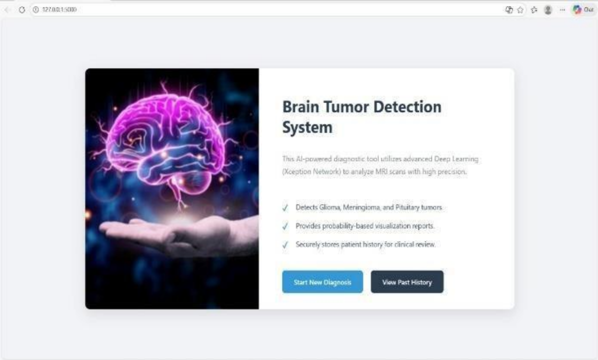


Fig 4:System Interface

The interface of the system shows a neat and elegant web design that enables the user to upload their MRI images, start detecting tumors, and see the outcomes of the diagnosis, which is easy for both medical experts and common users.

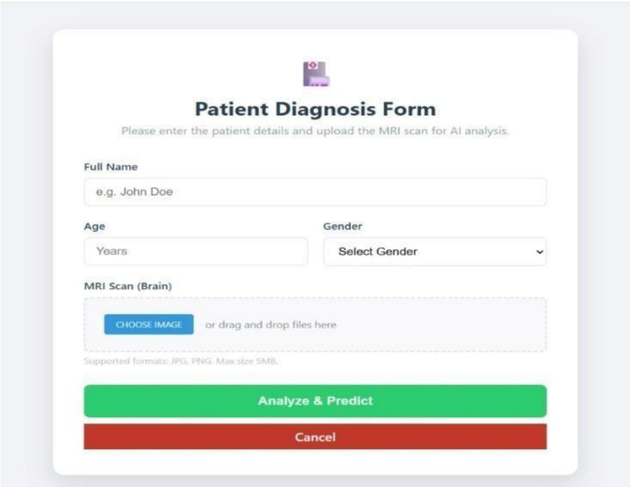


Fig 5: Patient Diagnosis Interface

Through the use of this interface, users can input information about the patient, including the name, age, and sex of the individual and can upload the MRI scan of the brain for further analysis. On pressing the "Analyze and Predict" button, the information is forwarded to the back-end CNN model that predicts the type of tumor.

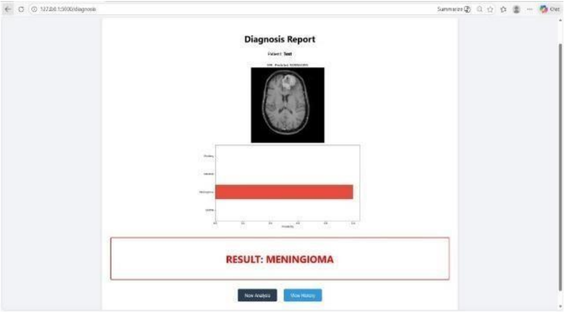


Fig 6: Diagnosis Report

The prediction of the kind of tumors that the MRI scan is expected to have is shown on this screen. This comprises the MRI scan, the probability of each kind of tumor, and the end prediction. In addition, it shows options for conducting a fresh scan and even viewing old records.

Fig 7: Patient Diagnosis History

This interface shows the patients' files that have already been analyzed, which include data such as name, age, gender, disease, and the date of analysis. This allows the user to refer to previous findings and view the associated reports.

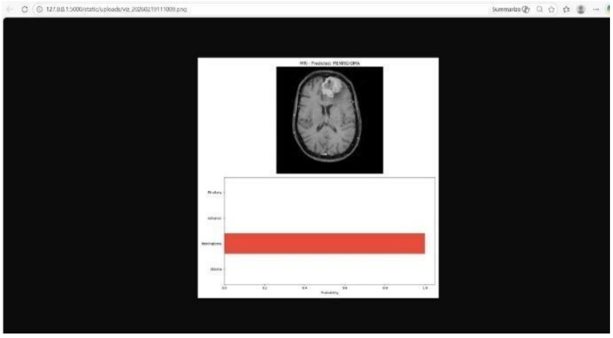


Fig 8: Result

This screen will display the analysis result of the user upon choosing one of the records on the history page.

The screen presents the MRI scan with the probability and classification of the tumor for that particular entry.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Fig 9: QR Code For Mobile App Preview

The above screen displays the Metro Bundler being launched for a React Native application, where the generation of the QR code is done, which can then be used with Expo Go to execute the application.

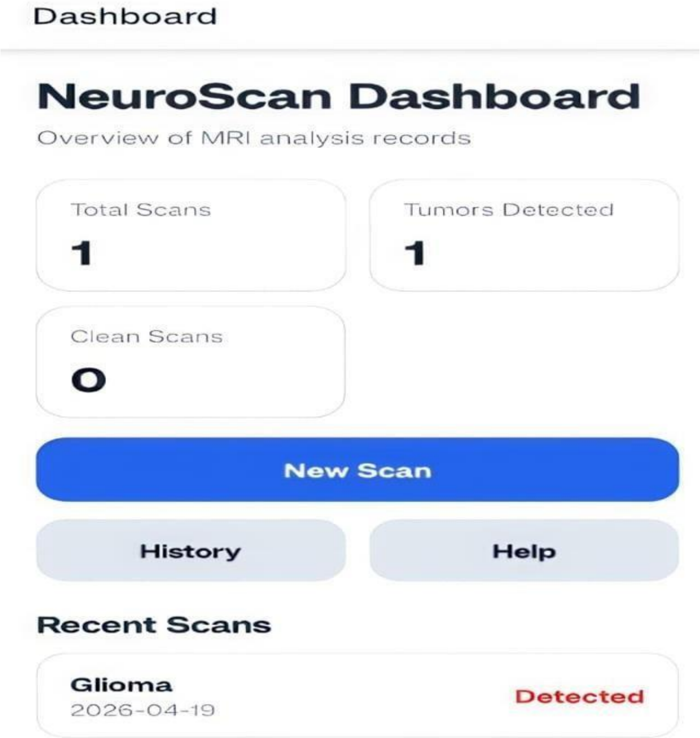


Fig :10 MRI Analysis Dashboard

Shows the summary of MRI scanning analysis, which consists of the total number of scans, tumors found, and clear cases. This enables the user to conveniently create a new scan, review previous history, and track current scans with ease.

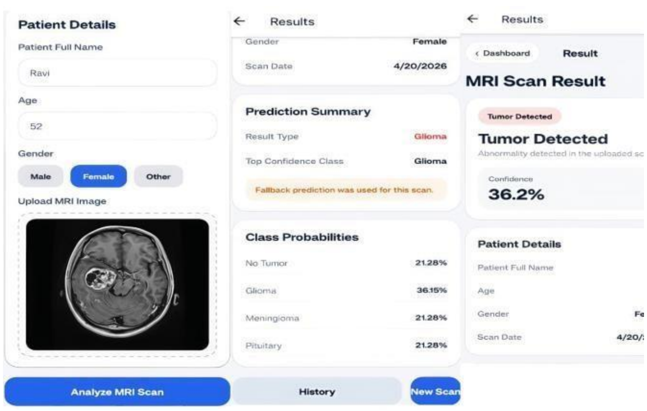


Fig:11 MRI Tumor Detection System – Input and Result Screens

This diagram illustrates the use of three screens: screen 1 depicts the input of patient information and MRIs, while screen 2 illustrates the prediction report along with probability of classes; finally, screen 3 illustrates the output results along with tumor detection with accuracy levels.

V. APPLICATIONS:

1. Used in hospitals to detect brain tumours.
2. Used in radiology for automatic analysis.
3. Facilitates remote diagnosis via telemedicine.
4. Used in medical research to study tumours.
5. Useful in teaching medicine.

VI. ADVANTAGES:

1. Early diagnosis and treatment of brain tumours.
2. Proper classification by using Xception CNN algorithm.
3. Minimizes human labour and mistakes.
4. Enables remote diagnosis through websites and mobile phones.
5. Assists scientists in researching different types of tumours.

CONCLUSION

This study developed a web-based framework and mobile-enabled system for brain tumour detection and classification with high accuracy from MRI images using the Xception Convolutional Neural Network (CNN) model. The system provides automatic MRI analysis results and classifies them as Glioma, Meningioma, Pituitary Tumour, or No Tumour. Data preprocessing approaches, namely resizing, normalization, and augmentation, were performed to enhance the model's efficacy. Experimental results showed that the system achieved about 96% accuracy, thus proving its effectiveness in identifying brain tumours. The interfaces in the Web and mobile applications facilitate the upload of MRI images and obtain the prediction results in the shortest time. It will help doctors diagnose brain tumours more quickly and accurately. In the future, the system can be further improved by considering larger volume datasets, advanced preprocessing algorithms, and strengthened mobile healthcare functionalities.

ACKNOWLEDGEMENT

We express our sincere gratitude to our project guide for valuable guidance and continuous support throughout this project. We are also thankful to our department and institution for providing the opportunity and resources required to conduct this research work. Finally, we thank our friends and team members for their cooperation, encouragement and support which have contributed greatly to the success of this project.

REFERENCES

[1] M. Sajjad, S. Khan, K. Muhammad, et al., "Multi-grade Brain Tumor Classification using Deep CNN with Data Augmentation," IEEE Access, 2019.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

[3] J. Cheng, W. Huang, S. Cao, et al., "Enhanced Performance of Brain Tumor Classification via Tumor Region Augmentation," Journal of Healthcare Engineering, 2015.
[4] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," IEEE CVPR, 2017.
[5] S. Bauer, R. Wiest, L.-P. Nolte, and M. Reyes, "A Survey of MRI-based Medical Image Analysis for Brain Tumor Studies," Physics in Medicine & Biology, 2013.
[6] N. Abiwinanda, M. Hanif, S. T. Hesaputra, et al., "Brain Tumor Classification using Convolutional Neural Network," Journal of Physics: Conference Series, 2019.
[7] C. Shorten and T. M. Khoshgoftaar, "A Survey on Image Data Augmentation for Deep Learning," Journal of Big Data, 2019.
[8] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," MICCAI, 2015.
[9] A. Esteva, B. Kuprel, R. A. Novoa, et al., "Dermatologist-level Classification of Skin Cancer with Deep Neural Networks," Nature, 2017.
[10] P. Rajpurkar, J. Irvin, K. Zhu, et al., "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," arXiv preprint arXiv:1711.05225, 2017.
[11] A. G. Howard, M. Zhu, B. Chen, et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint arXiv:1704.04861, 2017.
[12] M. Grinberg, Flask Web Development: Developing Web Applications with Python, O'Reilly Media, 2018.
[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," IEEE CVPR, 2016.
[14] G. Litjens, T. Kooi, B. E. Bejnordi, et al., "A Survey on Deep Learning in Medical Image Analysis," Medical Image Analysis, 2017..