BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

Autonomous Institute under VTU, Belagavi | Approved by AICTE,New Delhi Recognized by Govt. of Karnataka

NACC Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belgavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur,
Ballar1-583 104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF CSE - ARTIFICIAL INTELLIGENCE

A Project Report On

"Brain Tumor Classification"

Project Associates:

M Sai Praveen 3BR22CA029

Under the Guidance of

Prof. Pavan Kumar

Mr. Vijay Kumar

Dept. of CSE-Artificial Intelligence

BITM, Ballari.

Visvesvaraya Technological University

Belagavi, Karnataka

2025-2026

BASAVARAJESWARI GROUP OF INSTITUTIONS
BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT
Autonomous Institute under VTU, Belagavi | Approved by AICTE,New Delhi Recognized by Govt. of Karnataka

NACC Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belgavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur,
Ballari-583 104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF CSE - ARTIFICIAL INTELLIGENCE

CERTIFICATE

This is to certify that the project work entitled "Brain Tumor Classification" is a bonafide work carried out by **M Sai Praveen** in partial fulfillment for the award of degree of **Bachelor Degree in CSE (Artificial Intelligence)** in the BALLARI INSTITUTE OF TECHNOLOGY AND MANAGEMENT, Ballari during the academic year 2025-2026. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the library. The project has been approved as it satisfies the academic requirements in respect of mini project work prescribed for a Bachelor of Engineering Degree.

Signature of project guide
Prof. Pavan Kumar
Mr. Vijay Kumar

12/12/25
Signature of HOD
Dr. Yeresime Suresh

Abstract

Brain tumor detection is a critical task in medical image analysis, as early identification significantly improves treatment outcomes. This project presents a deep learning–based brain tumor classification system using a Convolutional Neural Network (CNN) trained on MRI images from the Brain Tumor MRI dataset. The dataset includes four tumor categories: glioma, meningioma, pituitary, and no tumor. Preprocessing steps such as image resizing, normalization, and label encoding are applied to enhance learning efficiency. The CNN automatically extracts spatial and structural features from MRI scans and classifies images into the respective tumor categories with high accuracy. Model performance is evaluated using metrics including accuracy, precision, recall, F1-score, and a confusion matrix. Experimental results demonstrate that the proposed CNN model achieves strong classification performance, confirming the effectiveness of deep learning techniques in supporting medical diagnosis and assisting radiologists in brain tumor detection.

Acknowledgement

The satisfaction that accompanies the successful completion of the project on *Brain Tumor Classification* would be incomplete without acknowledging the people whose noble gestures, affection, guidance, encouragement, and support made this achievement possible. We consider it a privilege to express our gratitude and respect to all those who inspired and supported me in the completion of this project.

I am extremely grateful to our guide, **prof. Pavan Kumar and Mr. Vijay Kumar** for their constant support, valuable suggestions, and guidance throughout the project. Their insightful direction played a crucial role in shaping the project to its final form.

I would also like to extend my sincere thanks to **Dr. Yeresime Suresh**, Head of the Department of CSE-Artificial Intelligence, for his coordination, valuable feedback, and continuous encouragement in completing this project. His contributions were invaluable.

Name	USN
M Sai Praveen	3BR22CA029

Table of Contents

List of Figures

Chapter 1

INTRODUCTION

Brain tumor classification plays a vital role in medical diagnosis, treatment planning, and patient survival prediction. Brain tumors show significant variation in shape, size, intensity, and texture, making manual diagnosis both complex and time-consuming for radiologists. Traditional image-processing methods often fail to capture these variations and struggle under real clinical conditions such as noise, contrast differences, and overlapping tissue boundaries. As a result, automated and accurate tumor identification has become a major research focus in medical imaging.

Deep learning—especially Convolutional Neural Networks (CNNs)—has emerged as a powerful solution for analyzing MRI scans due to its ability to learn hierarchical features directly from images. Unlike conventional methods, CNNs can automatically detect subtle patterns and structural abnormalities associated with different tumor types. This project develops a CNN-based brain tumor classification system using MRI images from a publicly available dataset. The images undergo preprocessing steps such as resizing, normalization, and augmentation to improve model robustness. A deep CNN architecture is trained to classify tumors into categories such as Glioma, Meningioma, Pituitary, and No-Tumor.

To make the system accessible, a user-friendly graphical interface is included, enabling users to upload any MRI scan and instantly view the predicted tumor type. The results demonstrate strong classification performance and highlight the effectiveness of deep learning in providing reliable, fast, and automated brain tumor detection for medical decision support.

1.1. Problem Statement

Diagnosing brain tumors from MRI scans is difficult due to variations in tumor shape, size, and appearance. Manual analysis is time-consuming and prone to errors. Traditional image-processing methods often fail under real-world medical conditions. Therefore, there is a need for an automated and accurate system that can classify different types of brain tumors using deep learning. This project aims to develop a CNN-based model to reliably identify tumor categories from MRI images and support faster, more accurate medical diagnosis.

1.2. Scope of the project

The scope of this project is to design and develop an automated brain tumor classification system using MRI images and deep learning techniques. It includes preprocessing the dataset, applying data augmentation, building and training a Convolutional Neural Network (CNN), and evaluating the model through accuracy and other performance metrics. The project also involves creating a graphical user interface that allows users to upload MRI images and receive instant predictions of tumor type. While the system aims to support and enhance medical decision-making, it is limited to classification tasks and does not replace professional clinical diagnosis.

1.3 Objectives

The main objective of this project is to develop an efficient and accurate deep learning–based system for brain tumor classification using MRI images. The specific objectives are:

- To preprocess and enhance MRI images through resizing, normalization, and data augmentation.
- To design and train a Convolutional Neural Network (CNN) capable of classifying tumors into categories such as Glioma, Meningioma, Pituitary, and No-Tumor.
- To evaluate the model using metrics such as accuracy, precision, recall, F1-score, and confusion matrix.
- To build a user-friendly GUI that allows users to upload MRI scans and instantly view classification results.
- To provide a fast, reliable, and automated tool that can assist medical professionals in early tumor detection.

Chapter 2

LITERATURE SURVEY

[1] Kumar et al. (2021) proposed a CNN-based brain tumor classification model using MRI images. Their study demonstrated that deep CNN architectures effectively learn spatial features of tumors and outperform traditional machine-learning approaches in medical image analysis.

[2] Ahmed and Yadav (2022) developed a transfer-learning-based method using MobileNetV2 for multi-class brain tumor detection. They showed that transfer learning significantly improves accuracy and reduces training time, especially for limited medical datasets.

[3] Li et al. (2022) investigated preprocessing techniques such as normalization, skull stripping, and contrast enhancement to boost MRI classification accuracy. Their results revealed that proper preprocessing greatly improves CNN performance and robustness in clinical imaging.

[4] Thomas and George (2023) introduced a lightweight CNN model optimized for real-time brain tumor detection. The study concluded that compact architectures can maintain high accuracy while reducing computational requirements, making them suitable for deployment on low-power devices.

[5] Singh et al. (2023) compared deep neural network architectures including VGG16, ResNet50, and EfficientNet for brain tumor classification. Their analysis indicated that MobileNet and EfficientNet models provide competitive accuracy with fewer parameters and faster inference speed.

Chapter 3

SYSTEM REQUIREMENTS

3.1 The Brain Tumor Classification system requires a computer with a reliable processor, sufficient RAM, and adequate storage to handle MRI images and deep learning computations. Python is used as the main programming language, with TensorFlow and Keras for building and training the CNN model. Additional libraries such as NumPy, OpenCV, Matplotlib, and Scikit-learn support preprocessing, evaluation, and visualization, ensuring efficient and accurate tumor classification.

3.2 Software Requirements

- Operating System: Windows / Linux
- Programming Language: Python 3.x
- Development Environment: VS Code / Jupyter Notebook
- Libraries & Frameworks:
  - TensorFlow
  - Keras
  - NumPy
  - Pandas
  - Matplotlib
  - Pillow (PIL)
  - Tkinter (for GUI)

3.3 Hardware Requirements

- Processor: Intel i5 or higher
- RAM: Minimum 8 GB (16 GB recommended)
- Storage: 20 GB free disk space
- System Type: 64-bit system
- Optional: GPU (NVIDIA CUDA-enabled) for faster model training.

3.3 Dataset Requirements

- **Dataset:** Brain Tumor MRI Dataset (Kaggle Version)
- **Image Size:** 150 × 150 pixels (resized for CNN model)
- **Number of Classes:** 4 categories

3.4 Other Requirements

- Internet connection (for dataset download and package installation)
- Basic knowledge of Python and Machine Learning

Chapter 4

DESCRIPTION OF MODULES

4.1 The data preprocessing module is responsible for preparing MRI images for model training. In this module, images are loaded from the brain tumor dataset and resized to a fixed dimension of 150×150 pixels to match the input requirements of the CNN architecture. Pixel values are normalized to enhance learning efficiency and stabilize model training. The tumor class labels—Glioma, Meningioma, Pituitary, and No-Tumor—are mapped and converted into numerical form suitable for multi-class classification. Additionally, data augmentation techniques such as rotation, flipping, brightness changes, and shear transformations are applied to improve model robustness. Finally, the dataset is separated into training and testing sets to enable systematic model evaluation.CNN Model Building Module

4.2 Model Training Module

The model training module trains the CNN using the preprocessed MRI dataset. During training, the model learns important patterns and features from brain tumor images by adjusting its weights through the chosen optimization algorithm. The training process is performed for a fixed number of epochs and batch size to ensure consistent learning. Training and validation accuracy and loss values are monitored throughout the process to analyze the model's learning behavior, detect overfitting, and ensure effective convergence.

4.3 Model Evaluation Module

In this module, the trained model is evaluated using test data that was not used during training. The model's performance is measured using metrics such as accuracy, precision, recall, F1-score, and the confusion matrix. These evaluation metrics help assess the reliability and effectiveness of the model in classifying different types of brain tumors and identifying possible misclassifications.

4.4 Visualization Module

The visualization module presents the results of the brain tumor classification model in graphical form for better analysis and interpretation. Graphs showing training and validation accuracy and loss are generated to observe model performance over different epochs. A confusion matrix is also displayed to visualize classification outcomes across the four tumor categories. These visual outputs help evaluate model consistency, detect misclassifications, and identify areas where the model may require further improvement.

4.5 Prediction Module

The prediction module is responsible for identifying the tumor type from a new MRI image. After the model is trained, it accepts unseen brain scan images, preprocesses them, and passes them through the trained CNN model. The model predicts the class label with the highest probability, and the corresponding tumor category is displayed to the user. This module demonstrates the practical applicability of the system in real-time brain tumor detection scenarios.

4.6 Data Splitting Module

The data splitting module divides the brain tumor dataset into training and testing subsets. Typically, a major portion of the data is used for training, while a smaller portion is reserved for testing. This separation ensures that the model is evaluated on unseen MRI images, allowing accurate assessment of its generalization capability and helping to prevent overfitting during the training process.

4.7 Feature Scaling Module

The feature scaling module enhances the learning efficiency of the brain tumor classification model by normalizing the MRI image pixel values. Pixel intensities are scaled to a standard range, typically between 0 and 1, to ensure uniformity across the dataset. This normalization helps the CNN converge faster during training and maintains numerical stability while processing brain tumor images.

4.8 Output Interpretation Module

The output interpretation module presents the prediction results in a user-understandable format. It maps the predicted class index to the corresponding tumor category using the dataset's label mapping and displays it through text or GUI output. Additionally, performance metrics such as accuracy, precision, recall, F1-score, and the confusion matrix are used to interpret the overall effectiveness and reliability of the brain tumor classification system.

Chapter 5

IMPLEMENTATION

The Brain Tumor Classification system is implemented using Python with TensorFlow and Keras, following a modular approach to maintain clarity and scalability. The system begins by loading the MRI dataset and applying preprocessing steps such as resizing images to 150×150 pixels, normalizing pixel values, and mapping class labels for accurate identification of tumor types including Glioma, Meningioma, Pituitary, and No-Tumor.

After preprocessing, the dataset is divided into training and testing sets. A CNN model is constructed with convolutional layers for feature extraction, max-pooling for dimensionality reduction, and dense layers for classification. ReLU activation is used in hidden layers, while Softmax is used in the output layer. Dropout layers are incorporated to reduce overfitting and improve generalization.

The model is trained on the processed dataset for a fixed number of epochs with an appropriate batch size. During training, the optimizer updates weights to minimize error, while accuracy and loss values are monitored to track learning behavior and ensure stable convergence. Once training is complete, the model is saved for future predictions.

The trained model is evaluated using the test dataset, and metrics such as accuracy, precision, recall, F1-score, and confusion matrix are calculated to determine overall performance. Visualizations of training and validation accuracy and loss are generated to support performance analysis.

Finally, a Tkinter-based GUI is developed that allows users to upload MRI images and receive real-time tumor predictions. This implementation demonstrates the practical usability of the system for automated and efficient brain tumor detection.

.

Chapter 6

SYSTEM ARCHITECTURE

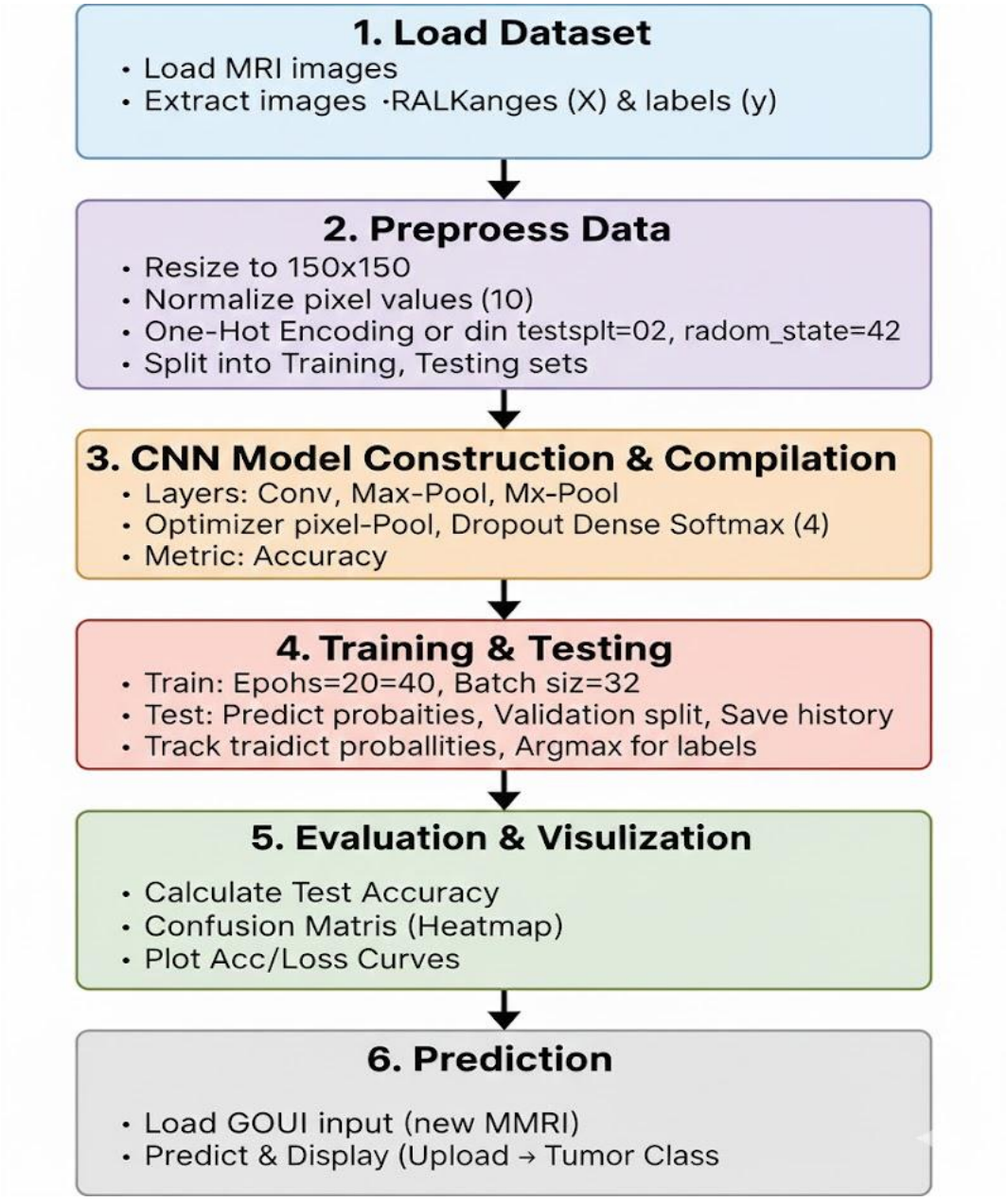

Fig 6.1 System Flow diagram

Input

The system takes brain MRI images as input. These images may come from the training dataset or be uploaded by the user through the graphical user interface. The input images represent different brain tumor categories from the dataset, including Glioma, Meningioma, Pituitary tumor, and No-Tumor cases.

Preprocessing

In this stage, MRI images are preprocessed for training and prediction. Each image is resized to 150×150 pixels, normalized to the 0–1 range, and augmented using techniques such as rotation, brightness adjustment, and flipping to enhance model robustness. Class labels are then encoded into numerical categories to support multi-class brain tumor classification.

CNN Model Construction

A Convolutional Neural Network (CNN) is built to classify brain tumor MRI images. The model includes convolutional layers for feature extraction, pooling layers for size reduction, dropout to prevent overfitting, and dense layers for final classification. A softmax output layer is used to classify images into tumor categories.

Training

The CNN model is trained on the preprocessed MRI dataset, learning important tumor-related features by minimizing loss through an optimizer. Training and validation accuracy and loss are monitored to ensure proper learning and prevent overfitting.

Visualization and Prediction

Accuracy and loss graphs are generated to assess model performance, and a confusion matrix evaluates classification quality. The trained model then predicts new MRI images, and the identified tumor type is displayed to the user through the GUI.

Chapter 7

CODE IMPLEMENTATION

Input: Brain Tumor MRI Dataset (4 categories: Glioma, Meningioma, Pituitary, No-Tumor)

Output: Predicted Tumor Class (1–) and performance metrics

1. Start

2. Load Dataset
   2.1 Load MRI images from the Brain Tumor dataset folders (Glioma, Meningioma, Pituitary, No-Tumor).
   2.2 Extract images along with their corresponding tumor class labels.
   2.3 Store the image data in the feature set X and the labels in the target vector y.

3. Preprocess Data
   3.1 Resize all MRI images to **150 × 150 pixels** (required for the CNN architecture).
   3.2 Normalize pixel values to the 0–1 range.
   3.3Encode class labels into numerical or one-hot format for multi-class classification.
   3.4 Split the dataset into training and testing sets using:
   - test_split = 0.2
   - random_state = 42

4. Build CNN Model
   4.1 Construct a CNN model with multiple convolutional layers for feature extraction.
   4.2 Add max-pooling layers for dimensionality reduction.
   4.3 Add dropout layers to reduce overfitting.
   4.4 Add dense layers for classification.
   4.5 Add a Softmax output layer to classify MRI scans into **4 tumor categories**.

5. Compile Model
   5.1 Use the Adam optimizer for efficient gradient-based learning.
   5.2 Set the loss function to Categorical Cross-Entropy .
   5.3 Use Accuracy as the primary evaluation metric.

6. Train Model
   6.1 Train CNN model using the training dataset with settings such as:
   - Epochs = 20-40
   - Batch size = 32
   - Validation data = validation split

   6.2 Save the training history, accuracy and loss values, for performance analysis and visualization.

7. Test Model
   7.1 Generate predicted class probabilities for test MRI images.
   7.2 Convert predicted probabilities into class labels using argmax.

8. Evaluate Performance
   8.1 Calculate test accuracy.
   8.2 Generate a classification report with precision, recall, and F1-score.
   8.3 Compute a confusion matrix to analyze class-wise prediction performance.

9. Visualize Results
   9.1 Plot training and validation accuracy curves.
   9.2 Plot training and validation loss curves.
   9.3 Display confusion matrix as a heatmap.

10. Prediction
    10.1 Load the trained brain tumor classification model.
    10.2 Accept a new MRI image from the user through the GUI.
    10.3 Predict and display the corresponding tumor class.

11. End

Chapter 8

RESULT

```
Epoch 39/40
178/178 ━━━━━━━━━━━━━━━━━━━━ 40s 226ms/step - accuracy: 0.9944 - loss: 0.0141 - val_accuracy: 0.9891 - val_loss: 0.037
3 - learning_rate: 9.0000e-05
Epoch 40/40
178/178 ━━━━━━━━━━━━━━━━━━━━ 2s 13ms/step - accuracy: 1.0000 - loss: 5.1291e-04 - val_accuracy: 0.9891 - val_loss: 0.0
371 - learning_rate: 9.0000e-05
40/40 ━━━━━━━━━━━━━━━━━━━━ 2s 61ms/step - accuracy: 0.9891 - loss: 0.0371

Test Accuracy: 0.9891
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This fi
le format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')
` or `keras.saving.save_model(model, 'my_model.keras')`.
✅ Model saved at outputs\brain_tumor_model.h5
41/41 ━━━━━━━━━━━━━━━━━━━━ 3s 61ms/step
```

Fig 8.1 Result of model training

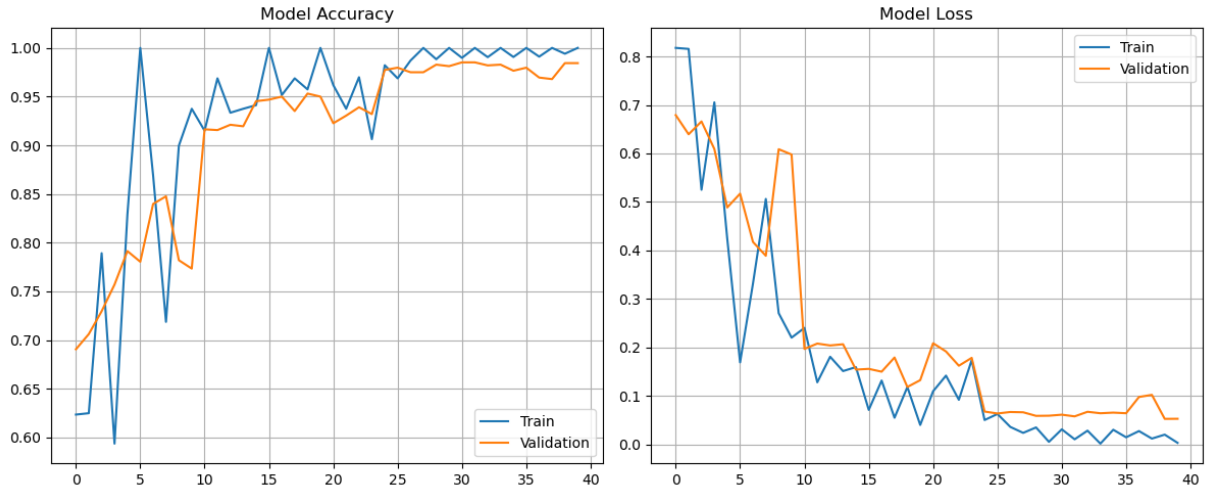

Fig 8.2 Model accuracy and Model loss

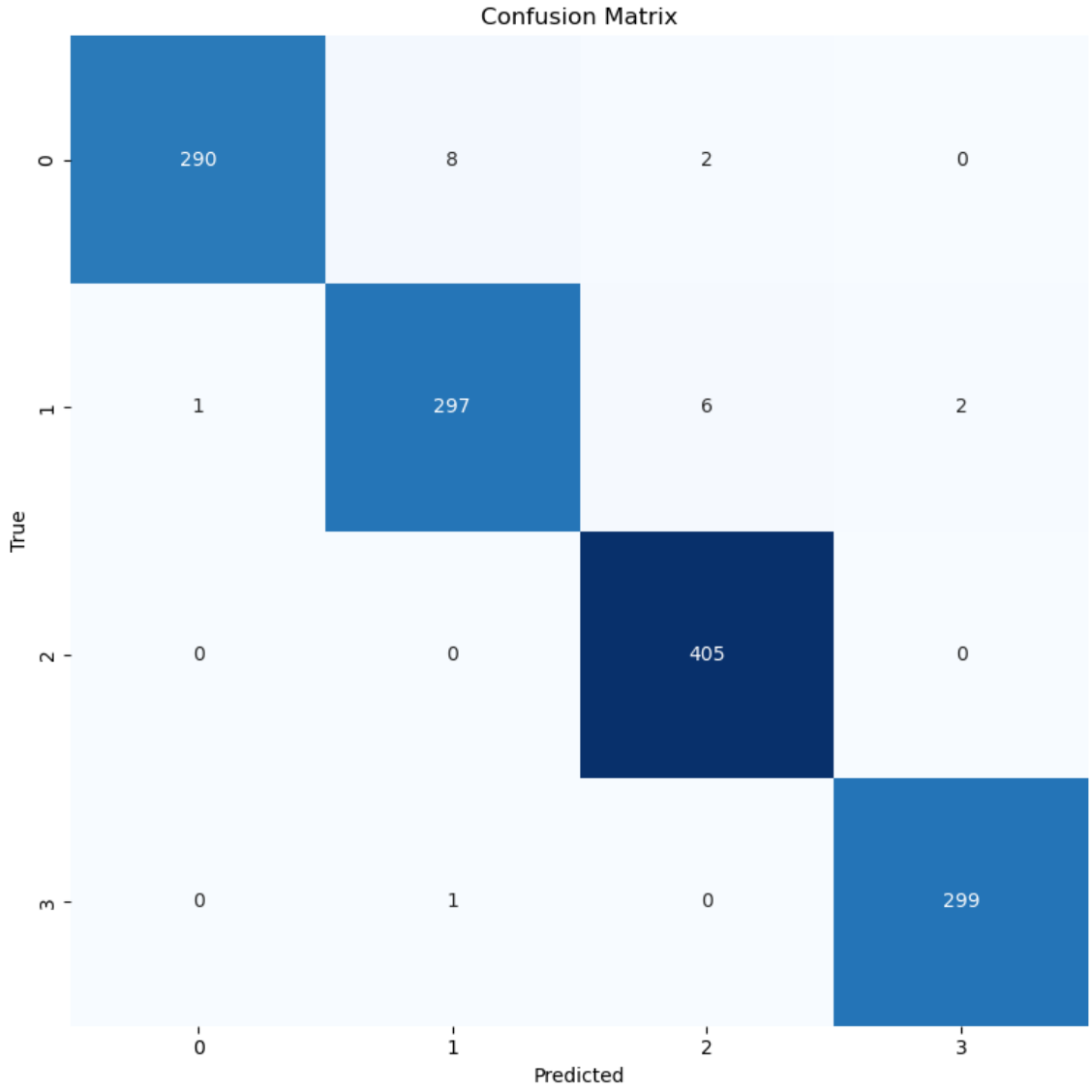

Fig 8.3 Confusion Matrix

Conclusion

This project successfully demonstrates the effectiveness of deep learning for brain tumor classification using a Convolutional Neural Network. By training the model on an MRI-based tumor dataset, the system is able to accurately identify and classify brain tumors into four categories: Glioma, Meningioma, Pituitary, and No-Tumor. Preprocessing techniques such as image resizing, normalization, and data augmentation significantly enhanced the model's learning efficiency and generalization capability.

The experimental results show strong training and validation accuracy, indicating that the model effectively extracts meaningful features from MRI images. Performance evaluation through accuracy and loss graphs, along with the confusion matrix, provides clear insights into the model's stability, reliability, and ability to distinguish between different tumor types.

Furthermore, the integration of a user-friendly graphical interface enables users to upload MRI scans and instantly receive the predicted tumor category, making the system practical and easy to use. Overall, this project highlights the potential of CNN-based deep learning models in medical image analysis, supporting clinical decision-making, early diagnosis, and improving healthcare outcomes.

References

[1] Kumar et al. (2021). CNN-based brain tumor classification model using MRI images, demonstrating that deep CNN architectures effectively capture spatial tumor features and outperform traditional machine-learning methods in medical image analysis.

[2] Ahmed and Yadav (2022). Transfer learning approach using MobileNetV2 for multi-class brain tumor detection, showing improved accuracy and reduced training time, especially for smaller medical datasets.

[3] Li et al. (2022). Study on preprocessing techniques such as normalization, skull stripping, and contrast enhancement, revealing that proper preprocessing significantly boosts CNN performance and robustness for MRI tumor classification.

[4] Thomas and George (2023). Lightweight CNN framework designed for real-time brain tumor detection, concluding that optimized models maintain high accuracy while reducing computational load, enabling use on low-power devices.

[5] Singh et al. (2023). Comparative analysis of deep neural architectures including VGG16, ResNet50, and EfficientNet, reporting that MobileNet and EfficientNet deliver competitive accuracy with fewer parameters and faster inference.