Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Brain Tumor Detection using CNN and Medical Imaging

Ruchith D
Dept. of CSE
SIET, Tumkur, India
1SV23CS092

Varshini M
Dept. of CSE
SIET, Tumkur, India
1SV23CS116

Guide: Dr. Sridhara T, Associate Professor and HOD, Dept. of Cybersecurity

Abstract - Brain tumors are serious neurological abnormalities that require accurate identification and timely clinical evaluation. Magnetic Resonance Imaging (MRI) is widely used to visualize brain structures and detect abnormal tissue. However, manual examination of MRI scans requires expertise and may be time-consuming when large numbers of images must be reviewed. This paper presents a brain tumor classification approach using Convolutional Neural Networks (CNNs) and medical imaging. The implemented training pipeline uses MobileNetV2, a lightweight convolutional architecture, with transfer learning to classify MRI images into four categories: glioma, meningioma, pituitary tumor, and no tumor. The dataset contains 3,264 MRI images, including 2,870 training images and 394 testing images. Image resizing, normalization, and data augmentation are employed during preprocessing. The model is trained using the Adam optimizer and categorical cross-entropy loss for ten epochs. The recorded final training accuracy is 92.41%, while validation accuracy reaches 72.59%. A separate Flask-based application provides image-upload and classification functionality through a vision-language API. The results demonstrate the feasibility of CNN-based MRI classification while identifying limitations related to generalization and deployment consistency. The system is intended as a research prototype rather than a substitute for clinical diagnosis.

Index Terms - Brain Tumor Detection, Convolutional Neural Network, MRI, Medical Image Classification, Deep Learning, MobileNetV2, Transfer Learning, Flask, Artificial Intelligence.

I. INTRODUCTION

Brain tumors are abnormal tissue growths occurring within or around the brain. Depending on their location and biological characteristics, tumors may affect neurological functions and require specialized medical assessment.

Magnetic Resonance Imaging (MRI) provides detailed images of brain anatomy and is commonly used for the investigation of suspected brain abnormalities. The interpretation of MRI scans is primarily performed by trained medical professionals. Automated image analysis can provide additional decision-support information, particularly for organizing and screening large collections of medical images.

Deep learning has become an important technique in medical image analysis. Convolutional Neural Networks (CNNs) automatically learn image features through multiple convolutional layers, reducing the need for manually designed features. The objective of this work is to investigate a CNN-based system capable of categorizing brain MRI images into four predefined classes. The main contributions are:

- Development of a four-class brain MRI classification pipeline.
- Application of MobileNetV2-based transfer learning for feature extraction.
- Preprocessing and augmentation of MRI images for model training.
- Experimental analysis using training and validation accuracy.
- Examination of a Flask-based web application for image submission and classification.

II. LITERATURE REVIEW

Deep learning techniques have received significant attention in medical image analysis due to their ability to learn hierarchical visual representations.

A. Convolutional Neural Networks

LeCun et al. discussed gradient-based learning techniques and the effectiveness of convolutional architectures for image recognition tasks [1]. CNNs use convolution operations, nonlinear activation functions, and feature aggregation to identify visual patterns.

B. Deep Learning in Medical Imaging

Litjens et al. reviewed deep learning applications in medical image analysis, including classification, detection, segmentation, and computer-aided diagnosis [2]. Their work demonstrates the relevance of deep neural networks in interpreting complex medical imaging data.

C. Brain Tumor Classification

Cheng et al. investigated brain tumor classification using MRI images and image-based feature representations [3]. Their research highlights the potential of computational methods for distinguishing different brain tumor categories.

D. Transfer Learning

Transfer learning allows knowledge learned from a large image dataset to be reused for another classification problem. MobileNetV2 introduces inverted residual structures and linear bottlenecks, providing a computationally efficient CNN architecture [4]. This makes it a suitable candidate for experimental medical image classification systems.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

III. PROBLEM STATEMENT

Manual MRI interpretation requires specialized expertise and careful examination of anatomical structures. The visual similarities between certain tumor categories may make automated classification challenging.

The problem addressed in this research is the development and evaluation of a CNN-based classification model that can assign an input brain MRI image to one of four categories:

1) Glioma
2) Meningioma
3) Pituitary Tumor
4) No Tumor

The classification system should process MRI images, extract relevant features, and produce a predicted class.

IV. PROPOSED METHODOLOGY

The experimental methodology consists of dataset preparation, image preprocessing, CNN feature extraction, classification, and performance evaluation.

A. System Architecture

Fig. 1 illustrates the principal stages of the CNN classification pipeline.

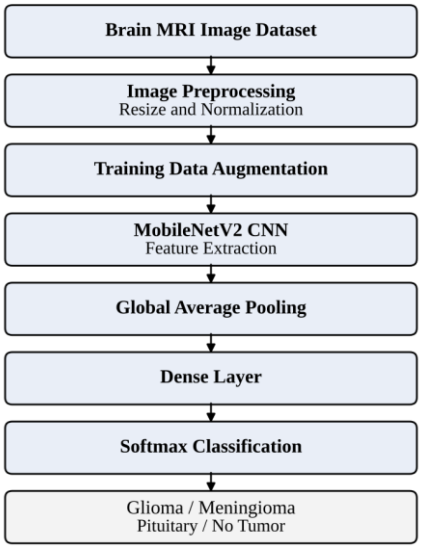


Fig. 1. Architecture of the CNN-based brain MRI classification pipeline.

B. Dataset Description

The supplied dataset contains MRI images organized into separate Training and Testing directories. The images are grouped into four classification categories, as summarized in Table I.

TABLE I. DISTRIBUTION OF BRAIN MRI DATASET

Category	Training	Testing
Glioma Tumor	826	100
Meningioma Tumor	822	115
Pituitary Tumor	827	74
No Tumor	395	105
Total	**2870**	**394**

The complete dataset contains 3,264 images. The class distribution is not perfectly balanced, particularly in the training subset. This imbalance should be considered when interpreting classification performance.

C. Image Preprocessing

The CNN notebook resizes input images to a spatial resolution of 224 × 224 pixels. The images are represented using three color channels. Pixel values are normalized by dividing each intensity by 255:

$$I_n(x, y, c) = I(x, y, c) / 255 \quad (1)$$

where I represents the original pixel intensity and I_n is the normalized value. Normalization maps the pixel intensities into the range [0, 1].

D. Data Augmentation

The training pipeline applies horizontal flipping and random zoom augmentation. The configured zoom range is 0.3. These transformations increase the variation of training inputs and can reduce sensitivity to certain image changes. The testing generator applies normalization without the training augmentation settings.

V. CNN MODEL DESIGN

A. MobileNetV2 Architecture

The implemented model uses MobileNetV2 as its convolutional feature extraction backbone. The pretrained network is initialized using ImageNet weights [5]. The original classification head is removed by setting include_top=False. During the recorded experiment, the backbone layers are frozen. This allows the newly added classification layers to be trained without updating the pretrained convolutional feature extractor.

B. Classification Layers

Global average pooling is applied to the feature maps produced by MobileNetV2. A fully connected layer containing 256 neurons uses the ReLU activation function. The output layer contains four neurons with softmax activation. For an input vector z, the softmax probability for class i is:

$$P(y = i \mid z) = exp(z_i) / \Sigma_{j=1}^{4} exp(z_j) \quad (2)$$

The predicted class is:

$$\hat{y} = argmax_i \, P(y = i \mid z) \quad (3)$$

C. Loss Function

The model uses categorical cross-entropy as its loss function. For N samples and C classes, the loss is:

$$L = -(1/N) \Sigma_{n=1}^{N} \Sigma_{c=1}^{C} y_{nc} \log(p_{nc}) \quad (4)$$

where y_{nc} is the ground-truth indicator and p_{nc} is the predicted class probability.

D. Training Configuration

The model is implemented using TensorFlow [7] and optimized with the Adam optimizer [6]. The CNN training notebook uses the configuration given in Table II.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

TABLE II. CNN TRAINING CONFIGURATION

Parameter	Configuration
Architecture	MobileNetV2
Input Size	224 × 224 × 3
Pretrained Weights	ImageNet
Batch Size	16
Epochs	10
Optimizer	Adam
Loss Function	Categorical Cross-Entropy
Hidden Dense Layer	256 Neurons
Output Classes	4
Output Activation	Softmax
Backbone Training	Frozen

VI. WEB APPLICATION IMPLEMENTATION

A Flask-based web application is also included in the project files. The application provides a landing page, user registration, login, image upload, and result display functionality.

A. User Authentication

The prototype stores user credentials in a local JSON file. A session is used to control access to the protected application pages. This implementation is appropriate only as an initial prototype. Production deployment requires secure password hashing, stronger session management, and protected credential storage.

B. MRI Image Upload

The application accepts an image uploaded by an authenticated user. The image is saved to a server-side upload directory before being processed.

C. Prediction Interface

The supplied Flask code uses an external vision-language model through the OpenRouter API. The model receives an encoded MRI image and a prompt requesting one of the four tumor categories. The returned response contains a predicted category and a short textual description.

This inference mechanism is distinct from the locally trained MobileNetV2 classifier. Consequently, the CNN validation results reported in this paper must not be interpreted as measured accuracy of the Flask API prediction service.

VII. EXPERIMENTAL RESULTS

The training notebook records a ten-epoch MobileNetV2 experiment. The recorded training and validation accuracies are presented in Table III.

A. Accuracy Analysis

Classification accuracy is calculated as:

$$Accuracy = (N_{correct} / N_{total}) \times 100 \quad (5)$$

where $N_{correct}$ and N_{total} are the numbers of correct and total predictions, respectively.

At the final recorded epoch, training accuracy reaches 92.41%. The corresponding validation accuracy is 72.59%.

TABLE III. RECORDED CNN TRAINING RESULTS

Epoch	Train Acc. (%)	Val. Acc. (%)
1	67.72	61.42
2	84.01	59.39
3	87.22	58.38
4	87.42	67.26
5	88.88	69.54
6	90.39	69.80
7	90.34	68.53
8	91.01	68.53
9	92.30	64.72
10	92.41	72.59

The validation accuracy is substantially lower than the training accuracy. This difference suggests that the model does not generalize to the held-out images as effectively as it fits the training data. The recorded results therefore indicate a need for further model optimization and independent evaluation.

B. Loss Analysis

At epoch ten, the recorded training loss is 0.1964. The validation loss at the same epoch is 1.0584. The higher validation loss further indicates a generalization gap. Possible contributing factors include dataset imbalance, differences between training and testing images, and limited adaptation of the frozen feature extractor. These factors require additional controlled experiments before definitive conclusions can be drawn.

C. Performance Limitations

The current notebook reports overall accuracy and loss values. A verified confusion matrix, class-wise precision, recall, and F1-score are not included in the supplied training outputs. Therefore, class-specific diagnostic performance cannot be established from the available results.

In addition, the testing directory is used as validation data during model training. A separate untouched test dataset would provide a more reliable final assessment.

VIII. DISCUSSION

The experiment demonstrates that transfer learning can be applied to four-category MRI classification using a relatively lightweight CNN architecture. MobileNetV2 reduces the need to train a large convolutional network from scratch. However, the gap between training and validation performance shows that the current implementation requires further refinement.

The imbalance between the tumor and no-tumor training categories may influence the learned decision boundaries. Class-balanced evaluation is therefore important.

Another practical consideration is the difference between the trained CNN model and the Flask application's external API classifier. A unified deployment should load the trained CNN weights directly and apply the same preprocessing procedure used during training. Without such integration, the accuracy of the deployed application cannot be inferred from the CNN experiment.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

IX. ADVANTAGES AND APPLICATIONS

The proposed experimental approach offers several potential advantages:

- Automated extraction of MRI image features.
- Classification into four predefined categories.
- Reuse of pretrained convolutional representations.
- Support for image-based research and educational analysis.
- Possibility of integration into a web-based interface.

Potential applications include medical imaging research, academic demonstrations, and development of computer-aided image analysis tools. Any clinical use would require extensive external validation, regulatory consideration, and qualified medical oversight.

X. FUTURE SCOPE

Future development may focus on the following improvements:

1) Fine-tuning selected MobileNetV2 layers to improve generalization.
2) Applying class weighting or other imbalance-handling techniques.
3) Evaluating precision, recall, F1-score, and confusion matrices.
4) Using an independent test set and external MRI datasets.
5) Adding Grad-CAM visualizations for model interpretability [8].
6) Integrating the trained CNN directly with the Flask application.
7) Improving security and privacy controls for medical images.
8) Investigating tumor localization and segmentation.

XI. CONCLUSION

This paper presented a CNN-based brain tumor classification approach using MRI images and transfer learning. The experimental model uses a frozen MobileNetV2 feature extractor with a four-class softmax classification head. The supplied dataset contains 3,264 MRI images covering glioma, meningioma, pituitary tumor, and no-tumor categories.

After ten epochs, the recorded model achieved 92.41% training accuracy and 72.59% validation accuracy. These results establish a working experimental baseline but also reveal a meaningful generalization gap. Further optimization, class-wise evaluation, independent testing, and integration of the trained CNN with the web application are necessary before making stronger performance claims. The work provides a foundation for continued research in CNN-assisted brain MRI analysis.

ACKNOWLEDGMENT

The authors express their sincere gratitude to Dr. Sridhara T, Associate Professor and Head of the Department of Cybersecurity, Shridevi Institute of Engineering and Technology (SIET), Tumkur, for his guidance and support. The authors also thank the Department of Computer Science and Engineering and the institution for providing an academic environment for project development.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] G. Litjens et al., "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[3] J. Cheng et al., "Enhanced performance of brain tumor classification via tumor region augmentation and partition," *PLOS ONE*, vol. 10, no. 10, 2015.

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[5] J. Deng et al., "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[6] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learning Representations*, 2015.

[7] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Systems Design and Implementation*, 2016, pp. 265–283.

[8] R. R. Selvaraju et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Computer Vision*, 2017, pp. 618–626.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)