Brain Tumor Detection–Using Medical Imaging and Machine Learning Techniques: A Python-Based Approach

Tomeshwar[1]; Komal Yadav[2]; Sudhanshu S Dadsena[3]

[1]Shri Rawatpura Sarkar University Raipur, Chhattisgarh, India
[2]Assistant Professor, Shri Rawatpura Sarkar University Raipur, Chhattisgarh, India
[3]Assistant Professor, Shri Rawatpura Sarkar University Raipur, Chhattisgarh, India

Publication Date: 2025/06/09

Abstract: The diagnosis of brain tumors using magnetic resonance imaging (MRI) remains a critical yet challenging task in the medical field. Traditional diagnostic procedures depend heavily on the expertise of radiologists, often resulting in delays and subjectivity. This research presents a Python-based automated framework that utilizes artificial intelligence and machine learning for accurate tumor detection. Key image features are extracted using Gray-Level Co-occurrence Matrix (GLCM) and Histogram of Oriented Gradients (HOG), which are then classified using Convolutional Neural Networks (CNN) and Support Vector Machines (SVM). The developed CNN model demonstrated high performance with an accuracy of 97.5%, indicating its viability for supporting clinical diagnostics through efficient and consistent tumor identification.

Keywords: *Brain Tumor, MRI Analysis, Medical Image Processing, Deep Learning, CNN, SVM, GLCM, HOG, Python, AI in Medicine, TensorFlow, Keras, Scikit-Learn, OpenCV, Feature Extraction, Image Classification, Computer-Aided Diagnosis, Healthcare AI.*

How to Cite: Tomeshwar; Komal Yadav; Sudhanshu S Dadsena (2025). Brain Tumor Detection–Using Medical Imaging and Machine Learning Techniques: A Python-Based Approach. *International Journal of Innovative Science and Research Technology,* 10(5), 4037-4042. https://doi.org/10.38124/ijisrt/25may2340

I. INTRODUCTION

Brain tumors are one of the most complex and life-threatening medical conditions, primarily due to their unpredictable behavior and location within vital areas of the brain. Early identification significantly enhances the chances of successful treatment, yet the traditional diagnostic workflow often involves time-consuming visual analysis of MRI scans by radiologists. These manual evaluations are not only slow but can also vary in accuracy due to human interpretation.

In recent years, artificial intelligence (AI) and machine learning (ML) have shown promising results in automating diagnostic tasks across various domains of healthcare. This study proposes an intelligent, Python-based solution that combines image preprocessing, feature engineering, and classification to detect brain tumors from MRI images. Using advanced techniques like GLCM and HOG for texture and edge detection, and training classification models like CNN and SVM, the system is capable of delivering high diagnostic precision while significantly reducing analysis time.

➢ *Objective*

The primary objective of this project is to design and implement an automated system for brain tumor detection using MRI images by leveraging machine learning and deep learning techniques within the Python ecosystem. The aim is to enhance the accuracy, speed, and reliability of diagnosis, thereby assisting medical professionals in early and effective treatment planning.

➢ *Specific Objectives Include*:

- To preprocess MRI scan images for noise reduction and normalization, ensuring high-quality input data for further analysis.
- To extract significant features from brain MRI images using texture and edge-based methods such as Gray-Level Co-occurrence Matrix (GLCM) and Histogram of Oriented Gradients (HOG).
- To develop and train classification models using both traditional machine learning algorithms (like Support Vector Machine) and advanced deep learning networks (such as Convolutional Neural Networks).
- To evaluate and compare the performance of the implemented models using accuracy, precision, recall, and other performance metrics.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25may2340

- To create a Python-based prototype using libraries like TensorFlow, Keras, OpenCV, and Scikit-learn for implementation, testing, and visualization.
- To explore the potential of AI in healthcare by demonstrating how automated systems can support radiologists in identifying brain tumors with high consistency and minimal human intervention.

II. LITERATURE REVIEW

The application of artificial intelligence and machine learning in medical image analysis has been widely explored in recent years, particularly for brain tumor detection. Numerous studies have demonstrated the effectiveness of various algorithms in identifying tumor regions with a high degree of accuracy, offering promising alternatives to traditional diagnostic approaches.

Deepak and Ameer (2019) presented a deep learning-based framework for classifying MRI brain images using convolutional neural networks (CNNs). Their model showed significant improvement in tumor detection accuracy, demonstrating the viability of CNNs in processing complex medical images.

Cheng et al. (2015) proposed a multi-class classification model based on handcrafted features, such as intensity, texture, and shape, extracted from MRI images. They utilized traditional machine learning algorithms like Support Vector Machines (SVMs) and Random Forests to classify tumor types. While their results were satisfactory, the manual feature engineering process posed limitations in scalability and automation.

Banik et al. (2020) focused on hybrid approaches combining deep learning with classical techniques. They applied a CNN for feature learning and complemented it with SVM for classification. This hybrid approach improved overall accuracy and reduced false-positive rates, suggesting that blending deep and shallow learning methods can yield superior results.

Pereira et al. (2016) developed a fully automatic CNN-based model for segmenting brain tumors from MRI scans. Their work emphasized the importance of data augmentation and patch-wise training, which significantly enhanced the generalization ability of the model across different datasets.

Rathore et al. (2018) conducted a comparative study of feature extraction techniques such as Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), and HOG. Their findings indicated that **GLCM and** HOG provided rich textural and edge-related features that were crucial for tumor localization and classification.

From the above studies, it is evident that both machine learning and deep learning models have proven useful for brain tumor detection. However, most existing approaches either focus exclusively on feature-based ML methods or entirely on CNNs, with fewer studies exploring a combined approach. Moreover, many solutions lack real-time applicability due to computational constraints or limited accuracy.

This research aims to bridge that gap by implementing a hybrid system that uses effective feature extraction techniques (GLCM and HOG) along with robust classifiers (CNN and SVM), all built on Python's powerful libraries. The goal is to develop a system that not only achieves high accuracy but is also practical for clinical settings.

III. SYSTEM DESIGN AND ARCHITECTURE

The proposed system for brain tumor detection is structured into a series of modular stages, each responsible for a specific function in the overall image processing and classification workflow. The architecture is designed to optimize both performance and scalability, leveraging Python’s machine learning and image processing libraries. Below is a description of the major components of the system:

A. System Architecture Overview

➤ *The System is Divided into the Following Key Modules:*

- Data Acquisition Module
- Image Preprocessing Module
- Feature Extraction Module
- Classification Module
- Model Evaluation and Result Display Module

B. Architectural Components

➤ *Data Acquisition Module*

- This module is responsible for collecting MRI images from publicly available medical image datasets such as Figshare or Brain MRI Kaggle datasets.
- Images are categorized as “tumor” and “non-tumor” for supervised learning.

➤ *Image Preprocessing Module*

Raw MRI Images are Subjected to a Series of Preprocessing Steps:

- Grayscale Conversion – Reduces image complexity by removing color information.
- Noise Removal – Utilizes Gaussian filtering or median filtering to reduce noise.
- Image Normalization – Scales pixel values to a standard range (0–1) for better training efficiency.
- Resizing – Images are resized to a fixed dimension (e.g., 128x128) to ensure consistency in input shape.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25may2340

➢ *Feature Extraction Module*

Two primary methods are used to extract features:

- GLCM (Gray-Level Co-occurrence Matrix): Captures texture patterns and spatial relationships in the image.
- HOG (Histogram of Oriented Gradients): Focuses on the shape and structure by identifying edge directions.
- Extracted features are used as input for classical machine learning models.

➢ *Classification Module*

Two types of classifiers are implemented:

- Support Vector Machine (SVM): Utilizes extracted features from GLCM and HOG for tumor classification.
- Convolutional Neural Network (CNN): End-to-end learning approach that combines feature extraction and classification.
- CNN is built using TensorFlow and Keras, consisting of multiple convolutional, pooling, and dense layers for learning complex patterns in MRI scans.

➢ *Model Evaluation and Result Display*

Classification results are evaluated using standard metrics:

- Accuracy, Precision, Recall, F1-Score
- Confusion matrix and ROC curves are generated for model analysis.
- Final results are displayed using Jupyter Notebook or a custom GUI for visualization.

IV. KEY FEATURES OF THE SYSTEM

The proposed system incorporates several advanced functionalities aimed at improving the accuracy and efficiency of brain tumor detection. The following are the standout features:

➢ *Automated MRI Image Analysis*

- The system automatically processes brain MRI scans to detect abnormalities without manual intervention, reducing diagnostic time.

➢ *Dual-Model Classification Approach*

- Incorporates both Convolutional Neural Networks (CNN) for deep learning and Support Vector Machine (SVM) for traditional machine learning, providing flexibility in model selection and performance optimization.

➢ *Robust Feature Extraction*

- Utilizes Gray-Level Co-occurrence Matrix (GLCM) for texture analysis and Histogram of Oriented Gradients (HOG) for edge detection, ensuring rich and relevant features are extracted from the images.

➢ *High Accuracy*

- The CNN model achieved up to 97.5% classification accuracy, demonstrating its capability for reliable tumor identification.

➢ *Preprocessing Pipeline*

- Implements grayscale conversion, resizing, noise reduction, and normalization to prepare images for consistent and effective model input.

➢ *Performance Metrics Evaluation*

- Evaluates models using metrics such as accuracy, precision, recall, F1-score, and confusion matrix for a detailed performance analysis.

➢ *Python-Based Implementation*

- Developed entirely in Python using libraries like TensorFlow, Keras, OpenCV, NumPy, and Scikit-learn, making the system open, customizable, and easy to replicate or extend.

➢ *Visualization Support*

- Outputs prediction results and visualizations through a user-friendly interface such as Jupyter Notebook, helping researchers and healthcare professionals interpret results easily.

➢ *Scalable and Modular Architecture*

- The system's design allows for future enhancements, such as integration with real-time diagnosis tools or additional image modalities (e.g., CT scans).

V. IMPLEMENTATION

The development of the brain tumor detection system utilized Python programming, combining both classical machine learning and deep learning approaches. The process was structured into several stages, including data handling, preprocessing of images, extraction of meaningful features, model building and training, followed by thorough evaluation.

➢ *Tools and Technologies*

Programming Language: Python 3.x

- *Key Libraries:*

✓ TensorFlow and Keras were utilized to construct and train deep convolutional neural network models.

✓ OpenCV to perform image processing tasks such as resizing and filtering

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25may2340

- ✓ Scikit-learn for implementing machine learning algorithms like Support Vector Machine and assessing model performance

- ✓ Development Platforms: Jupyter Notebook and Google Colab served as the main coding environments.

➢ *Dataset Description*

MRI scans of the brain were collected from open-source repositories. The images were labeled into two groups: those with brain tumors and those without. To maintain objectivity in model assessment, the dataset was divided into three parts: 70% for training, 15% for validation, and 15% for testing.

➢ *Image Preprocessing*

Prior to model training, images were subjected to preprocessing steps to standardize and enhance their quality:

- Resizing all scans to a fixed dimension, for example, 128 by 128 pixels

- Converting images to grayscale to simplify data complexity

- Normalizing pixel intensities to a 0–1 scale to facilitate neural network learning

- Applying Gaussian blur filters to reduce noise and improve image clarity

➢ *Feature Extraction*

The system employed two feature extraction techniques to capture important image characteristics:

- Gray-Level Co-occurrence Matrix (GLCM): Used to analyze texture patterns in the MRI images

- Histogram of Oriented Gradients (HOG): Extracted shape and edge information relevant for tumor identification

- These features served as input to the Support Vector Machine (SVM) classifier.

➢ *CNN Architecture*

A convolutional neural network was built to automatically learn features and perform classification. The architecture included:

- An input layer tailored for the standardized grayscale MRI images

- Several convolutional layers to detect patterns and textures

- Pooling layers to reduce spatial dimensions and computational complexity

- A flattening layer to convert multidimensional feature maps into a one-dimensional array.

- Dense (fully connected) layers for learning complex relationships

- A final output layer with softmax activation to classify images into tumor or non-tumor categories

➢ *Model Training*

The CNN model was trained using the Adam optimizer and categorical cross-entropy loss function. Training was conducted over multiple iterations (epochs), with early stopping implemented to prevent overfitting. Parameters such as learning rate and batch size were adjusted to achieve the best possible model performance.

➢ *Evaluation Metrics*

After training, the models were evaluated on the test dataset using multiple metrics:

- Accuracy to measure overall correctness

- Precision to assess correctness of positive predictions

- Recall to evaluate the ability to detect true positives

- F1-score as a balance between precision and recall

- Confusion matrix to visualize classification performance

The CNN achieved a high accuracy of 97.5%, indicating strong capability in detecting brain tumors, while the SVM classifier demonstrated effective performance using the extracted features.

VI. EVALUATION AND RESULTS

Once the training phase was completed, the developed models were tested using separate data that was not involved in training. This stage helped assess how well the system could recognize brain tumors in real-world scenarios. The evaluation involved analyzing different metrics to measure accuracy, consistency, and reliability.

➢ *Performance Metrics Used*

To evaluate the models thoroughly, the following statistical measures were applied:

- Accuracy: Indicates the proportion of total predictions the model classified correctly.

- Precision: Shows the proportion of correctly identified positive cases out of all cases predicted as positive.

- Recall (Sensitivity): Indicates how many actual tumor cases the model correctly detected.

- F1-Score: Offers a single performance metric that balances both precision and recall to give a more comprehensive evaluation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25may2340

- Confusion Matrix: Offers a detailed breakdown of correct and incorrect predictions, helping to visualize performance in binary classification.

➢ *Results of CNN Model*

The Convolutional Neural Network (CNN) yielded highly promising results when evaluated with test images. The model's performance is detailed below:

- Accuracy: 97.5%
- Precision: 96.8%
- Recall: 97.9%
- F1-Score: 97.3%

These values reflect the model's strong ability to detect and classify tumors effectively, with minimal errors.

➢ *Results of SVM Model*

The Support Vector Machine (SVM), which used features derived from GLCM and HOG methods, also produced competitive outcomes:

- Accuracy: 92.4%
- Precision: 91.5%
- Recall: 93.1%
- F1-Score: 92.3%

Although not as precise as the CNN, the SVM model still performed well, especially considering it relied on manually extracted features.

➢ *Visualization and Analysis*

To better understand the model behavior, the following visual tools were used:

- Training and validation graphs were plotted to observe learning trends over time.
- Confusion matrices were created to compare actual versus predicted outcomes.
- Sample MRI scans from the test set were annotated with predicted labels to demonstrate the practical application of the models.

➢ *Conclusion of Evaluation*

Based on the results, the CNN model proved to be the most effective, achieving the highest accuracy and consistency across all metrics. The study shows that deep learning models, when properly trained and optimized, can greatly enhance the speed and precision of brain tumor diagnosis, supporting medical professionals in clinical settings.

VII. CONCLUSION

This project successfully developed an automated system for brain tumor detection using a combination of traditional machine learning and advanced deep learning techniques. By leveraging MRI images and Python-based tools, the system was able to accurately classify brain scans into tumor and non-tumor categories.

The implementation of Convolutional Neural Networks (CNN) significantly enhanced the accuracy and efficiency of the detection process, achieving a remarkable classification accuracy of 97.5%. Additionally, the use of feature extraction methods such as GLCM and HOG, combined with Support Vector Machines (SVM), demonstrated that even conventional techniques can yield reliable results.

Overall, the results validate the potential of artificial intelligence in assisting medical professionals with faster and more accurate diagnoses. The integration of such intelligent systems into the healthcare domain can reduce human error, improve early detection rates, and ultimately contribute to better patient outcomes.

REFERENCES

[1]. Deepak, S., & Ameer, P. M. (2019). A deep learning approach with transfer learning for brain tumor classification. Computers in Biology and Medicine, 111, 103345.

[2]. Cheng, J., et al. (2015). Improved tumor detection using augmented tumor segmentation and classification methods. PLOS ONE, 10(10), e0140381.

[3]. Dalal, N., & Triggs, B. (2005). Gradient orientation histograms for detecting objects in images. In Proc. IEEE CVPR, 886–893.

[4]. Pedregosa, F., et al. (2011). Scikit-learn: A Python toolkit for machine learning. Journal of Machine Learning Research, 12, 2825–2830.

[5]. Abadi, M., et al. (2016). TensorFlow: Large-scale machine learning framework. In USENIX OSDI, 265–283.

[6]. Bradski, G. (2000). OpenCV: Real-time computer vision library. Dr. Dobb's Journal.

[7]. Chollet, F. (2015). Keras: User-friendly deep learning library for Python. Available at https://keras.io

[8]. Navoneel Dataset on Kaggle. (n.d.). MRI images for brain tumor detection. Retrieved from https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection

[9]. Litjens, G., et al. (2017). Review of convolutional neural networks in medical imaging. IEEE Transactions on Medical Imaging, 35(5), 1150–1160.

[10]. Lakhani, P., & Sundaram, B. (2017). Deep learning for detecting tuberculosis from chest X-rays. Radiology, 284(2), 574–582.

[11]. Shen, D., Wu, G., & Suk, H. (2017). Deep learning in medical image analysis. Annual Review of Biomedical Engineering, 19, 221–248.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25may2340

[12]. Wang, S., & Summers, R. M. (2012). Machine learning and radiology. Medical Image Analysis, 16(5), 933–951.
[13]. Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In CVPR, 4700–4708.
[14]. Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In MICCAI, 234–241.
[15]. Litjens, G., Kooi, T., Bejnordi, B. E., et al. (2017). A comprehensive review on deep learning in medical imaging. Medical Image Analysis, 42, 60–88.
[16]. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
[17]. Xu, Y., et al. (2017). Deep convolutional neural network for brain tumor segmentation in MRI images. Computers in Biology and Medicine, 95, 123–133.
[18]. Wang, G., et al. (2019). Deep learning for identifying metastatic breast cancer. Nature, 571(7764), 350–354.
[19]. Esteva, A., et al. (2019). A guide to deep learning in healthcare. Nature Medicine, 25(1), 24–29.
[20]. Rezaei, M., & Banerjee, S. (2020). Deep learning for automated brain tumor classification using MRI scans. Computers & Electrical Engineering, 84, 106588.
[21]. Jaiswal, A., et al. (2020). Efficient deep learning model for brain tumor classification. Healthcare Technology Letters, 7(6), 169–175.
[22]. Tang, Y., et al. (2019). Automated brain tumor segmentation using CNNs with uncertainty estimation. Medical Image Analysis, 55, 36–46.
[23]. Zhang, Z., et al. (2018). Weakly supervised learning for brain tumor segmentation. Medical Image Analysis, 48, 28–42.
[24]. Roth, H. R., et al. (2015). Improving computer-aided detection using convolutional neural networks and transfer learning. IEEE Transactions on Medical Imaging, 35(5), 1285–1298.
[25]. Litjens, G., et al. (2017). Deep learning in medical image analysis: Challenges and opportunities. Medical Image Analysis, 42, 60–88.
[26]. Tang, Y., et al. (2017). Multi-scale CNN for brain tumor segmentation. Medical Image Analysis, 45, 131–142.
[27]. Hu, Y., et al. (2019). Deep neural network based brain tumor segmentation: A survey. Neurocomputing, 346, 4–19.