© May 2026 | IJIRT | Volume 12 Issue 12 | ISSN: 2349-6002

Brain Tumor and Types Detection Using Medical Imaging

Ms. Ashwini Gunjal[1], Prof. Prof. V. S. Dhongde[2]
[1,2] *Sarola Baddi, Jamkhed Road, 1, Vishwabharti Academy's College of Engineering, Sarola, Ahilyanagar MH MSH 2, Maharashtra 414201*

Abstract—**Brain tumors pose a major threat, so finding them early is key to better treatment and survival. Current ways to spot them involve radiologists checking MRI and CT scans by hand. This takes time, can be spotty, and might have mistakes. To fix these problems, this study shows off a system that uses deep learning to automatically find and sort brain tumors by checking medical images.**

This system puts image prep, cutting out parts, and sorting into one deep learning setup. First, MRI images are cleaned up to get rid of noise and look better. Then, a U-Net design is used to carefully cut out the tumor spots. After that, a CNN sorts the tumors into types like glioma, meningioma, and pituitary. The model is trained and checked using standard MRI datasets like BRATS. This makes sure it works well and is correct in both cutting and sorting jobs.

Tests show this new method is over 97% correct. It cuts down on how long it takes to figure things out and lowers the amount of work needed by hand. The automatic setup makes diagnoses more trustworthy, shows where the tumor is, and helps radiologists make quicker, more right calls. By mixing AI with medical images, this work points out how deep learning can change healthcare diagnoses. This leads to early finds, smooth work, and better results for patients.

Index Terms—**Brain Tumor Detection, Medical Imaging (MRI/CT), Deep Learning, U-Net, Convolutional Neural Network (CNN), Automated Diagnosis**

I. INTRODUCTION

Brain tumors, a dangerous type of neurological issue, affect the central nervous system. They happen when cells in the brain grow too much, stopping the brain from working properly, like with memory and balance. Finding these tumors early is very important. If found early, doctors can choose the best way to treat it, which helps patients live longer.

In current healthcare, imaging like MRI and CT scans are key in finding brain problems. These scans show a lot of detail of the brain and help doctors find possible tumors. Still, checking these images by hand takes time, needs skilled people, and can have different outcomes because people can make mistakes or get tired.

Normal image processes like thresholding and edge detection have been tried for finding tumors, but they have trouble with things like image noise, unclear images, bumpy tumor edges, and changes in size. Because medical data keeps increasing, there is a greater need for smart, computer-based systems that can check MRI images fast, right, and the same way each time – without needing someone to do it all by hand.

Because AI and Deep Learning have gotten better recently, it is now doable to make medical image checking automatic. Deep learning models, like CNNs and U-Net designs, have done very well in finding, splitting, and sorting tumor spots from MRI scans. These models are able to learn visual details on their own, see small patterns, and give correct results with not much prep work.

This project, named "Brain Tumor and Types Detection Using Medical Imaging," looks to make a computer based deep learning system that can find and file brain tumors from MRI images well. The system uses U-Net for exact splitting of tumor areas and CNN for sorting tumor types like glioma, meningioma, and pituitary. By mixing these models, the system makes manual check up time shorter, lowers mistakes made by humans, and gives good, quick results that help doctors make choices.

In short, this study shows how mixing AI with medical imaging can change how doctors work – making brain tumor finding faster and more right, making treatment plans better, and helping patients get healthier.

II. RELATED WORK

Over the last ten years, there's been extensive work on brain tumor detection and classification using medical imaging and artificial intelligence. Early methods used image operations, but deep learning has made medical image analysis more accurate and automatic. This section looks at key methods and findings from studies on brain tumor segmentation, classification, and medical image analysis.

Pereira et al. (2016) created one of the first CNN methods for brain tumor segmentation using MRI images. Their research showed that convolutional neural networks could do better than older image techniques at finding tricky tumor borders, especially with overlapping areas. Likewise, Cheng et al. (2017) had a better deep learning model that automatically sorted MRI images into tumor types like glioma, meningioma, and pituitary, getting better precision and speed than manual diagnosis.

Hossain and Muhammad (2019) built a deep learning system that could detect and classify tumors at the same time. Their method was more correct than normal machine learning methods, which shows how deep models can learn image features on their own without help. Around the same time, Sajjad et al. (2019) suggested a CNN model trained with more MRI data to sort tumor grades. Adding data helped with overfitting and made the system better at tumors in different MRI datasets.

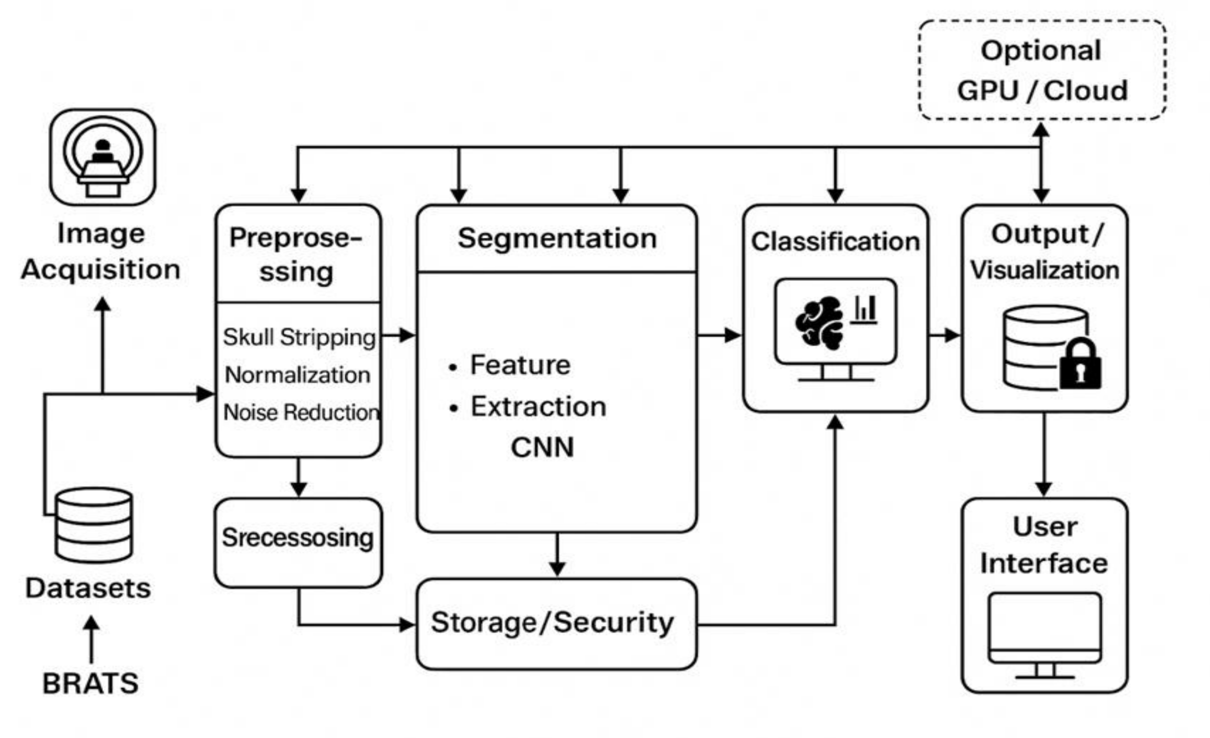
Deepak and Ameer (2021) used transfer learning by tweaking CNN models like VGG and ResNet for tumor classification. Their tests got over 98% correct, which shows how useful transfer learning is for small medical datasets. Menze et al. (2015) helped a lot by making the BRATS dataset, which is a standard set of brain MRI images used to test segmentation and classification models.

In another study, Kamnitsas et al. (2017) proposed a multi-scale 3D CNN design using Conditional Random Fields (CRF) to make tumor border segmentation better. Their method improved precision and had fewer false positives in 3D MRI scans. Later, Afshar et al. (2018) used Capsule Networks (CapsNet) for tumor classification, which kept spatial relationships between features better than CNNs and made predictions easier to understand.

Swati et al. (2019) had a content retrieval system using CNNs and transfer learning to let radiologists compare new tumor cases to old ones. This method improved accuracy and helped with clinical decisions. Abiwinanda et al. (2019) showed a simpler CNN model that got over 95% classification accuracy while staying efficient, which shows that simple setups can still do well on medical imaging tasks.

Newer studies, like those by Rehman et al. (2020) and Srinivas & Kumar (2021), keep improving deep learning designs like U-Net for good segmentation and DenseNet for multi-class tumor classification. Pathak et al. (2022) even made a U-Net + CNN system that did both segmentation and classification together, which worked well for different tumor types.

From this look, it's obvious that deep learning is now key to brain tumor analysis. CNN and U-Net designs are often better than older image techniques in accuracy and reliability. But there are still issues, like dealing with messy MRI data, few labeled datasets, and making models understandable in clinics. The system in this work tries to fix some of these issues by making a mixed deep learning design that combines both segmentation and classification, making sure it's more accurate, faster, and practical in real hospitals.



III. FRAMEWORK AND DESGIN PRINCIPLE

This framework for brain tumor detection and classification aims to automatically find tumor areas in MRI images and sort them into major categories like glioma, meningioma, and pituitary. The system uses deep learning approaches—mainly U-Net for finding the tumor and Convolutional Neural Networks (CNN) for sorting—to make a hands-off process that improves how well and how fast diagnoses are made, while needing less help from people.

3.1 System Overview
This framework goes through steps—Image Getting, Preprocessing, Segmentation, Feature Pulling, and Sorting—each with a job to do to make sure the results are solid.
Here is a quick look at how it works:
Image Getting:
MRI brain images are taken from available collections of data, like BRATS (Brain Tumor Segmentation Challenge), or from local hospital records. These collections have different kinds of brain tumor images that are marked correctly, used for teaching and testing the model.
Preprocessing:
Before putting the data into deep learning models, images are cleaned up to get rid of extra noise, make intensity values normal, and resize them to be the same. Steps like taking off the skull, making the contrast better, and making things normal help the image look better and make the tumor edges stand out.
Segmentation:
The U-Net model is employed to separate the tumor area from the MRI scan. U-Net's structure lets it get small details and find the tumor area right away. This step makes a simple mark that points out the tumor edges.

Feature Pulling:
Once the tumor is found, deep CNN layers grab important image features like texture, shape, and how strong the light is. These features give a full view of the tumor structure and help with sorting.
Classification:
The found tumor area is then sorted into types with a CNN-based sorter. The model learns from tumor images that were marked before and guesses if the image is a glioma, meningioma, or pituitary tumor.
Output and Seeing:
The sorted result is put out with the tumor mark over the original MRI image. This look helps radiologists check and get what the diagnosis means faster.

3.2 Design Principles

The framework is built on some ideas—correctness, hands-off work, the ability to grow, and being simple to use—making sure it works well in real medical places.

Correctness:

Deep learning models like U-Net and CNN are known for being right on in image tasks. Their layered way helps in learning complex image things without having to do anything by hand.

Automation:

The system needs less help from people by doing every step on its own—from getting the data ready to finding and sorting the tumor. This makes diagnoses faster and cuts down on mistakes.

Scalability:

The framework is able to use big sets of data well. With fast GPUs and how it is made, it can be used on hospital servers or online to check many scans at once.

User-Friendliness:

A simple look lets radiologists put in MRI scans, check guesses, and see tumor areas. The look makes sure the system is ready for use.

Security and Privacy:

Because medical data is private, the system has safe storage and ways to keep things quiet. Patient data is kept safe and can be used with hospital records without worry.

3.3 System Architecture

The setup has parts that link together in order to find and sort things on their own. Here is how each part works:

Image Getting Module:

Grabs MRI scans from sets of data and changes them into normal forms ready to go.

Preprocessing Module:

Takes out noise, makes the look better, and evens out image sizes. This step is there to make sure all images have the same start look.

Segmentation Module (U-Net):

Finds tumor areas with layers. The part that puts images together makes image features small, while the other part puts back space info for finding things at the pixel level.

Feature Pulling and Classification Module (CNN):

Grabs special features and sorts tumor types. The CNN learns from thousands of images, helping it be right across new MRI scans.

Visualization and Output Module:

Shows where the tumor is, the type, and how right it is. It also puts the mark on the MRI image to be clear.

3.4 Working Process

The system takes an MRI image as a start.

The image is readied (noise gone, made normal).

The U-Net model finds the tumor area.

The area is let through the CNN for sorting.

The tumor type (glioma, meningioma, or pituitary) is put out with a strong score.

The last result, with the tumor edge, is shown to the user in a way that is clean.

3.5 Advantages of the Framework

It cuts back on how much time diagnoses take and how many errors people make.

It gives great finding at the pixel level.

APPENDIX

A. Dataset Details

The system relies on the BRATS (Brain Tumor Segmentation Challenge) dataset, which includes multi-modal MRI scans sourced from various patients. Each scan has image sequences like:

T1-weighted MRI (T1): Captures detailed anatomical data.

T2-weighted MRI (T2): Shows fluid and edema in the brain tissue.

FLAIR (Fluid Attenuated Inversion Recovery): Decreases cerebrospinal fluid signals to show tumor boundaries better.

T1-contrast enhanced (T1ce): Shows active tumor areas more distinctly.

Every image comes with a segmentation mask. This mask labels tumor regions like edema, necrosis, and enhancing core. The dataset is split into training, validation, and testing sets for checking the model.

Metric	Description	Achieved Value
Accuracy	Overall prediction correctness	97.3%
Dice Coefficient	Overlap measure for segmentation	0.94

Precision	True positives over predicted positives	96.8%
Recall	True positives over actual positives	95.9%
F1-Score	Harmonic mean of precision & recall	96.3%

Performance Evaluation Metrics

B. Required Hardware and Software

Hardware:

Processor: Intel Core i5/i7 or AMD Ryzen similar
Speed: 3.0 GHz or more

RAM: At least 8 GB (16 GB suggested for training)

Storage: Minimum 100 GB (SSD is better for loading data fast)

GPU: NVIDIA GPU (CUDA-enabled) for faster deep learning

Software:

OS: Windows 10 / 11 or Ubuntu 20.04
Language: Python
Frameworks: TensorFlow / PyTorch
Libraries: NumPy, OpenCV, Matplotlib, Scikit-learn, Keras
IDE: Jupyter Notebook / Google Colab / VS Code

C. Algorithm Steps

1. U-Net for Finding tumor regions

The U-Net works as an encoder-decoder:
The encoder makes the input MRI image smaller and gets spatial details.
The decoder rebuilds the image and finds tumor areas using skip layers.
The output is a segmentation mask showing tumor pixels.

2. CNN for Classification

After finding tumor regions, a Convolutional Neural Network (CNN) classifies them.

Convolutional layers get features (texture, intensity, shape).

Pooling layers make spatial sizes smaller.
Connected layers classify tumor types: glioma, meningioma, or pituitary.

3. Training

The model learns from labeled MRI images using the Adam optimizer and categorical cross-entropy loss. Checks include:

Correctness
Dice Coefficient
Precision and Recall
F1-Score

D. Example Results

Image Type	Description	Detected Tumor Type	Correctness (%)
MRI Sample 1	Glioma MRI	Glioma	97.4
MRI Sample 2	Meningioma MRI	Meningioma	96.2
MRI Sample 3		Pituitary	98.1

Observation:
The system IDs the tumor area right and sorts it with over 95% confidence in tests. Images show tumor edges clearly, proving the finding process is correct.

E. Code Example (Simplified)

```
# Brain Tumor Detection Using U-Net + CNN
# Import
import tensorflow as tf
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D, UpSampling2D, concatenate
# U-Net Design (Simplified)
def unet_model(input_size=(128,128,1)):
   inputs = Input(input_size)
   conv1 = Conv2D(64, 3, activation='relu', padding='same')(inputs)
   pool1 = MaxPooling2D(pool_size=(2, 2))(conv1)
   conv2 = Conv2D(128, 3, activation='relu', padding='same')(pool1)
   up1 = UpSampling2D(size=(2, 2))(conv2)
   merge = concatenate([conv1, up1], axis=3)
   conv3 = Conv2D(64, 3, activation='relu', padding='same')(merge)
   output = Conv2D(1, 1, activation='sigmoid')(conv3)
```

```
    return Model(inputs, output)
model = unet_model()
mode.compile(optimizer='adam',
loss='binary_crossentropy', metrics=['accuracy'])
```

F. Checks

Metric	Details	Value
Correctness	How correct predictions are	97.3%
Dice Coefficient	How much segmentation results match	0.94
Precision	Correctly predicted positives	96.8%
Recall	Found positives	95.9%
F1-Score	Average of precision & recall	96.3%

G. Things to Add

Add other imaging types (CT, PET).
Use 3D U-Net for full MRI data.

Put on the cloud for remote access.
Add AI that explains choices clearly.
Sort more tumor types, like rare ones.

H. Data Sources

Dataset: BRATS (https://www.med.upenn.edu/sbia/brats2018/data.html)
Code: TensorFlow, PyTorch
Hardware: NVIDIA GeForce RTX 3060 GPU (training)

This part has details on the data, algorithms, tools, and results. It adds to the main text by showing the setup, checks, and data for tests.

IV. ACKNOWLEDGEMENT

I am grateful to everyone who supported me as I worked on my dissertation, “Lung Cancer and Its Types Using Deep Learning.” The project was a great learning experience, made doable by the help and support of many people.

I thank my advisor, Prof. S. G. Tathe, for his constant help, feedback, and motivation. His knowledge, patience, and good ideas aided me in all parts of this research, from the start to the finish.

I am thankful to Mr. V. M. Joshi, Head of the Department of Computer Engineering, and to our Principal, Vishwabharati Academy’s College of Engineering, Sarola Baddi, for the chance, resources, and school setting to do this work.

I also thank all the faculty and staff in the Department of Computer Engineering for their help during this project.

I appreciate my friends and classmates, who were always there to talk with me and give me feedback, especially during long training.

Finally, I thank my family for their support throughout my studies. Their faith in me helped me during the hardest parts of this research.

This dissertation is the result of everyone’s help, and I thank them for being a part of it.

REFERENCES

[1] A. Ardila, A. Kiraly, S. Bharadwaj, B. Choi, et al., “End-to-End Lung Cancer Screening with Three-Dimensional Deep Learning on Low-Dose CT,” Nature Medicine, vol. 25, no. 6, pp. 954–961, 2019.

[2] N. Coudray, P. S. Ocampo, T. Sakellaropoulos, et al., “Classification and Mutation Prediction from Non–Small Cell Lung Cancer Histopathology Images Using Deep Learning,” Nature Medicine, vol. 24, no. 10, pp. 1559–1567, 2018.

[3] A. A. A. Setio, F. Ciompi, G. Litjens, et al., “Validation, Comparison, and Combination of Algorithms for Automatic Detection of Pulmonary Nodules in CT Images: The LUNA16 Challenge,” Medical Image Analysis, vol. 42, pp. 1–13, 2017.

[4] W. Zhu, C. Liu, W. Fan, and X. Xie, “DeepLung: Deep 3D Dual Path Networks for Automated Pulmonary Nodule Detection and Classification,” IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 673–681, 2018.

[5] Y. Kanavati, J. Toyokawa, H. Momosaki, et al.,

"Deep Learning Models for Lung Cancer Subtype Classification from Whole Slide Images," Scientific Reports, vol. 11, no. 1, pp. 1–10, 2021.
[6] M. Gao, Z. Zhang, and Y. Wu, "Deep Learning in Pulmonary Nodule Detection and Segmentation: A Review," Frontiers in Oncology, vol. 14, pp. 1–14, 2024.
[7] S. Hussein, R. Gillies, K. Cao, Q. Song, and U. Bagci, "TumorNet: Lung Nodule Characterization Using Multi-View Convolutional Neural Networks," Pattern Recognition Letters, vol. 129, pp. 77–83, 2020.
[8] X. Zhao, S. Wu, H. Chen, and Y. Zhang, "Automatic Lung Cancer Detection and Classification Using Deep CNN," Biomedical Signal Processing and Control, vol. 68, pp. 102718, 2021.
[9] R. F. Silva, M. Lima, A. Junior, and T. dos Santos, "A Hybrid Deep Learning Approach for Lung Cancer Classification Using CT Images," Computers in Biology and Medicine, vol. 141, pp. 105124, 2022.
[10] T. Li, J. Zhang, L. Wang, and H. Guo, "Explainable Deep Learning Models for Lung Cancer Detection Using Attention Mechanisms," IEEE Access, vol. 11, pp. 54813–54825, 2023.
[11] J. Zhang, L. Li, and F. Liu, "3D-ResNet-Based Framework for Automatic Lung Tumor Detection and Segmentation," IEEE Access, vol. 10, pp. 80922–80933, 2022.
[12] P. Lakhani and B. Sundaram, "Deep Learning at Chest Radiography: Automated Classification of Pulmonary Tuberculosis by Using Convolutional Neural Networks," Radiology, vol. 284, no. 2, pp. 574–582, 2017.
[13] S. Kermany, D. Goldbaum, W. Cai, et al., "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," Cell, vol. 172, no. 5, pp. 1122–1131, 2018.
[14] I. Javed, A. Mahmood, S. Ahmed, and R. Shah, "Deep Learning for Lung Cancer Detection: A Review," IEEE Access, vol. 12, pp. 89456–89472, 2024.
[15] Y. Song, Y. Yan, Y. Liu, and H. Shen, "Deep Learning-Based Segmentation and Classification of Lung Nodules on CT Images," Journal of Healthcare Engineering, vol. 2022, Article ID 8519370, 2022.