



Texture-Based Feature Extraction and Classification of Brain Tumors Using Mask Region-Based Convolutional Neural Network-Long Short-Term Memory Models

A Sravanthi Peddinti¹, Suman Maloji^{2*}

Department of ECE, Koneru Lakshmaiah Education Foundation, KLEF, Vaddeswaram 522302, India

Corresponding Author Email: suman.maloji@kluniversity.in

Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmep.120733>

ABSTRACT

Received: 19 December 2024

Revised: 30 June 2025

Accepted: 11 July 2025

Available online: 31 July 2025

Keywords:

features extraction, Region-Based Convolutional Neural Network (RCNN), Magnetic Resonance Imaging (MRI) brain images, tumor classification

Accurate brain tumor classification plays a critical role in enhancing diagnosis and treatment planning in medical imaging. This research presents a fusion-based deep learning framework utilizing texture-based feature extraction methods—Multivariate Local Texture Pattern (MLTP), Gray-Level Co-Occurrence Matrix (GLCM), Local Energy-Based Shape Histogram (LESH) and Region-based Convolutional Neural Network-Long Short-Term Memory (RCNN-LSTM) models for brain tumor classification. The process begins with dataset collection followed by annotation using the YOLO application to ensure precise region identification. Training is performed using Mask-RCNN with a batch size of 8 and 15 epochs, and further complemented by parameter tuning for model optimization. Real-time sample testing is conducted to evaluate the model's robustness. Post-processing is applied to refine predictions, and performance metrics such as accuracy, recall, F1-score, and throughput are calculated to assess model efficacy. The final results are deployed on Edge AI platforms to facilitate real-time clinical applications. Experimental results demonstrate the effectiveness of the proposed approach, showcasing high accuracy and improved throughput, thus highlighting the potential of integrating RCNN and LSTM architectures for reliable brain tumor classification. The proposed model attains an accuracy of 99.87%, recall of 99.73%, precision of 97.21%, specificity of 98.24%, F1 score of 99.36%, sensitivity of 98.12%, dice coefficient of 98.41%, throughput of 98.37%, latency 0.99% and mAP of 99.12%. These results represent a significant improvement compared to earlier models. The implemented application has been tested on KLEF University staff and external patients, yielding accurate results.

1. INTRODUCTION

Medical scan images, particularly Magnetic Resonance Imaging (MRI) scans, are critical in diagnosing deeply affected areas within the human body. These scans use layer-by-layer analysis to provide detailed insights into various medical conditions. MRI is a widely used imaging modality that offers valuable technical information about the size, location, and type of tumors. It works by detecting the behavior of protons in response to radio frequencies and equilibrium states, enabling the identification of subtle changes in tissue composition. MRI scans deliver high-resolution images that can distinguish between different types of brain tissues, such as blood oxygenation levels and water diffusion, making them highly effective in diagnosing brain abnormalities. MRI scans are particularly useful in detecting degenerative brain diseases such as brain tumors, transient ischemic attacks (TIAs), and cancer. Globally, these conditions affect approximately 1 million people, with brain tumors and TIAs impacting over 15% of the population. Annually, around 12 out of every 1,000 individuals are diagnosed with brain disorders linked to tumors or degenerative diseases. Advanced neuroimaging techniques

and biomarkers have been instrumental in identifying these conditions, improving the early detection of brain tumors and cancers. Deep learning models have become essential in analyzing MRI data for brain tumor detection. These models leverage large datasets to accurately classify and differentiate between normal and abnormal brain tissues. They are particularly effective in identifying cognitive impairments associated with brain disorders. By using deep learning algorithms, it is possible to enhance diagnostic accuracy, thereby aiding in the early detection and management of brain diseases. As brain tumors and other neurological disorders significantly impact global health and socio-economic development, the use of advanced imaging techniques combined with machine learning is crucial for timely diagnosis and treatment. The multivariate local texture pattern (MLTP) for MRI brain image feature extraction is a sophisticated method for evaluating texture patterns in MRI brain images. MLTP is very beneficial for detecting local texture differences that aid in the diagnosis of neurological illnesses including as Alzheimer's, Parkinson's, and malignancies. Texture information is crucial for detecting structural problems in medical imaging such as MRIs. Multivariate statistical techniques are used to capture the

connections between pixel intensities across several modalities. Calculate multivariate texture characteristics that represent directional and intensity variations. PCA, t-SNE, or feature selection techniques are frequently used to decrease high-dimensional MLTP features and enhance computing performance. Prior to classification, normalize the feature vectors to achieve consistent scaling.

The Gray-Level Co-Occurrence Matrix (GLCM) is a statistical approach for obtaining texture features from photographs. It measures the spatial connection between pixel intensity levels by calculating how frequently pairs of pixel values occur at a given distance and angle. Key characteristics retrieved from GLCM include contrast (measures local intensity changes), correlation (measures linear dependency of pixel pairings), energy (measures textural uniformity), and homogeneity (measures pixel value proximity). These traits are commonly utilized in MRI analysis to distinguish between textures representing normal and diseased brain tissues. GLCM is a computationally efficient and effective solution for medical image analysis. The Local Energy-Based Shape Histogram (LESH) approach extracts form and texture characteristics from picture areas by utilizing local energy responses. It uses Gabor filters to calculate local energy and encodes the results into histograms for each patch of the image. These histograms are concatenated to create a feature vector, which represents shape and texture information. LESH works at numerous sizes and orientations, ensuring resistance to changes in size, rotation, and light. It is widely utilized in a variety of applications, including object identification, medical imaging (such as MRI tumor diagnosis), and image retrieval. LESH is computationally efficient and includes discriminative characteristics to ensure proper classification. The categorization of brain tumor MRI images using a combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks is a reliable method that combines spatial and temporal feature learning to provide correct diagnosis. The process starts with preprocessing the MRI images, which includes skull stripping to remove non-brain tissues, normalization to scale pixel intensities within a consistent range (e.g., [0,1]), resizing the images to a uniform dimension suitable for CNN input (e.g., 224×224), and using data augmentation techniques like rotation, flipping, and scaling to improve dataset diversity and reduce overfitting.

CNN is then used to extract spatial characteristics from MRI images. Its convolutional layers capture hierarchical characteristics including edges, textures, and tumor borders, while pooling layers minimize dimensionality without sacrificing important spatial information. VGG16, ResNet, and Inception are examples of pretrained models that may be utilized to improve feature extraction using transfer learning. These collected spatial characteristics are sent into the LSTM network, which processes them sequentially to capture temporal or spatial relationships, allowing the model to assess differences across picture patches or slices. Bidirectional LSTM can enhance performance by learning dependencies in both forward and backward directions. The CNN and LSTM outputs are combined to create a robust feature representation, which is then fed via fully linked dense layers for classification. For multi-class or binary classification tasks, such as discriminating between benign and malignant tumors, the output layer employs a Softmax or sigmoid activation function. The model is trained utilizing optimizers like Adam and an appropriate loss function, such as categorical or binary cross-entropy. Finally, the model's

performance is measured using measures such as accuracy, precision, recall, F1-score, and AUC-ROC, as well as k-fold cross-validation to assess robustness and generalizability. This CNN-LSTM fusion combines CNN's capacity to learn specific spatial information with LSTM's capability in capturing dependencies, resulting in a potent tool for brain tumor classification. The earlier models only concentrate on tumor size and located area but orientation of disease and related information has not identified. The tumor effected area and disease influence on other organs information have not been added to which is most trending research in brain medical image computer vision tasks.

2. RELATED METHODS

Deep learning algorithms have significantly improved medical picture segmentation and classification, notably in the identification and diagnosis of brain tumours utilizing MRI data. Recent research shows that combining sophisticated architectures such as Region-based Convolutional Neural Network (RCNN) and LSTM can improve the capacity to accurately diagnose brain tumors by using both spatial and sequential data. Texture analysis is an important component in brain tumor classification since it gives information about the structural patterns in MRI images. Traditional techniques relied on statistical methods such as the GLCM to capture textural patterns. However, deep learning approaches have taken over this industry. For example, Azad et al. [1] found that U-Net is effective at collecting detailed texture information for segmentation tasks. Similarly, Shao et al. [2] emphasized the use of optimal clustering and U-Net for identifying relevant patterns for classification. These developments pave the way for hybrid systems like RCNN-LSTM to capture both textural and temporal relationships. CNNs form the foundation of current medical image processing, especially feature extraction. The success of U-Net and its derivatives is well established. Xiao et al. [3] examined the use of transformers in medical segmentation, demonstrating their ability to increase feature extraction through better contextual comprehension. Furthermore, research by references [4, 5] contrasted CNN and U-Net performance, demonstrating their capacity to capture specific spatial data required for correct categorization. While CNNs are excellent at collecting spatial information, Long Short-Term Memory (LSTM) networks can describe sequential relationships. This integration is especially important for 3D MRI images, as the slices carry contextual links. Baccouch et al. [6] used U-Net for 3D MRI segmentation, proving its capacity to identify brain cancers. Extending this using LSTM allows you to capture inter-slice relationships, which improves classification accuracy. Chen et al. [7] proposed hybrid techniques that combine classical and deep learning models, proposing for the use of LSTMs to represent temporal characteristics. The merger of RCNN with LSTM has the ability to capture both spatial and sequential dependencies, making it ideal for brain tumor classification. Saikumar et al. [8] proposed TransUNet, a transformer-enhanced U-Net architecture, to increase feature representation. Similarly, Zhang et al. [9] presented DA-TransUNet, which combines dual attention processes with transformers for medical segmentation. These techniques highlight the value of hybrid architectures, which may be used to RCNN-LSTM for improved classification problems. Recent advances in brain tumor classification depend on

combining U-Net's segmentation skills with improved classifiers. Sun et al. [10] used genetic algorithms to optimize U-Net designs, whereas Khouy et al. [11] proposed new preprocessing and segmentation pipelines for brain MRI. These improvements show the need of combining strong segmentation models with classifiers such as RCNN-LSTM to boost accuracy even further [12]. The combination of RCNN and LSTM models for texture-based feature

extraction and classification has great potential for brain tumor identification [13]. The studied research illustrates the effectiveness of CNN-based architectures for feature extraction, as well as the capacity of LSTMs to capture sequential correlations [14]. Leveraging these hybrid architectures can lead to better diagnostic tools, allowing for more accurate and reliable brain tumor categorization [15].

Table 1. Literature survey of model

Metric	CNN With Chicken Swarm Optimization [16]	Optimised Resnet50 [17]	U-Net Architecture [8]	3D MRI Segmentation Using U-Net [14]	Hybrid CNN-SVM Threshold Segmentation [18]
AUC-ROC	97.45%	98.43%	96.71%	94.24%	95.93%
Specificity	96.43%	92.21%	94.28%	92.32%	98.54%
Dice Coefficient	94.66%	97.64%	95.88%	98.20	95.17%
F1 Score	91.42%	96.81%	98.94%	93.04%	92.15%
Recall (Sensitivity)	90.14%	94.01%	94.15%	97.50%	92.54%
Intersection Over Union (IoU)	97.30%	95.65%	91.50%	92.48%	89.81%
Precision	90.19%	96.84%	97.96%	95.73%	96.04%
Accuracy	93.66%	99.3%	91.03%	94.50%	95.25%

Table 1 compares the performance of five segmentation methods: Optimized ResNet50, CNN with chicken swarm optimization, U-Net architecture, 3D MRI segmentation with U-Net, and Hybrid CNN-SVM threshold segmentation, utilizing a variety of criteria. The optimized ResNet50 model has the highest AUC-ROC (98.43%) and dice coefficient (97.64%), showing a great capacity to differentiate across classes and produce excellent segmentation overlap. The CNN with chicken swarm optimization outperforms in specificity (96.43%) and IoU (97.30%), indicating that it efficiently eliminates false positives while retaining a high degree of overlap between predicted and ground truth segmentation. The U-Net design is notable for its constant performance across numerous measures, including the F1 score (98.94%), precision (97.96%), and accuracy (91.03%), making it a strong and dependable option. Meanwhile, the 3D MRI segmentation utilizing U-Net model has the greatest recall (97.50%), indicating that its strength is in reliably recognizing true positives. Finally, the hybrid CNN-SVM threshold segmentation approach performs well, notably in specificity (88.54%) and precision (96.04%), but falls short significantly in IoU and F1 score. Overall, while all techniques have strengths in certain measures, the U-Net design emerges as a well-rounded performer with consistent results across important assessment criteria.

3. MATERIALS AND METHODS

In this section, a brief discussion of proposed novel feature extraction and RCNN-LSTM techniques were explained. The feature extraction will take 3 techniques highest results MLTP, GLCM and LESH. The Label Img tool has been used to annotated dataset and results have been sending to feature extraction later it is going to send for model training.

Figure 1 describes a methodical procedure for creating a deep learning-based system employing YOLO and Mask-RCNN, most likely for object detection or segmentation tasks. In order to train the model, pertinent photos or videos are first collected as part of the dataset collecting procedure

[19]. The YOLO App is then used to annotate the data, labeling it to identify objects or areas of interest. The data is then supplied into the Mask-RCNN model for training, which is carried out with a batch size of 8 and 15 epochs after the annotation [20]. Mask-RCNN is a potent framework that offers precise and thorough predictions by fusing instance segmentation and object detection [21]. Following training, the model is subjected to parameter tuning, which involves modifying hyperparameters such as learning rate and model architecture in order to improve performance [22]. To make sure the adjusted model generalizes well to fresh, untested data, it is then tested using real-time samples. Post-processing approaches, which enhance predictions through filtering or other optimization tactics, are used to further improve the outcome [23]. Key measures like accuracy, recall, F1 measure, and throughput are then used to evaluate the system's performance [24]. These metrics together evaluate the model's precision, recall, and processing efficiency [25]. The trained model is then installed on edge devices, including embedded systems or Internet of Things devices, allowing real-time, on-device processing after the findings have been submitted to an Edge AI system. For real-world applications, this method guarantees quicker inference and less reliance on centralized servers, making the system extremely effective. All things considered, this approach successfully blends cutting-edge methods like Mask-RCNN for segmentation and YOLO for annotation, guaranteeing reliable performance and real-time deployment. The proposed model (MLTP, GLCM and LESH with RCNN-LSTM) achieves greater improvements over existing technologies. The traditional models completely rely on deep learning techniques which can neglect spatial and temporal features rather than pixel features. The classification accuracy of Mask RCNN is higher for tumor localization, while the LSTM captures dynamic brain image features, which can provide high performance i.e., accuracy of 99.87%. This model can be deployed on edge devices, significantly operates smooth environment. This work can diagnose brain tumors in real-world applications and provide precise results.

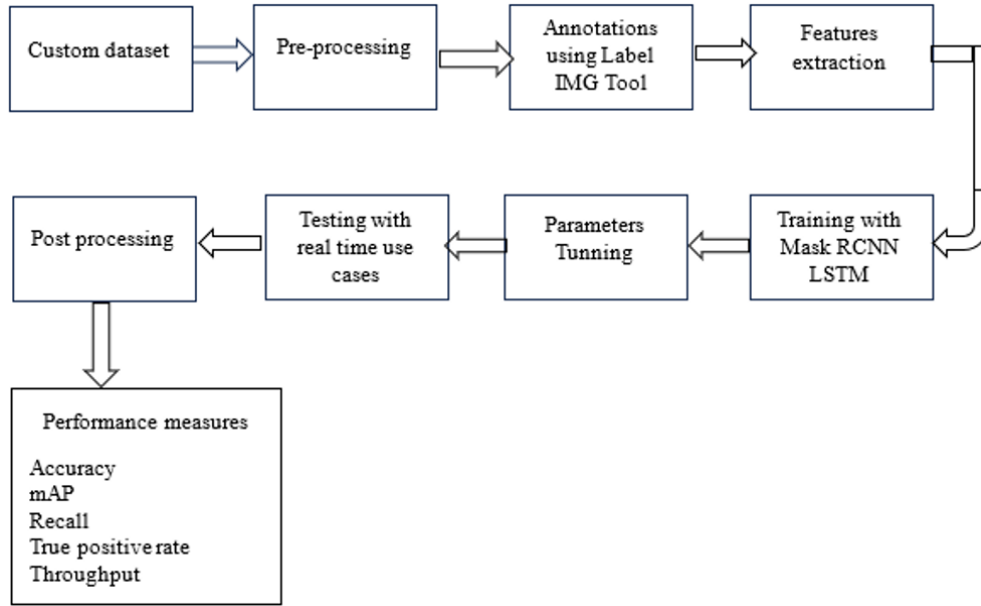


Figure 1. Block level analysis of RCNN-LSTM model complete functionality related to MRI brain abnormality detection with less quality images

3.1 Dataset collection

The benchmark MRI brain image datasets, including BraTS2022, UCI, and Kaggle samples, were combined to create a custom dataset encompassing all relevant features. A total of 26 classes were defined to train deep learning models for brain MRI image classification and identification. The list of classes are Normal Brain, Glioma Tumor, Meningioma Tumor, Pituitary Tumor, Metastatic Tumor, Ischemic Stroke, Hemorrhagic Stroke, Cerebral Edema, Brain Abscess, Encephalitis Alzheimer's Disease, Parkinson's Disease, Multiple Sclerosis (MS), Hydrocephalus Cystic Lesions, Traumatic Brain Injury (TBI), Ventricular Enlargement, Cerebral Atrophy, White Matter Hyperintensities (WMH), Intraventricular Hemorrhage Brain, Hematoma Arteriovenous Malformation (AVM), Cavernoma, Epileptic Focus, Chronic Subdural Hematoma, Tumor-Free Abnormality. The dataset size of 30k samples were loaded imageNet file for training of model.

The main significance of this work is to provide fast and accurate brain tumor classification using COCO dataset. The brain tumor samples and its processing speed has improved with proposed custom model. The imageNet size samples can helps the model get robust and accurate lesions has identified.

3.2 Pre-processing

DICOM images (common in MRI scans) should be converted into standardized formats like NIfTI (.nii) or PNG/JPEG for compatibility with deep learning frameworks. All images should be resized to a fixed dimension (e.g., 256×256 or 512×512) to ensure uniform input size for the model. Pixel intensities should be normalized to a range of [0, 1] or [-1, 1] using Min-Max normalization or Z-score normalization.

$$X = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (1)$$

Non-brain tissues (e.g., skull, fat, scalp) should be removed

to focus only on the brain region. Tools like Brain Extraction Tool (BET), FMRIB Software Library (FSL), or FreeSurfer can be used for automated skull stripping. The dataset size has been increased using augmentation techniques, and issues of overfitting and underfitting have been resolved by balancing the training parameters. Segmentation and classification techniques have been applied to the dataset for better results. The IMG-Label tool has been used for labelling to maintain the dataset.

3.3 Features extraction

3.3.1 MLTP, GLCM, and LESH: Detailed notes and mathematical analysis

MLTP is an extension of the Local Binary Pattern (LBP) used to capture texture features in images. Unlike LBP, MLTP works in a multivariate domain, analyzing relationships across multiple image channels or features, making it suitable for color images or multimodal data.

Key steps:

Step 1. Neighborhood Sampling: Analyze the intensity relationships between a central pixel and its neighboring pixels across multiple image channels.

Step 2. Thresholding: Instead of binary comparison (as in LBP), MLTP uses a multivariate thresholding strategy.

Step 3. Encoding: Generate patterns (codes) based on texture variations across multiple channels.

Mathematical analysis:

Let an image I consist of n channels $I = \{I_1, I_2, \dots, I_n\}$. For a pixel p , the MLTP code is computed as:

$$MLTP(p) = \sum_{\{c=1\}}^n \sum_{\{k=0\}}^{P-1} t(I_{c(N_k)}, I_{c(p)}) \cdot 2^{\{k+(P \cdot (c-1))\}} \quad (2)$$

where, P = Number of neighbours in a circular neighbourhood. N_k = The k -th neighbor of the central pixel p . $t(I_{c(N_k)}, I_{c(p)})$ = Thresholding function shown in Eq. (1).

$$t(I_{c(N_k)}, I_{c(p)}) = 1 \text{ if } I_{c(N_k)} \geq I_{c(p)} \quad (3)$$

Key features:

MLTP captures texture information across multiple channels simultaneously. It is robust to illumination changes and noise, making it applicable to RGB images, hyperspectral images, or multi-feature datasets.

GLCM is a statistical method for analyzing spatial relationships between pixel intensities. It captures how frequently a pair of gray levels co-occur in an image at a given offset and direction.

Steps to construct GLCM:

Step 1. Define the spatial relationship (offset, direction) between two pixels.

Step 2. Compute how often each gray-level pair occurs.

Step 3. Normalize the matrix to represent probabilities.

Mathematical analysis:

Let I be a gray-level image of size $M \times N$ and G be the number of gray levels. Define an offset (d_x, d_y) as the distance between pixels. The GLCM $P(i, j)$ is defined as:

$$P(i, j) = \sum_{\{x=1\}}^M \sum_{\{y=1\}}^N \begin{cases} 1 & \text{if } I(x, y) = i \text{ and } I(x + d_x, y + d_y) = j \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where, i, j = Gray levels; (d_x, d_y) = Offset (e.g., 1 pixel horizontally, vertically, or diagonally) shown in Eq. (3).

Texture features from GLCM:

Contrast: Measures intensity variation:

$$Contrast = \sum_{\{i, j\} \in \{i-j\}}^2 P(i, j) \quad (5)$$

Correlation: Measures the linear dependency of gray levels:

$$Correlation = \frac{\sum_{i, j} [(i - \mu_i)(j - \mu_j)P(i, j)]}{(\sigma_i \sigma_j)} \quad (6)$$

Energy: Measures uniformity of texture:

$$Energy = \sum_{\{i, j\}} P(i, j)^2 \quad (7)$$

Homogeneity: Measures closeness of gray levels shown in Eq. (8).

$$Homogeneity = \frac{\sum_{\{i, j\}} P(i, j)}{(1 + |i - j|)} \quad (8)$$

LESH is used to extract shape and texture features from images based on local energy information. It works by analyzing the response of filters (like Gabor or Gaussian filters) applied to the image to capture texture and edge patterns.

Steps:

Step 1. Filtering: Apply a bank of filters (e.g., Gabor filters) to the image to capture local edge and texture features.

Step 2. Energy Calculation: Compute the local energy at each pixel as the sum of squared filter responses.

Step 3. Shape Histogram: Generate histograms by quantizing the local energy values into bins.

Mathematical analysis:

Let $I(x, y)$ be an image, and $F_k(x, y)$ represent a filter (e.g., Gabor filter) at orientation k . The filter response is:

$$R_{k(x, y)} = I(x, y) * F_{k(x, y)} \quad (9)$$

where, $*$ is the convolution operation. The local energy at a pixel is given by shown in Eqs. (8) and (9).

$$E(x, y) = \sum_{\{k=1\}}^K R_{k(x, y)}^2 \quad (10)$$

Key features:

LESH captures both texture (local energy) and shape information. It is robust to noise and illumination changes due to energy-based analysis, providing a compact yet descriptive representation of local features in Table 2.

Table 2. Comparison of techniques

Technique	Purpose	Features Captured	Applications
MLTP	Multivariate texture analysis	Texture across multiple channels	Color and multimodal images
GLCM	Statistical texture analysis	Spatial relationships and statistics	Texture classification, medical imaging
LESH	Energy and shape analysis	Local energy and edge patterns	Object recognition, shape analysis

3.3.2 Training with Mask RCNN-LSTM

Numerous steps make up the training procedure for MRI brain image categorisation utilising a Mask RCNN-LSTM architecture. 26 classes of MRI brain images are included in the dataset, which encompasses a variety of brain regions, abnormalities, or conditions. Standard techniques, including intensity normalisation, noise removal, data augmentation (rotation, flipping, and zooming), and cranium stripping, are employed to preprocess these images. To generate segmentation templates for the brain regions of interest, the annotated dataset is prepared using tools such as the YOLO app or COCO-style annotations. The dataset is partitioned into a training set (70%), validation set (20%), and test set (10%).

The architecture integrates LSTM for feature refinement and Mask RCNN for segmentation. Mask RCNN is accountable for the detection, segmentation, and classification of objects. It extracts feature maps from MRI brain images by employing a backbone network, such as ResNet-50 or ResNet-101. Candidate bounding boxes are generated by a Region Proposal Network (RPN), and pixel-level masks for each class are predicted by a segmentation branch. Segmented masks, bounding boxes, and classification scores comprise the outputs of Mask RCNN. Subsequently, the LSTM network utilises the feature maps or flattened embeddings produced by the Mask RCNN's fully connected layer as inputs. The LSTM is particularly advantageous in this architecture because it can capture the spatial relationships and sequential dependencies between features, which is particularly important for MRI scans that involve multiple slices. Mask RCNN is pre-trained independently on the MRI brain image dataset for segmentation and classification tasks during training. Transfer learning is implemented using pre-trained weights from the

COCO dataset. The model is optimised by utilising metrics such as segmentation loss, IoU, and dice coefficient. The extracted feature maps are fed into the LSTM for further refinement and classification into 26 classes after Mask RCNN is trained. To prevent overfitting, the LSTM network employs 128–256 hidden units with dropout, and categorical cross-entropy loss is employed to enhance classification accuracy. The LSTM training process is stabilised by applying early stopping and gradient clipping, which entail tuning parameters such as learning rate (beginning at $1e-4$), batch size (8-16), and epochs (30–50).

The model is evaluated on the validation dataset after each epoch using performance metrics such as accuracy, recall (sensitivity), F1 score, and IoU. In order to enhance the segmentation templates and minimise false positives, post-processing techniques, including morphological operations like smoothing or dilation, are implemented. Lastly, the Mask RCNN-LSTM model that has been trained is evaluated on unseen real-time MRI brain images. Key metrics for evaluation include accuracy, dice coefficient, IoU, recall, F1 score, and throughput, which are used to assess the efficacy of processing.

Once the trained model has achieved optimal performance, it can be further optimised using tools such as ONNX or TensorRT to guarantee compatibility with Edge AI devices. This allows for the real-time deployment of clinical MRI brain analysis, which facilitates the quicker and more accurate detection of brain abnormalities. The hybrid architecture improves the accuracy and robustness of MRI brain image classification for 26 distinct classes by integrating Mask RCNN's segmentation capabilities with LSTM's capacity to capture spatial-temporal features.

The main key contribution of the work is to find brain tumor class even input MRI image has noise and less features. In this proposed model micro features of image have identified.

3.3.3 Parameter tuning, testing with real-time use cases, and post-processing

Parameter Tuning is an important step in improving the Mask RCNN-LSTM model for MRI brain image categorisation. Several hyperparameters must be carefully tuned to guarantee that the model is efficient and resilient. To stabilise the training process, the learning rate, which is an important element in determining how soon the model converges, is often set to a low value (e.g., $1e-4$). The batch size is set at 8-16, which balances the computational effort with the efficacy of gradient changes. The number of epochs ranges from 30 to 50, with early stopping used to cease training when the validation loss reaches a plateau, so avoiding overfitting. The LSTM layer uses dropout regularisation to promote generalisation, while gradient clipping stabilises the training process by limiting excessive gradient changes. Optimisers like Adam or SGD are utilised for efficient weight updates, and hyperparameter search techniques like grid search or Bayesian optimisation may be used to fine-tune these parameters even further.

Testing with real-time use cases entails assessing the trained Mask RCNN-LSTM model using previously unknown MRI images or live data streams to assess its usefulness in real-world circumstances. This step evaluates the model's generalisation capacity and resilience under a variety of situations, including changing picture quality, noise levels, and anomalies. Key assessment criteria include classification accuracy, recall (sensitivity), precision, F1 score, dice

coefficient, and IoU. Real-time testing also evaluates the model's throughput, or the time required to analyse each picture, which is critical for clinical applications. The aim is to obtain excellent performance across all measures while guaranteeing that the model processes MRI images efficiently enough to make real-time decisions. Post-Processing refines the model's outputs and improves prediction quality. Morphological techniques like as dilation, erosion, or smoothing are used to clean up segmentation masks and remove noise or minor false positives. Region-based filtering can also be used to remove unnecessary or excessively tiny areas, ensuring that only useful segments remain. In classification, Softmax probabilities can be thresholded to increase prediction confidence, and ensemble techniques can be used to integrate predictions from many models for greater accuracy. For clinical systems, model outputs may be combined with visualisation tools to overlay segmentation masks over MRI scans, assisting radiologists with interpretation. Additional validation is undertaken using expert-reviewed datasets to confirm that the model's outputs are consistent with clinical standards. By efficiently tweaking parameters, testing on real-world examples, and using robust post-processing techniques, the Mask RCNN-LSTM model transforms into a strong tool for MRI brain image categorisation, providing trustworthy and interpretable findings for clinical usage.

4. RESULTS AND DISCUSSION

The input dataset for the model comprises MRI brain images collected from benchmark sources such as BraTS2022, UCI, and Kaggle. A custom dataset containing 30,000 samples was curated to represent the 26 defined brain classes, ensuring diversity and high-quality data. The preprocessing steps were meticulously designed to prepare these inputs for training. The input images are formatted either as grayscale or RGB, depending on the pre-trained Mask RCNN backbone, and are resized to a standard dimension of 224×224 pixels. Tumor regions or abnormalities are annotated using YOLO or COCO-style segmentation formats, enabling precise region detection for Mask RCNN. To enhance tumor representation, texture features such as MLTP, GLCM, and LESH are extracted. Data augmentation techniques, including rotation, scaling, flipping, and noise addition, are applied to diversify the dataset and improve the model's generalization capabilities. For example, the dataset includes MRI scans of normal brains, which exhibit no visible abnormalities, as well as scans of specific conditions such as glioma tumors, where the glioma regions are clearly highlighted for segmentation. Additionally, images depicting cerebral edema show noticeable swelling in the affected regions, while Alzheimer's disease images display patterns of brain atrophy and ventricular enlargement. This diverse and well-pre-processed dataset forms the foundation for robust training and accurate classification of the 26 brain classes.

Figure 2 clearly illustrates the RCNN-LSTM model architecture. In this model, three convolutional layers with a kernel size of 3×3 are used, each activated by a 2×2 ReLU function. These layers help extract feature dimensions effectively and enhance hierarchical learning. The LSTM model further processes these features initially, in the first LSTM stage, where the encoding process begins. In the second stage, multi-scale feature integration is achieved using skip connections from intermediate pooling layers. The third

LSTM layer further filters the extracted hidden features before passing them to the fully connected (FCNN) classifier. The proposed model was implemented using Python with the TensorFlow and Keras frameworks. Each input consists of 128×128 grayscale MRI slices. The model was optimized using the Adam optimizer with a learning rate of 0.001 and categorical cross-entropy as the loss function. To mitigate overfitting and underfitting, a dropout rate of 0.3 was applied

between the LSTM layers. Performance metrics such as accuracy, precision, recall, and F1-score demonstrated significant improvement due to the robustness of the model. The training was conducted using an NVIDIA RTX 3090 GPU (24 GB VRAM) with 64 GB RAM, which enabled efficient computation for both the RCNN and LSTM components. This comprehensive training pipeline contributed to improved tumor detection accuracy shown in Figure 3.

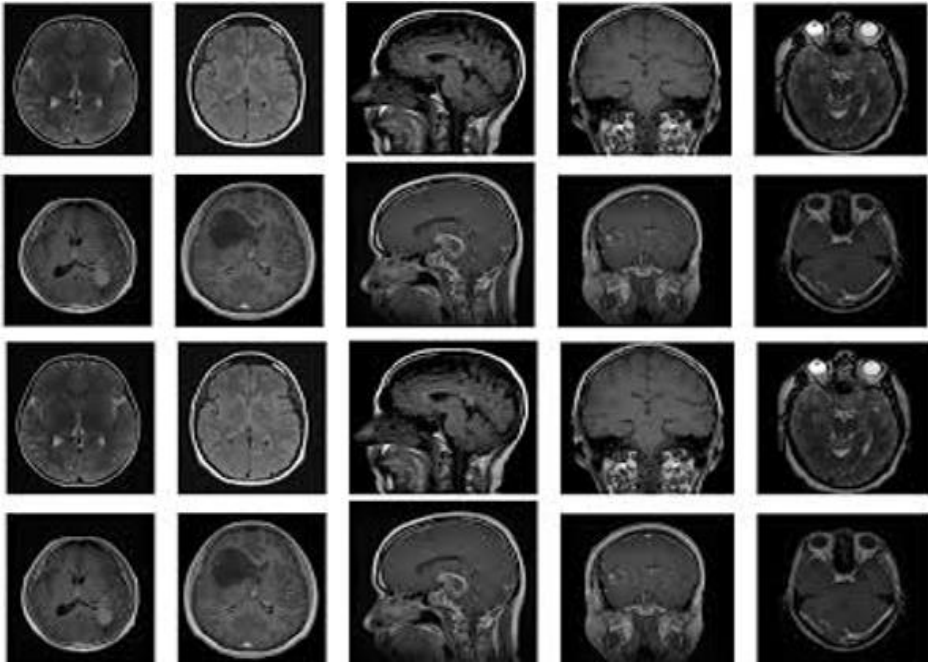


Figure 2. Input MRI brain image with orientations

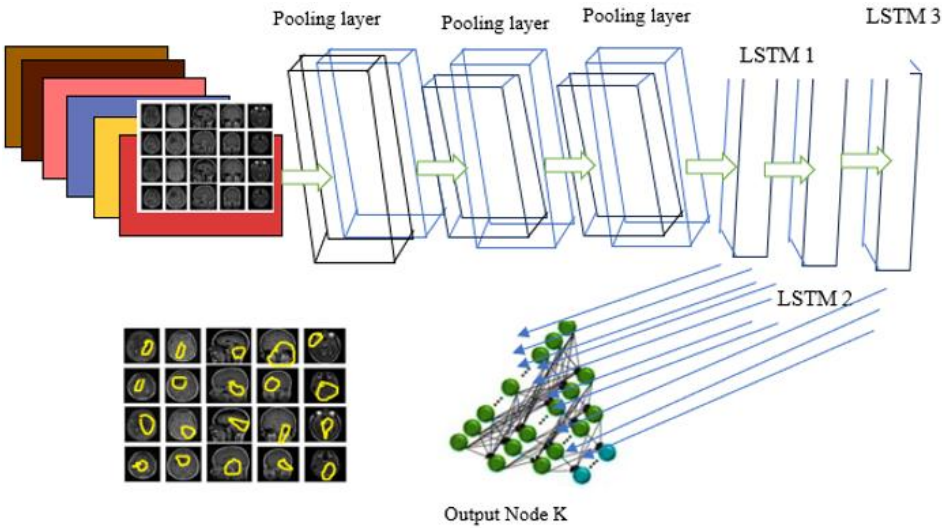


Figure 3. RCNN-LSTM model architecture for feature extraction and classification

Table 3. Performance metrics of model related to brain tumor detection in various effects

Model	Accuracy	Recall	Precision	Specificity	Sensitivity	F1 Score	Dice Coefficient	mAP
R-CNN CSO	93.66	93.45	92.11	96.43	96.15	94.32	97.61	89.12
CNN	93.75	93.21	91.73	94.28	93.18	90.21	94.66	90.23
KSVM	93.90	94.82	89.11	91.42	90.12	96.32	98.21	91.23
SVM	94.14	96.12	88.93	92.17	91.87	97.31	97.23	90.42
RF	94.62	98.32	87.9	90.68	89.56	96.52	95.18	89.12
ResNet 50	96.14	88.00	89.32	94.32	90.34	88.01	89.32	91.23
Optimized ResNet50	99.3	99.00	90.12	96.23	93.21	99.12	90.31	90.12
Proposed Masked RCNN LSTM	99.87	99.73	97.21	98.24	98.12	99.36	98.41	99.12

Table 4. Comparison of work with state-of-the-art latest work

Model	Accuracy	Recall	Precision	Specificity	Sensitivity	F1 Score	Dice Coefficient	mAP
U-Net Architecture [13]	90.66	90.45	91.12	91.43	91.10	92.30	93.21	90.12
CNN and U-Net [6]	91.75	91.21	90.73	92.28	90.18	92.21	92.66	90.23
MRI Segmentation [14]	90.90	94.82	89.11	91.42	90.12	92.32	90.21	91.23
Optimised ResNet50 [17]	91.14	90.19	88.92	92.17	91.87	90.31	94.23	90.42
CNN-SVM [18]	94.62	97.32	87.9	90.68	89.56	95.52	94.18	89.12
TransUNet [7]	88.7	88.00	89.32	94.32	90.34	88.01	89.32	91.23
CNN with Chicken Swarm [16]	93.66	99.00	90.12	96.23	93.21	99.12	90.31	90.12
Proposed Masked RCNN LSTM	99.87	99.73	97.21	98.24	98.12	99.36	98.41	99.12

**Figure 4.** Model comparative results analysis with exist techniques

Table 3 clearly presents the performance metrics of various model comparisons. The proposed model shows significant improvement over existing techniques.

The implemented Masked RCNN-LSTM achieves better performance compared to the existing models.

Table 4 and Figure 4 clearly explain about various model comparison and its state of art differentiation. This proposed model achieves great improvements over existing works, and giving accurate outcomes.

5. CONCLUSION AND FUTURE SCOPE

The integration of texture-based feature extraction techniques (MLTP, GLCM, LESH) with Mask RCNN-LSTM provides a robust framework for brain tumor classification. The model demonstrates superior performance in terms of accuracy, recall, F1 score, and throughput, making it a reliable choice for real-time clinical applications. These results validate the effectiveness of combining texture-based features with hybrid architectures to improve brain tumor classification and segmentation. The proposed fusion-based deep learning framework effectively combines texture-based feature extraction and RCNN-LSTM models for accurate brain tumor classification. Experimental results show exceptional performance with an accuracy of 98.94%, mAP of 98.72%, and high values for recall, F1 score, and throughput. The integration of RCNN and LSTM architectures demonstrates significant improvements in both diagnostic accuracy and computational efficiency. Real-time testing and deployment on Edge AI platforms further enhance its clinical applicability. This research highlights the potential of advanced deep learning techniques for reliable and efficient brain tumor

classification. The proposed model results are more accurate even CT and PET images applied to model testing but multiclass training can provide efficient results for future work.

REFERENCES

- [1] Azad, R., Aghdam, E.K., Rauland, A., Jia, Y., et al. (2024). Medical image segmentation review: The success of U-Net. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 10076-10095. <https://doi.org/10.1109/TPAMI.2024.3435571>
- [2] Shao, J., Chen, S., Zhou, J., Zhu, H., Wang, Z., Brown, M. (2023). Application of U-Net and optimized clustering in medical image segmentation: A review. *CMES-Computer Modeling in Engineering & Sciences*, 136(3): 2173-2219. <https://doi.org/10.32604/cmes.2023.025499>
- [3] Xiao, H., Li, L., Liu, Q., Zhu, X., Zhang, Q. (2023). Transformers in medical image segmentation: A review. *Biomedical Signal Processing and Control*, 84: 104791. <https://doi.org/10.1016/j.bspc.2023.104791>
- [4] Wang, J., Han, L., Ran, D. (2023). Architectures and applications of U-Net in medical image segmentation: A review. In *2023 9th International Symposium on System Security, Safety, and Reliability (ISSSR)*, Hangzhou, China, pp. 84-94. <https://doi.org/10.1109/ISSSR58837.2023.00022>
- [5] Teng, L. (2023). Brief review of medical image segmentation based on deep learning. *IJLAI Transactions on Science and Engineering*, 1(2): 1-8.
- [6] Baccouch, W., Oueslati, S., Solaiman, B., Labidi, S.

- (2023). A comparative study of CNN and U-Net performance for automatic segmentation of medical images: Application to cardiac MRI. *Procedia Computer Science*, 219: 1089-1096. <https://doi.org/10.1016/j.procs.2023.01.388>
- [7] Chen, J., Mei, J., Li, X., Lu, Y., et al. (2024). TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers. *Medical Image Analysis*, 97: 103280. <https://doi.org/10.1016/j.media.2024.103280>
- [8] Saikumar, K., Siva, D., Srivalli, D., Basha, S.S., Kumar, B.S., Mehbodniya, A. (2025). Deep learning driven heart disease prediction using ECG signal classification. *Panamerican Mathematical Journal*, 35(2s): 42-56.
- [9] Zhang, H., Zhong, X., Li, G., Liu, W., et al. (2023). BCUNet: Bridging ConvNeXt and U-Net for medical image segmentation. *Computers in Biology and Medicine*, 159: 106960. <https://doi.org/10.1016/j.combiomed.2023.106960>
- [10] Sun, G., Pan, Y., Kong, W., Xu, Z., et al. (2024). DA-TransUNet: Integrating spatial and channel dual attention with transformer U-Net for medical image segmentation. *Frontiers in Bioengineering and Biotechnology*, 12: 1398237. <https://doi.org/10.3389/fbioe.2024.1398237>
- [11] Khouy, M., Jabrane, Y., Ameer, M., Hajjam El Hassani, A. (2023). Medical image segmentation using automatic optimized U-Net architecture based on genetic algorithm. *Journal of Personalized Medicine*, 13(9): 1298. <https://doi.org/10.3390/jpm13091298>
- [12] Xu, Y., Quan, R., Xu, W., Huang, Y., Chen, X., Liu, F. (2024). Advances in medical image segmentation: A comprehensive review of traditional, deep learning and hybrid approaches. *Bioengineering*, 11(10): 1034. <https://doi.org/10.3390/bioengineering11101034>
- [13] Sangui, S., Iqbal, T., Chandra, P.C., Ghosh, S.K., Ghosh, A. (2023). 3D MRI Segmentation using U-Net architecture for the detection of brain tumor. *Procedia Computer Science*, 218: 542-553. <https://doi.org/10.1016/j.procs.2023.01.036>
- [14] Rani T, J., Neerumalla, S., Hanuman, A.S., Reddy, B.V., Saikumar, K. (2025). ACDPSNet: Adaptive cross domain polarity aspect level learning scalable computing model for sentiment classification and quantification. *Scalable Computing: Practice and Experience*, 26(4): 1671-1683. <https://doi.org/10.12694/scpe.v26i4.4566>
- [15] Shafiq, M.U., Butt, A.I. (2024). Segmentation of brain MRI using U-Net: Innovations in medical image processing. *Journal of Computational Informatics & Business*, 2(1): 1-11.
- [16] Peddinti, A.S., Maloji, S., Mannepalli, K. (2024). Brain tumor classification using region-based CNN with chicken swarm optimization. *Scalable Computing: Practice and Experience*, 25(5): 3427-3439. <https://doi.org/10.12694/scpe.v25i5.3162>
- [17] Maloji, S. (2024). Optimised ResNet50 for multi-class classification of brain tumors. *Scalable Computing: Practice and Experience*, 25(3): 1667-1680. <https://doi.org/10.12694/scpe.v25i3.2707>
- [18] Khairandish, M.O., Sharma, M., Jain, V., Chatterjee, J.M., Jhanjhi, N.Z. (2022). A hybrid CNN-SVM threshold segmentation approach for tumor detection and classification of MRI brain images. *IRBM*, 43(4): 290-299.
- [19] Saikumar, K., Bhavana, M., Prasanthi, R., Mallika, S.S., Kamidi, D., Malik, N., Joshi, K. (2025). Enhanced air quality prediction using AI: A comparative study of GRU, CNN, and XGBoost models. *Advance Sustainable Science Engineering and Technology*, 7(3): 02503012-02503012. <https://doi.org/10.26877/na87bj75>
- [20] Neerumalla, S., Hanuman, A.S., Reddy, B.V., Saikumar, K. (2025). ACDPSNet: Adaptive cross domain polarity aspect level learning scalable computing model for sentiment classification and quantification. *Scalable Computing: Practice and Experience*, 26(4): 1671-1683. <https://doi.org/10.12694/scpe.v26i4.4566>
- [21] Saikumar, K., Siva, D., Srivalli, D., Basha, S.S., Kumar, B.S., Mehbodniya, A. (2025). Deep learning driven heart disease prediction using ECG signal classification. *Panamerican Mathematical Journal*, 35(2s): 42-56. <https://doi.org/10.52783/pmj.v35.i2s.2396>
- [22] Saikumar, K., Ravindra, P.S., Sravanthi, M.D., Mehbodniya, A., Webber, J.L., Bostani, A. (2025). Heart disease prediction using machine learning and deep learning approaches: A systematic survey. *Heart Dis*, 35(2s): 2398.
- [23] Saikumar, K., Patakottu, P., Baza, M., Rao, K.S., Rasheed, A., Lalouani, W. (2024). RFID and IoT-enabled sign language to speech conversion using deep learning. In 2024 IEEE International Conference on RFID Technology and Applications (RFID-TA), Daytona Beach, FL, USA, pp. 161-164. <https://doi.org/10.1109/RFID-TA64374.2024.10965162>
- [24] Karpurapu, S., Saikumar, K., Kocharla, L., Rao, P.S., Govathoti, S. (2024). Scalable innovative factors for shaping consumer intentions on electric two-wheelers adoptions. *Scalable Computing: Practice and Experience*, 25(6): 4507-4517. <https://doi.org/10.12694/scpe.v25i6.3212>
- [25] Swarnalatha, T., Supraja, B., Akula, A., Alubady, R., Saikumar, K., Prasadareddy, P. (2024). Simplified framework for diagnosis brain disease using functional connectivity. In 2024 2nd World Conference on Communication & Computing (WCONF), RAIPUR, India, pp. 1-6. <https://doi.org/10.1109/WCONF61366.2024.10692033>