E-ISSN 2586-9027

Homepage : *https://tci-thaijo.org/index.php/SciTechAsia* Science & Technology Asia

Vol.30 No.2 April - June 2025 Page: [71-83]

Original research article

Brain Tumor Classification With Selective Fine Tuning Using Transfer Learning

Deepa AB[1,*], Varghese Paul[2]

[1] *Research Scholar, Department of Computer Science and Engineering, Rajagiri School of Engineering & Technology, APJ Abdul Kalam Technological University, Kerala 682039, India*
[2] *Department of Computer Science and Engineering, Rajagiri School of Engineering & Technology, APJ Abdul Kalam Technological University, Kerala 682039, India*

Received 27 November 2024; Received in revised form 20 April 2025
Accepted 30 April 2025; Available online 18 June 2025

ABSTRACT

The accelerated pace of contemporary life has led to a notable increase in cancer incidence, which poses a significant challenge in the field of oncology. This study introduces an innovative approach to brain tumor detection by employing fine-tuned pre-trained models with sparse data and comparing their performance to that of traditional convolutional neural networks (CNNs). The study addresses the challenge of limited medical imaging datasets in oncology, a discipline experiencing heightened demand due to rising cancer rates. By utilizing transfer learning techniques, the proposed method seeks to alleviate the overfitting issues that are commonly encountered. Fine-tuned models developed from pre-trained networks exposed to millions of diverse images have been adapted for tumor classification tasks by incorporating max-pooling and dense layers. A comparative analysis revealed that these refined models achieved superior accuracy, exceeding 90 percent even with limited data, thereby outperforming the conventional CNNs. This study evaluated the model performance using various metrics, including accuracy and precision, and demonstrated the efficacy of transfer learning in enhancing brain tumor detection capabilities. This approach holds promise for improving diagnostic tools in oncology, particularly in scenarios in which large-scale medical imaging datasets are unavailable.

Keywords: Brain tumor detection; Convolutional neural networks; Deep learning; Overfitting; Sparse data; Transfer Learning models

***Corresponding author:** deepabalancemp@gmail.com **doi:** 10.14456/scitechasia.2025.26

1. Introduction

Brain tumor(BT) detection remains a critical challenge in medical imaging, with increasing prevalence and mortality rates across all age groups.[1] Magnetic resonance imaging (MRI) is the most effective technique for detecting BT. Approximately 120 different types of tumors have been recorded thus far, appearing in various sizes and shapes, making it more difficult to identify them accurately [2, 3].

Recent advances in machine learning (ML) and neural networks (NN) have opened up new possibilities for automated tumor diagnosis and segmentation. Deep learning (DL) models, particularly convolutional neural networks (CNNs), have shown exceptional feature learning capabilities and offer improved precision compared with traditional artificial intelligence methods [4]. However, leveraging DL for tumor diagnosis presents challenges, including the acquisition of high-quality annotated data, diversity in patient populations and tumor types, sparse data, overfitting, and the complex nature of DL models [5]. Patient privacy must be carefully monitored, and stringent privacy regulations often hinder the sharing and utilization of diverse and well-annotated datasets.In the presence of limited data, DL models may struggle to comprehend the full spectrum of tumor heterogeneity [6, 7].

Transfer learning (TL) has emerged as a valuable technique for addressing these challenges, enabling the efficient reuse of learned representations and reducing the data and computational resources required for training. Several studies have demonstrated the effectiveness of TL in BT classification, achieving high accuracy using pre-trained models such as VGG16, VGG19, and ResNet50[8]. This study utilized nine pre-trained models for BT classification and achieved an accuracy of > 90 percent using TL[9]. This paper [10] employed three pre-trained models for BT classification and attained a maximum accuracy of above 90 percent using the fine-tuned VGG16 network. In this study, [11] used a CNN to develop a technique to categorize glioma BT MR images. This method accurately classifies glioma tumors and categorizes several glioma grades and two other common tumor types. It uses a pre-trained CNN to differentiate between normal and abnormal BT MRI images [12].

This study introduced an innovative approach for detecting and classifying brain tumors using transfer learning methods. This study focuses on the following:

- Performance analysis of different data augmentation techniques to address data scarcity challenges.
- Comparison of different fine-tuned pre-trained models and CNN in sparse data settings based on various evaluation metrics.
- The impact of different k-fold cross-validations on various transfer learning models was assessed.
- Conduct ablation studies on different data sizes to analyze the impact of sparse data on the output.

The proposed method extracts essential features from a standard dataset to improve classification accuracy. We evaluated the performance of four distinct deep learning models: selectively tuned VGG16, VGG19, ResNet50, and custom CNN. Section 2 includes dataset preparations, augmentation techniques, pre-trained model architecture, fine-tuning of TL models, k-fold validation, and the proposed architecture to automate binary tumor classification with a limited dataset. Section 3 includes a comparison of TL models with CNN based on accuracy, classification report, and confusion matrix, an ablation study of different augmentation techniques, an evaluation of optimal data size, and an ablation study of k-fold cross-validation. Section 4 presents conclusions and future enhancements.

2. Methods

This research focuses on improving the precision of brain MRI image classification using DL and TL methodologies. TL facilitates the application of knowledge from pre-trained models to new tasks, which is particularly advantageous in medical imaging because of the scarcity of labeled data. The proposed framework, depicted in Fig. 1, combines the capabilities of pre-trained CNNs, such as VGG16, VGG19, and ResNet50, which have been specifically fine-tuned for classifying brain tumors. In our method, pre-trained CNN models were employed owing to their strong feature extraction abilities. These models were adjusted to the current classification task through layer fine-tuning and the addition of dense layers. The final classification was achieved using a softmax layer to categorize the images as either tumor or no tumor. The classification process includes crucial steps, such as image denoising, data augmentation, train-test splitting, and feature extraction. Each step was designed to enhance the learning capacity of the model and ensure high prediction accuracy. Following the global average pooling layer in the CNN backbones, two fully connected dense layers with 1024 neurons and ReLU activation functions were added. A further dense layer with 512 neurons and ReLU was included before the final output layer, which had two neurons corresponding to the binary classes activated using the softmax function.

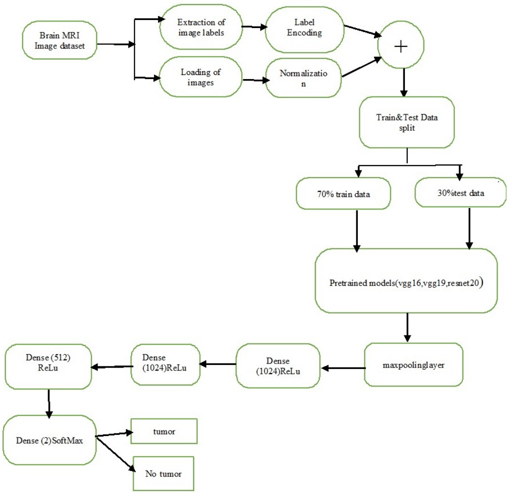


Fig. 1. Proposed Architecture.

2.1 Experimental Dataset

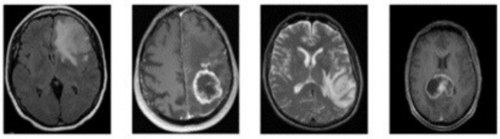


Fig. 2. experimental dataset.

The experimental dataset, sourced from Kaggle, consisted of 198 brain MRI images divided into two categories: ”Yes” (indicating the presence of a tumor) and ”No” (indicating the absence of a tumor), with each category containing 99 images. Fig. 2 shows a sample of malignant images. These images were annotated to support supervised learning in brain tumor classification tasks. The limited size of the dataset highlights the necessity for data augmentation and TL to develop an effective model.

2.2 Data preprocessing

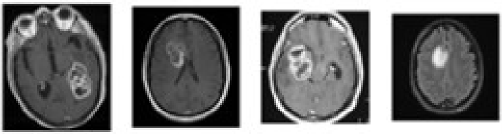


Fig. 3. Before preprocessing(278,236,3).

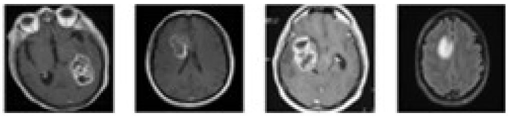


Fig. 4. Afterpreprocessing(224,224,3).

The labels were derived from the filenames, with ’Y’ representing Yes and N’ representing No. These categorical labels were then transformed into numerical forms (0 and 1) through label encoding, followed by one-hot encoding to create binary vectors suitable for training.Fig. 3 and Fig. 4 presents the image before and after resizing and normalization. A train-test split was executed, reserving 33 percent of the data for testing with a random seed applied to ensure reproducibility. This division allowed for a thorough evaluation of the model using new data. The final dataset shape was confirmed to be (224, 224, 3), indicating the presence of RGB color channels.

2.3 Data augmentation

Learning a relationship with a small set of data leads to overfitting. It is expensive to obtain medical data with annotations, and it is challenging to develop an accurate deep learning model with sparse data. Several augmentation techniques, such as rotation, have been used. The data volume increased with the use of these methods.

During training, we integrated a thorough data augmentation pathway to improve the model generalization and reduce overfitting[15]. Augmentations were applied dynamically using Keras’ ImageDataGenerator, along with custom preprocessing transformations. These included random rotations (±15°) to simulate variations in patient head positioning, horizontal and vertical flips to introduce orientation invariance, and zooming (range: 0.9–1.1) to mimic changes in tumor size. Additionally, shear transformations were used to slightly distort the shape of the images and model potential noise in clinical scenarios, whereas contrast adjustments were employed to replicate variations in MRI acquisition settings. These augmentations were strictly limited to the training set to maintain consistency in the evaluation of the validation and test performance. The augmented dataset was then used to fine-tune the pretrained convolutional neural networks such as VGG16 and ResNet50 for BT classification.

2.4 Architecture

CNN models VGG16, VGG19, and ResNet50 were selected for image classification. These models accept the input shapes of (224, 224, 3), which is consistent with the resized dataset. The original fully connected top layers of the pre-trained models were excluded and customized layers were added for binary classification.

2.4.1 Pre-trained Models

The VGG16, VGG19, and ResNet50 models were used, with weights transferred to avoid training from scratch. Some parts remained unchanged, whereas the other layers built the model. The model contains dense layers for predictions, max-pooling layers to reduce data, and convolutional layers to identify features. Feature maps were converted into one-dimensional vectors before entering the fully connected layers. Both 2D and 3D feature maps were flattened for the feedforward neural networks. Fig.5 shows that the CNNs use cross-entropy loss to measure the differences between the expected and actual results.

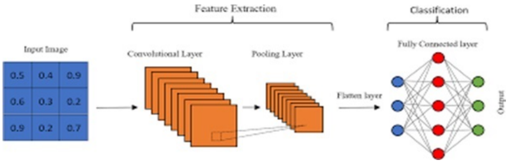


Fig. 5. CNN.

VGG16 is a deep convolutional neural network architecture that has achieved state-of-the-art performance on various visual recognition benchmarks.Fig. 6 shows 13 convolutional layers with a small receptive field (3x3) and max-pooling layers to minimize computational effort. VGG16 has sixteen weight layers, 13 convolutional layers, and three fully connected layers. It uses tiny 3 x 3 convolutional filters to extract finer details from input photos, making it efficient for image recognition tasks. The network terminates with three fully connected layers and a classification-focused softmaxlayer [13].

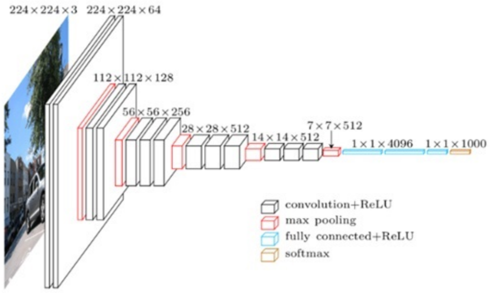


Fig. 6. VGG16.

The VGG19 model in Fig.7 is loaded with imagenet weights and configured to exclude its top layers, which are typically designed for broader image classification tasks VGG19 is an extension of VGG16[13].

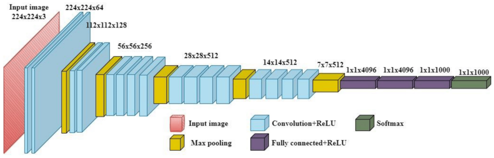


Fig. 7. VGG19.

ResNet50 is a deep convolutional neural network architecture shown in Fig. 8 that is designed to address the vanishing gradient problem and make it easier to train very deep neural networks. Residual blocks work by adding shortcut connections that allow the network to learn the residual information [14].

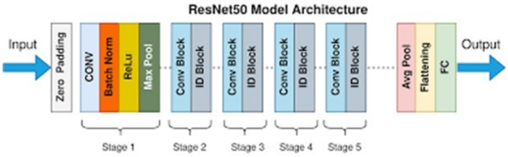


Fig. 8. RESNET50.

These three TL models were used as base models, and the layers were fine-tuned to adapt to the tumor classification task.

2.4.2 Fine-tuning of pre-trained Models

A downsampling layer lowers the spatial dimensions of the feature maps produced by the convolutional layers, thereby reducing the computational load and aiding the ability of the network to concentrate on more crucial features. A nonlinear procedure called MaxPooling chooses the largest value from a set of values within the input feature map, typically from a small rectangular region. The selected maximum values from each window create a downsampled version of the original feature map, and these values are organized into an output feature map. The dimensions of this output feature map are typically reduced compared with those of the input feature map. The max-pooling operation is mathematically represented as follows:

Input Feature Map: Assume that we have a 2D matrix comprising dimensions, referring Eq. (2.1)

$$[H \times W], \tag{2.1}$$

where H denotes the height of the feature map, and W is the width.

Output Feature Map: A feature map that has been down-sampled with dimensions, referring to the Eq. (2.2)

$$\left(\frac{H}{K}\right) \times \left(\frac{W}{K}\right), \tag{2.2}$$

is the outcome of MaxPooling; for a given height, K is used, and for a given width, W/K. Referring Eq.(2.3).

$$(Output(i, j) = max(Input(i \times K : (i \times K + K), j \times K : (j \times K + K)))). \tag{2.3}$$

Output (i, j) is the value at the ith row and jth column of the output feature map, and input $(i \times k : (i \times k + k), j \times k : (j \times k + k))$ represents the region within the input feature map defined by the i-th and j-th windows.

Dense layers are a fundamental part of artificial neural networks that connect and process information across all neurons in a given layer. A dense layer has three components: an input vector, a weight matrix, and a bias vector. The input vector, which is a 1D array of N dimensions, represents the input features or the activation of the previous layer. The weight matrix, W, has dimensions of (M, N), connecting every neuron in the previous layer to each neuron in the current layer. The bias vector B has the dimensions of M. Specialized neural networks can be built by replacing the fully connected top layers of the TL models and fine-tuning them. Linear activation in two dense layers with a size of 1024 is performed after a global average pooling layer that shrinks the size of the feature maps. A final dense layer with 512 neurons generalizes the prediction. All layers were fully connected. Max-pooling reduces the feature map size by selecting the maximum value for each local region. Mathematically, it can be expressed as referring to Eq. (2.4)

$$Y_{ij} = \max_{(m,n)} \left(x_{i+2m,\, j+2n}\right)). \tag{2.4}$$

Y_{ij} is the output at positions (i, j) & $(x_i + 2m, j + 2n)$ represents the input data in the local region. After each convolutional layer, the activation function, using Eq. (2.5), typically a Rectified Linear Unit (ReLU), is applied element-wise. The ReLU function is defined as

$$f(x) = max(0, x). \tag{2.5}$$

In binary classification problems, the output layer is defined by two number classes: tumor and non-tumor. This problem was solved using the softmax activation function, which computes the probability distribution over multiple classes. The softmax function can be defined by Eq. (2.6)

$$Softmax(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{n} e^{y_j}}. \tag{2.6}$$

Pro$(Y_i = y_i)$ is the probability of class i, and y_i is the score of the logit for class i.

Fine-tuning of the pre-trained models will freeze the base layers and add top layers to adapt the model to our problem.

2.5 Hyperparameter tuning

To improve the performance of the brain tumor classification model using selective fine-tuning, a structured hyperparameter optimization strategy was implemented through a Grid Search. Grid Search provides an exhaustive and interpretable method for exploring multiple combinations of hyperparameters, making it particularly effective for small-scale datasets often encountered in medical imaging tasks. The Configuring Grid Search parameters that are essential to the performance of the refined models were included in the hyperparameter search space[16, 17]. The following hyperparameters were considered: learning rates of 0.01, 0.001, and 0.0001. The sizes of the batches were 16 and 32. The Dropout are 0.3 and 0.5. The optimizers used were Adam and Stochastic gradient descent optimizers. There were 20 and 30 epochs. The adjusted layers are the last dense layer only, last convolutional block, and dense layer. Stratified 5-fold cross-validation was used to assess each combination to guarantee a small dataset.The average validation accuracy across all folds served as the main evaluation metric for choosing hyperparameters.We used early stopping with a 5-epoch patience to prevent overfitting.

3. Results and Discussion

3.1 Performance evaluation of different TL models

3.1.1 Accuracy

In this study, DL techniques were used to accurately classify BT types as benign or malignant. To increase the accuracy with sparse data, the system was trained using selectively fine-tuned pre-trained networks, such as VGG16, VGG19, and ResNet50. The softmax layers of the pre-trained networks were used to identify images that were malignant or benign. The CNN base was used to train the sparse dataset, resulting in a high accuracy during training and a decrease in performance during validation. This illustrates the overfitting problem in a typical network when using a small amount of data. We tuned our model using max-pooling and dense layers. The refined pre-trained network architecture that has been proposed performs better during the training and validation phases. A comparison chart of the typical training and validation accuracies of the normal CNN and fine-tuned architecture for different pre-trained models is displayed in Table 1. The training and test accuracies are provided.

Table 1. Accuracy comparison of TL models.

Model	training accuracy[%]	test accuracy[%]
CNN	0.86	0.71
$VGG16$	0.92	0.86
$VGG19$	0.89	0.88
$ResNet50$	0.90	0.90

The VGG16 accuracy was 92 percent of that of the fine-tuned pre-trained models. The training accuracy of the CNN model was 0.86, whereas its validation accuracy was 0.71, indicating potential overfitting. In the case of the CNN model, the significant disparity between the training and validation accuracies (0.86 vs. 0.71) indicates that the model may have memorized the training data to the extent that it struggles to perform well on new, unseen data, as shown in Fig. 9.

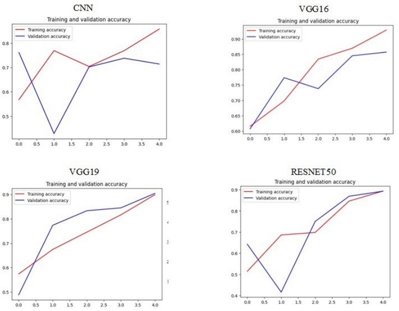


Fig. 9. Accuracyplot.

3.1.2 Confusion matrix

A confusion matrix is frequently used to depict the system accuracy for a given test dataset with known values. Fig. 9 displays a confusion matrix that summarizes the performance of the system in the classification of brain tumors. The x-axis of the matrix represents the target class and the y-axis represents the output class. This information can be used to identify areas in which the system performs well and where improvement is needed, ultimately leading to a more accurate and reliable classification system[18, 19].

The confusion matrices of various deep neural networks are compared in Fig.10. Eleven false-positive classifications in CNN networks occurred, along with other fine-tuned transfer-learning models, with seven false-positive cases and one false-negative case. The confusion matrices of all four models (VGG16, VGG19, CNN, and ResNet50) revealed varying distributions of the classification errors. Notably, VGG16 and ResNet50 exhibited fewer false negatives than the CNN baseline, indicating better sensitivity for identifying tumor-positive cases. False negatives are critical in medical diagnosis as they may result in missed tumors. For example, VGG19 misclassified 5 tumor-positive cases as negative, whereas ResNet50 reduced this to 3.

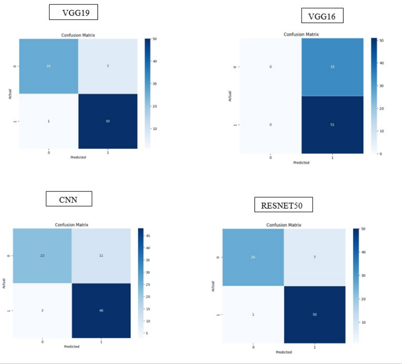


Fig. 10. confusion matrix.

By applying Grad-CAM, we observed that CNNs often focus on irrelevant regions during incorrect classifications. In contrast, ResNet50 more consistently attended to central tumor areas, suggesting better feature localization. This indicated that deeper networks with residual connections could learn more relevant spatial representations.

3.1.3 Classification report

The classification report in Table 2 shows various metrics such as precision, recall,F1 score, and support. The CNN model had the lowest precision, whereas the tuned VGG16, VGG19, and aResNet50 exhibited the best prediction accuracy. *Precision:* Among all the models, tuned Vgg16 had the highest precision (0.93), which suggests that it is highly accurate in predicting brain tumors when it identifies a positive class. Both the tuned VGG19 and ResNet50 models have the same precision (0.88), implying that they are equally effective in identifying positive cases. However, the CNN model had the lowest precision (0.76), indicating that it may have a higher rate of false positives than the other models.

Table 2. Classification report.

Model	Precision	Recall	F1 Score	Support
CNN	0.76	0.76	0.76	51
$VGG16$	0.93	0.82	0.87	51
$VGG19$	0.88	0.98	0.93	51
$ResNet50$	0.88	0.98	0.93	51

Recall: The recall metric evaluates how well a model can accurately detect all relevant instances. Among tuned VGG19 and ResNet50, both models show a high sensitivity to positive cases with a recall score of 0.98. However, tuned VGG16 has a slightly lower recall score of 0.82, which suggests that it may miss some positive instances compared with tuned VGG19 and ResNet50.In contrast, the CNN model had a recall score of 0.76, which aligns with its precision score and implies a balance between true positives and false negatives.

F1 Score: The performance of a model was evaluated using the F1 score, which is a balanced metric of precision and recall based on the harmonic mean. The F1score of Tuned VGG16, VGG19, and ResNet50 was 0.93, indicating that precision and recall were balanced in these models. In contrast, the CNN model has an F1 score of 0.76, which is the lowest among all models. This reflects the tradeoff between precision and recall, which is evident in the individual metrics of the CNN model.

Support: In this particular dataset, each class had an equal support value of 51, indicating consistency in the dataset size for each class. This means that the analysis is based on an equal number of instances for each class, which helps ensure that the results are not skewed in favor of any particular class.

3.2 Ablation studies

3.2.1 Ablation study on augmentation

The different augmentation techniques were evaluated based on the model evaluation metrics shown in Fig. 11. Combined augmentation provided better results than no augmentation did.

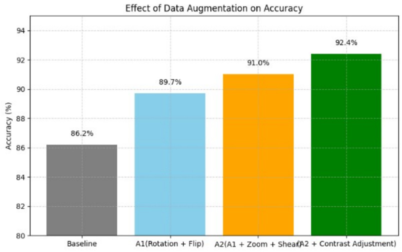


Fig. 11. Ablation study augmentation.

3.2.2 Ablation study on k cross validation

Evaluating model performance using a reliable validation strategy is essential, particularly in medical imaging tasks, such as brain tumor classification, where data are often limited and class distributions may be imbalanced[20]. In this study,we compared the traditional train-test split validation with k-fold cross-validation to understand their impact on model robustness and generalizability. The train-test split approach, with a commonly used ratio of 70:30, provides a simple and computationally efficient means of performance evaluation. However, this method is highly sensitive to the specific partitioning of data, which can lead to variability in performance metrics, particularly in datasets that are small or exhibit class imbalance.

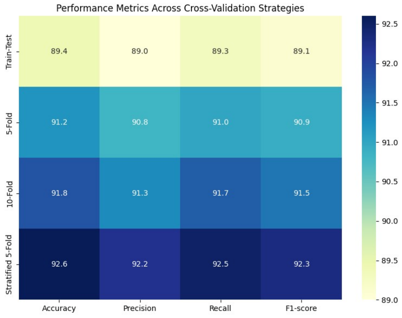


Fig. 12. Ablation cross validation.

In contrast, the k-fold cross-validation in Fig. 12 offers a more robust and statistically reliable alternative by systematically dividing the dataset into k subsets and iteratively using each fold for validation while training on the remaining folds. This procedure mitigates the risk of biased evaluation by ensuring that each sample contributes to both training and validation. The empirical results from this study demonstrated that 5-fold cross-validation improved the classification accuracy to 91.2 percent, whereas 10-fold cross-validation provided a slight further enhancement, achieving 91.8 percent accuracy. Among the strategies evaluated, stratified 5-fold cross-validation yielded the most consistent and superior performance, with an accuracy of 92.6 percent, owing to its ability to maintain the class distribution across folds. These findings underscore the importance of selecting appropriate validation techniques, particularly in the context of medical imaging tasks that involve limited and imbalanced datasets.

3.2.3 Ablation study on finetuned models

A comparison of the base model and fine-tuned models is shown in Fig. 13, and the base layers of the TL model were fine-tuned by adding max pooling and dense layers.

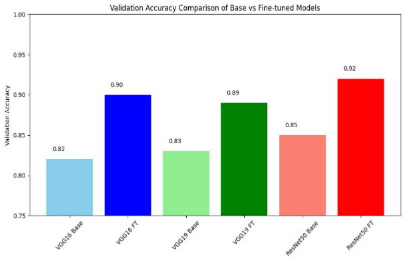


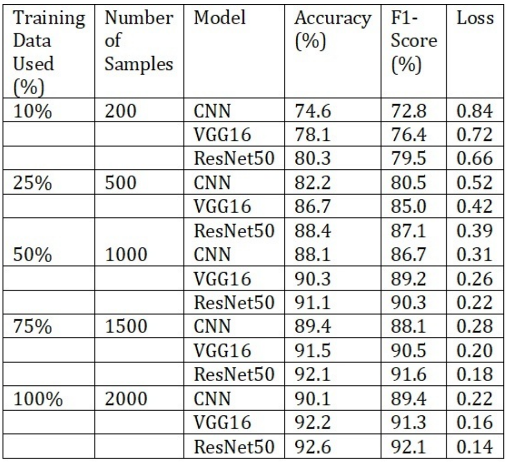
Fig. 13. Ablation fine-tuned models.

3.2.4 Ablation study on dataset

Deep learning models are highly data-dependent, and their generalization ability often scales with the volume and diversity of the training samples. In medical imaging, acquiring large annotated datasets is challenging because of privacy concerns, expert labeling requirements, and class imbalance. To explore the minimum data requirements for an effective model performance in brain tumor classification, we conducted a series of controlled experiments using incremental data subsets. We trained the selected architectures (VGG16, ResNet50, and a custom CNN) using progressively larger portions of the training set (10, 25, 50, 75, and 100 percent) while keeping the validation and test sets fixed. Performance was measured in terms of accuracy, F1-score, and loss across each subset.In Fig. 14, the results, visualized as a learning curve, indicated that at less than 25 percent of the training data, all models underfit, with accuracy below 80 percent and high variance. At 50 percent, the models began to stabilize with consistent validation accuracy of > 88 percent in CNN and > 90 percent in VGG16/ResNet50.Full dataset usage led to marginal improvements (1–2 percent) but significant stability in convergence and reduced overfitting.

3.2.5 hyperparameter tuning of the proposed architecture

The ideal configuration was obtained with a learning rate of 0.000132 and dropout rate of 0.5. We chose Adam as the optimizer, and 30 epochs were used. The final convolutional block+ Dense Layer was selected as the fine-tuning approach. The average performance indicators for all cross-validation folds were above 90 percent above.



Training Data Used (%)	Number of Samples	Model	Accuracy (%)	F1-Score (%)	Loss
10%	200	CNN	74.6	72.8	0.84
		VGG16	78.1	76.4	0.72
		ResNet50	80.3	79.5	0.66
25%	500	CNN	82.2	80.5	0.52
		VGG16	86.7	85.0	0.42
		ResNet50	88.4	87.1	0.39
50%	1000	CNN	88.1	86.7	0.31
		VGG16	90.3	89.2	0.26
		ResNet50	91.1	90.3	0.22
75%	1500	CNN	89.4	88.1	0.28
		VGG16	91.5	90.5	0.20
		ResNet50	92.1	91.6	0.18
100%	2000	CNN	90.1	89.4	0.22
		VGG16	92.2	91.3	0.16
		ResNet50	92.6	92.1	0.14

Fig. 14. Comparison of different datasize.

3.3 Discussion

The fine-tuned VGG16 model consistently outperformed the other models in terms of precision and other evaluation metrics, indicating that it is highly accurate in predicting positive cases. In contrast, the tuned VGG19 and ResNet50 models exhibited similar levels of effectiveness in detecting positive instances. When evaluating the models based on metrics such as precision and F1 score, it was observed that the performance of the tuned CNN model lagged behind that of our proposed architecture, indicating a lower overall performance of the model. In contrast, as shown in Fig. 15, the tuned VGG16, VGG19, and ResNet50 models demonstrated a more balanced performance, with training and validation accuracies that were closer in magnitude. For instance, the tuned VGG16 model exhibited a training accuracyof 0.92 and a test accuracy of 0.86, indicating better generalization to new data compared to the CNN model.

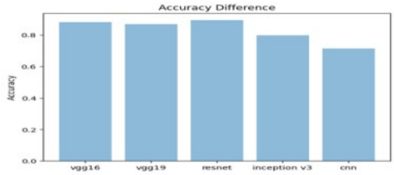


Fig. 15. The tuned VGG16, VGG19, and ResNet50 models.

The results from the CNN model underscore the importance of addressing overfitting in neural-network architectures. Our proposed architecture can be improved using regularization, data generation, and tuning. In summary, although the CNN model exhibited a relatively high training accuracy, its lower validation accuracy indicated susceptibility to overfitting. The comparison with VGG16, VGG19, and ResNet50 highlights the significance of developing models that not only perform well on training data but also generalize effectively to new and unseen datasets.

4. Conclusion

This study addressed overfitting in sparse datasets, which often results in a suboptimal validation performance. It explores the efficacy of transfer learning models by comparing architectures, such as VGG and ResNet. The results show that transfer learning models can significantly enhance the sparse data performance, thereby mitigating overfitting. The model evaluation used comprehensive metrics including a confusion matrix and other measures. This provides an understanding of the training and testing accuracy differences for CNN and how fine-tuned pre-trained models perform on sparse data. Overfitting problems can be avoided by using strategies such as data augmentation, hyperparameter tuning, and generative adversarial networks, and there are ways to make research more useful in healthcare. First, hospitals can use classification models to help radiologists quickly analyze images. Second, creating easy-to-use AI tools with visual aids helps to build trust in AI decisions. Third, using simple models on devices can speed up diagnosis in places with fewer resources. Testing these models in different hospitals showed that they work well with various MRI machines and patients. Finally, federated learning allows hospitals to improve models while keeping patient data private, speeding up AI use in healthcare.

5. Acknowledgements

- ***Funding Statement***: No funding was provided for this study.

- ***Ethical Compliance***: All procedures performed in studies involving human participants were in accordance with the ethical standards of the institution.

- ***Data Access Statement***: We thank ”Navoneel/brain MRI images for brain tumor detection” for providing the publicly available brain tumor dataset on Kaggle, which was used in this study.

- ***Conflict of Interest declaration***: The authors declare that they have no conflict of interest.

- ***Author Contributions***: Author1 contributed to the design, implementation, and analysis of the results and to the writing of the manuscript. Author2 conceived the original study and supervised the project.

References

[1] Wang S, Qureshi MA, Miralles-Pechuán L, Huynh-The T, Gadekallu TR, Liyanage M. Explainable AI for 6G use cases: technical aspects and research challenges. IEEE Open J Commun Soc. 2024;5:2490–540.

[2] Goel SD. Textbook of hospital administration. Elsevier Health Sciences; 2013.

[3] Rudin C, Chen C, Chen Z, Huang H, Semenova L, Zhong C. Interpretable machine learning: Fundamental principles and ten grand challenges. Stat Surv. 2022;16.

[4] Chengoden R, Victor N, Huynh-The T, Yenduri G, Jhaveri RH, Alazab M, et al. Metaverse to healthcare: A survey of potential applications, challenges, and future directions. IEEE Access. 2023;11:12765–95.

[5] Alzubaidi L, Bai J, Al-Sabaawi A, Santamaría J, Albahri AS, Al-Dabbagh BSN, et al. A survey on deep learning tools dealing with data scarcity: Definitions, challenges, solutions, tips, and applications. J Big Data. 2023;10(1).

[6] Müller T, Evans A, Schied C, Keller A. Instant neural graphics primitives with multiresolution hash encoding. ACM Trans Graph. 2022;41(4):1–15.

[7] Li J, Liu Y, Wang Q, Xing Z, Zeng F. Rotating machinery anomaly detection using data reconstruction generative adversarial networks with vibration energy analysis. AIP Adv. 2022;12(3).

[8] Rafiq G, Rafiq M, Choi GS. Video description: Comprehensive survey of deep learning approaches. Artif Intell Rev. 2023;56(11):13293–372.

[9] Sriprateep K, Khonjun S, Golinska-Dawson P, Pitakaso R, Luesak P, Srichok T, et al. Automated classification of agricultural species through a parallel artificial intelligence system, ensemble deep learning. Mathematics. 2024;12(2):351.

[10] Ashafuddula NIM, Islam R. ContourTL-Net: contour-based transfer-learning algorithm for early stage brain tumor detection. Int J Biomed Imaging. 2024;2024:1–20.

[11] Abdusalomov AB, Mukhiddinov M, Whangbo TK. Brain tumor detection based on deep learning approaches and magnetic resonance imaging. Cancers (Basel).2023;15(16):4172.

[12] Dahan F. Transformation of MRI images to three-level color spaces for brain tumor classification using Deep-Net. Intell Autom Soft Comput. 2024;39(2):381–95.

[13] Rasool N, Bhat JI. Unveiling the complexity of medical imaging using deep learning approaches. Chaos Theory Appl. 2023;5(4):267–80.

[14] Kumar NS, Deepika G, Goutham V, Buvaneswari B, Reddy RVK, Angadi S, et al. HARNet in the deep learning approach: A systematic survey. Sci Rep. 2024;14(1).

[15] Kolluri J, Das R. Evaluation of Deep Learning-Based Object identification. Int J Recent Innov Trends Comput Commun. 2022;10(1s):52–80.

[16] Wojciuk M, Swiderska-Chadaj Z, Siwek K, Gertych A. Improving classification accuracy of fine-tuned CNN models: Impact of hyperparameter optimization. Heliyon. 2024;10(5):e26586.

[17] Asiri AA, Aamir M, Irfan M, Shaf A, Ali T, Alqahtani S. Enhancing brain tumor diagnosis: an optimized CNN hyperparameter model for improved accuracy and reliability. PeerJ Comput Sci. 2024;10:e1878.

[18] Khan F, Mir MS, Soomro AB, Majid M, Ayoub S, Gulzar Y. Least-square support vector machine-based brain tumor classification system with multi-model texture features. Front Appl Math Stat. 2023;9:1324054.

[19] Chicco D, Jurman G, Tötsch N. The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in the two-class confusion matrix evaluation. BioData Min. 2021;14(1):10.

[20] Dewage TW, Rehman B, Hasan R, Mahmood S. Enhancing brain tumor detection through custom convolutional neural networks and interpretability-driven analysis. Information. 2024;15(10):653.

[21] Ullah MS, Khan MA, Alturki N, Mzoughi O, Saidani O, Masood A. Brain tumor classification from MRI scans: a framework of hybrid deep learning model with Bayesian optimization and quantum theory-based marine predator algorithm. Front Oncol. 2024;14:1335740.