

Efficient Feature Learning Based Xception CNN Model Optimization for MRI Brain Tumor Image Classification



Shaik Nasreen¹, Prakash Balasubramanian², Saleena Badarudeen², Mamoon Rashid^{3*}, Sultan S. Alshamrani⁴, Aisha Banu Wahab⁵, Ahmed Saeed AlGhamdi⁶

¹ Department of Computer Science and Technology, Madanapalle Institute of Technology and Science, Madanapalle 517325, India

² School of Computer Science and Engineering, Vellore Institute of Technology, Chennai 600127, India

³ School of Information Communication and Technology, Bahrain Polytechnic, Isa Town 33349, Kingdom of Bahrain

⁴ Department of Information Technology, College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia

⁵ Department of Computer Science and Engineering, BS Abdur Rahman Crescent Institute of Science and Technology, Chennai 600048, India

⁶ Department of Computer Engineering, College of Computer and Information Technology, Taif University, Taif 21994, Saudi Arabia

Corresponding Author Email: mamoon873@gmail.com

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/ts.420124>

ABSTRACT

Received: 14 April 2024
Revised: 19 September 2024
Accepted: 25 January 2025
Available online: 28 February 2025

Keywords:

brain tumor, Xception model, ImageDataGenerator, Convolution Neural Network (CNN), hyperparameter tuning

Brain tumors are abnormal growths of brain cells that can be benign (non-tumor) or malignant (tumor). These tumors can arise from different types of brain cells and occur in various brain regions. Timely detection is crucial for reducing the severity and improving prognosis. However, the traditional human examination suffers in early tumor detection due to the irregular patterns in MRI scans. Additionally, Machine learning and deep learning-based frameworks detect brain tumors more accurately than human analysis. This work introduces an efficient diagnostic approach with improved accuracy to classify the benign and malignant from MRI scans. This diagnostic approach consists of three levels. In the first level, the majority and minority samples are increased to train the framework with more subjects using ImageDataGenerator with real-time data augmentation. In the second level, a pre-trained Convolution Neural Network (CNN), namely the Xception framework, is utilized to learn comprehensive information about images. The hyperparameter tuning process improves the multi-class classification accuracy in the third level. The proposed framework classifies brain tumors into multiple such as glioma, meningioma, no tumor, and pituitary. The experimental dataset is obtained from the Kaggle repository to train the framework. The outcomes attained by the proposed framework deliberate higher accuracy compared with other CNN frameworks. The proposed framework proves its efficiency in the fine-grained classification of brain tumors with a validation accuracy of 99.87%. Thus, this framework may be employed in clinical services to diagnose brain MRI tumors.

1. INTRODUCTION

Brain tumors are among the illnesses that pose a severe risk to life due to the difficulty in identifying the factors that cause them. Machine learning and deep learning can help identify tumors at earlier stages more quickly and effectively than human examination on the scans. The frameworks are trained with a bulk of magnetic resonance imaging (MRI) scans; this allows the frameworks to recognize and classify the tumors accordingly. These machine learning (ML) and deep learning (DL) approaches attracted researchers to utilize them in medical diagnosis. There are three possible types of tumors: malignant, non-cancerous, and occasionally precancerous. Brain tumors lead to fatalities, with the most significant growth rate among people over 65. They are the second

leading origin of cancer-related mortality in children under 15, with the youngest average age among cancer-related deaths. Around 71% of all brain tumors are classified as benign or non-cancerous, while malignant or cancerous tumors account for about 29% [1].

Brain tumors are categorized into two main types: primary tumors and secondary tumors. The early tumors develop in the brain and do not travel to other body areas; nevertheless, they may grow to different brain and spinal cord sections. Additionally, secondary tumors develop in other areas of the body and eventually spread to the brain. Primary tumors are more likely to spread than secondary tumors. Glioma, meningioma, and pituitary are some of the primary brain tumors that occur the most. Glioma refers to cell growth in the brain or spinal cord, known as glial cells. Glioma can be either

cancerous or non-cancerous. Meninges that serve as membranes around the brain are the origin of the meningioma tumor; these tumors evolve slowly, typically taking years without showing any signs of it. The abnormal development of cells in the pituitary gland is called a pituitary tumor [2].

These primary brain tumors are difficult to identify in MRI images by human examination due to their different patterns. Several types of research have been done to ease the difficulty in detecting the various types of brain tumors from MRI images by indulging the concepts of ML and DL. ML uses image processing, segmentation, feature selection, and augmentation to eliminate the noise in MRI images and make them look clearer to categorize them better [3]. DL is a subfield of ML that provides various frameworks that use artificial neural networks to work like a human brain [4]. Deep learning has enhanced the advancement in detecting multiple medical diseases, including brain tumors. Several deep learning frameworks proposed for tumor detection have achieved a different accuracy range.

1.1 Problem statement

Brain tumors present a major health challenge, and early detection is essential for enhancing patient outcomes. However, the challenges posed by the irregular patterns in MRI scans make accurate detection difficult for traditional human examination methods. In contrast, ML and DL-based frameworks have demonstrated superior accuracy in brain tumor detection compared to human analysis. Thus, there is a critical need to develop an efficient diagnostic approach that combines the expertise of medical professionals with the power of DL approaches to classify benign and malignant brain tumors from MRI scans with enhanced accuracy. This research aims to design and evaluate such an efficient diagnostic framework to provide reliable and efficient support to medical practitioners in diagnosing brain MRI tumors. This method's success may enhance patient care and outcomes by enabling early detection and appropriate intervention in cases of brain tumors.

1.2 Contribution

This study presents a novel and effective method for the precise classification of brain tumors using MRI scan data. The proposed system consists of three levels of processing to achieve improved accuracy:

In the first level, our approach increases the majority and minority samples using ImageDataGenerator with real-time data augmentation. This step helps mitigate class imbalance in the dataset, enhancing the framework's capacity to handle various tumor types effectively.

In the second level, a CNN framework, specifically the pre-trained Xception architecture, is utilized. The Xception framework is chosen because it can learn comprehensive information from images, making it well-suited for brain tumor classification.

The third level involves hyperparameter tuning to optimize the multi-class classification accuracy further. Fine-tuning the framework's hyperparameters allows for better generalization and performance on the test data.

The proposed framework categorizes brain tumors into various types, such as glioma, meningioma, pituitary tumors, and non-cancerous cases. The dataset utilized to train the framework was sourced from the Kaggle repository.

The proposed framework's results show higher accuracy than other CNN frameworks. Specifically, the proposed framework achieves a validation accuracy of 99.87% in fine-grained classification of brain tumors. This high accuracy determines the efficiency and effectiveness of the framework in accurately diagnosing brain MRI tumors.

The structure of the article is planned as follows: The literature survey is presented in Section 2 of the article, Section 3 details the materials and methods applied in this research, and Section 4 assesses the study's methodology. Section 5 discusses the findings and provides an analysis, and Section 6 offers a conclusion along with suggestions for future research directions.

2. RELATED WORKS

Numerous studies have explored early brain tumor detection through various ML and DL techniques [5]. According to Siar and Teshnehlab [6], Gaussian filter was used in the pre-processing step to reduce the noise in the images, which were then normalized to minimize the differences in the size of each image. A VGG16 CNN framework was utilized as a feature extractor, paired with an SVM classifier for classification on a public dataset, achieving a 98.8% accuracy. In another study, Rehman et al. [7] proposed a fine-tuned, pre-trained ResNet50 CNN framework to classify tumors including meningioma, glioma, pituitary, and no tumor. The framework was optimized with an SGD optimizer, which achieved an accuracy of 98.69% on the MRI scans of the ImageNet dataset.

Khan et al. [8] proposed deep learning, k-means clustering, and data augmentation approaches for fine-tuned classification. The U-Net CNN has been employed for the segmentation. Several pre-trained CNN frameworks like ResNet-50, VGG16, and Xception were tested to obtain higher accuracy, where the Xception framework outperformed with the highest accuracy of 97.83% on the BRATS 2018 dataset. Noreen et al. [9] evaluated DensNet201 and InceptionV3 pre-trained frameworks as feature extractors using the SoftMax classifier, which showed an accuracy of 99.51% and 99.34%, respectively. Saleh et al. [10] used five pre-trained frameworks of CNN for tumor classification. Xception, InceptionV3, MobileNet, and ResNet50 were used to classify the tumor. The Xception framework outperformed with an accuracy of 98.75%.

Toğaçar et al. [11] used the hypercolumn approach in the layers of the convolution framework and presented a BrainMRNet framework. The framework was used to extract the characteristics. An SGDR optimizer was used for optimization, and the activation function chosen was ReLu. The framework's accuracy in determining the kind of tumor was 96.05%. The dataset was first pre-processed and normalized before being fed to the framework. Siddique et al. [12] utilized a pre-trained VGG16 approach within a Deep CNN framework to classify tumor images. When applied to an MRI image dataset, the framework demonstrated enhanced performance over conventional methods, achieving an accuracy rate of 96%.

Sharif et al. [13] introduced a DL framework using a fine-tuned Densenet201 pre-trained architecture, integrating feature extraction enhanced through an improved genetic algorithm and an Entropy–Kurtosis-based High Feature Value (EKbHFV) approach. The extracted features were classified

using a multi-class SVM cubic classifier. The framework has attained a precision of 95% on the BRATS dataset. İncir and Bozkurt [14] compared the pre-trained CNN frameworks. MobileNetV2, InceptionV3, and VGG19 were tested on a public dataset. That was pre-processed by resizing them. The CNN framework used ReLu and Soft-Max activation functions in the dense layers. The MobileNetV2 framework outpaced with an accuracy of 92%.

Agarwal et al. [15] developed a framework combining an orthogonal and Berkeley wavelet transform (BWT) with a DL classifier. To extract features, they employed the grey-level co-occurrence matrix (GLCM) technique. The genetic algorithm was used for the feature selection. CNN obtained an accuracy of 97.3% on the Health Insurance Probability and Accountability Act (HIPAA) dataset. Rasool et al. [16] employed a pre-trained GoogleNet framework with a fine-tuned GoogleNet framework that included a SoftMax classifier as a feature extractor in the SVM classifier. On the MRI dataset, the modified GoogleNet framework was employed to extract features in the SVM classifier, and it achieved a high precision of 98.1%.

Raza et al. [17] projected a hybrid CNN approach. DeepTumorNet is a hybrid framework that Google developed based on the architecture of GoogleNet. Within the framework was included the usage of the leaky ReLu activation function. With an accuracy of 99.67%, this framework outperformed Resnet50, AlexNet, darknet53, ShuffleNet, SqueezeNet, GoogLeNet, ResNet101, Exception Net, and MobileNetv2. A hybrid technique was presented for the categorization of tumors by Senan et al. [18]. ML and DL are both types of learning combined in this approach. The AlexNet and ResNet-18 were used for the feature extraction in the SVM algorithm with SoftMax activation function to classify the brain tumor. AlexNet, as a feature extractor in the SVM algorithm, got a better precision of 95.10% in the classification process.

Arefin et al. [19] compared the accuracy of ResNet50 and InceptionV3 in classifying brain tumors. The LGG dataset images underwent pre-processing and augmentation before being input into CNN frameworks. In these frameworks, the ResNet50 and InceptionV3 encoders were integrated with a U-net architecture. The framework combining the ResNet50 encoder with the U-net architecture achieved superior performance, reaching a precision of 99.77%. Bashkandi et al. [20] introduced a combination of a CNN, a particle swarm optimizer to optimize the hyperparameters, and a political optimizer to select the informative features. The data was pre-processed by histogram equalization, gamma correction, normalized, and augmented. The images were then fed to the framework to categorize the brain tumor, which showed an exactness of 97.09%.

Nanda et al. [21] propose a social spider Basis Neural Network (SSO-RBNN) in which the data was pre-processed using Median filters. After the pre-processing step, the features were extracted by first-order intensity statistical and segmentation-based features. The authors have mentioned the usage of Saliency-K-means segmentation and the (SSO-RBNN) for classification, which achieved an accuracy of 96% on the MRI images. Rahman et al. [22] conducted the category of brain tumor using the Parallel Deep CNN, which used the ReLu activation function in the convolutional layers. SoftMax was used in the classification network. This framework gave an accuracy of 97.30% after augmentation. After applying pre-processing techniques, Satyanarayana et al. [23] used a DCNN to categorize brain tumors on the dataset. A mass correlation-

based pre-processing method helped achieve a better accuracy of 93.62% while classifying the high-grade glioma tumor using the DNN framework.

Kurdi et al. [24] proposed a Harris Hawks optimized convolution network (HOCNN), a meta-heuristic optimized CNN that attained a precision of 98%. The quality of the images in the dataset was improved by applying histogram equalization and a median filter. The framework was accomplished on a small dataset which limits the framework's performance on other datasets. Jaspin and Selvan [25] proposed a Multi-class Classification Neural Network in which the data was pre-processed to eliminate the noise and normalized. The normalized data were then augmented with techniques like flipping and resizing. The data was then fed to the MCNN framework, which achieved an accuracy of 99% respectively.

2.1 Extract from literature

The survey mentioned above elaborates on the various machine learning frameworks like InceptionV3, DensNet201, ResNet50, VGG16, and Xception with their efficacy in handling various issues with its accuracy. The frameworks were fine-tuned using multiple techniques for optimization, like SGDR, SGD, MGA, and other optimizers. SoftMax and ReLu activation functions were used to bring the non-linearity into more complex neural network framework functions. However, dense frameworks tend to perform better on more complex tasks and larger datasets, but they require more computational resources for training and inference. Hence, Xception is designed to be more computationally efficient compared to traditional Inception frameworks and ResNet variants.

Based on the recent investigation, the Xception framework outperformed in classifying the brain tumor. In addition, the Xception framework received high accuracy in several studies. According to the study by Khan et al. [8], the Xception framework achieved a higher concert associated with other pre-trained frameworks, reaching a precision of 97.83% on the BRATS 2018 dataset. The Xception framework was utilized for categorizing tumors by Saleh et al. [10] and achieved a 98.75% accuracy. In this approach, we used the data augmentation technique to increase the minority classes, thereby improving the training process and achieving better multi-class classification accuracy.

3. METHODS AND MATERIALS

3.1 Dataset

This study uses the brain tumor MRI dataset found on the Kaggle repository [26]. In all, the collection contains 3264 different files. Testing and training data have previously been extracted from the dataset and separated. This dataset from Kaggle categorizes glioma, meningioma, pituitary, and no tumor. The testing data comprises 105 photographs with no tumor, 100 images with glioma, 115 with meningioma, 74 with pituitary, and 100 with no tumor. The training set includes 395 photos that do not have tumors in addition to 822 meningiomas, 826 gliomas, and 827 pituitary tumors. The scattering of various forms of brain tumors in the sample is shown in Figure 1.

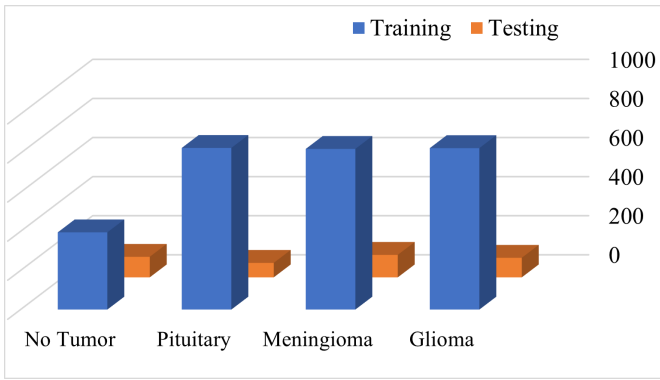


Figure 1. Dataset proportion

3.2 Image pre-processing and augmentation

Pre-processing is a crucial step to enrich accuracy by enhancing the image quality. This study applies various pre-processing steps to the dataset [27]. In the first step, we use a Gaussian blur filter to smoothen the image's appearance by minimizing noise and blurring the image. In the second step, we did thresholding to separate the image into background and

foreground by setting the pixels with values less than 45 to black and greater than 45 to white. Then, we erode the images twice to remove the tiny noises and make the tumor's edges more defined. The borders of the cancer are detected, and the outer counters of the tumor are found using a computer vision library. The extreme points of the counter are determined to crop the image. The extraction of ROI removes the unwanted background of the image.

In this study, the dataset is not satisfactory for the framework to train accurately. Hence, data augmentation is introduced to raise the dataset size, and thereby, it also reduces the overfitting issue. The ImageDataGenerator class of Keras is used to perform various augmentation techniques. Later, the images are rotated to improve the framework's training by providing multiple angles for each image. Then, the shifting of images is performed to get different tumor positions. Shearing is applied to achieve robustness by generating various views of the image. The Brightness of the images is adjusted by adjusting the brightness range. Horizontal and vertical flipping increases the data and the framework's generalization. The procedure of data preprocessing and data augmentation is demonstrated in Figure 2. In addition, the dataset before and after augmentation is illustrated in Figure 3.

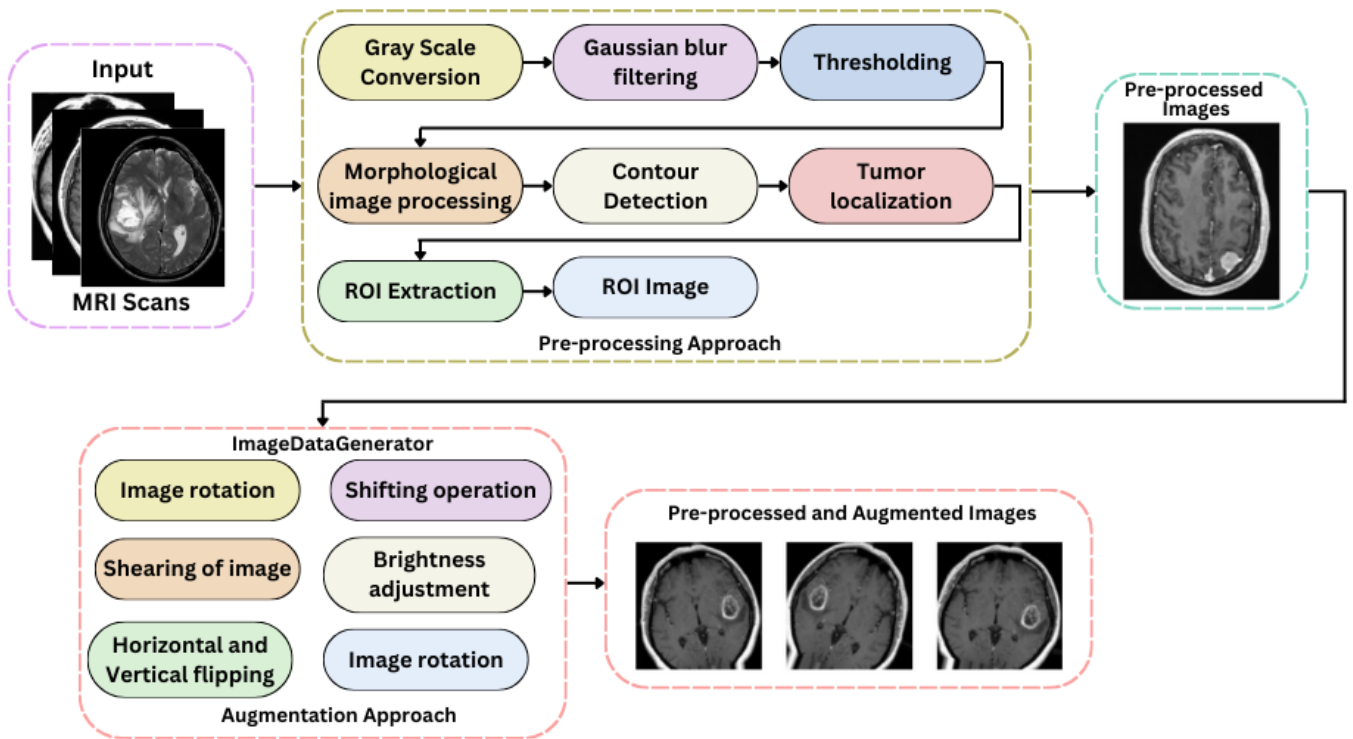


Figure 2. A block diagram depicting image pre-processing and augmentation techniques

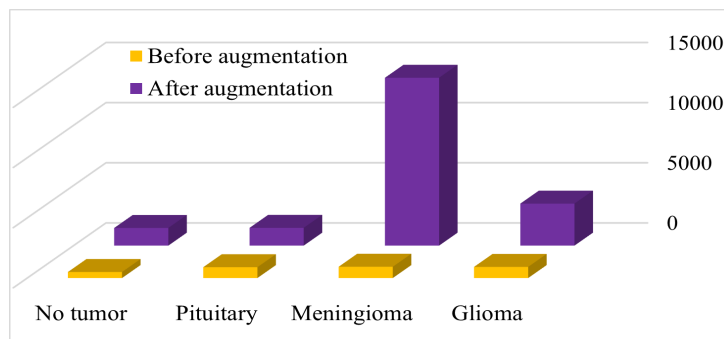


Figure 3. Dataset before and after augmentation

4. PROPOSED METHODOLOGY

The methodology in this paper follows a sequence in which the dataset was collected from Kaggle, which is first augmented to get all the possible dimensions of the image. The expanded dataset images are then pre-processed using various techniques mentioned in section 3 to extract the region of interest. The augmented and pre-processed dataset is then fed to the CNN framework as input to train the framework and get the results. The framework is finally evaluated by analyzing the performance metrics.

4.1 Proposed framework

This study used a pre-trained CNN framework, namely Xception [28]. The framework is 71 layers deep, out of which 14 are depth-wise separable layers and is an extension of the Inception framework. It was pre-trained on the large-scale ImageNet dataset. The framework utilizes channel-wise and spatial convolutional layers. As the initial layer in the framework, the depth-wise convolutional layer processes each input channel separately using a filter. This is commonly referred to as channel-wise convolution. The output from this layer is subsequently passed to the pointwise convolutional layer, which performs a 1×1 convolution and applies linear transformations to each individual feature pixel. This process is referred to as spatial convolution. The Xception framework

architecture has three stages: entry, middle, and exit flow. The architecture in Figure 4 describes the three different flows of the framework.

4.2 Hyperparameter tuning

The Xception framework has been utilized in this work due to its outstanding performance in various studies. The framework is hyper-tuned to get a more accurate brain tumors classification. Adam optimizer is employed in the Xception framework to improve the accuracy with efficient optimization. Adam optimizer dynamically adjusts the learning rate, making it suitable for the framework. The SoftMax activation function is used in the Xception framework for the multi-class classification of images. It helps the network produce normalized probability distribution among the classes. A batch size of 30 is used in training to improve convergence and good generalization. The framework is trained with 12 epochs to learn the underlying patterns of the data.

4.3 Working process of framework

Figure 5 illustrates the operational workflow of the Xception neural network used for brain tumor classification. The systematic approach of the Xception framework is presented as follows.

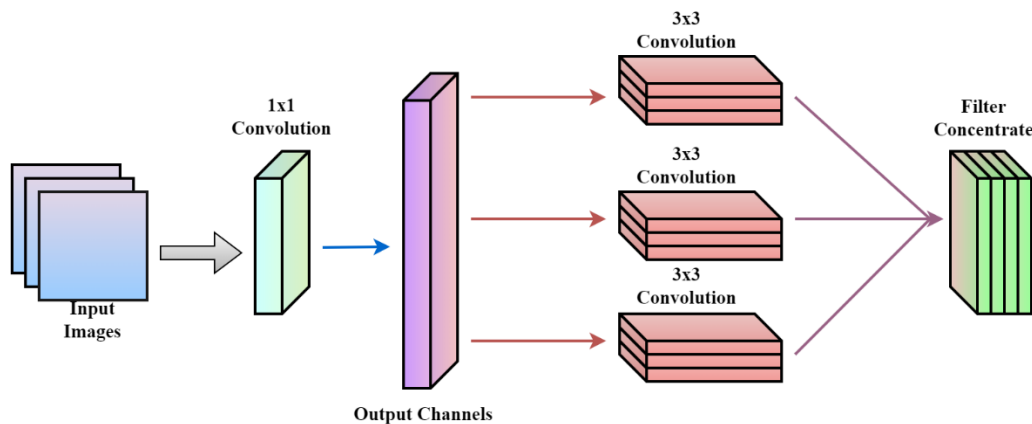
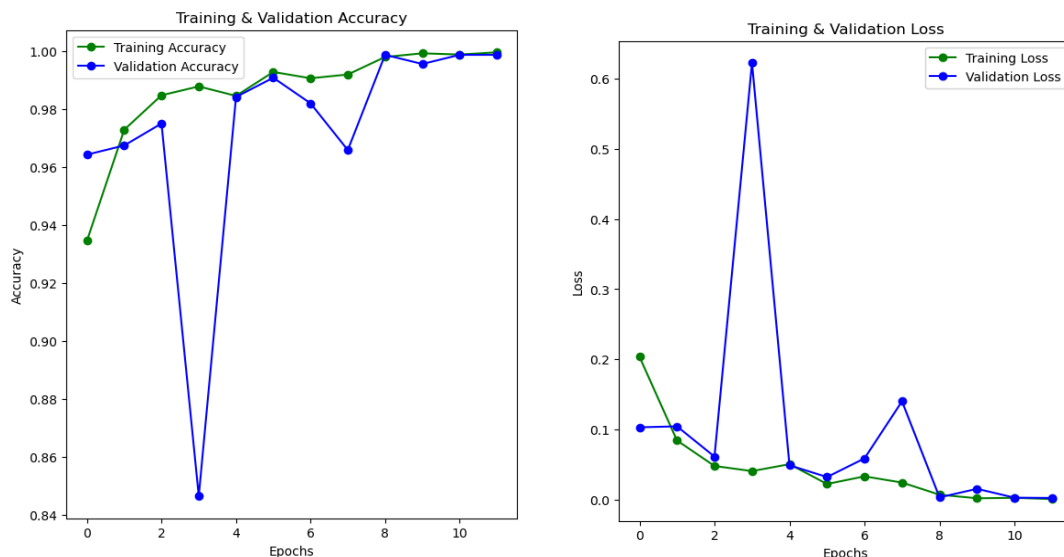
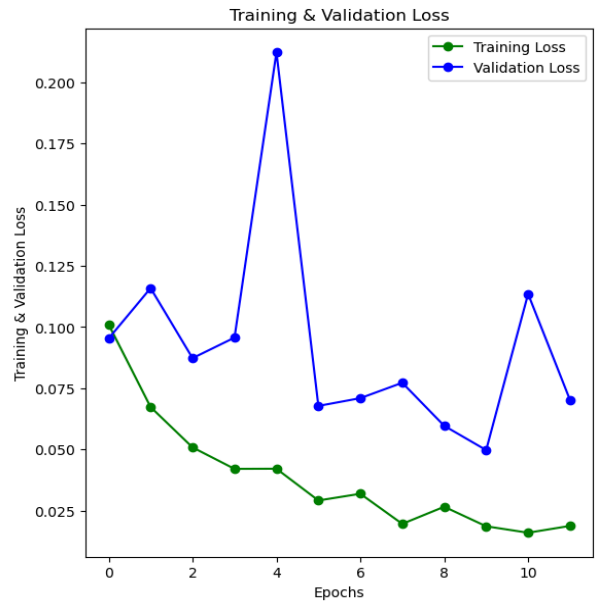
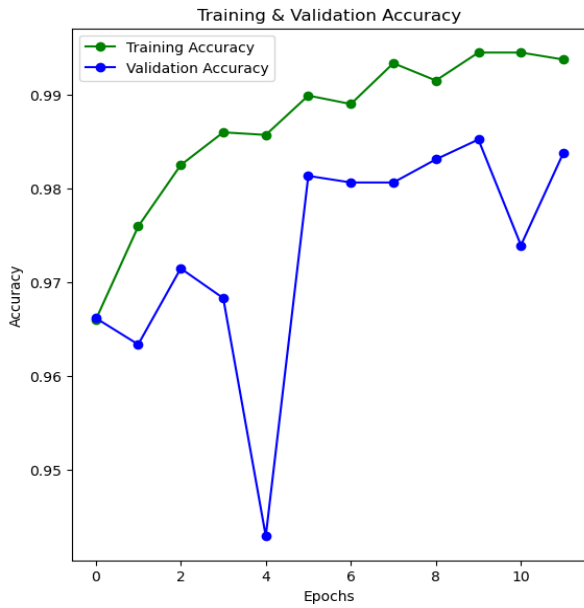


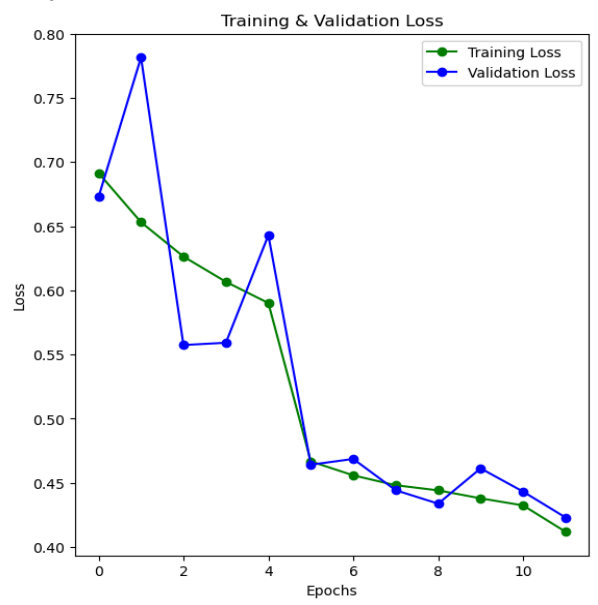
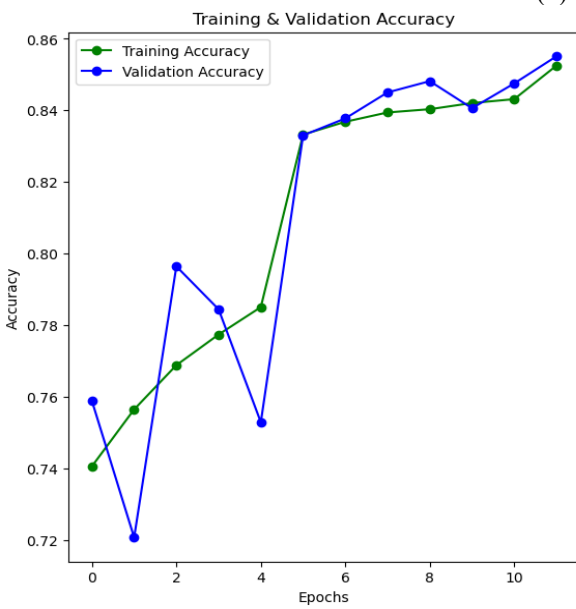
Figure 4. The general architecture of the Xception framework



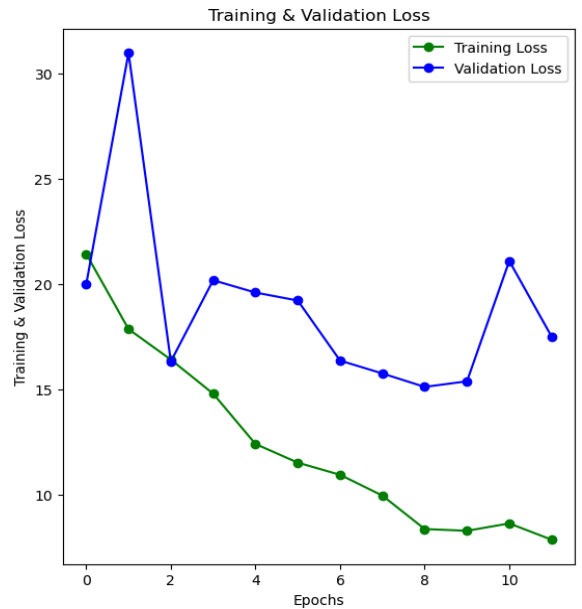
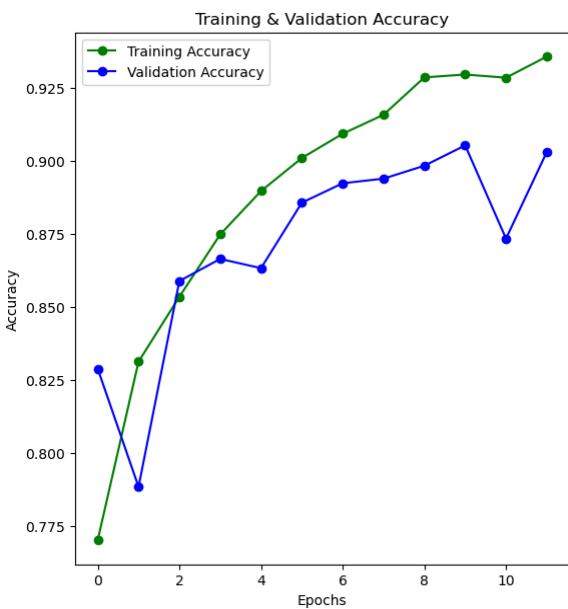
(a) Proposed Framework



(b) VGG 19



(c) ResNet 50



(d) InceptionV3

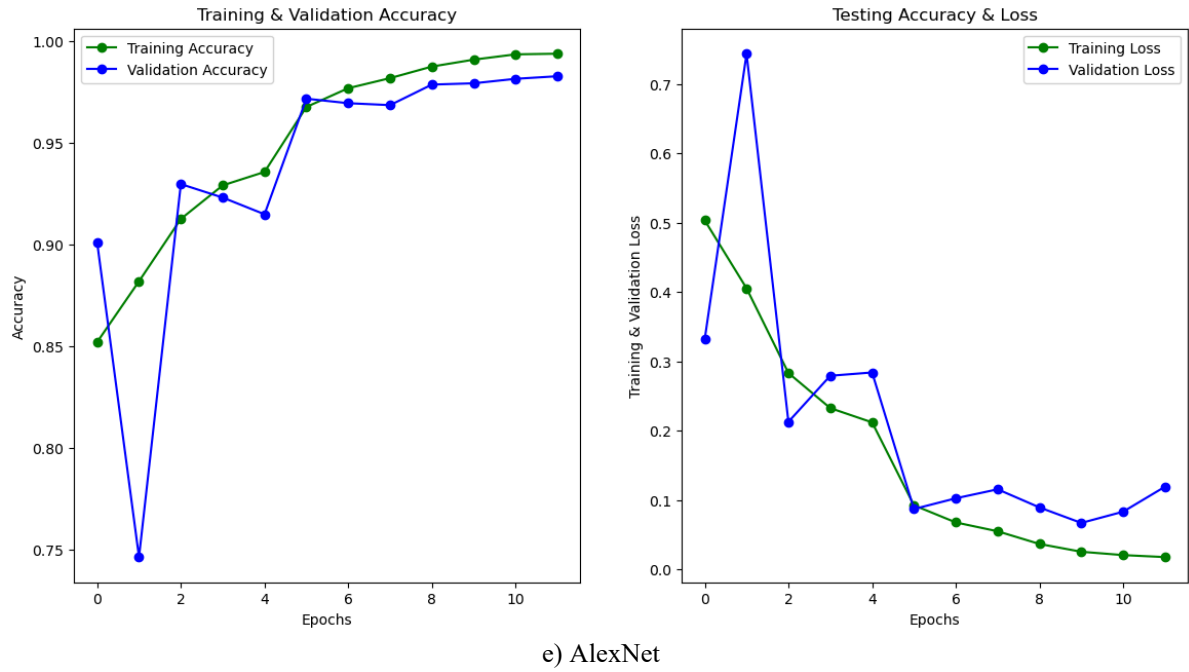


Figure 5. Illustration of the framework's validation loss, training loss, validation accuracy, and training accuracy

(1) The input image of size $\varepsilon \times \mu \times \omega$ is represented as a tensor $\chi \in R^{\varepsilon \times \mu \times \omega \times \gamma}$.

(2) Convolutional Layers: In the Xception architecture, each depthwise separable convolutional layer is implemented by first applying a depthwise convolution, followed by a pointwise convolution. The depth-wise convolution can be represented as:

$$\delta(\chi)_{ij\gamma} = \sum_{\iota=0}^{\mathcal{H}-1} \sum_{\kappa}^{\mathcal{L}-1} \chi(i + \iota, j + 1, \gamma) \times \omega d_{\iota\kappa}$$

where, $\delta(\chi)_{ij\gamma}$ is the output of the depthwise convolution at location (i, j) and channel γ , $\chi(i + \iota, j + 1, \gamma)$ is the input value at the location $(i + \iota, j + 1, \gamma)$ and channel γ , $\omega d_{\iota\kappa}$ is the depthwise convolutional filter at depth d and spatial location (ι, κ) , where \mathcal{L}, \mathcal{H} are the width and height of the filter, respectively.

The pointwise convolution can be represented as:

$$\rho(\delta(\chi))_{ij\gamma} = \sum_{d=0}^{\delta-1} \delta(\chi)_{ij\gamma d} \times \vartheta d_{\gamma}$$

where, $\rho(\delta(\chi))_{ij\gamma}$ is the output of the pointwise convolution at location (i, j) and channel γ , $\delta(\chi)_{ij\gamma d}$ is the output of the depthwise convolution at location (i, j) and depth d , ϑd_{γ} is the pointwise convolutional filter at depth d and channel γ , and δ is the numeral of output channels of the depthwise convolution.

Together, the depth-wise separable convolution can be represented as:

$$\sigma(\chi)_{ij\gamma} = \rho(\delta(\chi))_{ij\gamma}$$

where, $\sigma(\chi)_{ij\gamma}$ is the output of the convolutional layer at location (i, j) and channel γ .

(3) Skip Connections: Xception uses skip connections to connect some convolutional layers directly to the output. Let

$\sigma(\chi)$ and $\acute{\sigma}(\chi)$ denotes the output of two convolutional layers in Xception, and let Y be the output of the skip connection. The skip connection can be represented as:

$$\psi = \Delta(\sigma(\chi) + \acute{\sigma}(\chi))$$

where, Δ is an activation function (usually ReLU), and $+$ denotes element-wise addition.

(4) Global Average Pooling: A global average pooling layer is applied to the output feature maps, computing the average value for each feature map across all spatial positions. The global average pooling can be represented as:

$$\mathcal{G}(\sigma(\chi))_{\gamma} = \frac{1}{(\varepsilon \times \omega)} \times \sum_{i=0}^{\varepsilon-1} \sum_{j=0}^{\omega-1} \sigma(\chi)_{ij\gamma}$$

where, $\mathcal{G}(\sigma(\chi))_{\gamma}$ is the global average pooling layer produces an output for each channel γ .

(5) Fully Connected Layers: The feature vector is sent through a set of fully connected layers to produce the output probabilities for each class in the classification task. Let F be the feature vector and let ψ be the output probabilities. The fully connected layers can be represented as:

$$\mathcal{F} = [\mathcal{G}(\sigma(\chi))_1, \mathcal{G}(\sigma(\chi))_2, \dots, \mathcal{G}(\sigma(\chi))_{\delta}]$$

$$Y = \text{softmax}(\lambda \times \mathcal{F} + \phi)$$

The SoftMax function converts the output of fully connected layers into probability values. In this context, λ and ϕ represent the weight and bias matrices of these layers, respectively; δ denotes the number of output channels in the final convolutional layer.

4.4 Hyperparameter tuning and selection criteria

This study employed hyperparameter tuning to enrich the performance of the Xception framework for brain tumor classification. Key hyperparameters were optimized to

achieve the best trade-off between accuracy and generalization. The learning rate was initially configured at 0.001 and was adaptively modified throughout training by the Adam optimizer, which is recognized for its ability to adjust learning rates and promote stable convergence. Different batch sizes were evaluated, leading to the selection of a batch size of 30 to achieve an optimal balance between memory efficiency and training speed. Training epochs were limited to 12 to avoid overfitting, based on early convergence observed in validation trials.

The SoftMax activation function was applied in the final layer to generate normalized probability distributions across classes, ensuring effective multi-class classification. Transfer learning was leveraged by fine-tuning pre-trained weights from ImageNet, providing a solid foundation for accurate classification. Categorical cross-entropy was employed as the loss function, appropriate for multi-class tasks by measuring the difference between predicted probabilities and true class labels.

The hyperparameters were chosen after several rounds of k-fold cross-validation, aimed at enhancing framework robustness and reducing the risk of overfitting. The combination that yielded the highest validation accuracy and minimal loss was adopted. This tuning process contributed to the framework's superior performance, with the proposed approach achieving a validation accuracy of 99.87% and minimal validation loss (0.0026%). By systematically optimizing these parameters, the framework was able to balance accuracy, efficiency, and generalization effectively.

5. RESULT ANALYSIS AND DISCUSSION

In this section, we explore the experimental setup, assess the framework's performance, and compare it with other frameworks.

5.1 Experimental setup

This study seeks to categorize brain tumors into four distinct types: Meningioma, Glioma, Pituitary Tumor, and No Tumor, using Python as the primary programming language. TensorFlow and Keras are utilized to build, compile, and train the frameworks, with Keras serving as a high-level neural network API running on TensorFlow. The experiment was carried out on a system equipped with a 12th Gen Intel Core i7-12700F processor, running at a base clock speed of 2.10 GHz, with 32GB of RAM, and operating on a 64-bit version of Windows 10, with Jupyter Notebook serving as the development environment. The dataset comprises 3624 tumor files, which are pre-processed and augmented using Python libraries. The data is split into training (80%), validation (10%), and testing (10%) subsets. To enhance generalization and prevent overfitting, k-fold cross-validation is employed. This setup leverages high computational power and robust data management to achieve accurate brain tumor classification.

In addition, we use SoftMax as an activation function in the Xception framework for multi-class classification of images as it helps the network produce normalized probability distribution among the classes. SoftMax cross-entropy loss function helps in reducing overfitting by penalizing the framework. It assigns a greater probability in the case of the correct type and a lesser chance in the case of the wrong class.

The frameworks used to compare with the proposed framework of this study are described as follows.

5.1.1 InceptionV3

InceptionV3 is a convolutional neural network (CNN) introduced by Google. The inception framework served as the building block for the inceptionV3 framework. This pre-trained deep learning framework uses max pooling, convolutional layers, and average pooling layers to extract input image features [29].

5.1.2 ResNet50

ResNet50 is a Convolutional neural network framework introduced by Microsoft. It was proposed to overcome the limitations of ResNet architecture, such as vanishing gradient Descent. The introduction of ResNet50 made the training of the framework easy. ResNet50 contains five blocks in which the convolutional layers keep adding at the last block of five convolutional layers, which helps in better feature extraction [30].

5.1.3 VGG19

VGG19 was found at the University of Oxford. VGG19 is a 19-layer deep convolutional layer. It consists of 16 convolutional layers and three fully connected layers. VGG19 extracts more complex features from the images than VGG16 because of its depth [31].

5.1.4 AlexNet

AlexNet is an eight-layer deep CNN framework proposed in 2012. AlexNet has five layers of convolution with three fully connected layers. AlexNet has a dropout regularization that drops neurons preventing the overfitting issues in the framework [32].

The frameworks are tuned to achieve better accuracies in classifying brain tumors. These frameworks' accuracy is then compared with the Xception framework, which outperformed the other frameworks.

5.2 Performance metrics

Performance metrics are used as a standard for measuring the performance and efficiency of a framework. Metrics like sensitivity, Accuracy, Precision, Specificity, and False Positive Rate (FPR) are used to monitor the performance of a particular framework. The mathematical representation of the metrics is depicted in the equations below.

Sensitivity (Se), also known as the true positive rate, measures how accurately the system identifies positive cases. High sensitivity ensures most tumors are correctly detected, with very few false negatives.

$$Se = \frac{(T_p)}{(T_p) + (F_N)} \times 100$$

Precision (P) determines the percentage of tumor classes classified correctly out of all predictions made for that class. High precision indicates the excellent performance of the framework.

$$P = \frac{(T_p)}{(F_p) + (T_N)} \times 100$$

False Positive Rate (FPR) determines the percentage of

cases in which the tumor was classified incorrectly into a particular class as a positive classification out of all the pessimistic predictions that belong to that class. Low FPR determines the better performance of the framework.

$$FPR = \frac{(F_p)}{(F_p) + (T_N)} \times 100$$

Specificity (SP), Or True Negative Rate, determines the percentage of classifications predicted as negative for a particular class.

$$SP = \frac{(T_N)}{(T_N) + (F_p)} \times 100$$

F1-Score combines precision and recall values. F1-Score measures the framework's performance. A high F1-Score determines a good version of the framework. The mathematical formulation of the metric is represented as:

$$P = \frac{PXSE}{P + SE} \times 100$$

Accuracy (ACC) determines the percentage of tumor classifications done correctly out of all the predicted

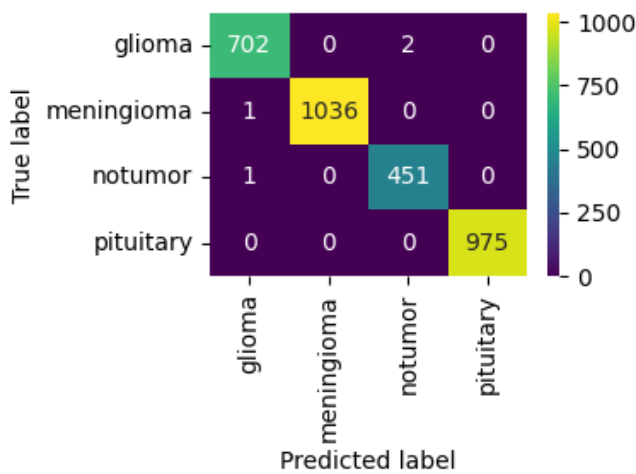
categories. A high accuracy determines the efficiency of the system. It is mathematically represented as

$$Accuracy = \frac{(T_N) + (Tp)}{(T_p) + (T_N) + (F_p) + (F_N)} \times 100$$

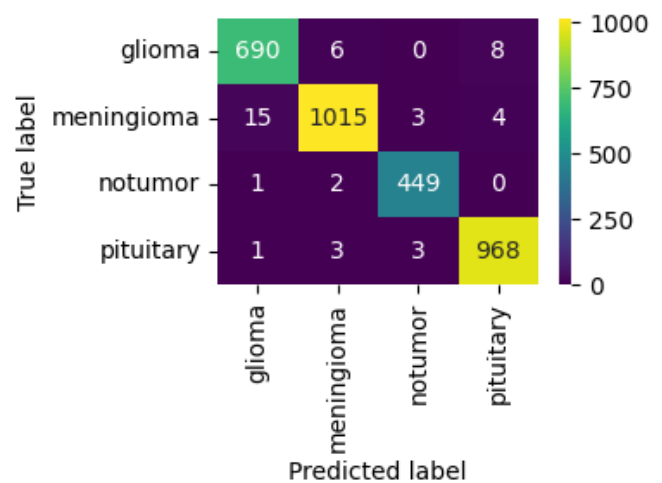
The abbreviations TN, TP, FN, and FP in the equations represent True Negative (TN), True Positive (TP), False Negative (FN), and False Positive (FP), respectively. True positive describes the actual valid values that are predicted correctly as accurate, True negative describes the actual false values that are precisely expected as false, False positive describes the actual false values that are incorrectly classified as true, and false negative represents the actual valid values that were incorrectly predicted as false.

5.3 Analysis of frameworks

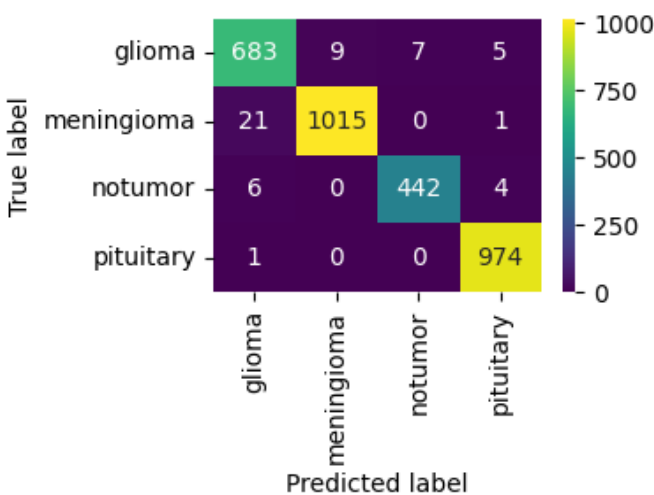
This study evaluates the Xception framework alongside other pre-trained CNN frameworks, including ResNet50, AlexNet, VGG19, and InceptionV3. Comparing the Xception framework to other state-of-the-art frameworks demonstrates the efficacy of our proposed mod-el. Figure 4 illustrates the achieved precision and loss for each framework.



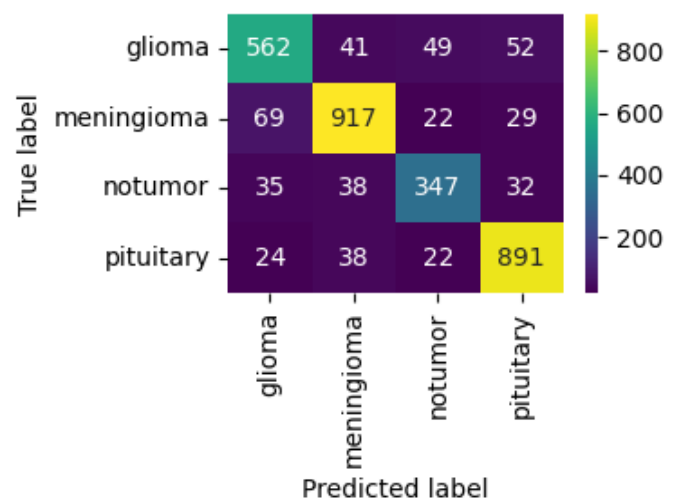
(a) Proposed Xception framework



(b) VGG19



(c) AlexNet



(d) ResNet50

Figure 6. Confusion matrix obtained by the proposed Xception framework, VGG19, AlexNet and ResNet50 frameworks

The confusion metrics that analyze the predictions of a classification framework that compares them to the actual values of the data are calculated for the evaluated CNN frameworks, as represented in Figure 5. The performance metrics for each framework are calculated from the confusion matrix described in Table 1.

Table 1. Performance outcome of the proposed framework

Class	Precision (%)	Recall (%)	F1-score (%)
Glioma	99.9	99.9	99.9
Meningioma	99.8	99.9	99.8
No tumor	99.9	99.9	99.9
Pituitary	100	100	100

Confusion metrics (CM) are vital in analyzing frameworks like Xception, AlexNet, VGG19, and ResNet50 performance, as illustrated in Figure 6. CM represents the predicted values on the x-axis and the y-axis. In addition, it also helps to calculate metrics like precision, accuracy, recall, and F1-score equations as described in equations 1 to 6, as specified in the earlier section.

The performance metrics are calculated from the confusion metrics as specified in Figure 6 of each framework to estimate the effectiveness of the frameworks.

From the above performance metrics of the Xception framework (Figure 7), we can say that it has a good precision value depicting that the framework has hardly made false optimistic predictions. A good recall value of this framework shows that the framework has attained good accuracy by

correctly classifying the positive instances, and a high f1-score depicts that our framework has achieved perfect precision and recall.

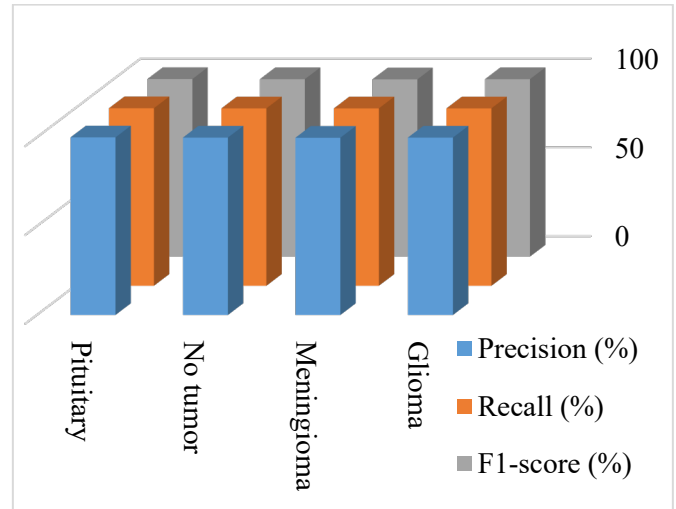


Figure 7. Graphical representation of the Xception framework's performance metrics

The performance outcomes of the other frameworks are also calculated to compare with the proposed Xception framework, as described in Tables 2 and 3. The graphical representation of the F1 score, precision, and recall performance scores are presented in Figures 8 and 9.

Table 2. F1 Score of the proposed framework and other CNN frameworks

Model	F1-Score (%)				Overall Accuracy (%)
	Glioma	Meningioma	No Tumor	Pituitary	
AlexNet	96	98	98	99	98
ResNet50	80.2	88.8	76.09	89.68	85.5
VGG19	97	98	99	99	98
Proposed model	99.9	99.8	99.9	100	99.87

Table 3. Precision and recall (%) of the proposed model and other CNN models

Model	Precision (%)				Recall (%)			
	Glioma	Meningioma	No Tumor	Pituitary	Glioma	Meningioma	No Tumor	Pituitary
AlexNet	96	99	98	99	97	98	98	100
ResNet50	75.4	91.8	75	90.97	85.7	85.99	77.22	88.43
VGG19	97	99	98	99	98	98	99	99
Proposed model	99.9	99.8	99.9	100	99.8	99.8	99.9	100

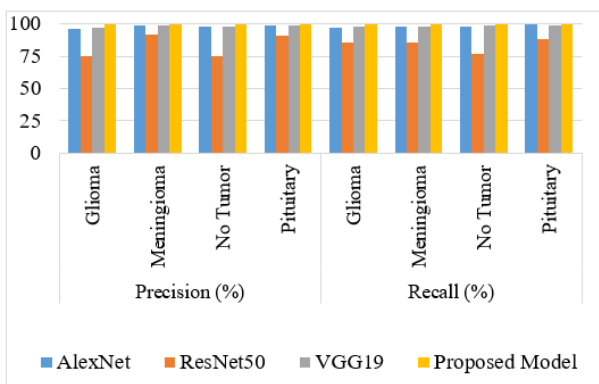


Figure 8. Graphical representation of precision and recall values obtained by the proposed framework and other CNN frameworks

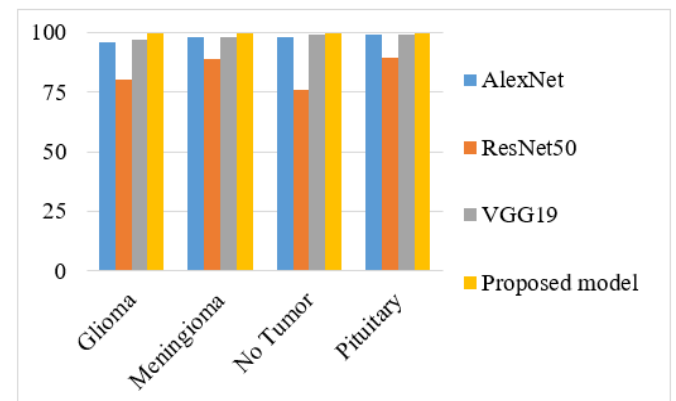


Figure 9. Graphical representation of F1 Score obtained by the proposed framework and other CNN frameworks

From the analysis, it is observed that the Xception framework, which is the proposed framework of our study, has obtained precision and recall of 100% for all the classes. The framework has hardly fallen into false pessimistic or optimistic predictions. The Xception framework gained an f1-score of 100%. The other CNN frameworks were compared with the f1-score of Xception. AlexNet and VGG19 were close to the perfect score, whereas the ResNet50 came after the remaining frameworks.

5.4 Comparison of the proposed framework concerning validation loss and accuracy

The performance of the proposed Xception framework is evaluated against four other pre-trained CNN frameworks—ResNet50, InceptionV3, AlexNet, and VGG19—using the same dataset to demonstrate its superiority in fine-grained classification. The Xception framework is built on the Inception framework with depth-wise separable convolutional layers as an additional performing both spatial and channel-wise convolution resulting in powerful feature extraction. The depth-wise convolutional layers in Xception are computationally more effective than the traditional convolution layers of AlexNet, VGG19, and ResNet50. The Xception framework uses parameters effectively because of the depth-wise separable convolutional layers, whereas other state-of-art frameworks evaluated in this paper require more parameters leading to more training time.

Table 4. Validation accuracy and loss of the proposed framework and other CNN frameworks

CNN Frameworks	Validation Accuracy (%)	Validation Loss (%)
AlexNet	98.26	0.11
ResNet50	85.51	0.42
VGG19	91.13	0.27
InceptionV3	90.53	1.3
Proposed framework	99.87	0.0026

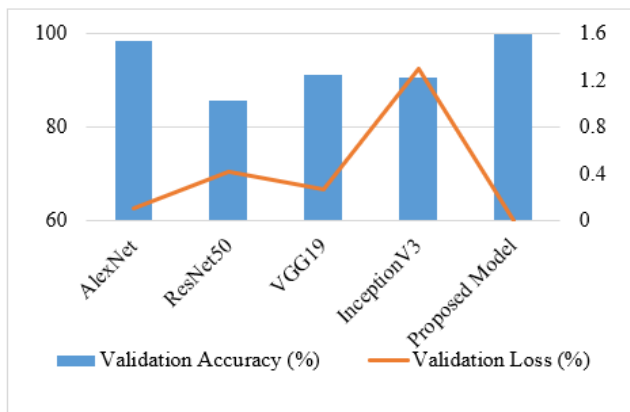


Figure 10. Comparison of the proposed framework with various deep learning architectures with respect to validation accuracy and loss

In this paper, the Xception framework performed fine-grained classification. The proposed framework achieved a validation accuracy of 99.87%, surpassing the performance of other CNN frameworks in terms of accuracy. Table 4 and Figure 10 depict the validation performance of the proposed

and other CNN frameworks. The proposed framework demonstrated exceptional performance, attaining a validation accuracy of 99.87% and a very low validation loss of just 0.0026%. AlexNet, VGG19, and InceptionV3 show their performance in classifying tumors a bit closer to Xception, whereas ResNet50 comes after all these frameworks in ranking.

5.5 Discussion

In this study, we compared our proposed framework with other CNN frameworks whose accuracies have been calculated and compared in Table 4, as presented earlier in this section. The confusion and performance metrics have been computed in Tables 2 and 3. In addition, the graphical illustration is provided in Figures 8 and 9. The main problem in the tumor classification was determining the irregular shapes of the tumor. The feature extraction methods can evaluate the odd shapes. Our proposed Xception framework, a powerful feature extractor with a depth-wise separable convolutional layer, has depth-wise and pointwise convolutional layers, extracting more features than the standard convolutional layers. The Xception framework was hyper-tuned in this study because of its ability to retain spatial information by which the object can be easily identified.

The framework achieved a perfect classification score and a minor validation loss in the fine-grained classification of brain tumors on the MRI dataset obtained from Kaggle. The dataset in this study was pre-processed to extract the region of interest and then augmented to get all the dimensions of the MRI image so that the framework can train on all the various angles of the MRI images. Adam optimizer used in this study dynamically adjusted the learning rate. The optimizer fine-tuned the weights, which helped improve the framework's training process speed. In Figure 10, the validation accuracy and loss of different frameworks are compared. The proposed framework outperforms AlexNet, ResNet50, VGG19, and Inception V3, achieving the highest validation accuracy while maintaining a low validation loss. The framework took less time in training and effectively used the parameters while maintaining high validation accuracy than the other CNN frameworks, such as AlexNet, InceptionV3, VGG19 and ResNet50, which use a more significant number of parameters. Feature selection could have increased the framework's performance for real-world data use. The powerful feature extraction capabilities of the Xception framework proved sufficient to achieve high validation accuracy in accurately classifying brain tumors.

6. CONCLUSION

A brain tumor ranks among the most rapidly progressing and life-threatening conditions. Much research has been done to eradicate the problem of differentiating among various types of brain tumors using ML and DL frameworks. The hyper tuning of parameters of the Xception framework utilized in this study gave an exceptional performance in the fine-grain classification of the brain tumors into four classes: glioma, meningioma, pituitary, and no tumor. Pre-processing and augmenting the MRI images helped the framework train better on various images. The proposed Xception framework demonstrated outstanding performance in comparison to other CNN frameworks, including AlexNet, ResNet50,

InceptionV3, and VGG19. The high validation accuracy and accurate classification of the proposed Xception framework make it fit for predicting brain tumors. The framework can be more promising when tested on various datasets; performing segmentation and localizing the exact position of the tumor will provide better treatment to the subject. The work can be expanded by employing optimization algorithms to select the most relevant features, thereby minimizing computational complexity and enhancing the framework's performance in real-time analysis.

FUNDING

This research was funded by Taif University, Saudi Arabia (Grant No.: TU-DSPP-2024-52).

REFERENCES

- [1] Ostrom, Q.T., Adel Fahmideh, M., Cote, D.J., Muskens, I.S., Schraw, J.M., Scheurer, M.E., Bondy, M.L. (2019). Risk factors for childhood and adult primary brain tumors. *Neuro-oncology*, 21(11): 1357-1375. <https://doi.org/10.1093/neuonc/noz123>
- [2] Zulfiqar, F., Bajwa, U.I., Mehmood, Y. (2023). Multi-class classification of brain tumor types from MR images using EfficientNets. *Biomedical Signal Processing and Control*, 84: 104777. <https://doi.org/10.1016/j.bspc.2023.104777>
- [3] Behura, A. (2021). The cluster analysis and feature selection: Perspective of machine learning and image processing. In *Data Analytics in Bioinformatics: A Machine Learning Perspective*, pp. 249-280. <https://doi.org/10.1002/9781119785620.ch10>
- [4] Sharma, S., Mittal, R., Goyal, N. (2022). An assessment of machine learning and deep learning techniques with applications. *ECS Transactions*, 107(1): 8979. <https://doi.org/10.1149/10701.8979ecst>
- [5] Shaik, N.S., Cherukuri, T.K. (2022). Transfer learning based novel ensemble classifier for COVID-19 detection from chest CT-scans. *Computers in Biology and Medicine*, 141: 105127. <https://doi.org/10.1016/j.combiomed.2021.105127>
- [6] Siar, M., Teshnehlal, M. (2019). Brain tumor detection using deep neural network and machine learning algorithm. In *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, Mashhad, Iran, pp. 363-368. <https://doi.org/10.1109/ICCKE48569.2019.8964846>
- [7] Rehman, A., Naz, S., Razzak, M.I., Akram, F., Imran, M. (2020). A deep learning-based framework for automatic brain tumors classification using transfer learning. *Circuits, Systems, and Signal Processing*, 39(2): 757-775. <https://doi.org/10.1007/s00034-019-01246-3>
- [8] Khan, A.R., Khan, S., Harouni, M., Abbasi, R., Iqbal, S., Mehmood, Z. (2021). Brain tumor segmentation using K - means clustering and deep learning with synthetic data augmentation for classification. *Microscopy Research and Technique*, 84(7): 1389-1399. <https://doi.org/10.1002/jemt.23694>
- [9] Noreen, N., Palaniappan, S., Qayyum, A., Ahmad, I., Imran, M., Shoaib, M. (2020). A deep learning model based on concatenation approach for the diagnosis of brain tumor. *IEEE Access*, 8: 55135-55144. <https://doi.org/10.1109/ACCESS.2020.2978629>
- [10] Saleh, A., Sukaik, R., Abu-Naser, S.S. (2020). Brain tumor classification using deep learning. In *2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech)*, Gaza, Palestine, pp. 131-136. <https://doi.org/10.1109/iCareTech49914.2020.00032>
- [11] Toğaçar, M., Ergen, B., Cömert, Z. (2020). BrainMRNet: Brain tumor detection using magnetic resonance images with a novel convolutional neural network model. *Medical Hypotheses*, 134: 109531. <https://doi.org/10.1016/j.mehy.2019.109531>
- [12] Siddique, M.A.B., Sakib, S., Khan, M.M.R., Tanzeem, A.K., Chowdhury, M., Yasmin, N. (2020). Deep convolutional neural networks model-based brain tumor detection in brain MRI images. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India, pp. 909-914. <https://doi.org/10.1109/I-SMAC49090.2020.9243461>
- [13] Sharif, M.I., Khan, M.A., Alhussein, M., Aurangzeb, K., Raza, M. (2021). A decision support system for multimodal brain tumor classification using deep learning. *Complex & Intelligent Systems*, 8: 3007-3020. <https://doi.org/10.1007/s40747-021-00321-0>
- [14] İncir, R., Bozkurt, F. (2024). Improving brain tumor classification with combined convolutional neural networks and transfer learning. *Knowledge-Based Systems*, 299: 111981. <https://doi.org/10.1016/j.knosys.2024.111981>
- [15] Agarwal, M., Rani, G., Kumar, A., Kumar, P., Manikandan, R., Gandomi, A.H. (2024). Deep learning for enhanced brain tumor detection and classification. *Results in Engineering*, 22: 102117. <https://doi.org/10.1016/j.rineng.2024.102117>
- [16] Rasool, M., Ismail, N.A., Boulila, W., Ammar, A., Samma, H., Yafooz, W.M., Emara, A.H.M. (2022). A hybrid deep learning model for brain tumour classification. *Entropy*, 24(6): 799. <https://doi.org/10.3390/e24060799>
- [17] Raza, A., Ayub, H., Khan, J.A., Ahmad, I., et al. (2022). A hybrid deep learning-based approach for brain tumor classification. *Electronics*, 11(7): 1146. <https://doi.org/10.3390/electronics11071146>
- [18] Senan, E.M., Jadhav, M.E., Rassem, T.H., Aljaloud, A.S., Mohammed, B.A., Al-Mekhlaifi, Z.G. (2022). Early diagnosis of brain tumour MRI images using hybrid techniques between deep and machine learning. *Computational and Mathematical Methods in Medicine*, 2022(1): 8330833. <https://doi.org/10.1155/2022/8330833>
- [19] Arefin, A.S.S.M.N., Ishti, S.M.I.A.K., Akter, M.M., Jahan, N. (2023). Deep learning approach for detecting and localizing brain tumor from magnetic resonance imaging images. *Indonesian Journal of Electrical Engineering and Computer Science*, 29(3): 1729-1737. <https://doi.org/10.11591/ijeecs.v29.i3.pp1729-1737>
- [20] Bashkandi, A.H., Sadoughi, K., Aflaki, F., Alkhazaleh, H.A., Mohammadi, H., Jimenez, G. (2023). Combination of political optimizer, particle swarm optimizer, and convolutional neural network for brain tumor detection. *Biomedical Signal Processing and Control*, 81: 104434. <https://doi.org/10.1016/j.bspc.2022.104434>
- [21] Nanda, A., Barik, R.C., Bakshi, S. (2023). SSO-RBNN

- driven brain tumor classification with Saliency-K-means segmentation technique. *Biomedical Signal Processing and Control*, 81: 104356. <https://doi.org/10.1016/j.bspc.2022.104356>
- [22] Rahman, T., Islam, M.S. (2023). MRI brain tumor detection and classification using parallel deep convolutional neural networks. *Measurement: Sensors*, 26: 100694. <https://doi.org/10.1016/j.measen.2023.100694>
- [23] Satyanarayana, G., Naidu, P.A., Desanamukula, V.S., Rao, B.C. (2023). A mass correlation based deep learning approach using deep convolutional neural network to classify the brain tumor. *Biomedical Signal Processing and Control*, 81: 104395. <https://doi.org/10.1016/j.bspc.2022.104395>
- [24] Kurdi, S.Z., Ali, M.H., Jaber, M.M., Saba, T., Rehman, A., Damaševičius, R. (2023). Brain tumor classification using meta-heuristic optimized convolutional neural networks. *Journal of Personalized Medicine*, 13(2): 181. <https://doi.org/10.3390/jpm13020181>
- [25] Jaspin, K., Selvan, S. (2023). Multiclass convolutional neural network based classification for the diagnosis of brain MRI images. *Biomedical Signal Processing and Control*, 82: 104542. <https://doi.org/10.1016/j.bspc.2022.104542>
- [26] Brain Tumor Classification (MRI). <https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri>, accessed on 1 April 2023.
- [27] Mangs. (2022). How to find extreme outer points in an image with Python, OpenCV. <https://devpress.csdn.net/python/63045d28c67703293080bddf.html>
- [28] Shaheed, K., Mao, A., Qureshi, I., Kumar, M., Hussain, S., Ullah, I., Zhang, X. (2022). DS-CNN: A pre-trained Xception model based on depth-wise separable convolutional neural network for finger vein recognition. *Expert Systems with Applications*, 191: 116288. <https://doi.org/10.1016/j.eswa.2021.116288>
- [29] Wang, C., Chen, D., Hao, L., Liu, X., Zeng, Y., Chen, J., Zhang, G. (2019). Pulmonary image classification based on inception-v3 transfer learning model. *IEEE Access*, 7: 146533-146541. <https://doi.org/10.1109/ACCESS.2019.2946000>
- [30] Kumar, R.L., Kakarla, J., Isunuri, B.V., Singh, M. (2021). Multi-class brain tumor classification using residual network and global average pooling. *Multimedia Tools and Applications*, 80(9): 13429-13438. <https://doi.org/10.1007/s11042-020-10335-4>
- [31] Awan, M.J., Masood, O.A., Mohammed, M.A., Yasin, A., Zain, A.M., Damaševičius, R., Abdulkareem, K.H. (2021). Image-based malware classification using VGG19 network and spatial convolutional attention. *Electronics*, 10(19): 2444. <https://doi.org/10.3390/electronics10192444>
- [32] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems-Volume 1*, Lake Tahoe, Nevada, USA, pp. 1097-1105.