TÜRKİYE CUMHURİYETİ
ADANA ALPARSLAN TÜRKEŞ SCIENCE AND TECHNOLOGY UNIVERSITY

GRADUATE SCHOOL
COMPUTER ENGINEERING DEPARTMENT

A HYBRID TRANSFER LEARNING MODEL FOR BRAIN TUMOR CLASSIFICATION

EZGİSU AKAT

M.Sc.

ADANA 2024

TÜRKİYE CUMHURİYETİ
ADANA ALPARSLAN TÜRKEŞ SCIENCE AND TECHNOLOGY UNIVERSITY

GRADUATE SCHOOL
COMPUTER ENGINEERING DEPARTMENT

A HYBRID TRANSFER LEARNING MODEL FOR BRAIN TUMOR CLASSIFICATION

EZGİSU AKAT

M.Sc.

THESIS ADVISOR
Prof. Dr. Serdar YILDIRIM

SECOND ADVISOR
Assoc. Prof. Dr. Yasin KAYA

ADANA, 2024

DECLARATION OF CONFORMITY

In this thesis study, which was prepared following the thesis writing rules of Adana Alparslan Türkeş Science and Technology University Institute of Graduate School, I declare that I provide all the information, documents, evaluations and results in accordance with scientific ethics and moral codes without resorting to any means or assistance that would be contrary to scientific ethics and traditions. I also declare that I refer to all of the articles I used in this study with appropriate references and accept all moral and legal consequences if a situation is found contrary to my statement regarding my work.

16/01/2024

Ezgisu AKAT

ÖZET

BEYİN TÜMÖRÜ SINIFLANDIRMASI İÇİN HİBRİT TRANSFER ÖĞRENME MODELİ

Ezgisu AKAT
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Serdar YILDIRIM
II. Danışman: Doç. Dr. Yasin KAYA
Ocak 2024, 49 sayfa

Beyin tümörleri dünya çapında en yaygın ve ölümcül hastalıklardan biridir. Hastanın yaşam beklentisini uzatmak için beyin tümörlerini erken aşamada tespit etmek çok önemlidir. Ancak, beyin tümörlerini manuel olarak sınıflandırmak zor ve zaman alıcı bir iştir. Bu çalışmada beyin tümörlerini otomatik olarak sınıflandırmak için hibrit transfer öğrenme modeli önerilmektedir.Önerilen yaklaşımın dört adımı ön işleme ve veri artırma, derin özellik çıkarımlarının birleşimi, ince ayar ve sınıflandırmadır. Önerilen model, halka açık dört genel veri setinde doğrulanmıştır: Br35H, Nickparvar, Figshare ve Sartaj. Önerilen model tüm veri setlerinde en yüksek doğruluk değerlerine ulaştı: Br35H'de %99.66, Figshare'de %97.56, Nickparvar'da %97.08, ve Sartaj'da %93.74. Önerilen model, beyin tümörlerinin sınıflandırılmasında diğer modern modellere göre daha etkili bir performansa sahiptir.

Anahtar Kelimeler: Beyin Tümörü Sınıflandırması, Birleştirilmiş CNN, Öğrenme Aktarımı

ABSTRACT

A HYBRID TRANSFER LEARNING MODEL FOR BRAIN TUMOR CLASSIFICATION

Ezgisu AKAT
Department of Computer Engineering
Supervisor: Prof. Dr. Serdar YILDIRIM
Co-Supervisor: Assoc. Dr. Yasin KAYA
January 2024, 49 pages

Brain tumors are one of the most common and fatal diseases globally. It is crucial to detect brain tumors at an early stage to increase the patient's life expectancy. Nonetheless, classifying brain tumors manually is a difficult and time-consuming task. A hybrid transfer learning model is suggested in this study to classify brain tumors automatically. The four steps of the suggested approach are preprocessing and data augmentation, fusion of deep feature extractions, fine-tuning, and classification. VGG16, ResNet50, and MobileNetV2 CNN pre-trained models are fused to increase the number of informative features and reduce overfitting. The suggested model is validated on four public available datasets: Br35H, Nickparvar, Figshare, and Sartaj. The proposed model achieved the highest accuracy values in all datasets: 99.66% on Br35H, 97.56% on Figshare, 97.08% on Nickparvar, and 93.74% on Sartaj. The suggested model is more effective than other cutting-edge models in classifying brain tumors.

Keywords: Brain Tumor Classification, Fusion of CNN, Transfer Learning

I sincerely dedicate this thesis to my family who supported me everytime …

ACKNOWLEDGEMENTS

Thank you to my supervisors, Prof. Dr. Serdar YILDIRIM and Assoc. Prof. Dr. Yasin KAYA, for providing guidance and feedback throughout this thesis. I also thank my friends and family for their support during my thesis studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADAM	: Adaptive Moment Estimation
AI	: Artificial Intelligence
AUC	: Area Under Curve
BF	: Bilateral Filter
CBMIR	: Content-Based Medical Image Retrieval
CFML	: Closed-Form Metric Learning
CNN	: Convolutional Neural Networks
CT	: Computer Tomography
DL	: Deep Learning
EKbHFV	: Entropy–Kurtosis-Based High Feature Value
ELM	: Extreme Learning Machine
FLOPs	: Floating Point Operations
FN	: False Negative
FP	: False Positive
GA	: Genetic Algorithm
GAN	: Generative Adversarial Network
GAP	: Global Average Pooling
GLCM	: Gray-Level Co-Occurrence Matrix
ILSVRC	: ImageNet Large Scale Visual Recognition Challenge
IRV2	: InceptionResNetV2
KNN	: K-Nearest Neighbors
LGG	: Low-Grade Glioma
LOPO	: Leave-One-Patient-Out
LSTM	: Long Short-Term Memory
MAG	: Modified Genetic Algorithm
mAp	: Mean Average Precision
ML	: Machine Learning
MLP	: Multilayer Perceptron
MRI	: Magnetic Resonance Imaging
NSGA	: Non-Dominated Sorted Genetic Algorithm
PCA	: Principal Component Analysis

PET : Positron Emission Tomography
PFpM : Parametric Flatten-P Mish
RBM : Restricted Boltzmann Machine
ReLU : Rectified Linear Unit
ResNet : Residual Networks
RF : Random Forest
RFE : Recursive Feature Elimination
RMSprop : Root Mean Square Propagation
RNN : Recurrent Neural Networks
ROC : Receiver Operating Characteristic Curve
SGD : Stochastic Gradient Descent
SNN : Siamese Neural Network
SVM : Support Vector Machine
TL : Transfer Learning
TN : True Negative
TP : True Positive

1. INTRODUCTION

A tumor is a mass of abnormal cells that grow uncontrollably due to unregulated division (Vankdothu, Hameed, & Fatima, 2022). Normally, cells in good health divide in an organized and systematic way, creating new cells. Tumors are classified into two main categories depending on their characteristics: benign and malignant (Toğaçar, Cömert, & Ergen, 2020). Benign tumors, known as non-cancerous growths in their original location, do not expand to other parts of the body. Patients with benign tumors typically do not face life-threatening consequences due to the tumor. On the other hand, malignant tumors are forms of cancer that are characterized by cells that grow uncontrollably and can spread to nearby or distant areas, causing serious health complications (Kumar, Kakarla, Isunuri, & Singh, 2021; Toğaçar, Cömert, & Ergen, 2020).

The brain is a complex organ made up of nerve cells and tissues that control vital bodily functions such as breathing, muscle movement, and sensory perception (S. Lu, Wang, & Zhang, 2020; Rehman, Naz, Razzak, Akram, & Imran, 2019). If tumors develop in the brain, they can be fatal due to its crucial role in the body (Togacar, Ergen, & Comert, 2021). A brain tumor can cause significant symptoms such as difficulty concentrating, memory problems, frequent headaches, seizures, changes in speech, and coordination issues (Alhassan & Zainon, 2021). Of all brain tumors, 70% are primary tumors, with secondary tumors making up the remaining 30%. One of the main characteristics of primary brain tumors is that cells within the brain's tissue grow and divide uncontrollably. Secondary brain tumors are formed when cancer cells from other parts of the body spread to the brain utilizing the bloodstream. This can cause the growth of tumors in the brain (Sharma et al., 2023).

There are various types of brain tumors based on their shape, location, and texture. There are three main types of brain tumors: Pituitary, Gliomas, and Meningioma. In clinical practice, the incidence percentages of brain tumors that are Glioma, Meningioma, and Pituitary tumors are about 45%, 15%, and 15%, respectively (Abdelaziz Ismael, Mohammed, & Hefny, 2020). The most prevalent kind of primary brain tumors that occur from glial cells, particularly astrocytes, are called gliomas, and they are located in both cerebral hemispheres. Meningiomas are usually benign growths that originate from the meninges. The pituitary gland, located at the base of the brain, can develop tumors called pituitary tumors (Abdelaziz Ismael, Mohammed, & Hefny, 2020; Kesav & Jibukumar, 2022; Sharma et al., 2023).

Based on the morphology of the cells, astrocytomas are categorized into four grades by the World Health Organization. Grade I tumors are benign, resemble normal brain cells, and grow slowly. Grade II tumors are malignant and have abnormal cells compared to Grade I. Grade III tumors have anaplastic cells that differ significantly from normal cells and are actively growing. Grade IV represents the most malignant tissue, characterized by highly abnormal cells that grow rapidly (Brunese, Mercaldo, Reginelli, & Santone, 2020). Additionally, Brain tumors are categorized into two stages: low-grade glioma (LGG) and high-grade glioma (HGG), due to their high mortality rate (Amin et al., 2020). The survival rate of HGG is around two years, whereas LGG has a faster survival rate and requires prompt treatment. It is critical to identify the grade of the tumor to determine the best course of medical care (Brunese et al., 2020).

Each year, over 500,000 people die from brain tumors in the UK. However, at least 102,000 children and young people are estimated to survive. Around 23,000 people were detected with brain tumors in the USA in 2015. Two years later, statistics show that brain cancer is the leading cause of death in children and adults worldwide (Ayadi, Elhamzi, Charfi, & Atri, 2021). Based on 2017 cancer statistics, brain tumors constitute a significant percentage of cancer-related disease, mortality, and morbidity worldwide in both adults and children (Rehman et al., 2019). In 2021, 84,170 people were diagnosed with brain tumors, with 69,950 of them being adults over 40 years of age (Muhammad Imran Sharif, Khan, Alhussein, Aurangzeb, & Raza, 2021). Therefore, it’s vital for a patient’s well-being to receive an early and accurate diagnosis.

Different techniques are used in clinics for treating brain tumors. During the benign stage, radiotherapy can eliminate the need for surgery, but in the cancerous stage, chemotherapy and radiotherapy can be effective treatments (Muhammad Imran Sharif et al., 2021). Medical professionals use several techniques to diagnose tumor type and stage for precise treatment. The medical field widely uses various techniques such as biopsy, PET, CT, and MRI. MRI is a commonly used imaging technique with high-resolution and non-invasive properties (Ayadi et al., 2021; Kesav & Jibukumar, 2022; R. Singh, Goel, & Raghuvanshi, 2020; Toğaçar et al., 2020; Vankdothu et al., 2022). Manual tumor diagnosis and classification is a time-consuming process, requiring expertise from doctors and radiologists. In addition, due to the increase in patient numbers, analyzing a large amount of data depending on visual analysis is costly and erroneous. This situation can lead to a severe problem, decreasing the patient's chance of survival. Automated image-processing systems are being developed to overcome the limitations of manual diagnosis (Ayadi et al., 2021). Thanks to advances in computer technology, the diagnosis and classification of brain tumors can be made automatically (Abdelaziz Ismael et al., 2020; Cinar & Yildirim, 2020; Sharma et al., 2023). A Content-Based Medical Image Retrieval (CBMIR) system retrieves the most similar medical images from a database to aid radiologists in disease diagnosis (Deepak & Ameer, 2020). Lately, a developed computer-assisted diagnosis (CAD) system utilizes artificial intelligence to classify and diagnose tumors accurately (Abdelaziz Ismael et al., 2020; Bodapati, Shaik, Naralasetti, & Mundukur, 2020). A typical CAD system consists of three stages: lesion segmentation, extraction of tumor characteristics, and the application of ML algorithms (Abdelaziz Ismael et al., 2020). The feature extraction stage is crucial in the development of an effective CAD system. Extracting good features from the problem domain is crucial for achieving high classification accuracy. The conventional techniques for feature extraction can be categorized into three groups: Wavelet and frequency features, Spatial domain features, and Contextual and Hybrid features. The use of deep learning in new CAD methods improves performance (Ayadi et al., 2021). The CAD system produces findings faster and without human intervention (Sharma et al., 2023).

Machine learning (ML) and Deep learning (DL) methods are sub-sets of artificial intelligence (AI) (Cinar & Yildirim, 2020; Sharma et al., 2023). There are two main types of machine learning: supervised and unsupervised learning. The supervised techniques require knowledge of classes and the use of prior models from training sets. In contrast, unsupervised techniques do not require any specific training data and are generally faster but may be less efficient due to large dataset sizes (Alhassan & Zainon, 2021). ML is a statistical method that identifies patterns and trends to perform actions without explicit instructions (Sharma et al., 2023). Several ML algorithms, including Random Forest (RF), Nearest Neighbor, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes, and C 4.5, have been utilized in prior research (Aslan, Dogan, & Koca, 2023; Brunese et al., 2020). DL is a sub-field of ML that removes the requirement for manually-crafted features. DL has significantly reduced the disparity between computer vision and human for pattern recognition, and it can outperform traditional techniques (Ayadi et al., 2021). However, DL algorithms have been shown to be effective in producing accurate results in large databases (Cinar & Yildirim, 2020).

Various DL algorithms such as convolutional neural networks (CNN), recurrent neural networks (RNN), transfer learning, and fine-tuning have been used in numerous studies for brain tumor diagnosis and classification. CNN and Long Short-Term Memory (LSTM) are types of DL (Vankdothu et al., 2022). CNN architecture comprises various layers, with its fundamental layers being convolutional, pooling, and fully connected. During feature extraction, input patterns undergo small filter convolution to identify the most distinctive feature (Abdelaziz Ismael et al., 2020). CNNs are commonly utilized in various applications such as video, image, speech processing, and natural language processing. When delving deeper, experts must extract more complex characteristics to obtain fundamental information. LSTM networks, a type of RNN, can learn order relationships in challenging forecasting sequence problems. LSTM is often used in sophisticated domains like speech recognition and machine translation to predict behavior (Abdelaziz Ismael et al., 2020; Sharma et al., 2023; Vankdothu et al., 2022).

Researchers have carried out numerous studies using DL and ML algorithms to classify brain tumors. However, they have faced several limitations. Ismael et al. employed a small dataset with three classes to evaluate their model (Abdelaziz Ismael et al., 2020). Similarly, Lu et al. noted that the dataset used in his study had two classes and little data. It is crucial to assess the model’s effectiveness and carry out research on more enormous datasets that contain more of a variety of tumor types (S. Lu et al., 2020). Brunese et al. specified that the least amount of variance from the learned behavior could lead to the incorrect classification of instances that are not in the training dataset, even though machine learning algorithms can address classification tasks without earlier domain knowledge (Brunese et al., 2020). Bodapati et al. that there are certain drawbacks to utilizing two pre-trained models for feature extraction (Bodapati et al., 2020). The performance of the model may be enhanced by using more pre-trained models to increase deep features.

Various studies have demonstrated that the fusion method lowers the classification errors and enhances the model’s performance. Kaya and Gürsoy and Adrian et al. used fusion images in their work (Adrian, Sagan, & Maimaitijiang, 2021; Kaya & Gürsoy, 2023b). Whereas Adrian et al. fused SAR and optical images to map crop types (Adrian et al., 2021), Kaya and Gürsoy fused RGB and segmented images to detect plant diseases (Kaya & Gürsoy, 2023b). He et al. used the fusion CNN approach using a pre-trained VGG16 model to identify smoke in foggy weather (L. He, Gong, Zhang, Wang, & Li, 2021). Inspired by these studies, we introduced a novel hybrid transfer learning model by fusing pre-trained VGG16, ResNet50, and MobileNetV2 models in order to detect brain tumors in this study. We used pre-trained models for the fusion process which can produce successful results with shorter training times and small datasets. Our study has made several important contributions as follows:

- A new deep transfer learning model was recommended to classify brain tumor images by concatenating features from VGG16, MobileNetV2, and ResNet50.
- The model's performance was enhanced through the integration of a fine-tuning technique.
- Experiments were carried out on four publicly available datasets, and performance analysis was compared.
- The model achieved an accuracy of 99.66%, 97.56%, 97.08%, and 93.74% on the Br35H, Figshare, Nickparvar, and Sartaj datasets, respectively.

2. LITERATURE REVIEW

Recently, DL and ML algorithms have been utilized extensively to diagnose brain tumors and identify their type automatically. Polat and Gungen utilized transfer learning algorithms with DenseNet21, ResNet50, VGG16, and VGG19 models to classify a Figshare dataset accurately. This study employed various optimization algorithms, such as Adadelta, adaptive moment estimation (ADAM), root mean square propagation (RMSprop), and stochastic gradient descent (SGD), to train models and then compare their results. ResNet50 with the Adadelta optimization algorithm achieved the highest accuracy of 99.02% among the proposed models (Polat & Güngen, 2021). Ismael et al. utilized data augmentation techniques such as rotation, horizontal and vertical flips, shifting, zooming, etc., after splitting the Figshare dataset into different ratios. With a learning rate of 0.00001 and the Adam optimizer, ResNet50 was trained for 500 epochs. The process of classification yielded a 99% accuracy rate (Abdelaziz Ismael et al., 2020). Sharma et al. aimed to classify a dataset obtained from Kaggle using a model that combined transfer learning with fine-tuning. Geometric alterations and tone space methods were applied to MR images. The ResNet50 model had its last layer replaced with four new layers. The modified model was then trained using the Adam optimizer algorithm, resulting in accuracy values of 92% and 100% (Sharma et al., 2023).

Çinar and Yildirim suggested a modified ResNet50 architecture by replacing the last five layers of it with ten new layers. The new layers included BatchNormalization, Relu, Dropout, Maxpooling, Fullyconnected, Softmax, and Classification layers. Moreover, the classification was performed using five different models: Densenet201, GoogleNet, Alexnet, ResNet50, and InceptionV3. A dataset of 253 tumor images was used in a classification study, where 98 were tumor-free, and 155 had tumors. The hybrid model achieved an accuracy rate of 97.01% (Cinar & Yildirim, 2020). Kumar et al. conducted a study with the objective of classifying a dataset of brain tumors into multiple categories. The ResNet50 architecture is used with global average pooling and softmax layers replacing the final layer to solve the issue of vanishing gradient. The accuracy achieved on the Figshare dataset is 97.08% with data augmentation and 97.48% without (Kumar et al., 2021).

Bodapati (2020) et al. recommended a composite two-channel DNN model that trains jointly on separate sets of features extracted from Xception and InceptionResNetV2 (IRV2). In this article, two datasets, Figshare and BraTS 2018, were used for classifying tumors. The accuracy values for Xception 96.30% and InceptionResNetV2 95.43% were obtained using a single channel on the Figshare dataset. The suggested model achieved 98.04% and 93.69% accuracy on the Figshare and BRATS 2018 datasets, respectively (Bodapati et al., 2020). Deepak and Ameer used dimension reduction techniques on the GoogleNet model. Sequentially, Autoencoder, Closed-Form Metric Learning (CFML), Principal Component Analysis (PCA), and Siamese Neural Network (SNN) were applied to features obtained from GoogleNet for feature dimension reduction. The best result was obtained using SNN. The recommended model yielded a mean average precision (mAp) value of 97.64 on the Figshare dataset (Deepak & Ameer, 2020).

Rehman et al. utilized transfer learning methods, such as fine-tuning and freezing, to classify the Figshare dataset. The high-resolution contrast images were generated using the contrast stretching algorithm, and data augmentation techniques including rotation and flipping were applied. For feature extraction, three pre-trained CNN architectures (VGGNet, GoogLeNet, and AlexNet) were utilized, followed by linear classification using SVM. The best result was achieved with a 98.69% accuracy value with fine-tuned VGG16 architecture (Rehman et al., 2019). Lu et al. put forward pre-trained AlexNet, modified and fine-tuned to identify abnormal brains in MRI. Batch normalization was applied to AlexNet, and an Extreme Learning Machine (ELM) was used to replace the last layers. ELM has the advantage of training faster than traditional backpropagation neural networks because it does not require iteration. Moreover, chaotic bat algorithm was employed to optimize the extreme learning machine for better classification performance. The suggested model achieved an accuracy of 96.43% (S. Lu et al., 2020).

Sharif et al. proposed pre-trained CNN Densenet201, which was fine-tuned to classify brain tumor MR images using BRATS2018 and BRATS2019 datasets. Two methods, which were Entropy–Kurtosis-based High Feature Value (EKbHFV) and a modified genetic algorithm (MGA), were employed for feature selections. Later, using a non-redundant serial-based approach, EKbHFV, and MGA-based features are fused and classified with a multiclass SVM cubic classifier. The proposed model attained an accuracy of more than 95% (Muhammad Imran Sharif et al., 2021). Togağcar et al. aimed to make a binary classification on the dataset created by Chakrabarty, consisting of 155 tumors and 98 tumor-free MR images. Firstly, data augmentation was applied to the images. A CNN model consisting of AlexNet and VGG16 architectures and hypercolumn technique is suggested. AlexNet and VGG16 architectures were used for feature extraction. Thanks to the hypercolumn technique, the original image became more clear, and the number of features increased. Later, the Recursive Feature Elimination (RFE) method is utilized for feature reduction. Finally, the SVM method was employed to classify. The suggested model achieved an accuracy of 96.77% (Togacar et al., 2021).

Swati et al. attempted to classify the Figshare dataset using the pre-trained VGG19 model and a block-wise fine-tuning strategy. They could compare the outcomes by gradually increasing the quantity of training datasets. Besides, they could compare the classification performance of the VGG16, AlexNet, and VGG19 models. It was observed that the VGG19 pre-trained model gave the best result with 94.82% accuracy (Swati et al., 2019). Mondal and Shrivastava recommended the BMRI-Net model, which composed of three phases. The first phase contains one convolutional layer, batch-normalization layer, and activation layer. The second phase consists of seven convolutional blocks; the last phase is a max-pool layer. Since the ReLU activation function has a few drawbacks, such as bias shift effect and neuron death, the parametric flatten-p mish (PFpM) activation function is utilized. Additionally, the model has more flexibility to learn complex patterns from the data more accurately, thanks to the parametric approach of PFpM. The model was tested on Br35H and Figshare datasets. Data augmentation technique and contrast enhancement process were applied to the images. The proposed model was compared by using various activation functions such as ReLU, Leaky ReLU, swish, and GELU and with several pre-trained models, including DenseNet101, InceptionV3, MobileNetV2, ResNet50, and VGG19. The proposed model achieved the highest accuracy of 99.57% on Figshare and 99.00% on Br35H (Mondal & Shrivastava, 2022).

Amran et al. classified the Br35H dataset using the GoogleNet model in their study. The last 5 layers of the GoogleNet model were removed, and 14 layers were added. The data augmentation technique was used. Performance comparison was made with different pre-trained models. The accuracy of the recommended model was 99.51% (Amran et al., 2022). Aurna et al. introduced an ensemble of deep CNN models depending on two-phase feature levels. As creating the model, it was created from 5 pre-trained models, including VGG19, InceptionV3, Xception, ResNet50, and EfficientNetB0, and one proposed CNN model. The feature-level ensemble process was implemented in two phases. In the first phase, the best ensemble models were tried to be found by combining suggested CNN, ResNet50, and EfficientNetB0 among six models. In the second phase, from the first phase, EfficientNetB0 + ResNet50 and ResNet50 + suggested CNN were selected as models. The dense layer was added before concatenating these two models. After the concatenate process, dropout was used with a ratio of 0.5. Additionally, PCA was used to obtain feature reduction and informative features. Four datasets, the Figshare, the Sartaj, a 4-class dataset, and the merged dataset, were used to test the proposed model. The data augmentation technique was applied to the images. The datasets' accuracy percentages are as follows: Sartaj- 98.16%, Figshare- 99.67%, 4-class- 99.76%, and merged- 98.96% (Aurna, Yousuf, Taher, Azad, & Moni, 2022).

Vankdothu et al. proposed a method that combines CNN and LSTM models to train a dataset with four classes on Kaggle: pituitary tumor, meningioma tumor, glioma tumor, and no tumor (benign). The CNN-LSTM model achieved the highest accuracy value of 92%, while the CNN and RNN models achieved 89.39% and 90.02%, respectively (Vankdothu et al., 2022). Alhassan and Zainon introduced using a hard swish-based Rectified Linear Unit (ReLU) activation function in CNN for classification. The normalization technique was employed on T-weighted CE-MRI images. Next, MR images were processed with a Histogram of Oriented Gradients (HOG) for feature extraction. The hard swish-based ReLU activation function effectively addresses the vanishing gradient problem and prevents overfitting by carefully selecting the number of hidden neurons. An accuracy of 98.6% was achieved with the suggested CNN model using the hard swish-based ReLU activation function (Alhassan & Zainon, 2021). Singh et al. proposed classifying the BraTS 2017 dataset, consisting of 210 high-grade Glioma (HGG) and 75 low-grade Glioma (LGG) MR images. In this study, a classifier model based on Gabor-modulated convolutional filters was presented. Gabor filters help in reducing the number of features and hence require fewer parameters for learning in models. The cross-validation technique called leave-one-patient-out (LOPO) was employed so as to ensure a larger training dataset. An accuracy of around 98.68% was achieved by the Gabor-modulated CNN model (R. Singh et al., 2020).

Anaraki et al. carried out a classification study on two datasets. One of them is the Figshare dataset, and the other is the dataset consisting of glioma grades. The datasets were applied normalization and data augmentation techniques. They created a model using CNN and genetic algorithm (GA) and used the bagging method to minimize the error rate. Different models were formed for two datasets. Whereas a model consisting of five convolutional layers, max-pooling, and one fully connected layer was formed for glioma grades classification, to classify the Figshare dataset, a model consisting of six convolutional layers, max-pooling, and one fully connected layer was formed. The dropout method was used in both models. The glioma grades dataset yielded an accuracy of 90.9%, while the Figshare dataset yielded an accuracy of 94.2% (Anaraki, Ayati, & Kazemi, 2019).

Sharif et al. performed classification and detection processes on brain tumor MR images from BRATS 2018, 2019, and 2020 data sets. The suggested model consists of several distinct stages. The first was reducing noise in images using a homomorphic wavelet filter. Secondly, feature extraction was done using the pre-trained Inceptionv3 model, and optimal features were obtained using a non-dominated sorted genetic algorithm (NSGA). Thirdly, after classification, the YOLOv2-inceptionv3 model was utilized for localization. Finally, McCulloch's Kapur entropy method was used for lesion segmentation. The suggested model achieved prediction values greater than 0.90 (Muhammad Irfan Sharif, Li, Amin, & Sharif, 2021). Aslan et al. used Generative Adversarial Network (GAN) and Restricted Boltzmann Machine (RBM) methods to classify brain diseases as ischemia, normal, white matter density, and atrophy. After the features obtained from the RBM method were trained with the GAN method, they were classified using Tree, Linear discriminant, Naive Bayes, K-mean, SVM, KNN, Ensemble, and Neural Network classifiers. The highest accuracy is in the SVM algorithm, with 99.65%, and the lowest accuracy is 83.74% with the Naive Bayes algorithm using the RBM method (Aslan et al., 2023).

Brunese et al. aimed to detect I, II, III, and IV-grade brain cancer on MR images obtained from three different data sets. In this study, an ensemble learner, which consists of some machine learning algorithms, was considered. After applying ten different machine learning algorithms one by one, including Nearest Neighbor, Linear SVM, RBF SVM, Gaussian Process, C 4.5, Naive Bayes, RF, Neural Network, QDA, and Logistic, the results were obtained. According to the calibration plot, the best results were obtained from C 4.5, RF, Nearest Neighbors, and Neural Network algorithms. An ensemble learner was constructed using these four algorithms. The proposed ensemble learner model achieved a 99% accuracy rate (Brunese et al., 2020).

Arumugam et al. made an effort to detect a brain tumor. They suggested the Multilayer Perceptron (MLP) model. The crossover smell agent optimization algorithm was used to find the optimal set of weights and biases that minimize the MLP classification error. A two-class dataset was created from three datasets, among which was the Nickparvar. A bilateral filter (BF) was employed to remove noise in the images, and a gray-level co-occurrence matrix (GLCM) was employed as a feature extractor. According to the evaluation results, the CSA-MLP model had the highest accuracy value of 98.56% compared to CNN, RF, transfer learning, and YOLOv3 (Arumugam, Thiyagarajan, Adhi, & Alagar, 2024). Remzan et al. suggested an ensemble model for classifying the Nickparvar dataset. Images were smoothed using the BF, which preserved the edges. To improve image quality, gamma correction was also used. While creating the model, seven pre-trained CNNs, such as VGG-19, ResNet-50, EfficientNetV2B1, DenseNet-121, Inception V3, EfficientNetB1, and MobileNet and ML algorithms, including RF, MLP, SVM, AdaBoost, and k-NN were employed. As a result of comparing the use of pre-trained models with ML algorithms, the best results were obtained by using EfficientNetV2B1, VGG19, and ResNet50 pre-trained models with MLP. Therefore, a new model was established by ensembling EfficientNetB1, ResNet50, and VGG19 pre-trained models and MLP. The recommended model had 96.67% accuracy (Remzan, Hachimi, Tahiry, & Farchi, 2023).

Prior studies have demonstrated that DL algorithms, ML algorithms, and various approaches, such as transfer learning and fine-tuning, are commonly employed to classify and detect brain tumors. Various techniques, such as data augmentation, feature descriptors, and various filters, were utilized to process images, and feature reduction methods were applied to enhance the accuracy of the results. As a result, the studies conducted yielded positive outcomes.

3. MATERIALS AND METHODS

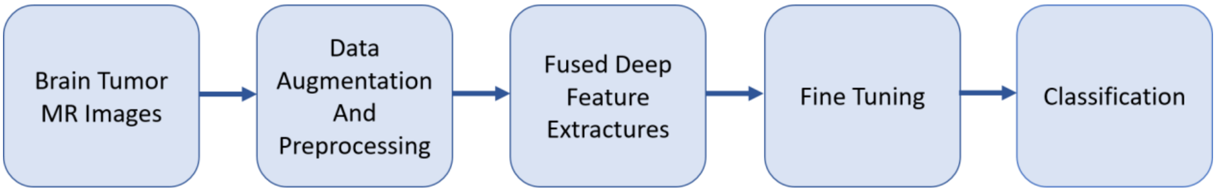

Figure 3.1. The block diagram of the suggested approach

Our approach consists of 4 stages, as depicted in the diagram Figure 3.1. data augmentation and preprocessing, concatenated feature extractions, fine-tuning, and classification.

3.1. Datasets

In this study, we used four different datasets that are open access on the site. The first dataset, the Br35H dataset set up by Hamada (Hamada, 2021), contains 3000 MR images, 1500 with tumors and 1500 without tumors. The second dataset formed by Sartaj Bhuvaji et al. contains a total of 3264 brain tumor MRI images, including 926 glioma, 937 meningioma, 901 pituitary, and 500 images with no tumor (Sartaj, 2019). The third dataset is the Figshare dataset, which was published by Deniz Kavi (Kavi, 2021). There are 3064 MR images with 708 meningioma, 1426 glioma, and 930 pituitary tumor. These three datasets are merged to yield the fourth: the Br35H, Sartaj, and Figshare datasets. This dataset was prepared by Masoud Nickparvar and includes 7023 human brain MR images classified into 926 glioma, 937 meningioma, 500 no tumor and 901 pituitary images (Nickparvar, 2021). In our study, we assigned names to four datasets: Br35H, Sartaj, Figshare, and Nickparvar. Figure 3.2. displays a collection of sample images taken from the Nickparvar dataset.

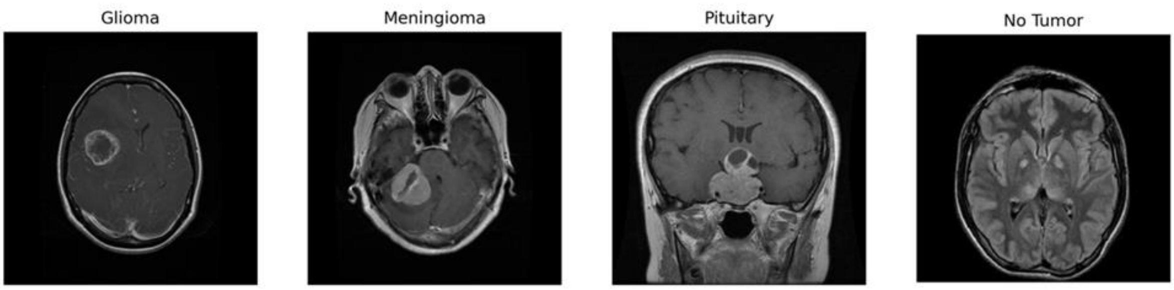

Figure 3.2. Sample images in the Nickparvar dataset

3.2. Pre-processing and Data Augmentation

We splited all the datasets into three parts: training, validation, and testing. Firstly, we used a seed value of 0 to randomly split datasets into 80% training and 20% testing datasets. Later, we separated 20% of the training dataset for the validation dataset using seed value 0. We resized all MR images to 224x224x3 and normalized the pixel intensity values for enhanced image clarity.

Table 3.1. Parameters used for data augmentation process

No	**Parameter**	**Value**
1	brightness_range	[0.05, 1.5]
2	zoom_range	0.15
3	width_shift_range	0.2
4	height_shift_range	0.2
5	fill_mode	nearest
6	rescale	1./255
7	horizontal_flip	True
8	vertical_flip	True

Data augmentation is one of several regularization techniques used to improve the performance of models (Mumuni & Mumuni, 2022). It involves a set of techniques used to increase the size and modify the shape of an image while preserving the label. It addresses two key issues for researchers: generating more data from limited samples and minimizing overfitting (Maharana, Mondal, & Nemade, 2022). Therefore, we trained it using augmented data so as to demonstrate the generalization property of the proposed model. All the parameters used for data augmentation are listed in Table 3.1.

3.3. Convolutional Neural Network

David Hubel and Torsten Wiesel conducted experiments in 1959 and later published their results in a work titled “Receptive Fields of Single Neurons in the Cat’s Striate Cortex”. They determined the organization of neurons in a cat’s striate cortex, which are layered in a tiered pattern. These layers can learn to identify visual patterns using local features that are extracted first. For more complex visual patterns, the extracted features are combined to create a higher-level representation. As a result, the extracted features are put together at this level of representation, which has become a fundamental principle of deep learning. In 1980, Kunihiko was inspired by T. Wiesel's proposal of a "Neocognitron". This work suggests a self-organizing neural network with multiple layers for hierarchical detection of visual patterns learned from data. This model is recognized as the first theoretical model of CNN (Salehi et al., 2023).

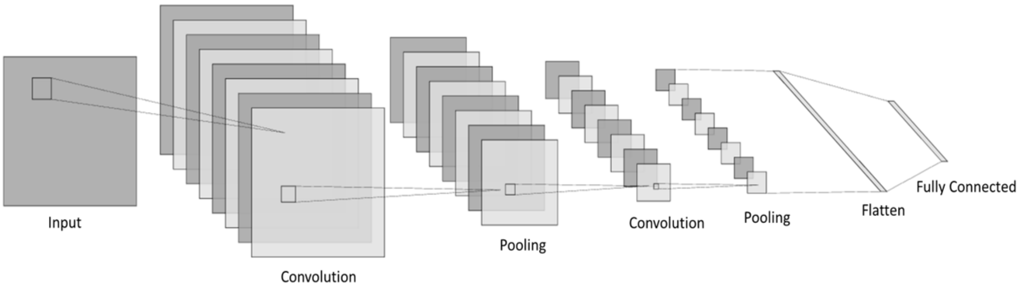

Figure 3.3. Basic CNN Layers

LeCun et al. were the first to introduce the concept of current convolution (Polat & Güngen, 2021). Traditional CNN architecture mainly comprises three layers: convolutional layer, pooling layer, and fully connected layer, as depicted in Figure 3.3. (Abdelaziz Ismael et al., 2020).

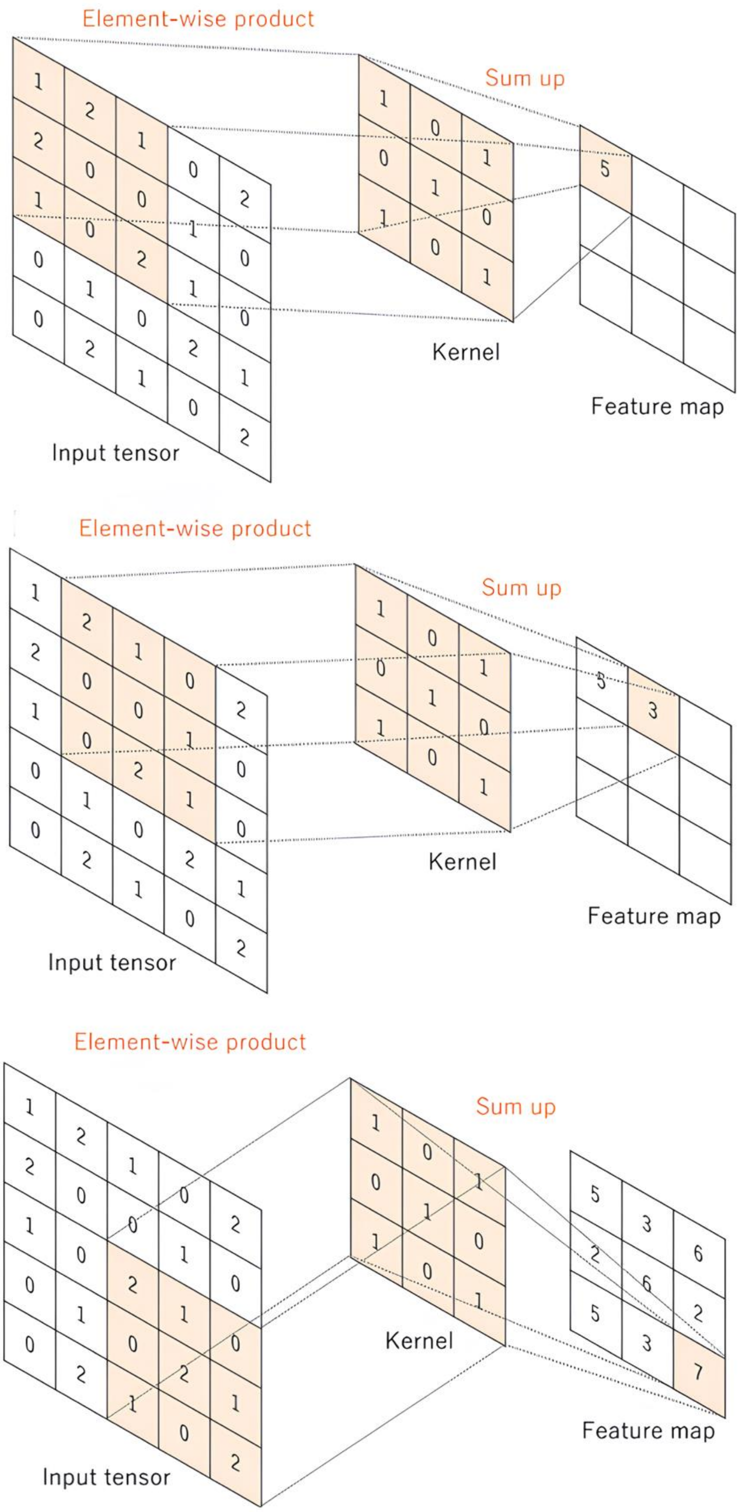

Figure 3.4. Convolution Operation with a Kernel Size (Yamashita, Nishio, Do, & Togashi, 2018)

The convolutional layer is an essential component of the operation of CNNs (Salehi et al., 2023). Convolution is a linear operation employed for feature extraction. A set of numbers, referred to as a kernel, is used to process the input tensor. This kernel is applied across the tensor to perform the desired operation. At every point of the input tensor, an element-wise product is calculated between the tensor and the kernel. This process involves applying multiple kernels to generate a variable number of feature maps, as displayed in Figure 3.4. The feature maps demonstrate different features of the input tensors, and each kernel can be considered as a unique feature extractor (Yamashita et al., 2018). The convolution operation extracts high-level features such as edges and corners from input images. Next, the feature maps are passed to the next convolutional layer to extract other high-level features from the input image (Polat & Güngen, 2021).

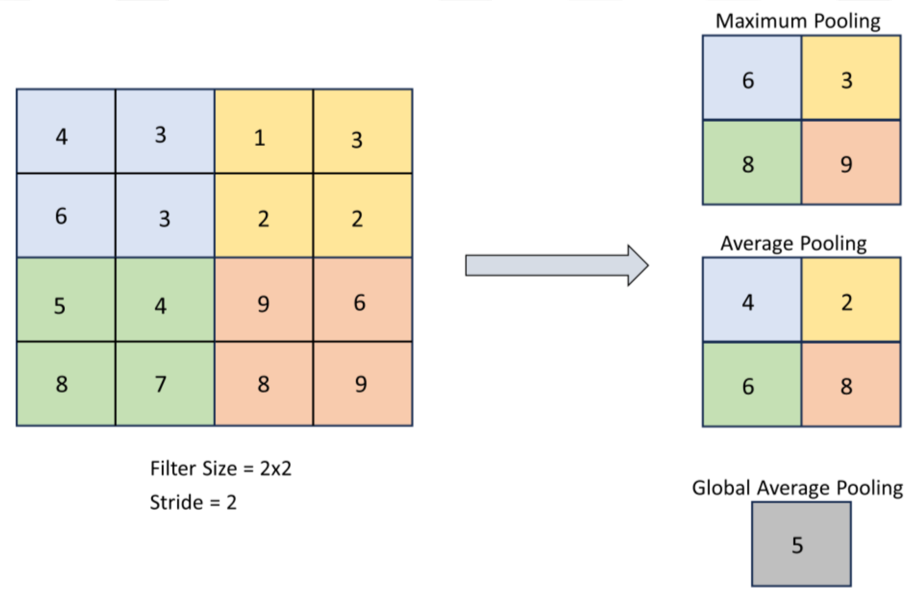

Figure 3.5. Maximum, Average, and Global Average Pooling

A pooling layer typically follows a convolutional layer to lower the size of feature maps and network parameters, reducing over-fitting and computational costs. It enables the network to remain invariant to minor transformations and distortions, allowing for a scale-invariant illustration of the image (Mane & Kulkarni, 2018; Salehi et al., 2023). Three common functions used for pooling are maximum, average and global average pooling (GAP), as demonstrated in Figure 3.5. The GAP carries out an extreme form of downsampling. The GAP differs from Average Pooling as it averages over the entire feature map from the previous layer rather than sliding a small window. The GAP layer transforms every W×H×C feature map input into a 1×1×C tensor. The number of Neural Network feature categories is equal to the number of feature map output channels generated by the last convolutional layer, which is also equivalent to the length of the vector produced by the GAP layer (R. L. Kumar, 2021). The GAP allows CNN to accept inputs of variable size and decrease the number of parameters that need to be learned (Yamashita et al., 2018).

The flattened layer connects the feature map of the convolutional layer with the fully connected hidden layer. This way, it can reshape multi-dimensional inputs into one-dimensional vectors (Kow, Hsia, Chang, & Chang, 2022).

The fully connected layers, sometimes called as dense layers, where each neuron is connected to all neurons of the previous layer, resulting in a high number of parameters (S.-H. Wang, Muhammad, Hong, Sangaiah, & Zhang, 2018; Yamashita et al., 2018). It multiplies the input by a weight matrix and adds a bias vector (Wang, 2020). It uses non-linear feature combinations to generate class prediction probabilities (Salehi et al., 2023). After every fully connected layer, a non-linear function is applied.

The activation function controls the activation of units in a layer, allowing better training, learning, and generalizing capabilities by working on a sum of weight and bias. The weights are updated depending on the output error, with activation functions providing gradients for back-propagation (B. Singh, Patel, Vijayvargiya, & Kumar, 2023). Adding activation functions after each convolutional and fully connected layer is necessary to transform non-linearity in the system (C. Wang, Peng, & De Baets, 2020). The sigmoid function becomes less effective when the input x grows large, causing the gradient to be close to zero. Rectified linear unit (ReLU) is now commonly used in deep learning instead of the traditional sigmoid function (C. Wang et al., 2020). ReLU is preferred over Sigmoid and Tanh functions due to its absence of the vanishing gradient problem (B. Singh et al., 2023). Nonetheless, during the training phase, certain ReLU neurons may permanently stop activating (C. Wang et al., 2020). In a multiclass classification task, an activation function that is commonly used is called the softmax function. This function allows the conversion of the output actual values from the last fully connected layer to probabilities for each target class. The probability values range between 0 and 1, with the sum of all values equalling to 1 (Yamashita et al., 2018).

Dropout is a regularization method that efficiently prevents overfitting and enables deep learning methods to work due to its low computational requirements (Piotrowski, Napiorkowski, & Piotrowska, 2020). During the training of a neural network, the standard dropout method randomly eliminates neurons. The dropout methods can reduce sensitivity to input variations and noise, improving deep learning model robustness (Salehin & Kang, 2023). Nevertheless, dropout in CNNs is limited to fully connected layers (Skourt, El Hassani, & Majda, 2022).

In 2012, a breakthrough occurred in the field of image recognition with the development of convolutional neural networks. The introduction of AlexNet marked a significant advancement in this technology, as it demonstrated the ability of deep neural networks to perform drastically in the domain of general-purpose image recognition. The success of AlexNet can be attributed to the flexible architecture of deep neural networks, which allows for the addition of layers of neurons and boosts the network's ability to learn (Jogin, Madhulika, Divya, Meghana, & Apoorva, 2018).

In our architecture, we employed three pre-trained CNNs which are VGG16, ResNet50, and MobileNetV2 models in the Keras framework.

3.3.1. VGG16

The VGG16, which represents ConvNet architecture, was introduced by Karen Simonyan and Andrew Zisserman in 2014 (Simonyan & Zisserman, 2014). The visual geometry group (VGG) network is commonly used due to its strong generalization capabilities and simple structure (Polat & Güngen, 2021). Simonyan and Zisserman formed five different models in their work with varying depths, as shown in Table 3.2. Model A has 8 convolution layers and 3 fully connected layers, whereas Model E has 16 convolution layers and 3 fully connected layers. All models, from A to E, are distinct from one another in terms of their depth. The parameters of models A to E are 133M, 133M, 134M, 138M, and 144M, respectively. They employed the ILSVRC-2012 dataset, a 1000-class dataset containing more than 1 million images.

Table 3.2. ConvNet Configuration (Simonyan & Zisserman, 2014)

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input(224 x 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

According to a single test scale, it was observed that the error decreased as the depth of the network increased. Models C and D have the same depth, but C has 3 1x1 convolution layers, and D has 3x3 convolution layers in the entire network. The D model performed better than the C. top-1 val error values, which are 29.6, 28.7, 27.3, 25.6, and 25.5 for A, B, C, D, and E, respectively. Besides, top-5 val error values are 10.4, 9.9, 8.8, 8.1, and 8.0 for A, B, C, D, and E, respectively (Simonyan & Zisserman, 2014).

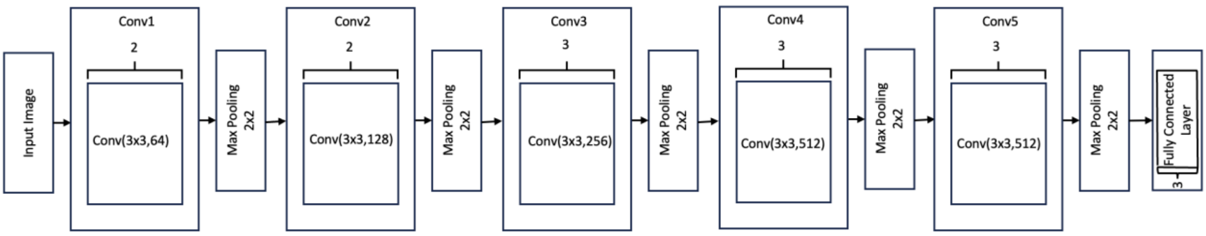

Figure 3.6. VGG16 architecture

The VGG16 network is comprised of 16 layers: 13 convolutional, 5 maximum pooling, and 3 fully connected, as depicted in Figure 3.6. The convolutional layer consists of five sections, and the number of convolutional kernels in each section is 64, 128, 256, 512, and 512. The pooling layers have a 2x2 window with stride 2, and the ReLU function is employed as the activation function for all hidden layers. There are three fully connected layers, the first two having 4096 nodes and the last having 1000 nodes. The last layer is the softmax activation layer (Azevedo, de Medeiros, Medeiros, Silva, & Costa, 2023). The VGG16 model accepts images with a size of 224 × 224 RGB pixels as input. The VGG16 network contains approximately 138 million trainable parameters (Polat & Güngen, 2021).

3.3.2. ResNet50

ResNet, which stands for Residual Networks, was introduced by He et al. (K. He, Zhang, Ren, & Sun, 2016). They utilized shortcut connections in their architecture to solve the degradation problem. This issue is that as the depth of the network increases, the accuracy value reaches saturation and then begins to decrease dramatically.

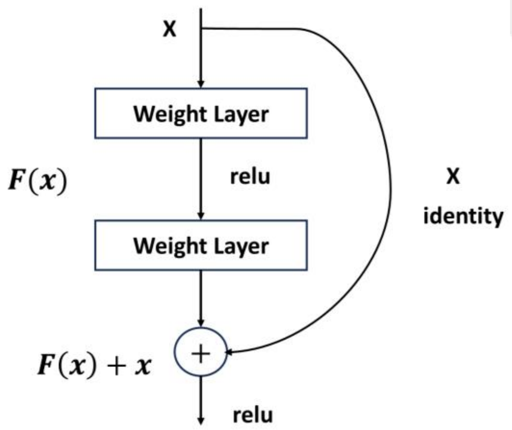

Figure 3.7. Residual learning: a building block (K. He et al., 2016)

There are two layers in Figure 3.7. This shortcut connection is formulated

$$y = F(x, \{W_i\}) + x \tag{1}$$

$$y = F(x, \{W_i\}) + W_s x. \tag{2}$$

where x, w, and y are input, weight, and output vectors, respectively. $F = W_2\sigma(W_1 x)$ was obtained as the output of the second layer, which represents residual mapping. σ and x represents ReLU, identity mapping. $F + x$ enables the transfer of important knowledge from the previous layer to the following layers. In this way, even if the gradient value approaches zero during backpropagation, the optimal function approaches identity mapping. In Eq (2), W_s is only employed for dimension matching.

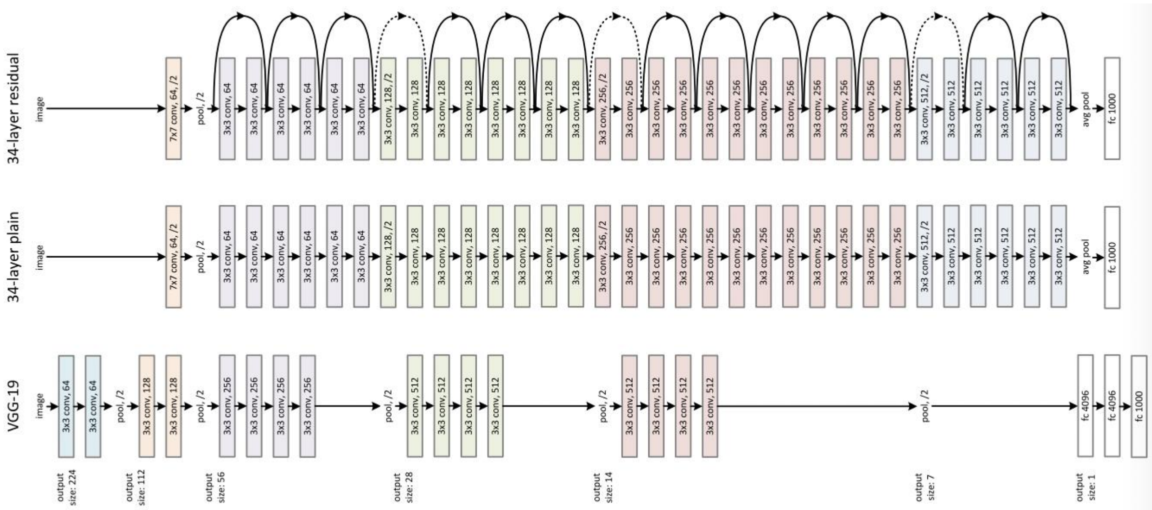

Figure 3.8. Top: The VGG19 Middle: a plain network Bottom: a residual network with 34 parameters (K. He et al., 2016)

Although their model has more layers than VGG19, as seen in Figure 3.8, their 34-layer baseline model with 3.6 billion float operations (FLOPs) has more trainable parameters than VGG19's with 19.6 billion FLOPs. Furthermore, based on the top-1 error, the 34-layer residual error rate (%28.54) was approximately 3.5% less than the 34-layer plain (%25.03). The bottleneck block is used to shorten the training time. To create the ResNet50 architecture, each 2-layer block in the 34-layer residual was replaced with a 3-layer bottleneck block. The top-1 error and top-5 error values of ResNet50 are 22.85% and 6.71%, respectively.

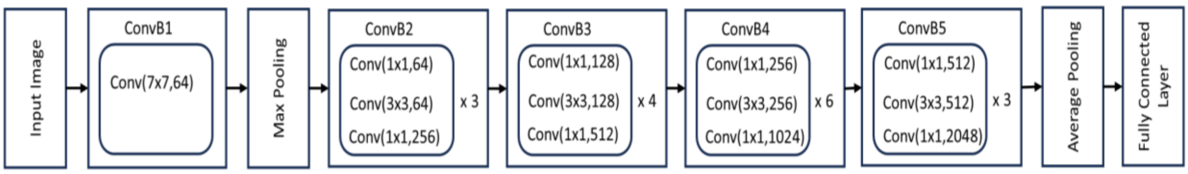

Figure 3.9. ResNet50 architecture

It is a deep neural network architecture composed of 50 convolutional layers trained on the ImageNet dataset that has more than 14 million images categorized into over 20 thousand classes for image recognition competitions (Emam, Samee, Jamjoom, & Houssein, 2023; Polat & Güngen, 2021). It won the 2015 ILSVRC Challenge (Emam et al., 2023). Additionally, bottleneck blocks are a key component in the ResNet architecture that enables faster training (Polat & Güngen, 2021). ResNet50 employs five different convolution blocks with varying convolution layers, filter sizes, and skip connections, as shown in Figure 3.9. While a 7x7 filter size was used in the first convolution block, three filter sizes were used in the remaining convolution blocks: 1x1, 3x3, and 1x1, respectively. The number of filters in every convolution block is different. The GAP and fully connected layers are employed as output layers (Kumar et al., 2021).

3.3.3. MobileNetV2

The MobileNetV2 introduced by Sandler et al. from Google in 2018 is a deep CNN (Maqsood, Damasevicius, & Maskeliunas, 2022; Zadeh Shirazi et al., 2020). It is specifically designed to operate efficiently on mobile and resource-limited devices. It employed the Depthwise Separable Convolutions approach, which comprised of a depthwise convolution layer and a pointwise convolution layer, which is a 1×1 convolution.

$$h_i . w_i . d_i \left(k^2 + d_j\right) \qquad (3)$$

The input of conventional convolution is $h_i \ x \ w_i \ x \ d_i$. When convolutional kernel $K \in R^{k \ x \ k \ x \ d_i \ x \ d_i}$ is applied to the input tensor, the output tensor is $h_i \ x \ w_i \ x \ d_i$. Its computational cost is $h_i \ x \ w_i \ x \ d_i \ x \ d_j \ x \ k \ x \ k$. When Depthwise Separable is used instead of standard convolution, computational cost decreases, as demonstrated in Eq (3).
The important feature of MobileNetV2 architecture is the use of linear bottleneck and inverted residual. Linear bottleneck is used to reduce input dimension without losing too much information. Shortcuts are also used between bottleneck blocks to prevent degrading and exploding gradient problems.

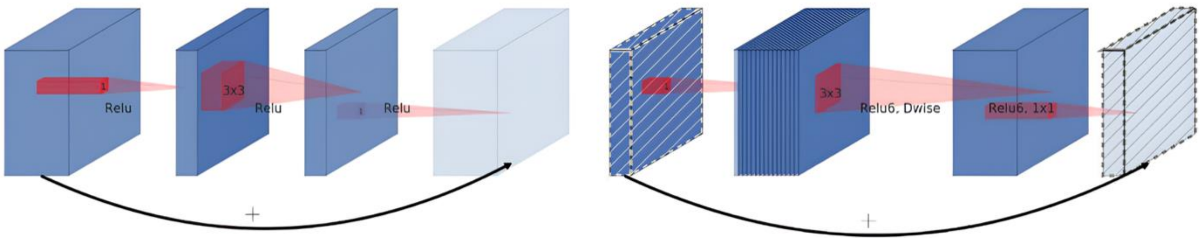

Figure 3.10. Residual block and inverted residual block (Sandler et al. 2018)

As illustrated in Figure 3.10**,** the inverted residual block is connected to bottlenecks, while the residual block is connected to layers that have high number of layers. The inverted residual block drops the memory cost (Sandler, Howard, Zhu, Zhmoginov, & Chen, 2018).

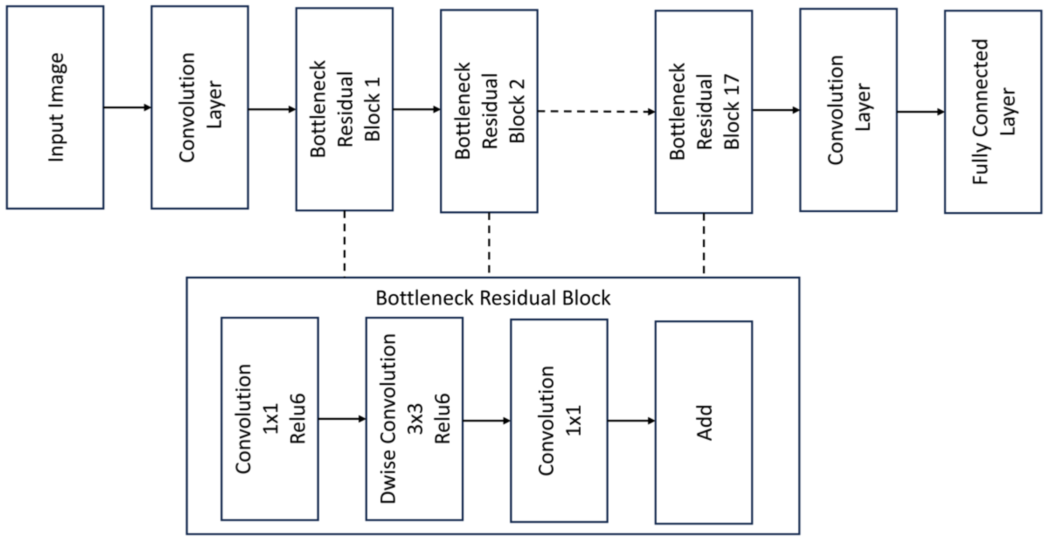

Figure 3.11. MobileNetV2 architecture (Kaya & Gürsoy, 2023a)

The MobileNetV2 architecture has 153 layers and takes input of size 224x224x3 (Maqsood et al., 2022). There are 32 initial convolution layers in it, and then there are 19 bottleneck layers (Kaya & Gürsoy, 2023a). The MobileNetV2 enables the reduction of the number of parameters, increased performance speed, lower latency, and smaller size (Maqsood et al., 2022).

3.4. Transfer Learning

Transfer learning (TL) is a machine learning technique gaining popularity in AI communities (Zhao, Hoi, Wang, & Li, 2014). Research on transfer learning has been conducted since 1995 under various descriptions: knowledge transfer, learning to learn, and lifelong learning more. In 2005, the Information Processing Technology Office of the Defense Advanced Research Projects Agency introduced a new mission for TL. This involves enabling a system to determine and make use of knowledge and skills acquired from earlier tasks to new ones. In transfer learning, knowledge is extracted from source tasks and applied to a target task. Conventional ML techniques attempt to learn each task from scratch. In contrast, TL techniques aim to transfer information from other tasks or domains to a target task with limited high-quality training data (J. Lu et al., 2015). TL can be classified as homogeneous or heterogeneous based on feature representation, where source and target domains have different feature fields (Zhao et al., 2014).

TL offers many advantages, including faster training, improved performance, and the ability to train with less data. The use of pre-trained model weights reduces training duration and accelerates learning. Besides, the reason for using the weights from a pre-trained model trained on a large amount of data is that they have already learned to recognize significant patterns and features for the given task. It is possible to achieve better performance and accurate results with less data (Polat & Güngen, 2021).

3.5. Fine-Tuning

Fine-tuning is a popular TL approach in neural networks (Vrbančič & Podgorelec, 2020). The fine-tuning allows some of the top layers of a frozen pre-trained model to unfreeze and simultaneously train the last layers of the pre-trained model, as displayed in Figure 3.12. A frozen convolutional base model's upper layers can be partially unfrozen through fine-tuning, which also trains the model's final layers concurrently (Kaya & Gürsoy, 2023a). In fine-tuning, it is unclear how many layers will be fine-tuned and how many will be left frozen (Vrbančič & Podgorelec, 2020).

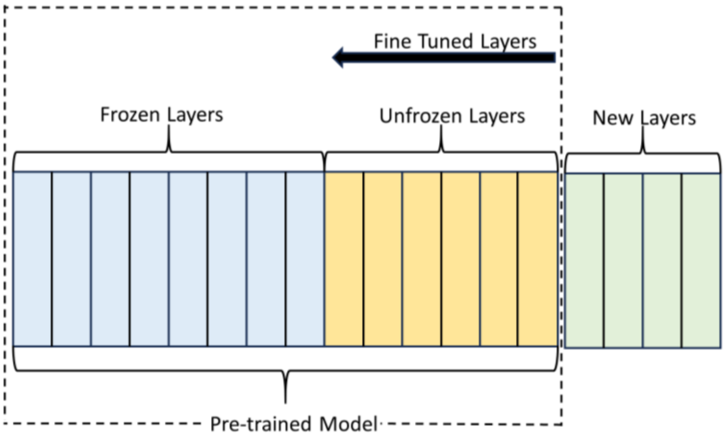

Figure 3.12. Fine-Tuning

An efficient fine-tuning method is to begin by updating the last layer and then gradually adding more layers until the required performance is achieved (Tajbakhsh et al., 2016). Therefore, in order to find the best performance in our study, we unfreezed from the last layer to the last eight layers of all pre-trained models, VGG16, ResNet50, and MobileNetV2. Later, all models are tested on all datasets. Finally, we found the best result in the model with the last two layers unfreezed in all datasets. Following the three pre-trained CNN models are fused, the total number of parameters is 41586978. Before fine-tuning, the number of trainable parameters was 1026594; after fine-tuning, the number of trainable parameters was 3388995.

3.6. Proposed Hybrid Model For Brain Tumor Classification

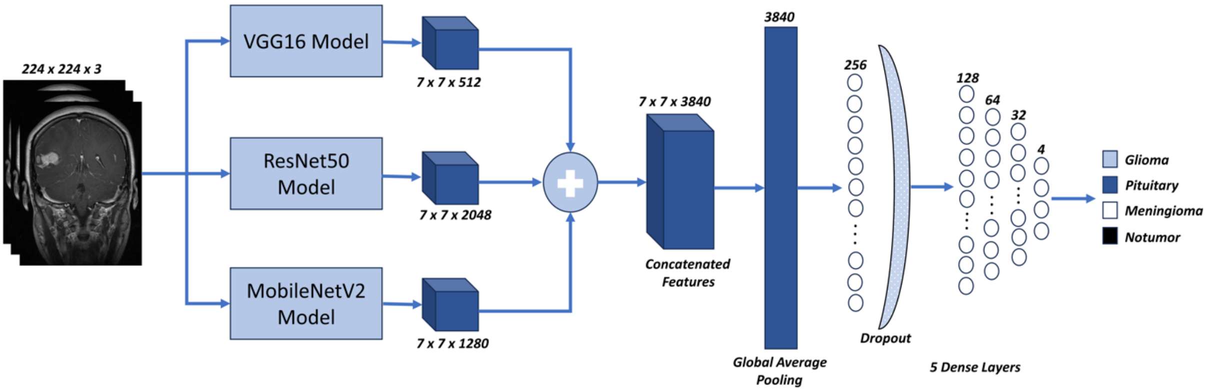

Figure 3.13. Structural design of proposed architecture

In this study, we fused deep feature extractions attained from pre-trained CNN VGG16, ResNet50, and MobileNetV2 models. In this manner, we are able to increase the number of comprehensive features so as to reduce overfitting and better the model's performance. We set a total of 3840 features by fusing 1280 from ResNet50, 512 from VGG16, and 2048 from MobileNetV2, as demonstrated in Figure 3.13.

$$F_{concatenated} = (F_{VGG16}^{T} \oplus F_{ResNet50}^{T} \oplus F_{MobileNetV2}^{T})^{T} \tag{4}$$

where, $\oplus$ represents the sum of two matrices. F_{VGG16}^{T}, $F_{ResNet50}^{T}$, and $F_{MobileNetV2}^{T}$ represent feature matrices obtained from VGG16, ResNet50 and MobileNetV2 models, respectively (Barai et al., 2023). F_{VGG16}^{T}, $F_{ResNet50}^{T}$, and $F_{MobileNetV2}^{T}$ are equal $7x7x512$, $7x7x1280$, and $7x7x2048$, respectively. When these matrices are summed, $F_{concatenated}$ is equal to $7x7x3840$.

We used GlobalAveragePooling2D to convert all extracted features into a vector, which was then fed into a dense layer as input. Following removing the last layer from all pre-trained models, we added 5 dense layers. The first four contain 256, 128, 64, and 32 neurons, respectively. We used ReLU as the activation function in all hidden layers. We included the Dropout with a rate of 0.5, after the first dense layer. It enables randomly removing nodes so as to avoid overfitting. In the last dense layer, we used the Sigmoid activation function for two classes and the Softmax activation function for more than two classes. Furthermore, the number of classes in the dataset determines the number of neurons in the final dense layer.

3.7. Model Parameters

We used the Adam optimizer with a learning rate of 0.001 for 100 epochs and a batch size of 32 to train our model. Moreover, we employed binary crossentropy loss for two classes and categorical crossentropy loss for more than two classes.

4. RESULTS AND DISCUSSION

4.1. Numerical Results

We performed all experimental tests on the T4 GPU using the 12 GB RAM provided by the Google Colab platform. We used Keras and Tensorflow frameworks. We randomly divide the datasets into 80% training and 20% testing datasets using a seed value of 0. Afterward, we employed seed value 0 to split into 20% of the training dataset for the validation dataset. The effectiveness of classification models is measured using different evaluation metrics. Some of these are accuracy, precision, recall, and F1-score metrics, as in Eqs (5), (6), (7), and (8). To provide a clearer visual representation of the class that is being incorrectly classified, we employed a confusion matrix in which each column represents the expected class and each row represents the actual class. False Positive Rate (X-axis) and True Positive Rate (Y-axis) are related, as indicated by a Receiver Operating Characteristic Curve (ROC) curve. The area under the ROC curve is represented by the AUC. The AUC is a performance metric that ranges from 0 to 1 and indicates the accuracy of a model's classification. A value closer to 1 indicates higher classification accuracy (Polat & Güngen, 2021).

We tested VGG16, ResNet50, and MobileNetV2 separately, fused them, and finally fine-tuned the fused model. We utilized four datasets in our experiments, namely Br35H, Figshare, Nickparvar, and Sartaj.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{Presicion} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{7}$$

$$\text{F1 Score} = \frac{2 * \text{Presicion} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}$$

- True Positive (TP) is the case where the speculated class is YES when the true class is YES,

- False Positive (FP) is the case where the speculated class is NO when the true class is YES.
- True Negative (TN) is the case where the speculated class is NO when the true class is NO.
- False Negative (FN) is the case where the speculated class is YES when the true class is NO.

4.1.1. Numerical Results For The Br35H Dataset

The training process of the fine-tuned fused model on the Br35H dataset is displayed in Figure 4.1. and Figure 4.2. While Figure 4.1. shows the validation accuracy value in the training process, Figure 4.2. shows the validation loss value in the training process. Through the training process of the model, 100 epochs of training were performed. Since we observed in previous training that the accuracy value reached saturation around 50 epochs, we performed fine-tuning at 50 epochs. As a result of 100 epochs of training, validation accuracy and validation loss values were 0.9875 and 0.1961.

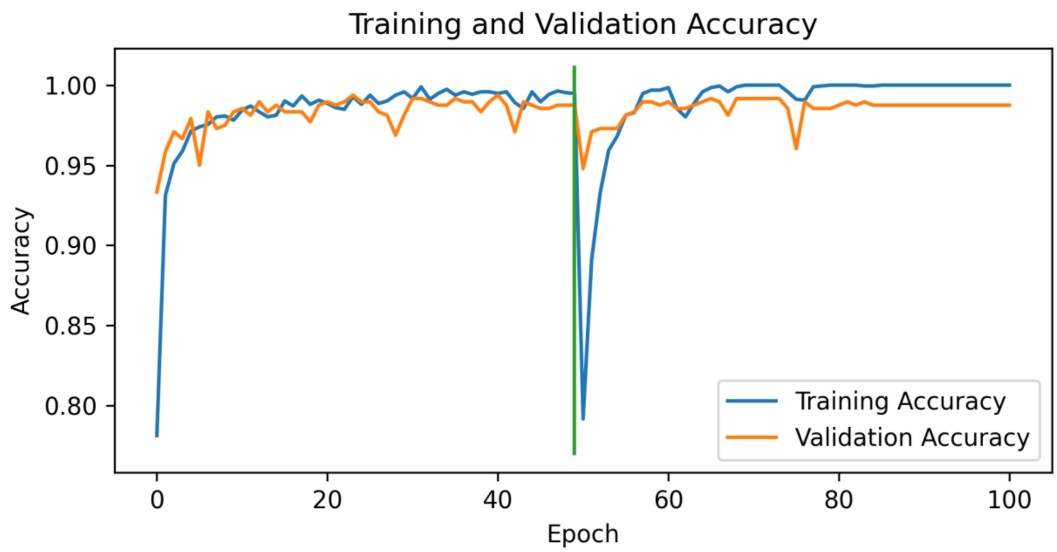

Figure 4.1. Accuracy over Epoch for fine-tuned fusion model

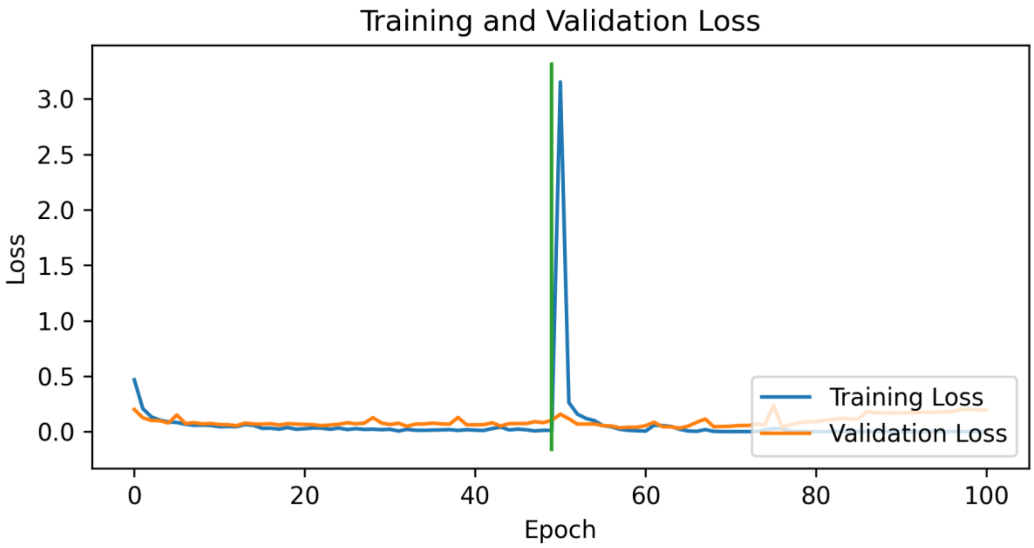

Figure 4.2. Loss over Epoch for fine-tuned fusion model

Table 4.1. demonstrates the performance metrics of all models for the Br35H dataset. While the VGG16 achieved 98.83% for accuracy, recall, and F1-score and 98.84% for precision, the ResNet50 achieved 99.17% for accuracy, precision, recall, and F1-score. The fused model had 99.00% for all metrics. Whereas the MobileNetV2 model yielded an accuracy of 96.50%, it had the lowest precision, recall, and F1-score. On the other hand, the fine-tuned fused model accomplished the highest accuracy, precision, recall, and F1-score, with a rate of 99.67%. There was an accuracy difference of around 3% between the MobileNetV2 and the fine-tuned fused models.

Table 4.1. Numerical results for the Br35H dataset

Method	**Accuracy**	**Precision**	**Recall**	**F1-score**
VGG16	0.9883	0.9884	0.9883	0.9883
ResNet50	0.9917	0.9917	0.9917	0.9917
MobileNetV2	0.9650	0.9650	0.9650	0.9650
VGG16+ResNet50+MobileNetV2	0.9900	0.9900	0.9900	0.9900
VGG16+ResNet50+MobileNetV2 (Fine-Tuning)	0.9967	0.9967	0.9967	0.9967

Figure 4.3 depicts the confusion matrix obtained from the Br35H dataset. We observed that while 299 No and 299 Yes were accurately classified, 1 No and 1 Yes were misclassified for

the fine-tuned fused model. The correct classification rates for No and Yes were 100% and 100%. The value of AUC was 1 for the model, as shown in Figure 4.4.

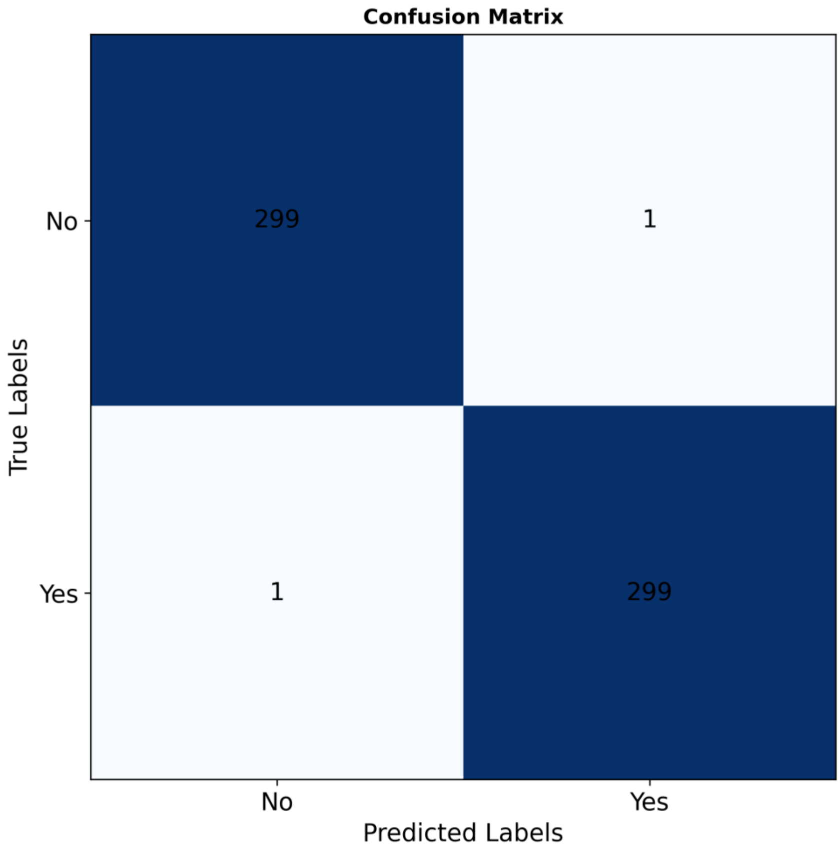

Figure 4.3. Confusion Matrix calculated on the Br35H dataset

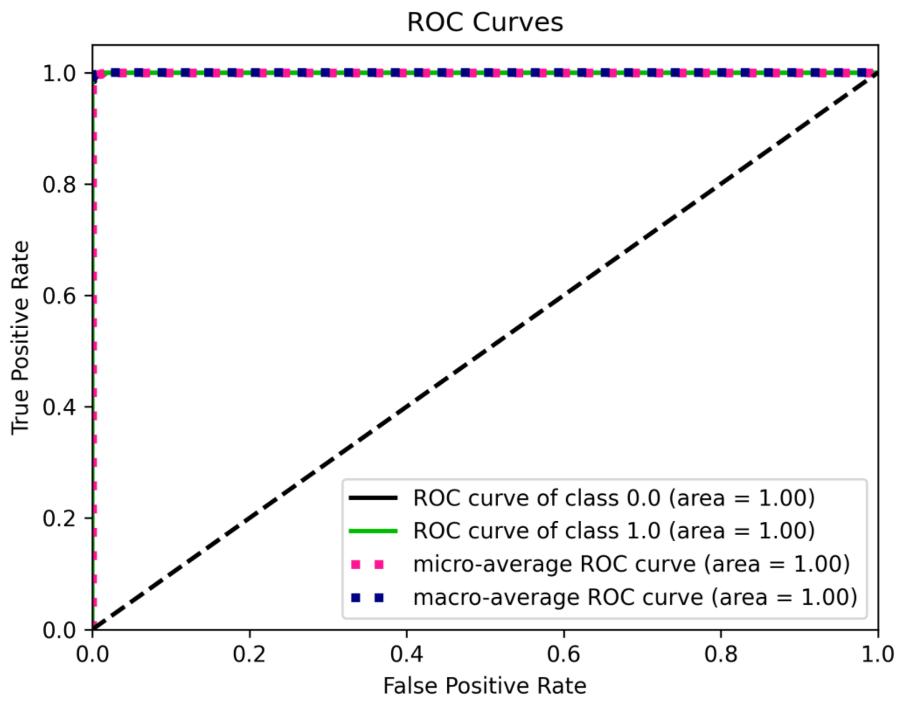

Figure 4.4. ROC Curve for the Br35H dataset

4.1.2. Numerical Results For The Figshare Dataset

Figure 4.5 and Figure 4.6 depicted validation accuracy and validation loss for the fine-tuned fused model during the training process. We saw that the accuracy value reached saturation around 50 epochs. We started fine-tuning at the 50th epoch to increase the performance of the model. Following 100 training epochs, the values of validation accuracy and validation loss were 0.9755 and 0.0909, respectively.

Figure 4.5. Accuracy over Epoch for fine-tuned fusion model

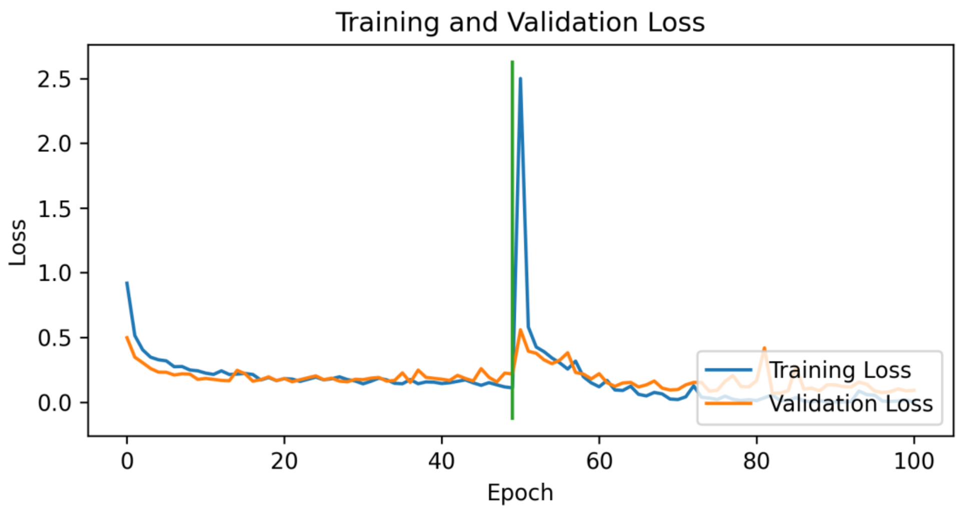

Figure 4.6. Loss over Epoch for fine-tuned fusion model

Evaluation metrics for the Figshare dataset are shown in Table 4.2. The MobileNetV2 had the lowest metrics. The MobileNetV2 achieved an accuracy of 91.37%, precision of 90.20%, recall of 91.31%, and F1-score of 90.63%. While the ResNet50 and the VGG16 models performed slightly better results than the MobileNetV2, the fused model performed better than the individual models. The fine-tuned model accomplished the best results, with 97.56% accuracy, 97.25% precision, 97.36% recall, and 97.31% F1-score. The fine-tuned fused model has an accuracy that is almost 4% higher than the accuracy of the fused model.

Table 4.2. Numerical results for the Figshare dataset

Method	Accuracy	Precision	Recall	F1-score
VGG16	0.9251	0.9162	0.9161	0.9159
ResNet50	0.9300	0.9210	0.9226	0.9217
MobileNetV2	0.9137	0.9020	0.9131	0.9063
VGG16+ResNet50+MobileNetV2	0.9332	0.9376	0.9137	0.9230
VGG16+ResNet50+MobileNetV2 (Fine-Tuning)	0.9756	0.9725	0.9736	0.9731

The confusion matrix evaluated on the Figshare dataset is viewed in Figure 4.7. Pituitary had the highest accurate classification rate (99%), followed by Glioma (98%), and Meningioma (95%). The number of correct classifications for Glioma, Meningioma, and Pituitary were 279, 135, and 185, respectively. There were 7, 1, and 7 misclassifications of meningioma, pituitary, and glioma, respectively. Figure 4.8. illustrates that AUC was 1 for the fine-tuned fused model.

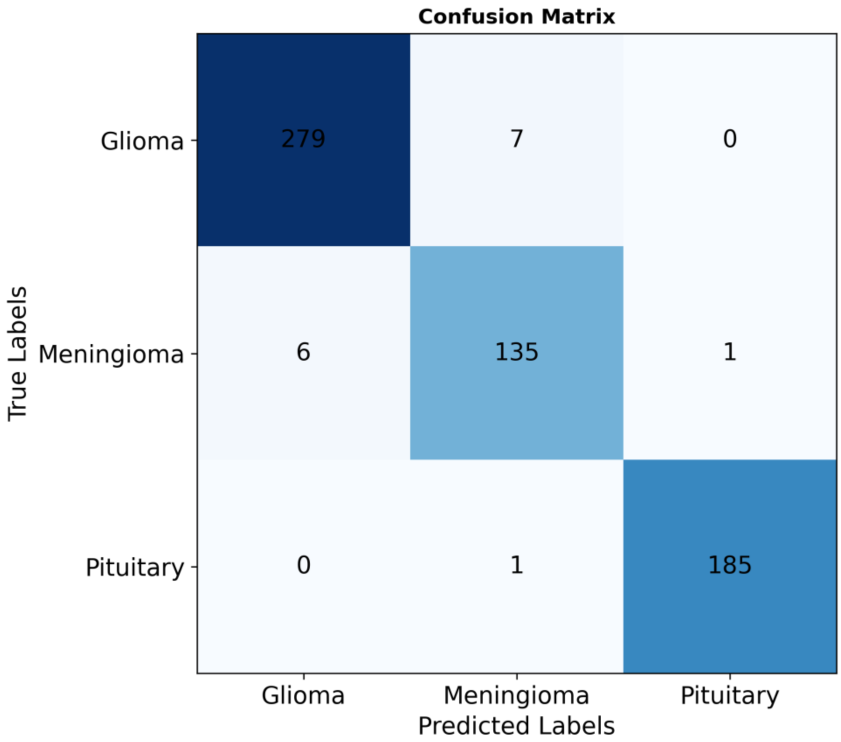

Figure 4.7. Confusion Matrix calculated on the Figshare dataset

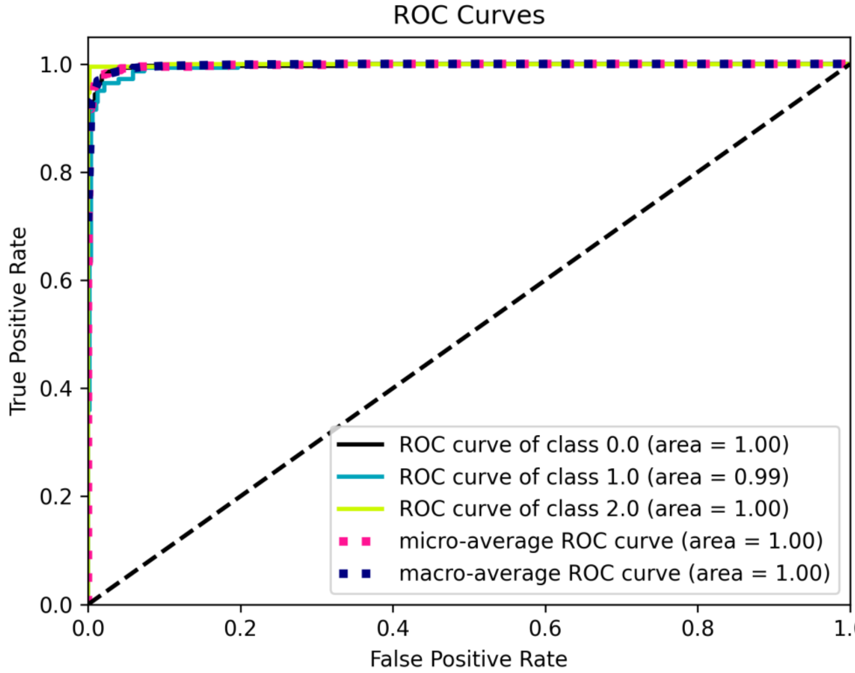

Figure 4.8. ROC Curve for the Figshare dataset

4.1.3. Numerical Results For The Nickparvar Dataset

Validation accuracy and validation loss values are illustrated in Figure 4.9. and 4.10. We started fine-tuning in the 50th epoch, and the validation accuracy value suddenly dropped to 0.9118. After 50 epochs of training with the last two unfrozen layers, the model achieved a validation accuracy of 0.9849 and a validation loss of 0.0689.

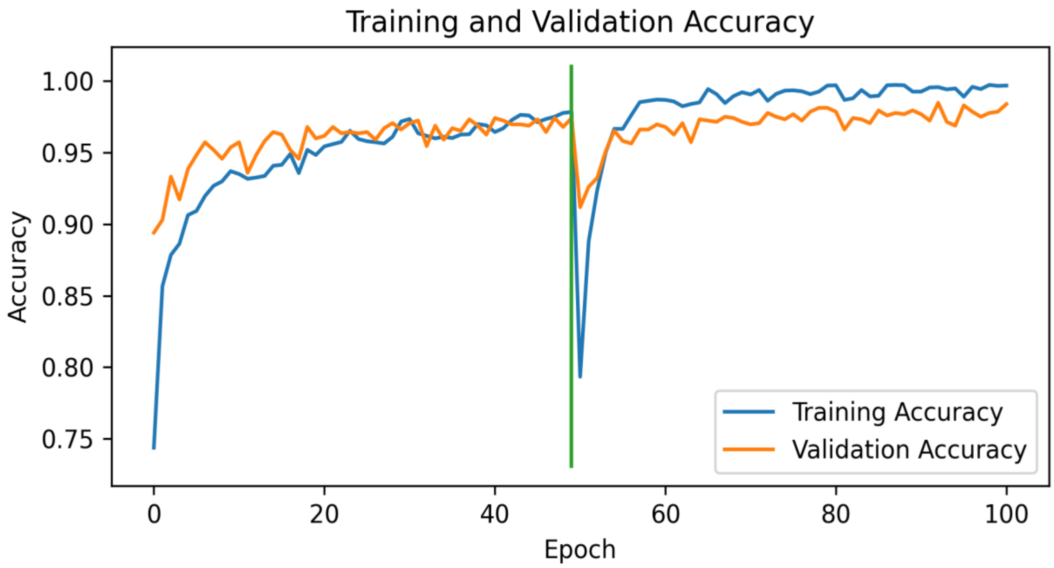

Figure 4.9. Accuracy over Epoch for fine-tuned fusion model

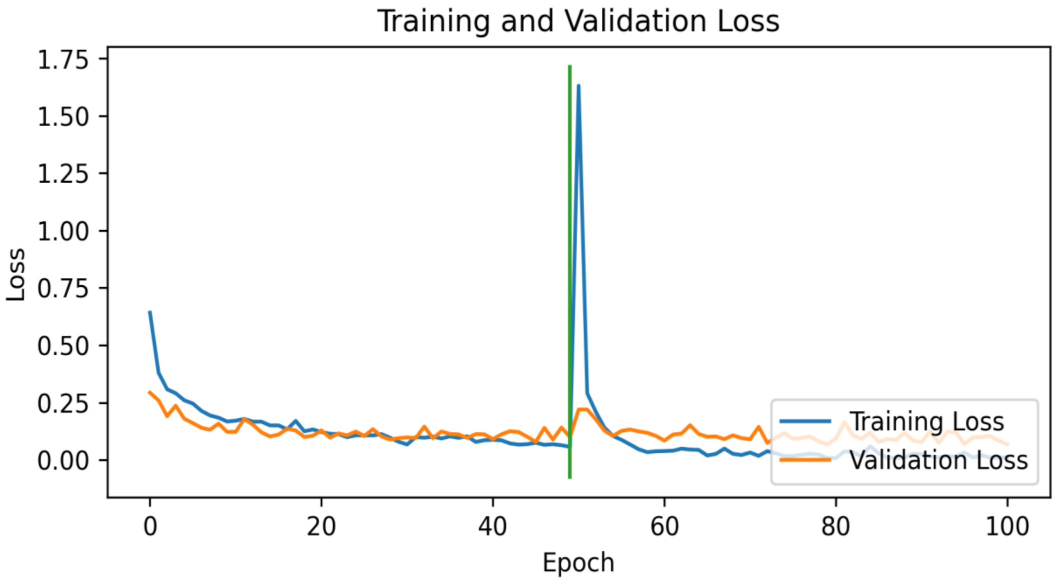

Figure 4.10. Loss over Epoch for fine-tuned fusion model

Table 4.3 depicts evaluation metrics for the Nickparvar dataset. The MobileNetV2 model showed the lowest performance. The ResNet50 and VGG16 models yielded reasonably similar results. The fused model outperformed the individual models by approximately 2%. The highest performance metrics were attained by the fine-tuned fused model: 97.08%, 97.02%, 96.94%, and 96.96% for accuracy, precision, recall, and F1-score, respectively.

Table 4.3. Numerical results for the Nickparvar dataset

Method	Accuracy	Precision	Recall	F1-score
VGG16	0.9488	0.9516	0.9462	0.9466
ResNet50	0.9467	0.9453	0.9444	0.9445
MobileNetV2	0.9324	0.9309	0.9294	0.9297
VGG16+ResNet50+MobileNetV2	0.9673	0.9669	0.9660	0.9660
VGG16+ResNet50+MobileNetV2 (Fine-Tuning)	0.9708	0.9702	0.9694	0.9696

Figure 4.11. demonstrated the confusion matrix obtained on the Nickparvar dataset. Glioma had the lowest accurate classification rate, 95%, whereas Pituitary and NoTumor had the highest rate, 100%. Meningioma had a 96% accuracy rate. Besides, the number of correct classifications of Glioma, Meningioma, NoTumor, and Pituitary was 309, 317, 399, and 400, respectively. Glioma, Meningioma, NoTumor, and Pituitary had 16, 12, 1, and 12 incorrect classifications, respectively. The AUC of 1 is displayed in Figure 4.12.

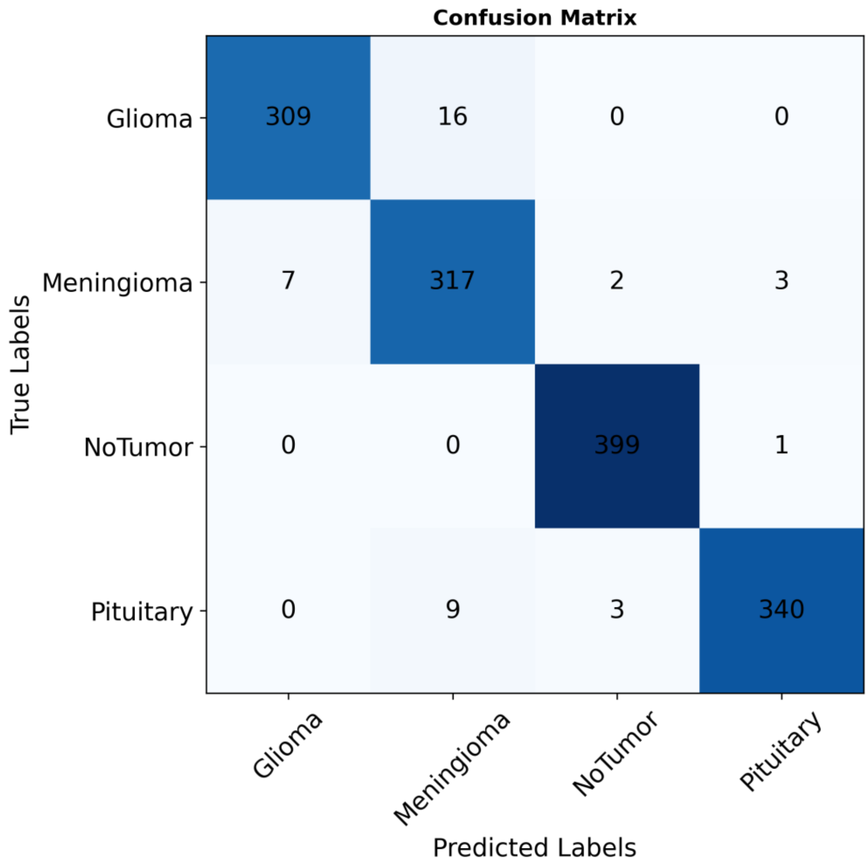

Figure 4.11. Confusion Matrix calculated on the Nickparvar dataset

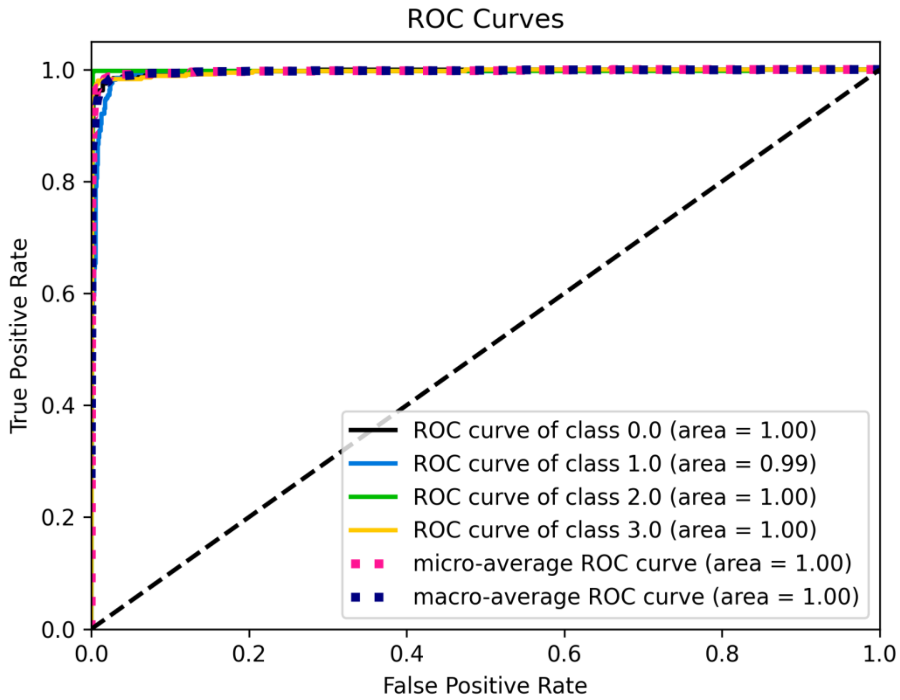

Figure 4.12. ROC Curve for the Nickparvar dataset

4.1.4. Numerical Results For The Sartaj Dataset

Figure 4.13 and Figure 4.14 displayed the training process of the fine-tuned fused model on the Nickparvar dataset. A total of 100 epochs of training were performed, the last 50 epochs of which were trained with fine-tuning. As a consequence of 100 epochs of training, validation accuracy and loss values were 0.9424 and 0.3581, respectively.

Figure 4.13. Accuracy over Epoch for fine-tuned fusion model

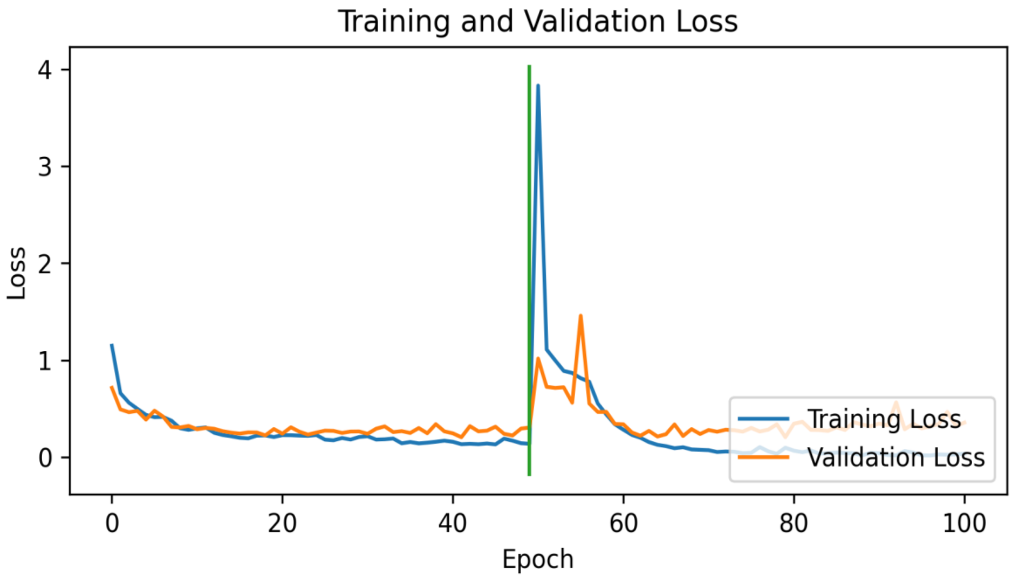

Figure 4.14. Loss over Epoch for fine-tuned fusion model

Performance metrics for the Sartaj dataset are presented in Table 4.4. The model with the lowest performance metric was The MobileNetV2, whose accuracy, precision, recall, and F1-score are 88.09%, 88.08%, 88.05%, and 87.96%, respectively. The accuracy attained by the VGG16, ResNet50, and fused models was very similar. Compared to the fine-tuned fused model with others, it has around 5% higher accuracy than the MobileNetV2 and about 2% higher than others.

Table 4.4. Numerical results for the Sartaj dataset

Method	**Accuracy**	**Precision**	**Recall**	**F1-score**
VGG16	0.9160	0.9176	0.9205	0.9188
ResNet50	0.9145	0.9195	0.9239	0.9213
MobileNetV2	0.8809	0.8808	0.8805	0.8796
VGG16+ResNet50+MobileNetV2	0.9130	0.9053	0.9188	0.9104
VGG16+ResNet50+MobileNetV2 (Fine-Tuning)	0.9374	0.9389	0.9404	0.9395

Figure 4.15 shows the confusion matrix tested on the Sartaj dataset. The most accurate classification was for Pituitary Tumor, with a success rate of 97%. The second most accurate classification was for No Tumor, with a success rate of 96%. The Meningioma_Tumor had the lowest classification rate of 90%, followed by the Glioma_Tumor, which had a classification rate of 93%. The number of accurate classifications for each tumor type were as follows: Glioma_Tumor (173), Meningioma_Tumor (169), No_Tumor (96), and Pituitary_Tumor (176). The number of incorrect classifications for each tumor type were as follows: Glioma_Tumor (14), Meningioma_Tumor (19), No_Tumor (4), and Pituitary_Tumor (5). We can see that the AUC was 0.99 in Figure 4.16.

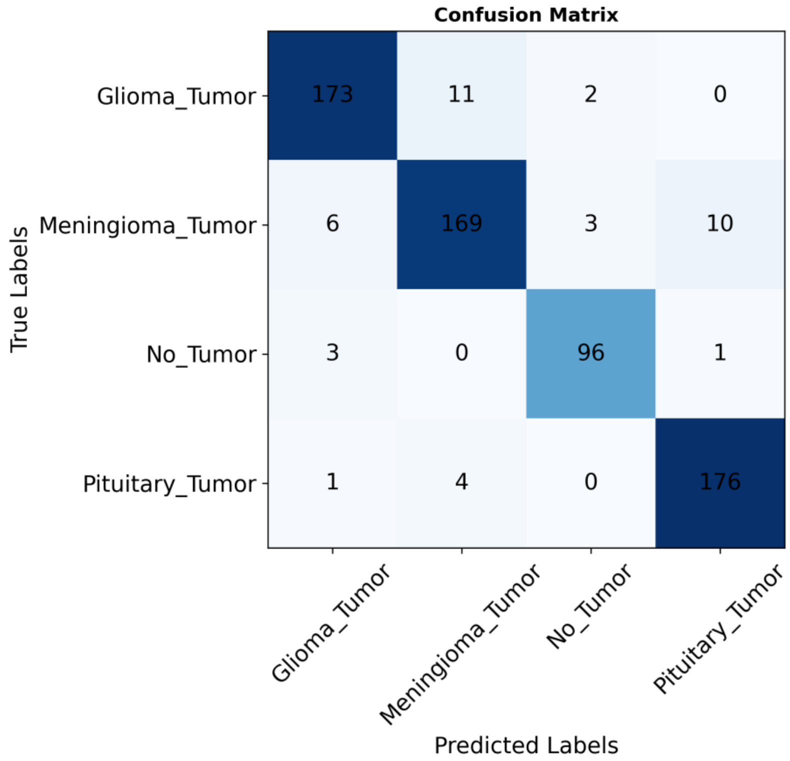

Figure 4.15. Confusion Matrix calculated on the Sartaj dataset

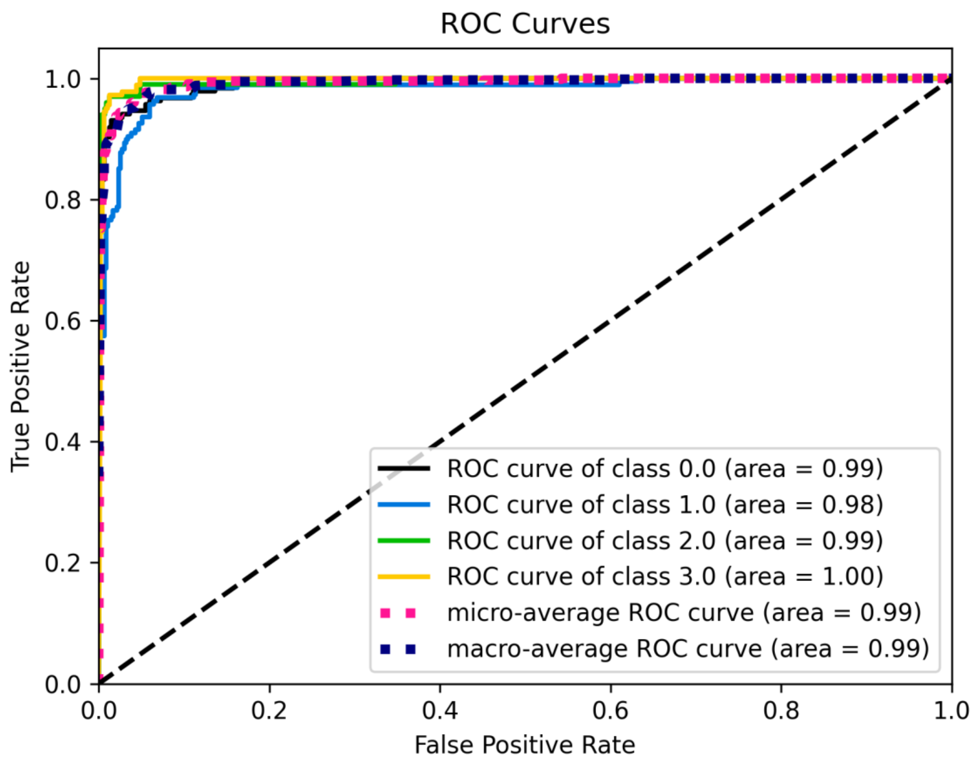

Figure 4.16. ROC Curve for the Sartaj dataset

4.2. Discussion

In this study, we have suggested a fine-tuned fused model to classify various types of brain tumors correctly. Tables 4.1, 4.2, 4.3, and 4.4 show that the MobileNetV2 model yielded the lowest results, whereas the suggested model yielded the best results across all datasets. The ROC curve shows an area of 1 in all datasets except for Sartaj (0.99), which indicates effective classification. Furthermore, we noticed that the suggested model was effective on all cross datasets. However, the fusion process and fine tuning increased the number of trainable parameters. As a result, it caused higher computational costs and longer training time. Upon comparison of the 4-class datasets, Sartaj (93.74%) and Nickparvar (97.08%), it was viewed that Nickparvar's accuracy rate is nearly 4% higher. This difference may be due to the number of images in the Nickparvar (7023) and Sartaj (3264) datasets, as the amount of data is very effective in deep learning (Cinar & Yildirim, 2020). Besides, the Nickparvar dataset was noticed to be the most effective dataset for the fused model. As complexity increases, fusion's impact may also increase.

Table 5.1. Comparison of the proposed model on the Figshare, Sartaj, Br35H, and Nickparvar datasets with the previous studies

Study	Dataset	Methodology	Accuracy
Swati et al. (2019)	Figshare	VGG19	94.82%
Anaraki et al. (2019)	Figshare	CNN	94.2%
Vankdothu et al. (2022)	Sartaj	CNN, LSTM	92%
Aurna et al. (2022)	Sartaj Figshare	CNN	99.76% 98.16%
Amran et al. (2022)	Br35H	GoogleNet	99.51%
Mondal and Shrivastava (2022)	Br35H Figshare	CNN	99% 99.57%
Remzan et al. (2023)	Nikparvar	ResNet50, VGG19 EfficientNetV2B1, MLP	96.67%
Arumugam et al. (2024)	Nickparvar Figshare	CNN, MLP	98.56%
Proposed model	Br35H Figshare Nickparvar Sartaj	Fusion of CNNs	99.67% 97.55% 97.08% 93.74%

As demonstrated in Table 5.1. many works have been carried out on Br35H, Figshare, Sartaj, and Nickparvar datasets until now. Swati et al. and Anaraki et al. achieved 94.82% (Swati et al., 2019) and 94.2% accuracy on the Figshare dataset (Anaraki et al., 2019), respectively. Swati et al. put forward a model that applies the block-wise fine-tuning method to the pre-trained VGG19 model. They also utilized fine-tuning techniques in their work, just like we do. Anaraki et al. introduced CNN architecture enhanced by employing the GA. The proposed model has a higher accuracy value compared to these models. As for the Sartaj dataset, Vankdothu et al. recommended a CNN-based model that combines a CNN with an LSTM, which achieved 92% accuracy (Vankdothu et al., 2022). The proposed model has a higher accuracy value than this model. Aurna et al. introduce a two-stage feature ensemble of deep CNN using Figshare and Sartaj datasets (Aurna et al., 2022). Accuracy values of 99.76% and 98.16% were obtained for the Sartaj and Figshare datasets, respectively. In their work, feature concatenating and data augmentation techniques were used on images, as we do. However, they performed concatenation twice using five different pre-trained models and a scratched CNN. They concatenated three pre-trained models like ours, they also added their own CNN model to the concatenation process to increase the model's performance. They utilized the PCA. The proposed model may not more effective than their model due to certain differences. Regarding the Br35H dataset, Amran et al. suggested a model using the GoogleNet model (Amran et al., 2022). An accuracy value of 99.51% was attained on the Br35H dataset, which is lower than that of the proposed model. Mondal and Shrivastava recommended a novel deep CNN model using the Parametric Flatten-p Mish activation function, which achieved 99% and 99.57% accuracy on Br35H and Figshare datasets (Mondal & Shrivastava, 2022). The proposed model’s accuracy is lower in the Figshare dataset but higher in the Br35H dataset. As to the Nickparvar dataset, Remzan et al. combined features attained from three pre-trained models, ResNet50, VGG19, EfficientNetV2B1, and used MLP as a classifier, similar to our study. Even though we utilized the same dataset, their work was less effective because they used fewer MR images. Arumugam et al. introduced a Crossover smell agent-optimized multilayer perceptron CNN model (Arumugam et al., 2024). They merged three datasets and formed a 2-class dataset to train their models. They also utilized BF and GLCM during preprocessing. Their model may have given better results than the proposed model, as it is easier to classify a 2-class dataset, and the model performs better in datasets with more data. To sum up, Table 5.1 displays promising outcomes for scratch CNN models, transfer learning and fine-tuning approaches, and machine learning algorithms.

5. CONCLUSIONS

Treating and surviving patients with brain tumors depends on correct classification. As a result, scientists and medical experts are actively looking for novel ways to treat this disease. Nonetheless, there are still certain areas in need of improvement. Hence, we recommended a hybrid transfer learning model to accurately classify brain tumors so as to advance the medical field and introduce a new model to the literature. Since pre-trained models allow successful results with short training times and small datasets, we employed a pre-trained model instead of scratch CNN in the fusion process. In our study, initially, we used normalization and data augmentation techniques on MR images. Secondly, we fused the features obtained from three pre-trained models, VGG16, ResNet50, and MobileNetV2, to enhance the model's performance. After the final layer was eliminated, five dense layers were added to the pre-trained models employing a dropout technique subsequent to the initial one. After training the recommended model for 50 epochs, we applied fine-tuning to our model. Following unfreezing the last two layers of all our pre-trained models, we retrained the recommended model for 50 epochs. Lastly, we carried out extensive experiments to assess the effectiveness of our recommended model utilizing the Br35H, Figshare, Nickparvar, and Sartaj public datasets. Additionally, we contrasted our model with other models on the basis of transfer learning. In conclusion, the recommended model accomplished the highest accuracy value across all datasets, with 99.66% accuracy on the Br35H dataset, 97.56% on the Figshare dataset, 97.08% on the Nickparvar dataset, and 93.74% on the Sartaj dataset.

6. RECOMMENDATIONS

For future studies, it will be possible to observe the fusion process effect more clearly by using the recommended model on various larger datasets. We hope to make the results better by experimenting with various pre-trained models and increasing the number of pre-trained models used for fusion. Due to the fusion process, there is an increase in trainable parameters, resulting in longer training times and higher computational costs. Therefore, different feature reduction approaches can be used.

REFERENCES

Abdelaziz Ismael, S. A., Mohammed, A., & Hefny, H. (2020). An enhanced deep learning approach for brain cancer MRI images classification using residual networks. *Artif Intell Med, 102*, 101779. doi: 10.1016/j.artmed.2019.101779

Adrian, J., Sagan, V., & Maimaitijiang, M. (2021). Sentinel SAR-optical fusion for crop type mapping using deep learning and Google Earth Engine. *ISPRS Journal of Photogrammetry and Remote Sensing, 175*, 215-235. doi: 10.1016/j.isprsjprs.2021.02.018

Alhassan, A. M., & Zainon, W. M. N. W. (2021). Brain tumor classification in magnetic resonance image using hard swish-based RELU activation function-convolutional neural network. *Neural Computing and Applications, 33*(15), 9075-9087. doi: 10.1007/s00521-020-05671-3

Amin, J., Sharif, M., Gul, N., Raza, M., Anjum, M. A., Nisar, M. W., & Bukhari, S. A. C. (2020). Brain tumor detection by using stacked autoencoders in deep learning. *Journal of medical systems*, *44*, 1-12. doi: 10.1007/s10916-019-1483-2

Amran, G. A., Alsharam, M. S., Blajam, A. O. A., Hasan, A. A., Alfaifi, M. Y., Amran, M. H., . . . Eldin, S. M. (2022). Brain Tumor Classification and Detection Using Hybrid Deep Tumor Network. *Electronics*, *11*(21), 3457. doi: 10.3390/electronic11213457

Anaraki, A. K., Ayati, M., & Kazemi, F. (2019). Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms. *biocybernetics and biomedical engineering, 39*(1), 63-74. doi: 10.1016/j.bbe.2018.10.004

Arumugam, M., Thiyagarajan, A., Adhi, L., & Alagar, S. (2024). Crossover smell agent optimized multilayer perceptron for precise brain tumor classification on MRI images. *Expert Systems with Applications*, *238*, 121453. doi: 10.1016/j.eswa.2023.121453

Aslan, N., Dogan, S., & Koca, G. O. (2023). Automated classification of brain diseases using the Restricted Boltzmann Machine and the Generative Adversarial Network. *Engineering Applications of Artificial Intelligence, 126,* 106794. doi: 10.1016/j.engappai.2023.106794

Aurna, N. F., Yousuf, M. A., Taher, K. A., Azad, A. K. M., & Moni, M. A. (2022). A classification of MRI brain tumor based on two stage feature level ensemble of deep CNN models. *Comput Biol Med*, *146*, 105539. doi: 10.1016/j.compbiomed.2022.105539

Ayadi, W., Elhamzi, W., Charfi, I., & Atri, M. (2021). Deep CNN for Brain Tumor Classification. *Neural Processing Letters*, *53*(1), 671-700. doi: 10.1007/s11063-020-10398-2

Azevedo, M. B., de Medeiros, T. d. A., Medeiros, M. d. A., Silva, I., & Costa, D. G. (2023). Detecting face masks through embedded machine learning algorithms: A transfer learning approach for affordable microcontrollers. *Machine Learning with Applications, 14*, 100498. doi: 10.1016/j.mlwa.2023.100498

Barai, A., Faruk, M. F., Shuvo, S. M., Srizon, A. Y., Hasan, S. M., & Sayeed, A. (2023, June). A late fusion deep cnn model for the classification of brain tumors from multi-parametric mri images. In *2023 International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)* (pp. 1-6). IEEE. doi: 10.1109/NCIM59001.2023.10212729

Bodapati, J. D., Shaik, N. S., Naralasetti, V., & Mundukur, N. B. (2020). Joint training of two-channel deep neural network for brain tumor classification. *Signal, Image and Video Processing, 15*(4), 753-760. doi: 10.1007/s11760-020-01793-2

Brunese, L., Mercaldo, F., Reginelli, A., & Santone, A. (2020). An ensemble learning approach for brain cancer detection exploiting radiomic features. *Comput Methods Programs Biomed, 185*, 105134. doi: 10.1016/j.cmpb.2019.105134

Cinar, A., & Yildirim, M. (2020). Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture. *Med Hypotheses, 139*, 109684. doi: 10.1016/j.mehy.2020.109684

Deepak, S., & Ameer, P. M. (2020). Retrieval of brain MRI with tumor using contrastive loss based similarity on GoogLeNet encodings. *Comput Biol Med, 125*, 103993. doi: 10.1016/j.compbiomed.2020.103993

Emam, M. M., Samee, N. A., Jamjoom, M. M., & Houssein, E. H. (2023). Optimized deep learning architecture for brain tumor classification using improved Hunger Games Search Algorithm. *Computers in Biology and Medicine, 160*, 106966. doi: 10.1016/j.compbiomed.2023.106966

Hamada, A. (2021). Br35H dataset. Retrieved 09.01.2023, from https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition. doi: 10.1109/CVPR.2016.90

He, L., Gong, X., Zhang, S., Wang, L., & Li, F. (2021). Efficient attention based deep fusion CNN for smoke detection in fog environment. *Neurocomputing, 434*, 224-238. doi: 10.1016/j.neucom.2021.01.024

Jogin, M., Madhulika, M., Divya, G., Meghana, R., & Apoorva, S. (2018). *Feature extraction using convolution neural networks (CNN) and deep learning.* Paper presented at the 2018 3rd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT). doi: 10.1109/RTEICT42901.2018.9012507

Kavi, D. (2021). Brain Tumor Image Dataset. Retrieved 08.01.2023, from https://www.kaggle.com/datasets/denizkavi1/brain-tumor

Kaya, Y., & Gürsoy, E. (2023). A MobileNet-based CNN model with a novel fine-tuning mechanism for COVID-19 infection detection. *Soft Computing, 27(9)*, 5521-5535. doi: 10.1007/S00500-022-07798-Y/TABLES/9

Kaya, Y., & Gürsoy, E. (2023). A novel multi-head CNN design to identify plant diseases using the fusion of RGB images. *Ecological Informatics, 75*, 101998. doi: 10.1016/j.ecoinf.2023.101998

Kesav, N., & Jibukumar, M. G. (2022). Efficient and low complex architecture for detection and classification of Brain Tumor using RCNN with Two Channel CNN. *Journal of King Saud University - Computer and Information Sciences, 34*(8), 6229-6242. doi: 10.1016/j.jksuci.2021.05.008

Kow, P. Y., Hsia, I. W., Chang, L. C., & Chang, F. J. (2022). Real-time image-based air quality estimation by deep learning neural networks. *J Environ Manage, 307*, 114560. doi: 10.1016/j.jenvman.2022.114560

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. doi: 10.1007/s11042-020-10335-4

Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., & Zhang, G. (2015). Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems, 80*, 14-23. doi: 10.1016/j.knosys.2015.01.010

Lu, S., Wang, S.-H., & Zhang, Y.-D. (2020). Detection of abnormal brain in MRI via improved AlexNet and ELM optimized by chaotic bat algorithm. *Neural Computing and Applications, 33*(17), 10799-10811. doi: 10.1007/s00521-020-05082-4

Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings, 3*(1), 91-99. doi: 10.1016/j.gltp.2022.04.020

Mane, D., & Kulkarni, U. (2018). Visualizing and understanding customized convolutional neural network for recognition of handwritten Marathi numerals. *Procedia computer science, 132*, 1123-1137. doi: 10.1016/j.procs.2018.05.027

Maqsood, S., Damasevicius, R., & Maskeliunas, R. (2022). Multi-Modal Brain Tumor Detection Using Deep Neural Network and Multiclass SVM. *Medicina (Kaunas), 58*(8). doi: 10.3390/medicina58081090

Mondal, A., & Shrivastava, V. K. (2022). A novel Parametric Flatten-p Mish activation function based deep CNN model for brain tumor classification. *Comput Biol Med, 150*, 106183. doi: 10.1016/j.compbiomed.2022.106183

Mumuni, A., & Mumuni, F. (2022). Data augmentation: A comprehensive survey of modern approaches. *Array*, 100258. doi: 10.1016/j.array.2022.100258

Nickparvar, M. (2021). Brain Tumor MRI Dataset. Retrieved 09.01.2023, from https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

Piotrowski, A. P., Napiorkowski, J. J., & Piotrowska, A. E. (2020). Impact of deep learning-based dropout on shallow neural networks applied to stream temperature modelling. *Earth-Science Reviews, 201*, 103076. doi: 10.1016/j.earscirev.2019.103076

Polat, Ö., & Güngen, C. (2021). Classification of brain tumors from MR images using deep transfer learning. *The Journal of Supercomputing, 77*(7), 7236-7252. doi: 10.1007/s11227-020-03572-9

Rehman, A., Naz, S., Razzak, M. I., Akram, F., & Imran, M. (2019). A Deep Learning-Based Framework for Automatic Brain Tumors Classification Using Transfer Learning. *Circuits, Systems, and Signal Processing, 39*(2), 757-775. doi: 10.1007/s00034-019-01246-3

Remzan, N., Hachimi, Y. E., Tahiry, K., & Farchi, A. (2023). Ensemble learning based-features extraction for brain mr images classification with machine learning classifiers. *Multimedia Tools and Applications*. doi: 10.1007/s11042-023-17213-9

Salehi, A. W., Khan, S., Gupta, G., Alabduallah, B. I., Almjally, A., Alsolai, H., . . . Mellit, A. (2023). A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope. *Sustainability, 15*(7), 5930. doi: 10.3390/su15075930

Salehin, I., & Kang, D.-K. (2023). A Review on Dropout Regularization Approaches for Deep Neural Networks within the Scholarly Domain. *Electronics, 12*(14), 3106. doi: 10.3390/electronics12143106

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). *Mobilenetv2: Inverted residuals and linear bottlenecks.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition. doi: 10.1109/CVPR.2018.00474

Sartaj. (2019). Brain Tumor Classification (MRI). from https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri

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*(4), 3007-3020. doi: 10.1007/s40747-021-00321-0

Sharif, M. I., Li, J. P., Amin, J., & Sharif, A. (2021). An improved framework for brain tumor analysis using MRI based on YOLOv2 and convolutional neural network. *Complex & Intelligent Systems, 7*(4), 2023-2036. doi: 10.1007/s40747-021-00310-3

Sharma, A. K., Nandal, A., Dhaka, A., Zhou, L., Alhudhaif, A., Alenezi, F., & Polat, K. (2023). Brain tumor classification using the modified ResNet50 model based on transfer learning. *Biomedical Signal Processing and Control, 86*, 105299. doi: 10.1016/j.bspc.2023.105299

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. doi: 10.48550/arXiv.1409.155

Singh, B., Patel, S., Vijayvargiya, A., & Kumar, R. (2023). Analyzing the impact of activation functions on the performance of the data-driven gait model. *Results in Engineering, 18*, 101029. doi: 10.1016/j.rineng.2023.101029

Singh, R., Goel, A., & Raghuvanshi, D. K. (2020). Computer-aided diagnostic network for brain tumor classification employing modulated Gabor filter banks. *The Visual Computer, 37*(8), 2157-2171. doi: 10.1007/s00371-020-01977-4

Skourt, B. A., El Hassani, A., & Majda, A. (2022). Mixed-pooling-dropout for convolutional neural network regularization. *Journal of King Saud University-Computer and Information Sciences, 34*(8), 4756-4762. doi: 10.1016/j.jksuci.2021.05.001

Swati, Z. N. K., Zhao, Q., Kabir, M., Ali, F., Ali, Z., Ahmed, S., & Lu, J. (2019). Brain tumor classification for MR images using transfer learning and fine-tuning. *Comput Med Imaging Graph, 75*, 34-46. doi: 10.1016/j.compmedimag.2019.05.001

Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., & Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging, 35*(5), 1299-1312. doi: 10.1109/ICoDT255437.2022.9787469

Togacar, M., Ergen, B., & Comert, Z. (2021). Tumor type detection in brain MR images of the deep model developed using hypercolumn technique, attention modules, and residual blocks. *Med Biol Eng Comput, 59*(1), 57-70. doi: 10.1007/s11517-020-02290-x

Toğaçar, M., Cömert, Z., & Ergen, B. (2020). Classification of brain MRI using hyper column technique with convolutional neural network and feature selection method. *Expert Systems with Applications, 149*, 113274. doi: 10.1016/j.eswa.2020.113274

Vankdothu, R., Hameed, M. A., & Fatima, H. (2022). A Brain Tumor Identification and Classification Using Deep Learning based on CNN-LSTM Method. *Computers and Electrical Engineering, 101*, 107960. doi: 10.1016/j.compeleceng.2022.107960

Vrbančič, G., & Podgorelec, V. (2020). Transfer learning with adaptive fine-tuning. *IEEE Access, 8*, 196197-196211. doi: 10.1109/ACCESS.2020.3034343

Wang, C., Peng, G., & De Baets, B. (2020). Deep feature fusion through adaptive discriminative metric learning for scene recognition. *Information Fusion, 63*, 1-12. doi: 10.1016/j.inffus.2020.05.005

Wang, S.-H., Muhammad, K., Hong, J., Sangaiah, A. K., & Zhang, Y.-D. (2018). Alcoholism identification via convolutional neural network based on parametric ReLU, dropout, and batch normalization. *Neural Computing and Applications, 32*(3), 665-680. doi: 10.1007/s00521-018-3924-0

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights Imaging, 9*(4), 611-629. doi: 10.1007/s13244-018-0639-9

Zadeh Shirazi, A., Fornaciari, E., Bagherian, N. S., Ebert, L. M., Koszyca, B., & Gomez, G. A. (2020). DeepSurvNet: deep survival convolutional network for brain cancer survival rate classification based on histopathological images. *Medical & biological engineering & computing*, *58*, 1031-1045. doi: 10.1007/s11517-020-02147-3

Zhao, P., Hoi, S. C., Wang, J., & Li, B. (2014). Online transfer learning. *Artificial intelligence*, *216*, 76-102. doi: 10.1016/j.artint.2014.06.003