INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT (IJSREM)
VOLUME: 07 ISSUE: 06 | JUNE - 2023 SJIF RATING: 8.176 ISSN: 2582-3930

Brain Tumor Classification Using Deep Hybrid Techniques

GUIDE
Prof.Nita Kulkarni
Computer Science And Engineering
SDM college of Engineering And
Technolgy(VTU)

Vaishnavi Indi
Computer Science And Engineering
SDM college of Engineering And Technology(VTU)
Dharwad ,India

Yasmeen Farha
Computer Science And Engineering
SDM college of Engineering And Technology(VTU)
Dharwad ,India
Yasmeenfarha423@gmail.co

Anushree KM
Computer Science And Engineering
SDM college of Engineering And Technology(VTU)
Dharwad ,India

N S Rachana
Computer Science And Engineering
SDM college of Engineering And Technology (VTU)
Dharwad , India
Rachanashettar366@gmail.co
m

***Abstract*— Brain tumour pose a substantial health risk, potentially resulting in serious neurological issues and even fatality. Magnetic Resonance Imaging (MRI) is a frequently employed diagnostic instrument for detecting and categorizing brain tumour; however, manually examining MRI images can be labour-intensive, subject to bias, and susceptible to human mistakes. Consequently, there is an increasing demand for creating automated and dependable techniques for classifying brain tumour using deep learning strategies.**

In this study, it was attempted to identify the disease's stage or whether it was a brain tumour using scans of the brain. To identify the illness, CNN architectures are employed. Additionally, a hybrid approach that we created has been suggested. Using the architectures currently in use, it is divided into 4 kinds based on the rate of disease progression. Alexnet, Resnet18, Googlenet, and the Hybrid technique we devised each produce the findings independently. The Resnet18 approach is the cornerstone of the proposed hybrid model. With the created hybrid model, an accuracy of 91% has been attained. Therefore, it is determined that the hybrid model created to diagnose brain tumours has attained the success achieved by previous CNN architectures and even delivers better outcomes after looking into other scientific papers in the literature.

Keywords—MRI, CNN Architectures, Alexnet, Resnet18, Googlenet.

I. INTRODUCTION

Brain tumors rank among the most prevalent forms of cancer that impact the central nervous system. They originate from irregular cell growth in the brain or adjacent tissues, which can result in serious neurological issues and even fatality. Prompt detection and precise categorization of brain tumors are crucial for enhancing patient prognosis and raising their chances of survival [1].

Magnetic Resonance Imaging (MRI) is a commonly employed diagnostic instrument for identifying and categorizing brain tumors. MRI generates intricate images of the brain, assisting in determining the tumor's size, position, and classification. Nonetheless, the manual examination of MRI images can be lengthy, biased, and susceptible to human mistakes. As a result, there is an increasing focus on creating automated and dependable methods for classifying brain tumors using deep learning approaches [2].

Deep learning falls under the umbrella of machine learning and employs neural networks with numerous hidden layers to learn from and extract features from data. Deep learning has demonstrated exceptional results across various domains, such as computer vision, natural language processing, and medical image assessment [3].

In recent times, multiple studies have explored the application of deep learning techniques for classifying brain tumors using MRI images. These investigations have showcased deep learning's potential to enhance the precision, speed, and consistency of brain tumor diagnosis. Deep learning methods are also able to manage extensive datasets and identify subtle distinctions between tumor types that may be imperceptible to the human eye.

Also, Machine learning (ML) has shown great advantages in the field of medical imaging, particularly in the classification of brain tumors. There are several types of brain tumors, like gliomas, meningiomas, pituitary and metastases, each with different characteristics that can be identified on studies related to imaging.

One of the approaches of brain tumor image classification using Machine learning is to train a model on a large dataset of labeled images. This can be done using various algorithms which are particularly well-suited for image classification tasks.

In this paper, the ResNet-18 is used as the base model and hybrid model is proposed.
Results were obtained with different networks like Alexnet, GoogleNet and ResNet-18 architectures. Later, the classification operating was executed with the developed hybrid ResNet-18 model where the highest performance rate was achievement. ResNet-18 model was used as a base in the developed hybrid model. Some layers of ResNet-18 model have been removed. In addition, new layers added to ResNet-18 model.
The aim of this research paper is to present a summary of cutting-edge deep learning techniques for MRI-based brain tumor classification. This research paper also aims to investigate the use of machine learning techniques for brain tumor image classification. The paper will discuss the current state-of-the-art in brain tumor imaging, the challenges of analyzing medical images, and the benefits of using machine learning algorithms for classification. The paper will also present a comparative analysis of various deep learning architectures for brain tumor image classification, and evaluate their performance on benchmark datasets. We will address the hurdles associated with classifying brain tumors and explain how deep learning can tackle these obstacles. Additionally, we will assess existing literature and emphasize the advantages and drawbacks of various deep learning models regarding brain tumor classification. Lastly, we will offer suggestions for prospective research avenues in this domain.

II. LITERATURE REVIEW

Similar research has previously been published in the literature. But two classifications were often used in the investigations. He is either absent from two lessons or he is ill. We have attempted to estimate the illness stage in this investigation. There are a total of 4 phases of the disease. For the diagnosis of the condition, this is crucial. Table 1 lists the earlier research on the topic that has been done.

Table 1. Literature review table

Author	Dataset	Methods	Accuracy
Dilip Ranjan Nayak [4]	Bra TS 2013	-Sunflower optimization Algorithm -Forensic based Investigation algorithm -Material Generation algorithm	90%
C. Narmatha [5]	BRATs 2018	A combination of fuzzy and brain-storm optimization techniques	93.85%
Komal Sharma [6]	-	Multi-Layer Perceptron (MLP) and Naïve bayes	98.6% and 97.6% respectively
Shilpa Rani [7]	Figshare, Brain MRI Kaggle, BraTS	3D Alexnet Classifier	98.88%
Zarin Anjuman Sejuti [8]	-	CNN-SVM based method	97.1%.
Arkapravo Chattopadhyay [9]	-	SVM classifier and other activation algorithms	99.74%

III. OVERVIEW FOR DEEP HYBRID MODEL

One of the key reasons that Deep Learning has become so popular is because it eliminates the need for labor-intensive, time-consuming human feature engineering on unstructured data, which is a requirement for practically all traditional machine learning methods. In traditional machine learning techniques, the algorithm's final performance and accuracy have been determined by the dataset's understanding and the capacity to execute feature engineering on it. Conversely, the final classification or clustering layer of a Deep Learning model powered by fully connected neural network layers may lead to over-fitting when fed with "less" data, or even most of the time,

these models require unnecessary usage of computational power and resources, which is not present in traditional machine learning algorithms.
This can be accomplished by deep hybrid learning, the fusion network that results from fusing deep learning and machine learning. Here, we'll demonstrate how to apply deep hybrid learning, in which we produce or extract features from unstructured data using deep learning techniques and then use traditional machine learning techniques to create highly accurate classification models based on the unstructured data. With Deep Hybrid Learning (DHL), we may thus combine the advantages of DL and ML while minimizing their respective shortcomings, resulting in solutions that are more precise and less resource-intensive. [10]

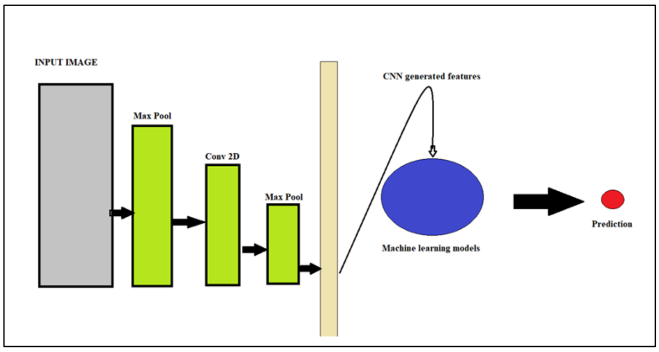


Figure 1. Deep Hybrid Model

IV. METHODOLOGY

Step 1: Acquiring (collecting) data sets
Step 2: Assessing it against a few pretrained networks to determine which is better for our dataset.
Step 3: We determined that resnet18 was superior to the other two models for our dataset. Therefore, we would like to proceed with further modification to enhance the accuracy. By altering various parameters, we added a few layers to Resnet 18 and calculated accuracy.
Step 4: Using our improved model, we now take the information from the feature layer and feed them to various machine learning algorithms in an effort to increase accuracy.
Step 5: Using this hybrid model, we can classify MRI scans of brain tumor.

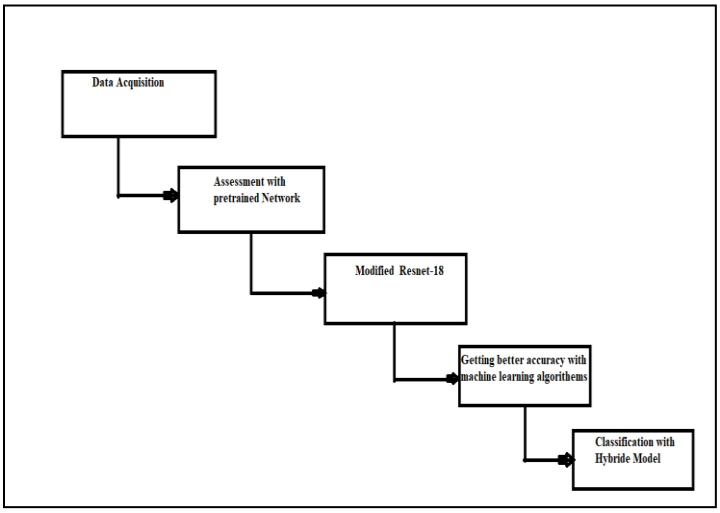
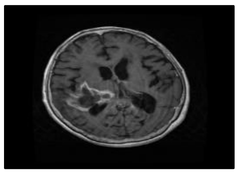
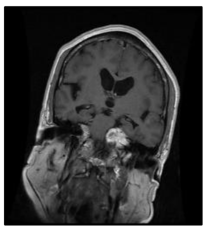


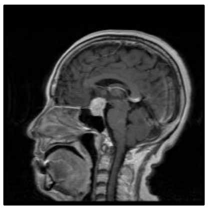
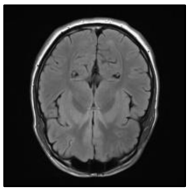
Figure 2. Main Model

A. *Dataset Acquisition*

Data set was obtained from the open access Kaggle website. The study made use of an Brain Tumor MRI image data set. There are four categories: Glioma Tumor, Meningioma Tumor, Pituitary Tumor, and No Tumor were found in the dataset. This data set has already been preprocessed. First and foremost, this data collection was used to train the various networks. It was then tested using test data. Table 2 depicts the amount of data and image samples used in the Study. 80% of the data are utilized for training, and 20% are used for testing. On a machine with an Intel i8 processor, the application was run in the Matlab environment.

Table 2. Images and Image Number

Type of tumor	Training Images	Testing Images	Image
Glioma Tumor	826	100	
Meningioma Tumor	822	115	

Pituitary Tumor	827	74	
No Tumor	395	105	

B. Assessing with different pretrained network

CNNs use neural networks to introduce the concept of hidden layers. When a single vector receives an input image, the hidden layers of the neural network perform a variety of neural changes. Each hidden layer has a large number of neurons, and each neuron's previous layer is linked to the next layer of neurons. Neurons within the same layer, however, are not linked. Each neuron has a specific function as well as a weighted input component. After applying functions and weights, the output of each neuron is skewed towards a positive or negative value. In order to reach a conclusion, this approach explores numerous hidden layers. The final layer is a fully connected layer that combines the results of all the concealed levels. [11]
As a deep learning-based feature extractor, we choose a CNN-based model since it can collect significant features without any human supervision.

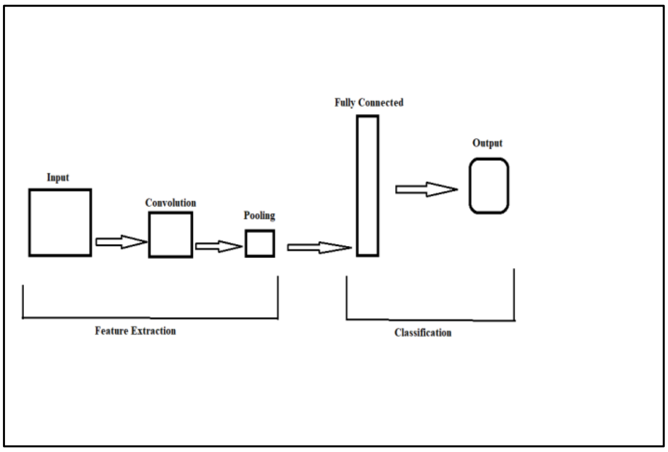


Figure 3. CNN Architecture
For classification purposes, three pretrained CNN architectures, GoogleNet, Alexnet, and ReseNet18, are employed in our research.

1) Googlenet
Googlenet was apparently created to accommodate for large feature representations by utilizing millions of daily item photos from the ImageNet collection. It can classify patterns in approximately 1000 photos. It employs 12 times less parameters than Alexnet. This model, like other neural networks used in computer vision applications, accepts images as input and outputs labels for one of its learnt classes along with the level of confidence. GoogLeNet's architecture is made up of 22 layers, including 9 inception modules. [12]

2) Alexnet
AlexNet competed in the ImageNet challenge (30) in 2012, and scored a top-5 error of only 15.3%, which was more than 10.8% better than the runner-up's result, which utilized a shallow neural network. AlexNet was originally run on two graphical processing units (GPUs). Currently, researchers use only one GPU to implement AlexNet. The structure of AlexNet is depicted in Figure. Only layers connected with learnable weights are counted in this investigation. As a result, AlexNet has five conv layers (CL) and three fully-connected layers (FCL), for a total of eight layers. [13]

3) Resnet-18
ResNet-18 is an 18-layer convolutional neural network. A pretrained version of the network trained on over a million photos from the ImageNet database can be loaded. The pretrained network can categorize photos into 1000 different object categories, including keyboards, mice, pencils, and various animals. As a result, the network has learned detailed feature representations for a diverse set of images. The network has an image input size of 224-by-224. [14]

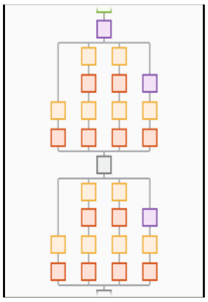


Figure 4.GoogleNet

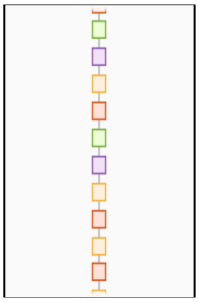


Figure 5.Alexnet

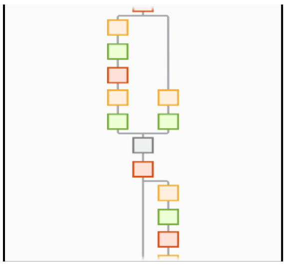


Figure 6.Resnet18

Using these three pre-trained networks, we train our dataset. Resnet-18 gives us the most accuracy, so we use it as our base model and continue to modify it.

C. *Modified Resnet-18*

Here now we use Resnet-18 as our base model and change some of the layers. In the improved model, the last five layers of Resnet18 have been removed. Ten new layers were added in place of these removed layers, and the number of layers increased from 71 to 75. This Time we have trained the improved network by varying different parameters like dropout factor, learning rate and optimizer. The architecture of the proposed hybrid model is as in Figure 7.

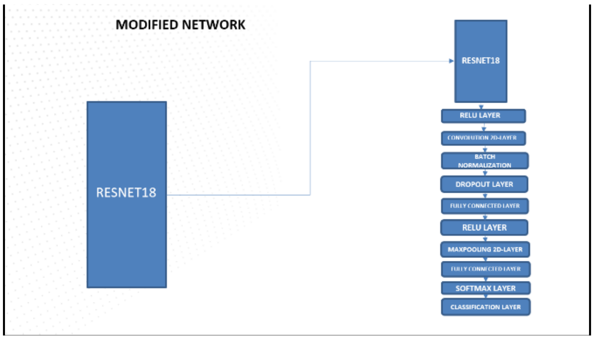


Figure 7.Modified Network

The layers used in the modified network are:

1) *Relu Layer :* Each input element is subjected to a threshold operation by a ReLU layer, which sets any value less than zero to zero. *[15]*

This action is comparable to

$$f(x) = \{ x , \quad x \geq 0 \; 0 , \quad x < 0$$

In order to obtain activation values for each neuron in neural networks, various common non-linear activation functions, such as sigmoid functions (or logistic) and hyperbolic tangent, are frequently utilized. Recently, the activation levels in conventional neural network or deep neural network paradigms have been calculated instead using the ReLu function. [16]

2) *Convulation-2d layer :* Sliding convolutional filters are applied to 2-D input by a 2-D convolutional layer. By moving the filters vertically and horizontally along the input, computing the dot product of the weights and the input, and then adding a bias term, the layer convolves the input. *[17]*

3) *Batch-Normalization* : A mini-batch of data is normalised across all observations for each channel separately by a batch normalisation layer. Use batch normalisation layers between convolutional layers and nonlinearities, such as ReLU layers, to accelerate convolutional neural network training and lessen the sensitivity to network initialization. Following normalisation, the layer moves and scales the input using learnable scale factors and offsets. *[18]*

4) *Dropout Layer :* A dropout layer randomly sets input elements to zero with a given probability. *[19]* By randomly removing nodes during training, a single model can be utilised to simulate having a huge variety of distinct network designs. Dropout is a regularisation technique that is incredibly computationally affordable and amazingly successful for reducing overfitting and enhancing generalisation error in all types of deep neural networks. *[20]*

5) *Fully-connected Layer*: A fully connected layer multiplies the input by a weight matrix and then adds a bias vector. Layers in a neural network that are fully linked have all of their inputs connected to every activation unit in the layer above them. The final few layers in the most of well-liked machine learning models are fully connected layers that combine the data gathered by earlier layers to create the output. The Convolution Layer takes the greatest time, followed by this layer. *[21]*

6) *Max-pooling 2d layer* : A 2-D max pooling layer performs downsampling by dividing the input into rectangular pooling regions, then computing the maximum of each region. There are other types of pooling that use the exact same methodology as what we just did, with the exception that they do a different operation on the areas rather than aiming for the maximum value. *[22]*

7) *SoftMax Layer :* A softmax layer applies a softmax function to the input. convolutional neural network that can distinguish between a cat and a dog in an image. The two groups are mutually exclusive because an image may only be either a cat or a dog and not both. This network would typically return numbers like [-7.98, 2.39], which are not normalised and cannot be understood as probabilities. It is possible to convert the numbers into a probability distribution by including a softmax layer in the network. This indicates that the user can see the output, for instance, the app is 95% certain that this is a cat. Additionally, since the output is guaranteed to reside between 0 and 1, it can be fed directly into other machine learning algorithms without the requirement for normalisation. It should be noted that if the network is configured to have only two output classes and is classifying photos into dogs and cats, it is compelled to

classify every image as either a dog or a cat, even if it is neither. The neural network must be reconfigured to incorporate a third output for miscellaneous if we need to account for this scenario. *[23]*

8) Classification layer : For classification and weighted classification problems with classes that are mutually exclusive, a classification layer calculates the cross-entropy loss. From the output size of the preceding layer, the layer infers the number of classes. Before the classification layer, for instance, you may include a fully connected layer with an output size of K and a softmax layer to specify the network's K number of classes. *[24]*

Parameters Used :

1) *Learning rate*: One of the most crucial hyperparameters to tune is learning rate, which is essential for the quick and efficient training of neural networks. The amount of the erroneous value that must be returned to the network's weights in order to proceed in the direction of lesser loss depends on the learning rate. *[25]*
2) *Dropout Factor*: In order to speed up processing and the time it takes to receive results, dropout refers to data or noise that is purposefully removed from a neural network. A network does this by removing all transmissions from its neuron nodes that aren't directly linked to the issue or training it's working on. Dropout is the technical word for this neuron node removal. [26]
3) *Optimizers* : Optimizers are techniques or algorithms used to reduce a loss function (error function) or increase production efficiency.

   SGDM: A momentum term is added to standard stochastic gradient descent in the stochastic optimisation technique known as SGD with Momentum. The direction of the previous update is kept to some extent during the update, while the current update gradient is utilised to fine-tune the final update direction. This simulates the inertia of an item when it is moving. This will allow you to learn more quickly and eliminate local optimisation. It will also boost stability to a certain level. *[27]*

   ADAM: Adam is an alternate optimisation approach that generates more effective neural network weights by repeatedly iterating "adaptive moment estimation." In comparison to many other optimisation programmes, Adam builds on stochastic gradient descent to solve non-convex problems more quickly and with less resource usage. It works best in extremely large data sets by maintaining "tighter" gradients during numerous learning iterations. *[28]*

D. Modification with Machine Learning

We currently have all the features in the network's feature layer, which were taken from it and employed using machine learning methods to increase accuracy.

The algorithms which we have used are:

1) SVM(support Vector Machine): Support Vector Machine (SVM) is a classification and regression prediction tool that automatically prevents over-fitting to the data while maximizing forecast accuracy. Support Vector Machines are systems that use the hypothesis space of linear functions in a high-dimensional feature space and are trained using an optimization theory-based learning method that incorporates a learning bias. Initially well-liked by the NIPS community, support vector machines are now a vital component of machine learning research all across the world. SVM gains notoriety when, when given pixel maps as input, it performs handwriting recognition tasks with accuracy on par with complex neural networks and extensive features. *[29]* Finding a decision boundary with the greatest margin of separation between data points from distinct classes is the fundamental goal of SVM. In other words, SVM seeks the best line or curve that divides data points into distinct classes with the least amount of overlap. Support vectors are the data points that are closest to the decision border.

2) Decision Tree: A decision tree is built by posing a series of queries regarding a record of the dataset at hand. Following up on each response, a new question is posed until the record's class label is determined. The series of queries and potential solutions can be arranged as a decision tree, a hierarchical structure made up of nodes and directed edges.

A tree has three different kinds of nodes:

- A root node with zero or more outward edges but no incoming edges.
- Each internal node has two or more outgoing edges and exactly one incoming edge.
- Each leaf or terminal node has one incoming edge and zero outgoing edges.

Each leaf node in a decision tree is given a class label. The root and other internal nodes are examples of non-terminal nodes that contain attribute test conditions to distinguish records with various attributes. *[30]*

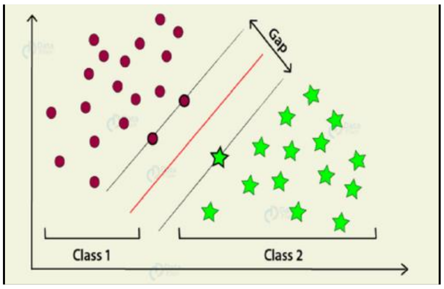


Figure 8.SVM

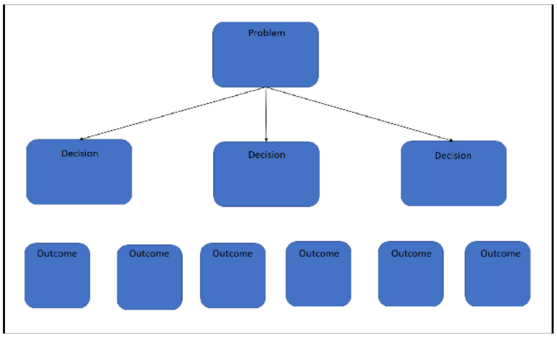


Figure 9.Decision Tree

We achieve 100% accuracy with the SVM model and 99.3 with the Decision Tree by using machine learning classifiers to train the retrieved features. As a result, we continue by testing the model and computing the performance metrics.

E. Classification and Performance metrics

Now we test the model and classify the images.
The performance of the classification can be scaled using a number of favored approaches. Utilizing the confusion matrix, these computations are made. The most recommended measures are F-Score, Accuracy, Precision, and Recall.

1) Confusion matrix: One of the key measures employed in the classification stage of CNN architectures is the confusion matrix. The Confusion matrix is used to calculate additional performance metrics. *[31]*

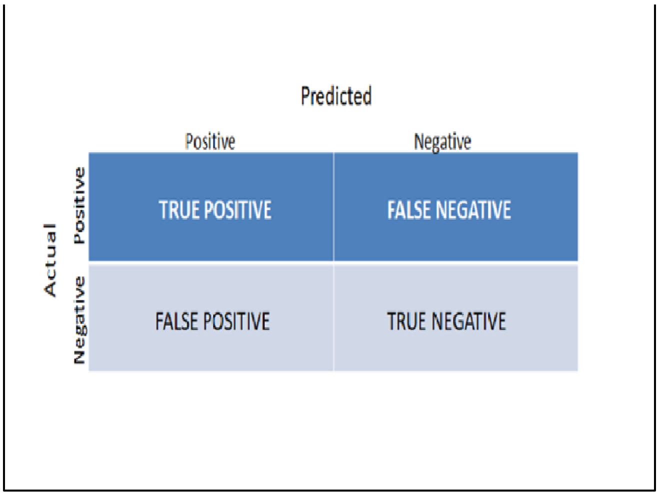


Figure 10. Confusion Matrix

- True Positive (TP): TP is the calculated number of data that are correct
- True Negative (TN): The fact that the data is estimated negatively is what is truly negative.
- False Positive (FP): Since the data is estimated positively, it is actually negative.
- False Negative (FN): The fact that the data is estimated negatively hides the fact that the result is actually positive.

2) Accuracy: The percentage of correctly classified data instances over all data instances is known as accuracy. *[32]*

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

3) Precision : A good classifier's precision should preferably be 1 (high). Only when the numerator and denominator are equal, or when TP = TP + FP, does precision become 1, which also implies that FP is zero. As FP rises, the precision value lowers and the denominator value rises, which is the opposite of what we desire. *[32]*

$$Precision = \frac{TP}{TP + FP}$$

4) Recall: A good classifier should have recall that is 1 (high). Recall only increases to 1 when the numerator and denominator are equal, or when TP = TP + FN, which also implies that FN is zero. As FN rises, the recall value falls (which is what we don't want) and the value of the denominator exceeds the numerator. *[32]*

$$Recall = \frac{TP}{TP + FN}$$

5) F-score : Therefore, the ideal precision and recall for a competent classifier are one, which also implies that FP and FN are zero. As a result, we require a statistic that considers both precision and recall. The definition of the F1-score, a statistic that considers both precision and recall. [32]

$$F\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

V. EXPERMENTAL RESULTS

The purpose of this study is to categorize the data according to the four stages of the brain tumor sickness. We employed CNN architectures along with the hybrid approach we created. These four classes are:

1.Pituitary tumor
2.Meningoma tumor
3.Glioma tumor
4.No tumor

The pre-trained networks GoogleNet, AlexNet, and ResNet have been contrasted. ResNet is the network here that displays the highest accuracy so we select ResNet.The graphic shows the accuracy and loss graphs for these networks.

Figure 11. GoogleNet accuracy and loss curves

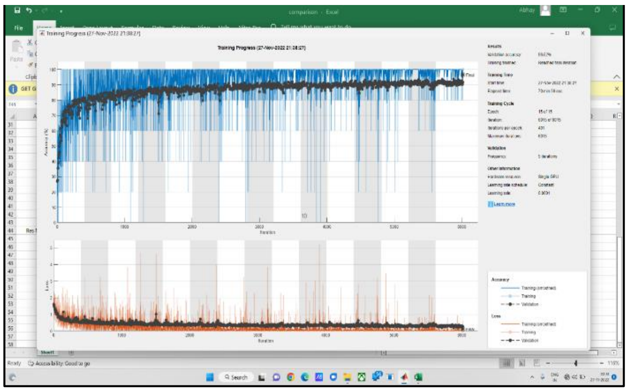


Figure 12. AlexNet accuracy and loss curves

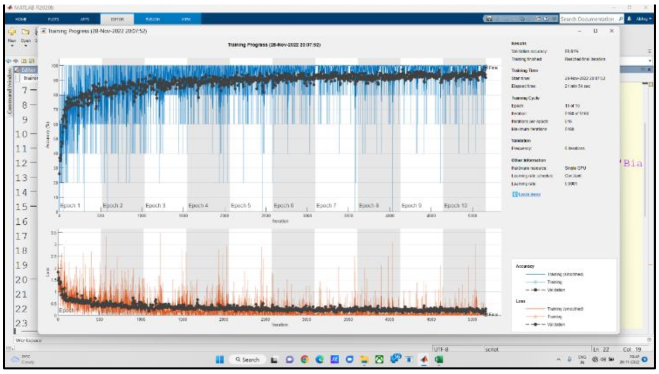


Figure 13. ResNet-18 accuracy and loss curves

Table 3. Comparison Table between accuracies of pretrained network

Network	%Training	%Validation	No. of Epochs	No. of Iterations/ Epoch	Mini Batch size	Time	Accuracy
	70	30	15	5	5	5m 48s	48.77%
Google Net	80	20	10	6	5	3m 57s	52.01%
	90	10	5	7	5	2m 19s	50.11%
Alex Net	80	20	10	459	5	59m 12s	94.76%
	90	10	15	516	5	48m 34s	93.03%
	70	30	15	401	5	70m 59s	96.50%
Res Net	80	20	15	459	6	71m 34s	95.64%
	90	10	10	156	7	21m 34s	98.61%

We altered several of the layers using the chosen ResNet.Resnet18's final five layers have been taken off. These lost layers were replaced with ten additional layers, bringing the total number of layers up to 75. By adjusting several parameters, including the dropout factor, learning rate, and optimizer, we trained the improved network. And we achieved the highest accuracy at dropout of 0.5 (93.2). Confusion matrix obtained after the network is trained and tested.

Table 4. Accuracy with Different optimization algorithm

OPTIMIZATION ALGORITHM	ACCURACY(%)
ADAM	28.62
SGDM	92.32

Table 5. Accuracy with various dropout factor

DROPOUT	ACCURACY(%)
0.3	91.80
0.5	93.2
0.7	90.75

Table 6. Accuracy with various Learning rate

LEARNING RATE	ACCURACY(%)
0.001	83.25
0.01	92.32
0.1	28.80

The network's feature layer currently contains all of the features that were extracted from it and used to boost accuracy using machine learning techniques. The algorithms that we employed are as follows:

1.SVM

2.Decision Tree

We employed these techniques to improve the efficiency of our model and to extract the features, and the SVM algorithm provided improved accuracy.

Table 7. Accuracy for machine learning algorithm

ALGORITHM	ACCURACY(%)
DECISION TREE	99.3
SVM	100

Figure 14. Confusion matrix for hybrid model

Table 8. Calculation of different parameters for hybrid model

Parameters	Score
Testing Accuracy	91
Precision	91
F1 Score	91
Recall	91

VI. CONCLUSION

When abnormal cells develop within the brain, a tumor is created. Headaches, nausea, and balance issues are some of the indications and symptoms that the tumor can produce as it spreads because it puts pressure on the surrounding brain tissue and alters how it functions. The ability of computer-aided tools to make an early diagnosis of this condition is crucial. One of the CNN-based designs, the Resnet18 model, served as the foundation for the method we enhanced in this research. With the Hybrid model we have created, the accuracy rate of 95% has grown to 100% due to the layers that have been added and deleted from the Resnet18 architecture.
This rate has a high success rate when compared to past studies. Nevertheless, it is thought to be a high-performance study.

VII. REFERENCES

[1] e. a. Hashemzehi R, "Detection of brain tumors from MRI images base on deep learning using htbrid model CNN and NADE," *Biocybern biomed Eng*, 2020.

[2] s. Chaplot, L. Patnaik and N. Jagannathan, "Classification of magenetic resonanace brain images using wavelets as input to support vector machine and neural network,," *biomed,signal processed control*, vol. 1, pp. 86-92, 2006.

[3] S. Y. F.P.Polly, "Detection and classification Of HGG and LGG brain tumor using machine learning," *International conference on Information Networking* , pp. 813-817, 2018.

[4] N. D.R, N. Padhy, P. Mallick, D. Bagal and Kumar, "Brain tumor classification using noble deep learning approach with parametric optimization through metaheuristic approaches," *computers*, 2022.

[5] S. C.Narmatha, "A hybrid fuzzybrain-storm optimization algorithm for thr classification of brain tumor MRI images," *Journal of Ambient Intelligence and humanized computing*, vol. 20, pp. 1-9, 2020.

[6] k. k. a. s. G. sharma, "Brain tumor detection based on machine learning," *International journal of computer applications*, vol. 103.1, 2014.

[7] d. g. k. p. k. ,. H. a. u. shilpa rani, "Efficient 3D Alexnet Architecture for object Recognization using syntactic patterns from medical images," *Computational Intelligence and Neuroscience* , vol. 10.1155, p. 19, 2022.

[8] z. a. M. S. Islam, "An efiicient method to classify brain tumor using CNN Architecture and SVM," *ICREST*, vol. 10.1109, pp. 644-648, 2021.

[9] "MRI Based brain tumor image detectation using CNN based deep learning method," *Neuroscience Informatics,* vol. 2, no. 4, 2022.

[10] "DEEP HYBRID LEARNING - a fusion of conventional ML with state of the art DL," [Online].

[11] S. S. P. G. F. I. A. S. I. M. M. K. S. B. S. A. I. H. A. Tahia Tazin, ""A Robust and Novel Approach for Brain Tumor Classification Using Convolutional Neural Network"," *CIN,* p. 11, 2021.

[12] M. S. &. S. A. S.] AL-Huseiny, " Transfer learning with GoogLeNet for detection of lung cancer.," *Indonesian Journal of Electrical Engineering and Computer Science,*, vol. 22, no. 2, pp. 1078-1086, 2021.

[13] S. X. X. C. Shui-Hua Wang, *Front. Psychiatry,*, 11 April 2019.

[14] [Online]. Available: www.mathworks.com/help/deeplearning/ref/resnet18.

[15] "Rectified Linear Unit (ReLU) layer - MATLAB - MathWorks India," [Online].

[16] "ReLu Definition | DeepAI," [Online].

[17] "2-D convolutional layer - MATLAB - MathWorks India," [Online].

[18] [Online]. Available: https://www.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.batchnormalizationlayer.html.

[19] [Online]. Available: https://www.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.dropoutlayer.html.

[20] "machinelearniy," [Online]. Available: https://https://machinelearniy.com/dropout-for-regularizing-deep-neural-networks/.

[21] "opengenius," [Online]. Available: https://iq.opengenus.org/fully-connected-layer/.

[22] "deeplizard," [Online]. Available: https://deeplizard.com/learn/video/ZjM_XQa5s6s.

[23] "deepai," [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/softmax-layer.

[24] "layers," matlab2022, [Online]. Available: https://www.mathworks.com/help/deeplearning/ref/classificationlayer.htm.

[25] "medium," [Online]. Available: https://medium.com/@gopi_/learning-rates-for-neural-networks-d359dabba4d6.

[26] "techtarget," [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/dropout.

[27] "medium," [Online]. Available: https://medium.com/mlearning-ai/optimizers-in-deep-learning-7bf81fed78a0.

[28] "deepai adam," [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/adam-machine-learning.

[29] "course css," [Online]. Available: https://course.ccs.neu.edu/cs5100f11/resources/jakkula.pdf.

[30] "towards science direct," [Online]. Available: https://towardsdatascience.com/understanding-decision-tree-classifier-7366224e033b.

[31] m. y. A. cinar, "Classification of Alzheimer's Disease MRI Images with CNN Based Hybrid Method," *IIETA,* vol. 25, no. 4, pp. 413-418, 2020.

[32] "MEDIUM," [Online]. Available: https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd.