

An Efficient M L Approach to Detect Brain Tumor Using MRI Images

Md. Arafat Muktadir, A.S.M Rahmat Ullah, Md. Jubayer Islam, Emdad Hossain, Tanjila Akter Munny, Syed Muaz Ali, and Md. Ashraful Alam

Department of Computer Science and Engineering, BRAC University, Dhaka, Bangladesh

Email- md.arafat.muktadir@g.bracu.ac.bd, a.s.m.rahmat.ullah@g.bracu.ac.bd, md.jubayer.islam@g.bracu.ac.bd, emdad.hossain@g.bracu.ac.bd, tanjila.akter.munny@g.bracu.ac.bd, syed.muaz.ali@g.bracu.ac.bd, ashraful.alam@bracu.ac.bd

Abstract—Brain tumors have become one of the most leading causes of death worldwide. Early diagnosis of brain tumor is really important as it spreads quickly. One of the most challenging task in medical field is tumor detection and classification. Medical data has revealed that manual tumor detection and classification with human assistance can result in an incorrect prediction and diagnosis. Recently, combining machine learning with deep learning algorithms have shown promising outcomes for enhancing the accuracy and efficiency of brain tumor detection and classification from MRI images. In our paper our main target is to find better accuracy as well as more efficient way to detect and classify brain tumor from the MRI images using Image Processing, Machine Learning and Deep Learning.

Index Terms—Brain Tumor, Machine Learning, Deep Learning, Image Processing.

I. INTRODUCTION

Brain Tumor is one of the most serious, widespread, and potentially deadly diseases around the world. A growth of abnormal brain cells is known as a brain tumor. Brain tumors come in a wide variety of forms. It can be benign (noncancerous) or malignant (cancerous). Symptoms of brain tumors are not always observable. In fact, meningioma, the most frequent brain tumor in adults, frequently progresses undetected because of its slow growth. Until a tumor is large enough to interfere with healthy brain tissue, symptoms might not appear. Magnetic resonance imaging (MRI) is generally the first step in the detection of a brain tumor as MRI pictures themselves have great resolution and can clearly depict the structure, size, and location of brain tumors[1]. In numerous scientific studies, machine learning and deep learning algorithms are used to detect brain tumors. It takes relatively little time to identify a brain tumor when these algorithms are applied to MRI scans, and the improved accuracy simplifies patient care[2]. CNN (Convolutional Neural Network), is a type of artificial neural network that is used in deep learning and are usually utilized for object recognition and classifications[3]. Deep learning have the potential to detect brain tumors since they can provide a visual depiction of the tumor's region. According to the tumor region, it will be easy to start treatment as soon as possible, and hopefully it will be feasible to lower the number of deaths caused by brain tumors[4].

This research aims for achieving better accuracy as well as a more efficient way of the detection and classification of

brain tumor from MRI brain images through image processing and combining machine learning and deep learning. Here we have given our future expectations that can be done from our work: Early detections will be possible that can improve the chances that the patient will recover after treatment, doctors can determine from classified images that in which region the tumor has grown and our work will ensure more accurate and efficient tumor discoveries.

II. LITERATURE REVIEW

Chattopadhyay, Maitra[5] proposed a methodology that consists of some customizations and some new operations along with the traditional way. To elaborate, they used a 9 layer CNN model consisting of 14 stages and hidden layers. Moreover, they created a convolutional kernel convoluted with input layer administering with 32 convolutional filters of size 2×2 with the support of 3 tensor channels. Also, a 2×2 Max pooling operation for the CNN has been used. Now, they again used convolutional filters and this time they increased them to 64. Moreover, they used flattening and 2 dense layers. The first dense layer has 512 hidden layers and the second dense layer has the final 2 layers. A note to be taken is that they also tried SVM and Sigmoid, but Softmax gave them better results. They showed that Sigmoid gave them 97.63% accuracy, 58.33% testing accuracy and evaluation of the model was 68.72% whereas Softmax gave them 99.74% accuracy, 93.78% testing accuracy and evaluation of the model was 97.71% when RMSProp was used as an extension of optimization algorithm. For AdaMax as an extension of the optimization algorithm, Softmax gave them 98.10% accuracy, 75% testing accuracy and evaluation of the model was 82.40%. Here, they used 2473 training images and 273 testing images.

Sajjad, Muhammad et al.[6] proposed an interesting methodology for multigrade tumor classification. To elaborate, they introduced a novel deep learning framework that can classify brain tumors into four different grades. This system consists of three main steps. The first step is tumor segmentation. Here, they used a fully automated deep learning based technique called InputCascadeCNN that uses two-way processing of images. The second step is data augmentation. Here, due to lack of satisfactory amounts of data, they extended their data through 30 parameters and 8 augmentation techniques

(transformational and noise invariance techniques). Thus, each sample of the dataset extended into 30 samples. The third step is delivering the images into their CNN architecture. Here, they used VGG-19, a pretrained model, where they changed only the weights of fully connected layers for fine-tuning the model that classifies brain tumors into four different grades. They achieved an overall accuracy of 90.67%.

Brindha, Kavinraj et al.[7] detected brain tumor using deep learning implementing through two techniques. They are ANN and CNN. Here, the ANN model has seven layers. The first layer is the flatten layer and dense layer has been used as the next five layers. The last layer is the output layer, which is also a dense layer. To elaborate about the layers, there are different layers that got applied on the input image with the ReLU an activation function. Now, when the ANN model is used on a testing dataset, it gave 65.21% accuracy, whereas CNN gave 89%. Also, for the same class of dataset, precision is 65% for ANN and 89% for CNN. This dictates that CNN technique is much better compared to ANN technique. The strength of this paper is that the researchers used both ANN and CNN techniques to build the models and showed which technique is better. The weakness of this paper is the self defined CNN model not being 100% accurate. Here accuracy can be improved by giving more image dataset, applying image augmentation techniques and improving self defined CNN model.

Ankita[8] proposed a framework that basically has five modules. Dataset, pre-processing, splitting the dataset, construct CNN and classification. In the dataset, they take numerous MRI pictures and take one as an input picture. In pre-processing, they encode the label and resize the picture. In splitting of the dataset, they set the picture as 80% training dataset and 20% testing dataset. Here, the execution of the proposed Profound CNN demonstration has been compared with prevalent exchange learning approach. The input of the 1 convolution layer is of settled measure 224×224 RGB picture. The strength of this paper is that the researchers used VGG-16 and other CNN techniques to build the models and showed which technique is better. The weakness of this paper is the CNN & VGG-16 model not being 100% accurate.

Gumaei, Hassan et al.[9] proposed a method that consists three steps for classification of brain tumors. First, conversion of intensity values from brain pictures are done through preprocessing. The most crucial characteristics are then retrieved using PCA-NGIST, an impressive brand-new hybrid technique. Then, the RLEM classifier is used to categorize brain cancers. This dataset includes 3064 brain scans. These are collected from 233 people who have three different forms of brain tumors. Holdout (70% training, 30% testing) and 5-folds cross validation procedures are used in the trials. The study found that the PCA-NGIST feature extraction method is received more accuracy than PCA-GIST, GIST, and NGIST methods. The proposed approach had an accuracy range of 91.51%–94.233%. We want to learn from the suggested technique in our research to resolve another biological classification issue.

III. RESEARCH METHODOLOGY

A. Proposed Workflow

Our proposed workflow is inspired of the work done by [10] where they have acquired the predictions from 13 CNN models. After that, they extracted features from those 13 CNN models and loaded them into 9 ML classifiers to evaluate the extracted features. Then they concatenated the extracted features and loaded into those 9 ML to get final predictions. One thing to note here is that they have used different dataset.

So, our proposed work plan consists of eight parts. The first part is data acquisition. The second part is data pre-processing. The third part is data splitting. The fourth part is training the four convolutional neural network models. The fifth part is testing the four convolutional neural network models and acquiring the prediction results of each model. Furthermore, the sixth part is taking the average of probability of each class from every model and acquire the accuracy. The seventh part is combining the extracted features and loading those along with the labels from the training and validation datasets into the KNN algorithm for training. Lastly, the eighth part is combining the extracted features from the testing dataset and loading those for the prediction. We have given Fig[1], Fig[2], Fig[3] for better understanding of the proposed workflow.

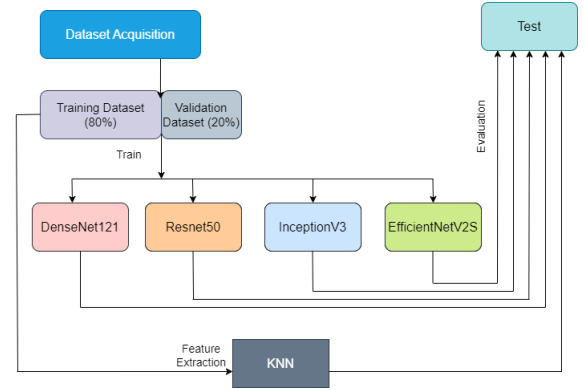


Fig. 1. Work Procedure

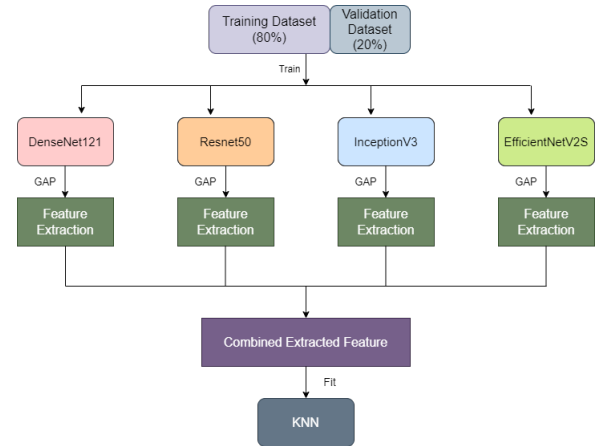


Fig. 2. Work Procedure of Feature Extraction From Training & Validation Dataset

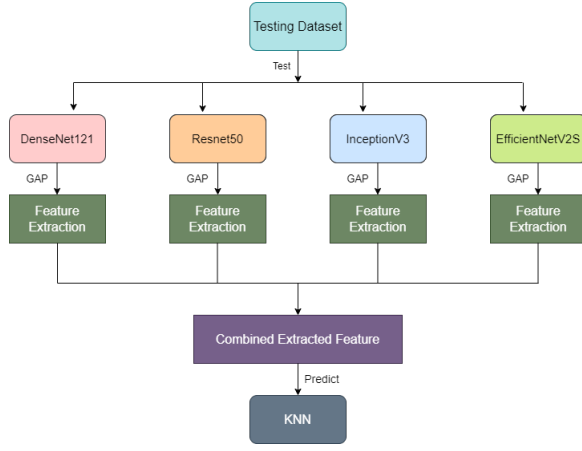


Fig. 3. Work Procedure of Feature Extraction From Testing Dataset

B. Dataset

1) *Dataset Details*: This dataset contains 7022 images of human brain MRI images which are classified into 4 classes: Glioma, Meningioma, Notumor, Pituitary. Meningiomas, which are typically benign tumors that arise from arachnoid cap cells, account for (13-26) percent of all cerebral tumors [11]. Gliomas account for over 30 percent of all primary brain tumors and 80 percent of all malignant brain tumors[12]. A Pituitary tumor, also known as a Pituitary adenoma, is a tumor that develops in the Pituitary gland[13]. The dataset is divided into two parts. One part is training which contains 5712 images. Here, class Glioma has 1321 images, class Meningioma has 1339 images, class Notumor has 1595 images and class Pituitary has 1457 images. Now, another part is testing which contains 1311 images. Here, class Glioma has 300 images, class Meningioma has 306 images, class Notumor has 405 images and class Pituitary has 300 images.

2) *Dataset Preprocessing*: In this part, we have set the shape of the all the images to $224 \times 224 \times 3$, we have normalized the images to have values between 0 and 1 by dividing with 255. Also, we have done the splitting. Here, the dataset was already in two parts. One is training data and another one is testing data. Now, we have split the training dataset into two parts. One is the training dataset, which is 80% of the main training dataset. Another in the validation dataset, which is 20% of the main training dataset.

3) *Dataset Augmentation*: We have used some augmentation techniques for the dataset. To elaborate, we have used horizontal flipping randomly, we have used rotation randomly from 0 to 40 degree, we have used width shift as well as height shift randomly from 0 to 20%, zooming randomly from 0 to 20% and lastly shearing randomly from 0 to 20%.

C. Convolutional Neural Network Architectures

Firstly, we have used the ResNet50 [14] architecture. ResNet50 is a deep convolutional neural network architecture introduced in 2015 by Microsoft Research. It belongs to the ResNet50 family and is widely used in computer vision tasks like classifying images, detecting objects, and image segmentation[15]. Secondly, we tried the EfficientNetV2S [16]

architecture. EfficientNetV2S is an advanced deep neural network architecture that improves efficiency and performance for computer vision tasks. The “S” variant has fewer parameters and is lighter, making it suitable for resource-constrained scenarios. EfficientNetV2S achieves state-of-the-art performance on benchmarks like ImageNet, making it ideal for real-time inference or environments with limited resources [17]. Thirdly, the DenseNet121 [18] architecture was used. DenseNet121 is a deep neural network architecture known for its efficient parameter usage and dense connectivity in image classification tasks. DenseNet121 achieves excellent performance with fewer parameters due to its parameter efficiency [19]. Lastly, we have used the InceptionV3 [20] architecture. InceptionV3 is a convolutional neural network architecture known for its effectiveness in object detection and image classification tasks. It incorporates dimensionality reduction techniques, like 1×1 convolutions, to reduce computational complexity [21].

D. Ensembling Predictions Through Mean

In this strategy, we have used averaging of the prediction from different convolutional neural network models to get a better prediction. To elaborate, every convolutional neural network model we have used has a dense layer as the output layer. For this experiment, this dense layer has four neural units as the final output layer. Here, we get a 1D matrix of dimension depending on the classes and the shape of the matrix is 1×4 . Now, every model gives us a prediction for every class of a number between 0 and 1 as we have used softmax as the activation function. Now, we have calculated the mean of the prediction of every model for each class and get the final prediction.

E. Combining Extracted Features and Using the KNN Algorithm

We have used The K-nearest neighbors (KNN) algorithm to find better accuracy. It is a form of supervised machine learning algorithm that can be used for both classification and regression tasks. To elaborate our strategy, we have combined the extracted features from four different convolutional neural network models and fitted them in the KNN algorithm. Also, we have added the validation dataset to the training dataset, which gave us a new dataset to work with. After that, we have removed the dense layer from the convolutional neural network models. This was done to remove the predictions of the four classes and to obtain the extracted features from the global average pooling (GAP) layer of the convolutional neural network models. This global average pooling layer gives us the extracted features as an output of an array of 1D that we can use with the KNN algorithm. So, ResNet50, EfficientNetV2S, DenseNet121, InceptionV3, the four convolutional neural network models gave us the extracted features of an output of shape $1 \times 1 \times 2048$, $1 \times 1 \times 1280$, $1 \times 1 \times 1024$, $1 \times 1 \times 2048$ for every image. After that, we have concatenated them. To elaborate, we have combined the extracted features that we got from the four models and the output size became $1 \times 1 \times 6400$. We have used these in the KNN algorithm to find better accuracy. So, we have fitted the combination of extracted features in

KNN algorithm along with their labels. Also, we have set the neighbor to 1. Now for the prediction part, we have loaded the testing dataset into the four convolutional neural networks, and they gave us the extracted features. Now, we have combined the extracted features and fitted into the KNN algorithm for prediction.

IV. RESULT AND ANALYSIS

Here, we have trained 4 convolutional neural network models. Furthermore, we have tested them and acquired the accuracy, precision, recall, F1 score and confusion matrix of each model. After that, we have predicted the mean of the probability of each class of every model and calculated an accuracy. Lastly, we have used KNN algorithm for prediction through extracted features to search for better accuracy, precision, recall, F1 score and confusion matrix.

A. Testing With ResNet50

We have started our testing with ResNet50. Here, we have imported weights from a convolutional neural network model trained on the ImageNet dataset. Also, we have set the batch size to 32 and learning rate to $1e-5$. Moreover, we have run 50 epochs for the convolutional neural network model. We have used Keras checkpoint for recording the best validation accuracy during the training period through every epoch. Here, the best validation accuracy the model gave us is 0.9668 on epoch 49. Now, we have saved the model that gave us best validation accuracy for testing purpose. Finally, we move to the testing period. We have encoded the labels. Here, 0 represents Glioma, 1 represents Meningioma, 2 represents Notumor and 3 represents Pituitary. Now, after the testing the model on the testing dataset, the model gave us an accuracy of 97.55%. We have given table[I] showing the precision, recall and F1-score for every class.

TABLE I
THE PRECISION, RECALL AND F1-SCORE OF RESNET50

Class	Precision	Recall	F1-score
Glioma	0.94	1.00	0.97
Meningioma	0.98	0.93	0.96
Notumor	0.99	0.99	0.99
Pituitary	0.99	0.98	0.98

B. Testing With EfficientNetV2S

After ResNet50, we have tested with EfficientNetV2S. Here, we have imported weights from a convolutional neural network model trained on the ImageNet dataset. Also, we have set the batch size to 32 and learning rate to $1e-5$. Moreover, we have run 50 epochs for the convolutional neural network model. We have used Keras checkpoint for recording the best validation accuracy during the training period through every epoch. Here, the best validation accuracy the model gave us is 0.9633 on epoch 48. Now, we have saved the model that gave us best validation accuracy for testing purpose. Finally, we move to the testing period. We have encoded the labels. Here, 0 represents Glioma, 1 represents Meningioma, 2 represents

Notumor and 3 represents Pituitary. Now, after the testing the model on the testing dataset, the model gave us an accuracy of 97.10%. We have given table[II] showing the precision, recall and F1-score for every class.

TABLE II
THE PRECISION, RECALL AND F1-SCORE OF EFFICIENTNETV2S

Class	Precision	Recall	F1-score
Glioma	0.98	0.97	0.97
Meningioma	0.98	0.91	0.94
Notumor	0.98	1.00	0.99
Pituitary	0.95	1.00	0.97

C. Testing With InceptionV3

After EfficientNetV2S, we have tested with InceptionV3. Here, we have imported weights from a model trained on the ImageNet dataset. Also, we have set the batch size to 32 and learning rate to $1e-5$. Moreover, we have run 50 epochs for the model. We have used Keras checkpoint for recording the best validation accuracy during the training period through every epoch. Here, the best validation accuracy the model gave us is 0.9650 on epoch 45. Now, we have saved the model that gave us best validation accuracy for testing purpose. Finally, we move to the testing period. We have encoded the labels. Here, 0 represents Glioma, 1 represents Meningioma, 2 represents Notumor and 3 represents Pituitary. Now, after the testing the model on the testing dataset, the model gave us an accuracy of 96.41%. We have given table[III] showing the precision, recall and F1-score for every class.

TABLE III
THE PRECISION, RECALL AND F1-SCORE OF INCEPTIONV3

Class	Precision	Recall	F1-score
Glioma	0.99	0.93	0.96
Meningioma	0.95	0.92	0.93
Notumor	0.98	1.00	0.99
Pituitary	0.93	1.00	0.96

D. Testing With DenseNet121

After InceptionV3, we have tested with DenseNet121. Here, we have imported weights from a model trained on the ImageNet dataset. Also, we have set the batch size to 32 and learning rate to $1e-5$. Moreover, we have run 50 epochs for the model. We have used Keras checkpoint for recording the best validation accuracy during the training period through every epoch. Here, the best validation accuracy the model gave us is 0.9624 on epoch 47. Now, we have saved the model that gave us best validation accuracy for testing purpose. Finally, we move to the testing period. We have encoded the labels. Here, 0 represents Glioma, 1 represents Meningioma, 2 represents Notumor and 3 represents Pituitary. Now, after the testing the model on the testing dataset, the model gave us an accuracy of 97.33%. We have given table[IV] showing the precision, recall and F1-score for every class.

TABLE IV
THE PRECISION, RECALL AND F1-SCORE OF DENSENET121

Class	Precision	Recall	F1-score
Glioma	0.98	0.97	0.97
Meningioma	0.97	0.92	0.95
Notumor	0.99	1.00	0.99
Pituitary	0.95	0.99	0.97

E. Hyperparameter Values of All Deep Learning Models

We have used the same hyperparameter values for all 4 CNN models that we have used. Here, we have shown them in table[V].

TABLE V
HYPERPARAMETER VALUES

Hyperparameters	Values
Batch Size	32
Learning Rate	1e-5
Epoch	50
Loss Function	Categorical Cross Entropy
Optimizer	Adam
Activation Function	Softmax

F. Ensembling Predictions Through Mean

This strategy gave us an accuracy of 98.32%. We have incorporated table[VI] below showing the precision, recall and F1-score of every class.

TABLE VI
AFTER ENSEMBLING THE PRECISION, RECALL AND F1-SCORE

Class	Precision	Recall	F1-score
Glioma	0.99	0.98	0.99
Meningioma	0.98	0.95	0.97
Notumor	0.99	1.00	0.99
Pituitary	0.97	1.00	0.98

G. Combined Extracted Features Fitted in KNN

We have extracted features and combined them. Then, we have fitted them into the KNN algorithm. It gave us an accuracy of 99.38% which is the highest accuracy we have achieved. We have incorporated table[VII] showing precision, recall and F1-score.

TABLE VII
THE PRECISION, RECALL AND F1-SCORE OF KNN

Class	Precision	Recall	F1-score
Glioma	0.99	1.00	0.99
Meningioma	0.99	0.98	0.99
Notumor	1.00	1.00	1.00
Pituitary	0.99	0.99	0.99

H. Comparison Between Results

Here, we have incorporated Fig[4] for comparison of training and validation accuracy curves and Fig[5] for comparison of confusion matrix of different models and algorithms as well

as we have shown table[VIII] below for the same purpose. We can see that the best accuracy we have achieved is 99.33% which is predicted from the KNN algorithm. Moreover, the lowest accuracy we have got is 96.41% which is predicted from InceptionV3. Furthermore, our best performing model is 97.55% which is predicted from ResNet50. Also, we have achieved an accuracy of 98.32% from mean prediction ensembling which is the second-best accuracy we have got.

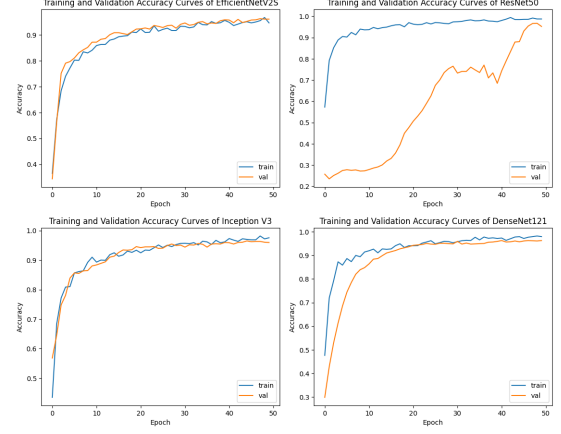


Fig. 4. Comparison of training and validation accuracy curves

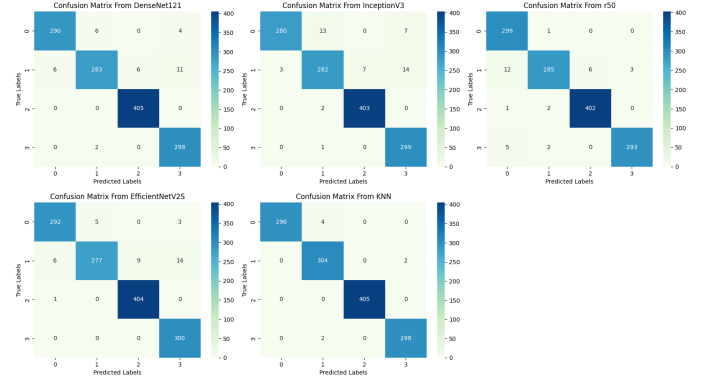


Fig. 5. Comparison of confusion matrix

TABLE VIII
COMPARISON OF ACCURACY, PRECISION, RECALL, F1-SCORE

Models	Accuracy	Precision	Recall	F1-score
ResNet50	97.55%	0.98	0.98	0.98
EfficientNetV2S	97.10%	0.97	0.97	0.97
InceptionV3	96.41%	0.96	0.96	0.96
DenseNet121	97.33%	0.97	0.97	0.97
Ensemble prediction	98.32%	0.98	0.98	0.98
KNN	99.38%	0.99	0.99	0.99

I. Comparison With Other Papers

Here, we have shown a comparison table[IX] that shows us a comparison of accuracy between our work and other's work. Here, every work that we have shared in the table used the same dataset that we have used, but their working procedure is different from ours. Now we can see that, our accuracy is the best among other's.

TABLE IX
COMPARISON WITH OTHER PAPERS

Paper	Accuracy
Mingzhe Hu,2023[22]	94.89%
Antonius Fajar Adinegoro,2023[23]	95%
Md Harun or Rashid,2022[24]	98.43%
Natalija Ivoš,2022[25]	97%
Our work	99.38%

V. CONCLUSION

The goal of this study was to examine the problem of a major, life-threatening medical emergency and see how combining machine learning and deep learning in this area could help healthcare professionals make more accurate and efficient clinical diagnosis. We have applied 4 convolutional neural network models to classify brain tumors into Glioma, Meningioma, Pituitary and Notumor classes using MRI dataset. We tried to find better accuracy through pre-processing, training, validating, testing, combining extracted features, averaging probabilities, using KNN machine learning algorithm. We have achieved 99.38% accuracy, and we aim to find better accuracy in the future works.

REFERENCES

- [1] H. Mohsen, *Classification using deep learning neural networks for brain tumors*. 2018.
- [2] J. A. K. N.Ullah, *An effective approach to detect and identify brain tumors using transfer learning*. 2022.
- [3] Ö. F. Brain tumor detection based on convolutional neural network with neutrosophic expert maximum fuzzy sure entropy. 2019.
- [4] Akbani, *Brain tumor detection using deep learning*. 2022.
- [5] M. M. Arkapravo Chattopadhyay, *Mri-based brain tumour image detection using cnn based deep learning method*. 2022.
- [6] S. K. M.Sajjad, *Multi-grade brain tumor classification using deep cnn with extensive data augmentation*. 2019.
- [7] P. M. P Gokila Brindha M Kavinraj and P. Prasanth, *Brain tumor detection from mri images using deep learning techniques*. 2021.
- [8] P. P. Ankita Parmar Mehfuza Holia, *A survey on brain tumor detection using deep learning*. 2020.
- [9] M. A.Gumaei, *A hybrid feature extraction method with regularized extreme learning machine for brain tumor classification*, 2019.
- [10] J. Kang, Z. Ullah, and J. Gwak, "Mri-based brain tumor classification using ensemble of deep features and machine learning classifiers," *Sensors*, vol. 21, no. 6, 2021, ISSN: 1424-8220. DOI: 10.3390/s21062222. [Online]. Available: <https://www.mdpi.com/1424-8220/21/6/2222>.
- [11] C. Marosi, M. Hassler, K. Roessler, et al., "Meningioma," *Critical Reviews in Oncology/Hematology*, vol. 67, no. 2, pp. 153–171, 2008, ISSN: 1040-8428. DOI: <https://doi.org/10.1016/j.critrevonc.2008.01.010>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S104084280800022X>.
- [12] A. Zbiciak and T. Markiewicz, "A new extraordinary means of appeal in the polish criminal procedure: The basic principles of a fair trial and a complaint against a cassatory judgment," en, *Access to Justice in Eastern Europe*, vol. 6, no. 2, pp. 1–18, Mar. 2023.
- [13] A. P. Amar and M. H. Weiss, *Pituitary anatomy and physiology*. 2003.
- [14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [15] A. Kaushik, *Understanding resnet50 architecture*, 2020.
- [16] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*, PMLR, 2021, pp. 10 096–10 106.
- [17] A. Sarkar, *EfficientNetV2—faster, smaller, and higher accuracy than Vision Transformers — towardsdata-science.com*, <https://towardsdatascience.com/efficientnetv2-faster-smaller-and-higher-accuracy-than-vision-transformers-98e23587bf04>, [Accessed 22-May-2023].
- [18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [19] *Architecture of DenseNet-121 — https*, <https://iq.opengenus.org/architecture-of-densenet121/>, [Accessed 22-May-2023].
- [20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
- [21] *Inception V3 Model Architecture — iq.opengenus.org*, <https://iq.opengenus.org/inception-v3-model-architecture/>, [Accessed 22-May-2023].
- [22] X. Yang, *End-to-end brain tumor detection using a graph-feature-based classifier — spiedigitallibrary.org*, <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/12468/124681C/End-to-end-brain-tumor-detection-using-a-graph-feature/10.1117/12.2654020.full?SSO=1>, [Accessed 22-May-2023].
- [23] A. F. A. a and G. N. S. a, *Classification and segmentation of brain tumor using efficientnet-b7 and u-net*, 2023.
- [24] *BrainNet-7: A CNN Model for Diagnosing Brain Tumors from MRI Images based on an Ablation Study - ProQuest — proquest.com*, <https://www.proquest.com/openview/44421bad128919eb3377cd5c312b9a31/1?pq-origsite=gscholar&cbl=5444811>, [Accessed 22-May-2023].
- [25] N. I. 1., *Application of convolutional neural networks for detection and classification of brain tumors*, 2022.