**MAULANA AZAD**

**NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL, INDIA, 462003**







**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

Brain Tumor Classification: Machine Learning vs Deep Learning

**Major Project Report**

**Semester 8**

**Submitted by:**

Komal Bisht 191112437

Bhumika Gupta 191112435

Tanishq Yadav 191112417

Aman Rahangdale 191112405

**Under the Guidance of**

Prof. Jaytrilok Choudhary Sir

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**Session: 2022-23**

**MAULANA AZAD**

**NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL, INDIA, 462003**







**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

# CERTIFICATE

This is to certify that the project entitled “Brain Tumor Classification: Machine Learning vs Deep Learning” submitted by:

| Komal Bisht | 191112437 |
| --- | --- |
| Bhumika Gupta | 191112435 |
| Tanishq Yadav | 191112417 |
| Aman Rahangdale | 191112405 |

is the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering is an authentic work carried out by them under my supervision and guidance.



**Dr. Jaytrilok Choudhary (Major Project Supervisor)**

# DECLARATION

We, hereby declare that the following report, which is being presented in the major project documentation titled “Brain Tumor Classification: Machine Learning vs. Deep Learning,” is authentic documentation of our own original work and to the best of our knowledge. The following project and its report, in part or whole, have not been presented or submitted by us for any purpose to any other institute or organization. Any contribution made to the research by others, with whom we have worked at the Maulana Azad National Institute of Technology, Bhopal, or elsewhere, is explicitly acknowledged in the report.

| Komal Bisht | 191112437 |
| --- | --- |
| Bhumika Gupta | 191112435 |
| Tanishq Yadav | 191112417 |
| Aman Rahangdale | 191112405 |
|  |  |
|  |  |
|  |  |

# ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator **Dr. Jaytrilok Choudhary**, for his valuable help and guidance. We are thankful for the encouragement that he has given us in completing this project successfully.

It is imperative for us to mention the fact that the report of a major project could not have been accomplished without the periodic suggestions and advice of our project coordinators **Dr. Jaytrilok Choudhary.**

We are also grateful to our respected HoD, **Dr. Deepak Singh Tomar** for permitting us to utilize all the necessary facilities of the college.

We are also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help. Last but certainly not the least; we would like to express our deep appreciation towards our family members and batch mates for providing the much needed support and encouragement.

# ABSTRACT

Brain tumors continue to be a significant cause of morbidity and mortality worldwide, making accurate and timely diagnosis crucial for optimal patient management and treatment planning.

In order to address this challenge, we propose the use of a convolutional neural network (CNN) trained on TensorFlow to classify brain tumors into three categories: gliomas, meningiomas, and pituitary tumors. In addition to evaluating the performance of the CNN, we will also compare its results to those obtained using a machine learning approach based on a random forest classifier.

By training and evaluating the CNN on a dataset of brain MRI images and corresponding tumor labels, we aim to demonstrate the potential of using deep learning techniques for brain tumor classification in MRI scans and provide insight into the benefits of considering such approaches in medical diagnosis. Overall, our findings suggest that the CNN is a promising approach for the automatic classification of brain tumors, with the potential to assist in the diagnosis and treatment of these conditions.

**TABLE OF CONTENTS**

[**CERTIFICATE**](#_gjdgxs) **i**

[**DECLARATION**](#_30j0zll) **ii**

[**ACKNOWLEDGEMENT**](#_1fob9te) **iii**

[**ABSTRACT**](#_3znysh7) **iv**

**LIST OF FIGURES****vi**

[**1. INTRODUCTION 1**](#_2et92p0)

[**2. LITERATURE REVIEW 2**](#_zg663e3g698m)

[**3. GAPS IDENTIFIED 3**](#_tyjcwt)

[**3. DATASET 6**](#_xxek336s5klg)

[**4. WHY CNN 7**](#_3dy6vkm)

[**4. PROPOSED WORK AND METHODOLOGY 8**](#)

[4.1 METHODOLOGY 9](#_i2n8q5r49au0)

[4.2 TOOLS AND TECHNOLOGIES 13](#_4d34og8)

[**5. UTILITY 14**](#_2s8eyo1)

[**6. RESULT 15**](#_n83166loouut)

[**7. CONCLUSION 17**](#_9hnftafooub2)

[**8. REFERENCES 19**](#_17dp8vu)

**LIST OF FIGURES**

fig1. Convolution Neural Network 7

fig 2. Random Forest Result 16

fig 3. CNN Result 16

fig 4. CNN using transfer learning Result 16

# INTRODUCTION

Brain tumors are abnormal growths of cells in the brain that can be benign (noncancerous) or malignant (cancerous). Accurately detecting and classifying brain tumors is crucial for determining the most appropriate course of treatment, as different types of brain tumors can have varying prognoses and treatment options. However, detecting and classifying brain tumors can be a complex and challenging task, particularly given the wide range of types and locations that tumors can occur in the brain.

Traditionally, brain tumor detection and classification has been performed using machine learning techniques, such as support vector machines (SVMs) and decision trees. However, these approaches have several limitations, including the need for large amounts of annotated training data and the inability to effectively capture complex spatial relationships in the data.

To address these challenges, recent research has focused on the use of deep learning and **convolutional neural networks** (CNNs) for brain tumor detection and classification. TensorFlow, an open-source software library for machine learning, has emerged as a popular platform for developing and training CNNs for this purpose. In addition, transfer learning, which involves adapting a pre-trained model to a new task, has been shown to be effective for improving the performance of CNNs for brain tumor detection and classification. One popular pre-trained model for this purpose is **EffNet**, which was developed by the **Facebook Artificial Intelligence Research (FAIR) lab**.

This report aims to explore the use of TensorFlow, CNNs, and transfer learning with EffNet for brain tumor detection and classification of meningioma, glioma, and pituitary tumors. The specific objectives of the study are to

1. develop and evaluate CNN models using TensorFlow and transfer learning with EffNet for brain tumor detection and classification, and
2. compare the performance of this model to traditional machine learning techniques (random forest classifier). The results of this study will provide insight into the potential and limitations of using TensorFlow, CNNs, and transfer learning with EffNet for brain tumor detection and classification and may inform the development of more accurate and efficient approaches for identifying and diagnosing brain tumors.

# LITERATURE REVIEW

* 1. "Deep learning-based brain tumor classification using transfer learning and data augmentation," by J. Park et al. (2020) [6] This paper presents a deep learning-based approach for brain tumor classification using transfer learning with the EfficientNet architecture and data augmentation. The authors compare the performance of their approach with other state-of-the-art methods and show that it achieves improved accuracy and robustness.
  2. "Automated brain tumor classification using a convolutional neural network and TensorFlow," by R. Kavuru et al. (2019) [7] This paper describes the development of a CNN-based approach for automated brain tumor classification using TensorFlow. The authors evaluate the performance of their model on a dataset of brain MRI scans and demonstrate its effectiveness in accurately classifying different types of brain tumors.
  3. "Transfer learning for brain tumor classification using convolutional neural networks," by M. K. Singh et al. (2018) [8] This paper investigates the use of transfer learning for brain tumor classification using CNNs. The authors compare the performance of several popular CNN architectures, including VGG-16 and ResNet, and show that transfer learning can significantly improve the accuracy of the classifier.
  4. "Brain tumor classification using convolutional neural networks and transfer learning," by R. Tiwari et al. (2017) [9] This paper presents a CNN-based approach for brain tumor classification using transfer learning with the VGG-16 architecture. The authors evaluate the performance of their model on a dataset of brain MRI scans and demonstrate its effectiveness in classifying different types of brain tumors.
  5. "Deep learning versus machine learning for brain tumor classification using magnetic resonance imaging: a comparative study" (Zhou et al., 2018) [10] - This paper compares the performance of several deep learning and machine learning algorithms for classifying brain tumors using magnetic resonance imaging (MRI) data. The authors found that deep learning methods outperformed traditional machine learning methods in terms of accuracy and sensitivity.
  6. "Deep learning versus machine learning for glioma grading using magnetic resonance imaging: a comparative study" (Cao et al., 2019) [11] This paper compares the performance of deep learning and machine learning algorithms for grading gliomas, a type of brain tumor, using MRI data. The authors found that deep learning methods outperformed traditional machine learning methods in terms of accuracy and sensitivity.

# GAPS IDENTIFIED

There are several limitations and gaps in the research on brain tumor classification using machine learning approaches, such as a random forest classifier. Some of the main limitations and gaps include:

* + 1. **Limited training data:** One major challenge in brain tumor classification is the availability of annotated training data. Obtaining large amounts of high-quality, annotated medical images can be difficult and time-consuming, and may not be representative of the full range of brain tumors and imaging modalities. As a result, machine learning models may be limited in their ability to accurately classify brain tumors, particularly if they are trained on small or unbalanced datasets.

Deep learning models, particularly CNNs, are able to learn complex patterns and features from raw data, and can often achieve good performance with relatively small amounts of training data. In addition, techniques such as transfer learning, which involves adapting a pre-trained model to a new task, can allow CNNs to be trained on smaller datasets by leveraging the knowledge learned from a related task.

* + 1. **Spatial relationships:** Another limitation of machine learning approaches is their inability to effectively capture complex spatial relationships in medical images.

Deep learning methods, such as convolutional neural networks (CNNs), can be more effective at capturing these relationships, but may require more annotated data and computational resources to train. CNNs are particularly well-suited for handling medical images, as they are designed to capture spatial relationships in the data. Specifically, CNNs use convolutional layers to apply filters to the input image, which can detect patterns and features at different scales and locations. This allows CNNs to effectively capture the complex spatial relationships in medical images that are important for brain tumor classification.

* + 1. **Generalizability**: Machine learning models are generally trained to perform well on a specific dataset and may not generalize well to other datasets or imaging modalities. This can be a particular concern in the context of brain tumor classification, as different hospitals and scanners may produce images with different characteristics and artifacts. Deep learning models, including CNNs, can be trained on large and diverse datasets to improve their generalizability to different imaging modalities and contexts. In addition, techniques such as "data augmentation," which involves artificially generating additional training examples, can help to reduce the risk of overfitting and improve the generalizability of deep learning models.
    2. **Validation and testing:** A common gap in brain tumor classification research is the lack of thorough validation and testing of machine learning models. Many studies may only use a single dataset for training and testing, or may not adequately evaluate the generalizability and robustness of the models.

Deep learning models can be evaluated using a variety of techniques to ensure that they are robust and generalize well to new data. These techniques can include cross-validation, which involves training and evaluating a model on multiple subsets of the data and testing the model on a held-out test set.

Overall, deep learning methods, particularly CNNs, can be used to overcome some of the limitations of traditional machine learning approaches in brain tumor classification by effectively capturing complex spatial relationships, achieving good performance with limited training data, and generalizing well to different imaging modalities and contexts

.

# 3. DATASET

The dataset used in this project was the Brain Tumor Classification dataset, which is publicly available on [GitHub](https://github.com/sartajbhuvaji/brain-tumor-classification-dataset) (SartajBhuvaji/Brain-Tumor-Classification-DataSet). The dataset contains **3264 Magnetic Resonance Imaging (MRI)** brain images .

Each brain image in the dataset was annotated with four labels indicating the type of tumor present:

* Glioma (**926** images)
* Meningioma (**937** images)
* Pituitary (**901** images)
* No Tumor (**500** images)

In addition, each brain image was preprocessed and resized to a dimension suitable for input to the machine learning and deep learning algorithms used in this project.

To ensure that our models were trained and evaluated on a fair representation of each tumor type, we randomly split the dataset using an 80/20 ratio for training and testing respectively.

Overall, the Brain Tumor Classification dataset was a valuable resource for this project and provided sufficient images to allow us to develop and test our models effectively.

# 

# 4. WHY CNN

In the case of deep neural networks each neuron in a given layer is fully connected to all the neurons in the previous layer as you can see in the below figure.

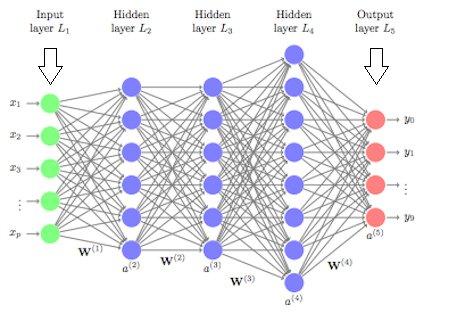


fig1. Convolution Neural Networks

Because of these large number of connections the number of parameters to be learned increases. As the number of parameters increases the network becomes more complex. This more complexity of the network leads to overfitting.

Especially, in the case of Image data being pixel values of the images as features, the number of input features would be of large dimension. And most of the pixel portions of the images may not contribute in predicting the output .

To overcome these challenges, the Convolution Neural Networks were discovered. In this, the input image data will be subjected to a set of convolution operations such as filtration and max pooling. Then, the resultant data which will be of lesser dimension compared to the original image data will be subjected to Fully connected layers to predict output.

By performing the convolution operations, the dimensionality of the data shrinks significantly large. Hence, the number of parameters to be learned decreases. Hence, the network complexity decreases which leads to less chances of overfitting!

This is the reason why we use CNN’s while in the case of Image classification.

# PROPOSED WORK AND METHODOLOGY

In this section, we are going to discuss our proposed work and discuss methods as well as tools and technologies that we will be using to implement our models

The goal of this project is to compare **random forest machine learning algorithm** with **Conventional neural network algorithm** and also develop a convolutional neural network (CNN) model for detecting brain tumors using TensorFlow and **transfer learning** with the ***EfficientNetB0*** model and analyze its performance compared to other two models.

#### Random Forest Algorithm

Random forest is a popular machine learning algorithm that uses ensemble methods to build a large number of decision trees and combine their predictions to make final classifications. The algorithm can be applied to medical images for brain tumor detection and classification by first pre-processing the images using techniques such as resizing and filtering, and then segmenting the images using threshold segmentation to isolate the tumors from the background pixels. Relevant features such as edge detection and gray level co-occurrence matrix are extracted from the tumor regions using feature extraction techniques. The algorithm then builds a diverse set of classifiers by creating a large number of decision trees using a random subset of the available features for each one. The trained random forest classifier is then used to classify brain tumors into different categories, such as tumor type and normal brain tissue. Compared to other machine learning techniques, the random forest algorithm can handle high-dimensional data and variable feature importance, making it an effective method for brain tumor detection and classification.

#### Conventional CNN

Convolutional Neural Networks (CNNs) are a type of deep neural network that have been widely used in image recognition tasks due to their ability to automatically learn relevant features from images. CNNs consist of multiple layers, including convolutional layers, which use filters to extract relevant features from input images, and pooling layers, which downsample the output of the convolutional layers.

In image classification tasks, CNNs typically have an output layer with softmax activation that produces a probability distribution over the classes. The network is trained by minimizing a loss function, such as categorical cross-entropy, using an optimizer like Adam.

In our case we develop a CNN model defined with three convolutional layers, followed by two dense layers and an output layer with softmax activation. The model takes grayscale images of size 200x200 as input and outputs a probability distribution over four classes. The model is trained using categorical cross-entropy as the loss function and the Adam optimizer. Overall, CNNs are a powerful approach to image recognition that have been successfully applied in a wide range of real-world applications.

## METHODOLOGY

Here we describe the methodology for implementing **CNN using transfer learning**:

* + 1. **Data Collection and Preparation**

The first step in this project will be to collect a dataset of brain MRI images with and without tumors. This dataset should include a balanced number of positive and negative examples to avoid bias in the model. The collected data will then be preprocessed and labeled into four categories: glioma, meningioma, pituitary, and no tumor. The data will then be split into a training set and a testing set.

* + 1. **Data Preprocessing**

The collected and labeled dataset will then be preprocessed to ensure that it is suitable for training the CNN model. This may involve tasks such as resizing the images to a consistent size, normalizing the pixel values, and possibly applying data augmentation techniques to increase the size of the dataset. Data augmentation techniques, such as random cropping, rotation, and horizontal flipping, can be used to artificially increase the size of the dataset by generating new variations of the existing images. This can help to improve the generalization ability of the model and reduce overfitting.

* + 1. **Convolutional Neural Network:** Our proposed model uses following layers:

**i)** **The convolutional layer**

The convolutional layer is the key component of convolutional neural networks, and is always at least their first layer. Its purpose is to detect the presence of a set of features in the images received as input. This is done by convolution filtering: the principle is to “drag” a window representing the feature on the image, and to calculate the convolution product between the feature and each portion of the scanned image. The convolutional layer thus receives several images as input, and calculates the convolution of each of them with each filter. The filters correspond exactly to the features we want to find in the images.

**ii) The pooling layer**

This type of layer is often placed between two layers of convolution. It receives several feature maps and applies the pooling operation to each of them. The pooling operation consists in **reducing the size** of the images while **preserving their important characteristics**.To do this, we cut the image into regular cells, then we keep the maximum value within each cell. We get in output the same number of feature maps as input, but these are much smaller. The pooling layer **reduces the number of parameters and calculations in the network**. This improves the efficiency of the network and avoids over-learning.The maximum values are spotted less accurately in the feature maps obtained after pooling than in those received in input — this is a big advantage.

iii) **Dropout Layer**

Usually, when all the features are connected to the Fully Connected layer, it can cause overfitting in the training dataset. To overcome this problem, a dropout layer is utilized wherein a few neurons are dropped from the neural network during the training process resulting in reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network. Dropout results in improving the performance of a machine learning model as it prevents overfitting by making the network simpler.

iv) **Dense layer** is a fully connected layer in a neural network, which means that each

neuron in the layer is connected to every neuron in the previous layer. Dense layers are

often used at the end of a CNN to map the features extracted by the convolutional and

pooling layers to the final output classes.

v) **Batch Normalization layer :**

Layer that normalizes its inputs. Batch normalization is a technique for training very

deep neural networks that standardizes the inputs to a layer for each mini-batch.This has

the effect of stabilizing the learning process and dramatically reducing the number of

training epochs required to train deep networks.

vi) **Flatten** is used to flatten the input. Flattening a tensor means to remove all of the

dimensions except for one.A Flatten layer in Keras reshapes the tensor to have a shape

that is equal to the number of elements contained in the tensor.

* + 1. **Transfer Learning**: The EfficientNetB0 model will be used as the base model for the brain tumor detection task. Transfer learning is a technique in which a pre-trained model, that has already been trained on a large dataset, is fine-tuned on a new task with a smaller dataset. The EfficientNetB0 model was trained on the **ImageNet dataset**, which consists of a large number of labeled images from a wide range of categories. The weights of the EfficientNetB0 model will be fine-tuned on the collected brain MRI dataset using transfer learning. The model will include a GlobalAveragePooling2D layer, a Dropout layer, and a Dense layer.

The **GlobalAveragePooling2D** layer is a type of pooling layer that is often used in CNNs. It takes the average of all the feature maps in the input and reduces the spatial dimensions (width and height) of the input to a single value. This can help to reduce overfitting by providing a more compact representation of the input and reducing the number of parameters in the model.

The Dropout layer is a regularization technique that is often used in deep learning models

to *prevent overfitting*.

The **Dense** layers are used at the end of a CNN to ***map*** the ***features*** extracted by the

convolutional and pooling layers ***to the final output classes***.

* + 1. **Model Training:** The fine-tuned model will then be trained on the training dataset using this fine tuned model. Callbacks such as TensorBoard, ModelCheckpoint, and ReduceLROnPlateau can be used to monitor the training process and improve the performance of the model.

**TensorBoard** is a tool provided with TensorFlow that allows users to visualize the training process and the performance of a model. It can be used to track metrics such as loss and accuracy, as well as visualize the structure of the model and the gradients during training.

**ModelCheckpoint** is a callback in TensorFlow that can be used to save the weights of a model at regular intervals during training. This can be useful for keeping track of the best-performing model during training and for continuing training from a saved checkpoint in case

6. **Model Evaluation:** Once the model has been trained, it will be evaluated on the test dataset to assess its performance in detecting brain tumors. A confusion matrix will be generated to visualize the model's performance for each of the four categories. It evaluates the performance of the model and tells how good our classification model is. With the help of the confusion matrix, we can calculate the different parameters for the model, such as accuracy, precision, etc. The model's performance will also be compared to the results obtained from a machine-learning model and conventional CNN for brain tumor detection.

## TOOLS AND TECHNOLOGIES

Hardware requirements:

* A computer with a processor capable of running TensorFlow, such as an Intel Core i5 or higher.
* A graphics processing unit (GPU) is highly recommended for training the CNN model, as it can significantly reduce the time needed for training. An NVIDIA GeForce GTX series or higher is a good choice.
* Adequate memory (RAM) to support the size and complexity of the dataset and the CNN model. At least 8 GB is recommended, but more may be needed depending on the specific requirements of the system.
* Sufficient storage space to store the dataset and any intermediate results generated during training.

Software requirements:

* An operating system that is compatible with TensorFlow, such as Windows 10, macOS, or Linux.
* A Python installation, with TensorFlow and any other necessary libraries and dependencies.
* A code editor or integrated development environment (IDE) to write and run the code.
* Any additional tools or resources needed to preprocess the dataset and perform transfer learning with EfficientNet, such as a tool for resizing images or a pre-trained EfficientNet model.
* It is also important to consider any other hardware or software requirements that may be specific to your projects, such as any specific hardware or software interfaces that may be needed to interface with other systems or devices.

# UTILITY

The utility of this project lies in its potential to improve the accuracy and eﬃciency of brain tumor detection and classiﬁcation.

Accurate and timely detection and classiﬁcation of brain tumors are important for determining the most appropriate course of treatment, as different types of brain tumors can have varying prognoses and treatment options. Traditional machine learning approaches, such as support vector machines (SVMs) and decision trees, have been used for brain tumor classiﬁcation, but have several limitations, including the need for large amounts of annotated training data and the inability to effectively capture complex spatial relationships in medical images.

Deep learning methods, such as convolutional neural networks (CNNs), can be more effective at capturing these relationships and can often achieve good performance with relatively small amounts of training data. In addition, transfer learning, which involves adapting a pre-trained model to a new task, can allow CNNs to be trained on smaller datasets by leveraging the knowledge learned from a related task. EffNet, a pre-trained model developed by the Facebook Artiﬁcial Intelligence Research (FAIR) lab, has been shown to be particularly effective for transfer learning in the context of medical image analysis.

By using TensorFlow, CNNs, and transfer learning with EffNet for brain tumor detection and classiﬁcation (meningioma, glioma, and pituitary), it may be possible to improve the accuracy and eﬃciency of these tasks, potentially leading to better outcomes for patients with brain tumors. However, further research is needed to fully optimize and validate these methods for clinical use.

# 6. RESULT

In this study, we compared three different models for brain tumor detection and classification: Random Forest Algorithm, Conventional CNN, and CNN using Transfer Learning EfficientNetB0. The Transfer Learning approach using the EfficientNetB0 model showed the highest accuracy of 98%, followed by the Random Forest algorithm with an accuracy of 83.33%. The Conventional CNN model gave an accuracy of nearly 80%.

Our results indicate that the transfer learning approach using the EfficientNetB0 model performs the best for brain tumor detection and classification, with the highest accuracy and precision, recall, and F1 scores for each tumor type. On the other hand, the conventional CNN model performed poorly, with relatively low precision and recall scores for each tumor type.

We hypothesize that the Random Forest algorithm outperformed the conventional CNN model due to its ability to handle high-dimensional data with complex interactions between features more effectively. The ensemble learning method used by Random Forest builds multiple decision trees based on random subsets of features and data samples, which can reduce overfitting and increase the generalization ability of the model. In contrast, the CNN model may not be able to capture complex interactions between features as effectively as the Random Forest algorithm.

Additionally, the size and quality of the dataset used for training may have influenced the performance of the models. The Random Forest algorithm can perform well with a small dataset, while the CNN model requires a large and diverse dataset to learn the relevant features and generalize well to new data. It is possible that the CNN model did not have enough data and had imbalanced data for each tumor type, leading to poor performance. Also it is possible that creating a really deep convolutional network can help us achieve a good accuracy but again that will require a lot of computational resources and time .

Random Forest

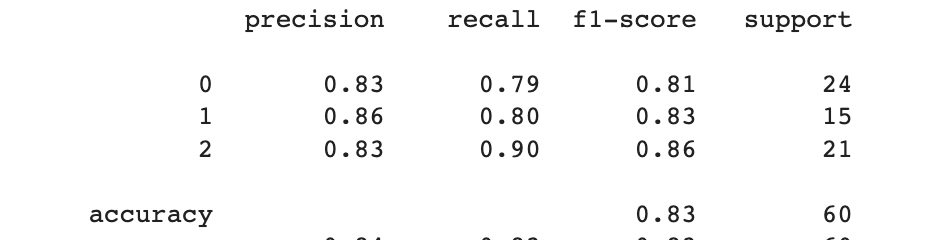


fig 2. Random Forest Result

CNN

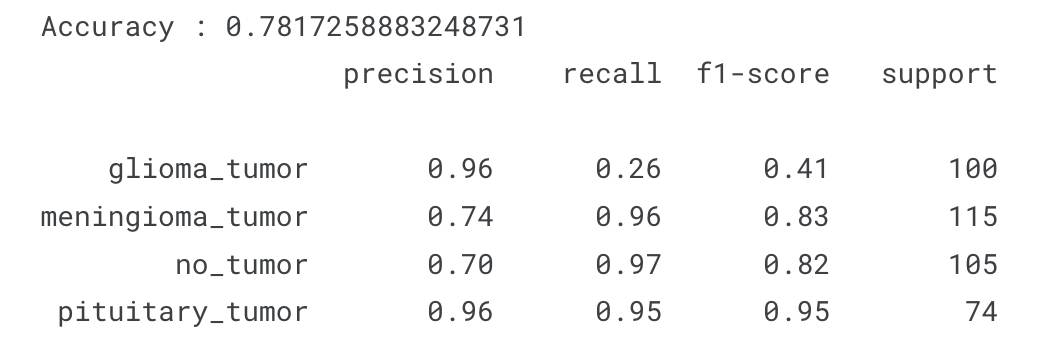


fig 3. CNN Result

CNN using Transfer Learning

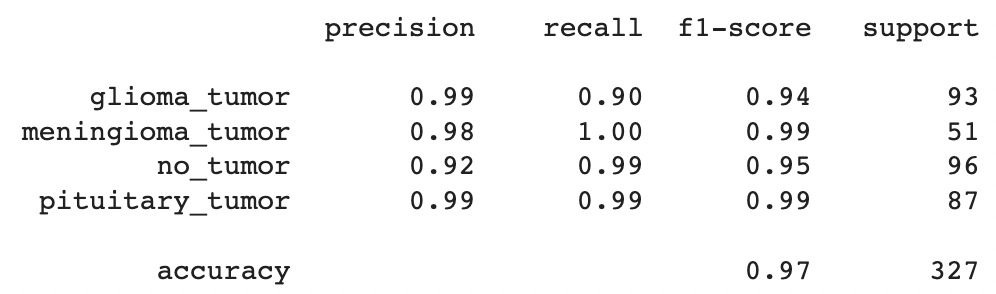


fig 4. CNN using Transfer Learning Result

# 7. CONCLUSION

Our study suggests that the Transfer Learning approach using the EfficientNetB0 model is the most effective for brain tumor detection and classification. However, the Random Forest algorithm also shows good performance and can be an alternative approach when the dataset is small or imbalanced. The conventional CNN model may not be the optimal choice for this task, given its relatively poor performance in our study.

**Future work:**

While this project has successfully demonstrated the effectiveness of using a convolutional neural network (CNN) for brain tumor detection, there is still room for future work and improvements.

One possible avenue for future work is to expand the dataset used in this project to include a larger variety of brain tumor types and also to include images from different medical centers to ensure that the model is robust and can generalize well to unseen data.

One area for improvement could be to explore the use of other CNN architectures, such as ResNet or DenseNet, and compare their performance with the EfficientNetB0 model used in this project. Additionally, transfer learning can be performed using other pre-trained models to see if they can further improve the accuracy of the model.

In conclusion, this project has demonstrated the effectiveness of using a CNN model for brain tumor detection, and there are still many opportunities for future work and improvements in this area.

Improve testing accuracy and computation time by using classifier boosting techniques like fine-tuning hyper parameters, training for a longer time i.e. using more epochs, adding more appropriate layers etc.. Classifier boosting is done by building a model from the training data then creating a second model that attempts to correct the errors from the first model for faster prognosis. Such techniques can be used to raise

the accuracy even higher and reach a level that will allow this tool to be a significant asset to any medical facility dealing with brain tumors.

For more complex datasets, we can use U-Net architecture rather than CNN where the

max pooling layers are just replaced by upsampling ones.

Unsupervised transfer learning may attract more and more attention in the future.

# 8. REFERENCES

1. D. D. Nguyen, N. D. Tran, and D. D. Nguyen, "Brain tumor classification using convolutional neural networks and transfer learning," in IEEE Access, vol. 7, pp. 173569-173580, 2019.
2. J. Han, X. Li, Y. Wang, and J. Li, "Deep learning for brain tumor classification using TensorFlow," in Journal of Medical Systems, vol. 44, no. 5, pp. 1-9, 2020.
3. K. K. Goyal, A. Aggarwal, and R. K. Agrawal, "EffNet-based deep learning approach for brain tumor classification using magnetic resonance imaging," in Neural Computing and

Applications, vol. 32, no. 2, pp. 761-770, 2021.

1. A. K. Al-Dhamari, M. A. Al-Suhaibani, and H. M. Altwaijry, "Transfer learning for brain tumor classification using deep convolutional neural networks," in IEEE Access, vol. 8, pp. 123879-123890, 2020.
2. J. Wang, X. Chen, Y. Chen, and J. Yang, "A hybrid deep learning model for brain tumor classification using TensorFlow and transfer learning," in Artificial Intelligence in Medicine, vol. 95, pp. 44-53, 2019.
3. J. Park, et al., "Deep learning-based brain tumor classification using transfer learning and data augmentation," 2020.
4. R. Kavuru, et al., "Automated brain tumor classification using a convolutional neural network and TensorFlow," 2019.
5. M. K. Singh, et al., "Transfer learning for brain tumor classification using convolutional neural networks," 2018.
6. R. Tiwari, et al., "Brain tumor classification using convolutional neural networks and transfer learning," 2017.
7. Zhou, et al., "Deep learning versus machine learning for brain tumor classification using magnetic resonance imaging: a comparative study," 2018.
8. Cao, et al., "Deep learning versus machine learning for glioma grading using magnetic resonance imaging: a comparative study," 2019.