

---

# Transfer Learning Approach for Brain Tumor Classification on Imbalanced MRI Dataset

---

**Raza Imam**  
MBZUAI  
Abu Dhabi, United Arab Emirates  
raza.imam@mbzuai.ac.ae

**Mohammed Talha Alam**  
MBZUAI  
Abu Dhabi, United Arab Emirates  
mohammed.alam@mbzuai.ac.ae

**Mohammed El-Amine Azz**  
MBZUAI  
Abu Dhabi, United Arab Emirates  
mohammed.azz@mbzuai.ac.ae

## 1 Introduction

Impactful solutions are being offered by the context-aware deployment of deep learning methodologies to enhance medical diagnostics. The World Health Organization (WHO) states that a correct diagnosis of a brain tumor entails its discovery, localization, and classification based on its degree, kind, and severity. This research comprises finding the tumor, grading it according to type and location, and classifying it according to grade in the diagnosis of brain tumors using magnetic resonance imaging (MRI). This approach has experimented with using several models rather than a single model for classification task in order to categorize brain MRI data [1]. Deep learning and Transfer learning models have been proposed with higher accuracies in recent literatures for classifying Brain Tumor MRI images. However most of such proposals are focused on balanced data. Hence, approaches to account for the imbalance in data, as well as focusing on the precise classification of brain cancer in real world scenarios, are needed. We introduce a 'Transfer Learning + CNN model' rather of just 'Transfer Learning model with Fine Tuning' and compared 8 of such Transfer Learning models using various approaches on imbalance MRI images[2].

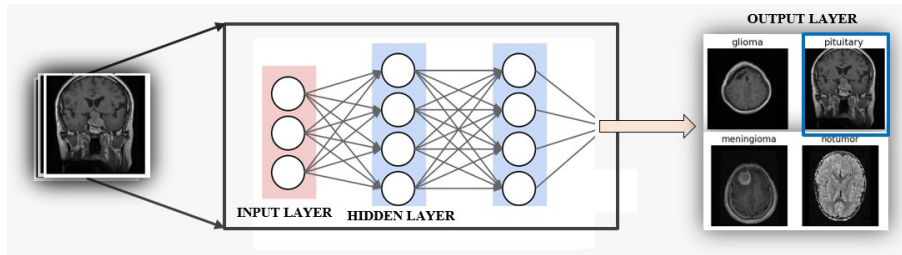


Figure 1: Classification of Brain Tumors from MRI images using CNN

Using three pathogenic forms of brain tumor (glioma, meningioma, and pituitary tumor), we aim to propose an accurate and automated classification scheme in this work. Towards this classification task, our goal is to empirically assess how well the most common benchmark models perform. Our goal of offering the finest classification model will open up new research directions in terms of choosing the right model for real-world brain tumor classification deployment. Utilizing tested models, the acquired characteristics are classified. The approach of our experiments that we will be acquiring is also represented in Figure 1. Subsequently, a thorough assessment of the suggested system is performed utilizing several effective evaluation metrics of classification tasks along with comparing several analytical factors, such as how well each model performs with less training samples from practicality aspect and how overfitting with lower training samples affects performance of the classifier.

## 1.1 Motivation

Early brain tumor identification and classification represent a significant area of research in the field of medical imaging. It helps in choosing the most appropriate line of action for treatments to save patients' lives. In both children and adults, brain tumors are regarded as one of the most severe disorders. Brain tumors account for 85% to 90% of all major malignancies of the Central Nervous System (CNS). An estimated 11,700 people receive a brain tumor diagnosis each year. For those with a malignant brain or CNS tumor, the 5-year survival rate is around 34% for males and 36% for women [3]. It is difficult to treat brain tumors as we know that our brain has a very complex structure having tissues that are linked to each other in a complex manner. Often, producing MRI results is very difficult and time-consuming in underdeveloped nations due to a shortage of skilled medical professionals and a lack of understanding of malignancies.

Depending on the tumor's severity—that is, its location, size, and type—different treatment methods are possible. The most common technique for treating brain tumors at the moment is surgery since it has no adverse implications on the brain. There are different medical imaging techniques that are used to view the internal structures of a human body in order to discover any abnormalities. The most often used of them to identify brain tumors is Magnetic Resonance Imaging (MRI) since it can show abnormalities that may not be seen or just dimly visible on computed tomography (CT) [4]. However, the rush of patients makes it difficult, chaotic, and perhaps error-prone to manually review these images. Automated classification methods based on machine learning and artificial intelligence have regularly outperformed manual classification in terms of accuracy in order to solve this issue. Therefore, recommending a system that does detection and classification utilizing deep learning algorithms employing the above-mentioned benchmark models would be helpful for radiologists and other medical professionals.

## 1.2 Contributions

In this article, we focused on imbalance problem of Brain Tumor Classification as it is a real-world scenario in deployment. To solve this imbalance problem, we experiment with several loss functions including focal loss. Alongwith, a comparative study is conducted on the performance of different oversampling methods including augmentation, SMOTE, and ADASYN. The models we comparatively experimented on are 8 Transfer Learning-CNN models that incorporates the base Transfer learning model, followed by the integration of proposed CNN layers. Following a detailed evaluation of each Transfer Learning-CNN model, we finally conclude VGG16-CNN is the novel proposal of this study. We employ 8 transfer learning models integrated with CNNs to increase the classification accuracy on different types of brain cancer (no tumor, glioma, meningioma, and pituitary cancer). We experiment with 5 different approaches to deal with imbalanced datasets such as- Changing loss functions: 1.Focal loss 2.Cross Entropy, and Oversampling methods: 3.Data Augmentation, 4.SMOTE 5.ADASYN. Our empirical evaluation of the models under the different approaches are assessed on Metrics including Accuracy, Precision, Recall, and F1 Score.

## 2 Related Works

The classification of brain tumors using MRI data has been the subject of numerous studies based on convolutional neural networks in recent years. Many of these methods make use of hybrid approaches, and many also offer technical variations on widely used deep learning models. In [5], Deepak and Ameer describe a classification method for the 3-class classification issue that combines transfer learning with GoogleNets. They used a number of evaluation metrics, with a mean classification accuracy of 98%, including area under the curve (AUC), precision, recall, F-score, and specificity. Kader et al. [6] have made use of the benefit of differential CNN by generating extra differential feature maps. With the capacity to categorize a sizable database of pictures with high accuracy of about 98%, their approach demonstrated a considerable improvement for the brain MRI classification problem. In [7], the authors proposed a hybrid architecture by adopting GoogleNet as a based CNN model while tweaking the last few layers for the specific Brain Tumor Classification. Their proposal attained the classification accuracy of 99.67%. Moreover, [8] conducted a multi-class study of Brain Tumor MRI Images as they propose a CNN model for early diagnoses purposes with fully optimized framework. Compared to the conventional CNN models, their solution attained an accuracy of 98.14%.

### 3 Methodology

#### 3.1 Transfer Learning and CNN

We observed that using the most recent transfer learning models that have been presented in the most recent literatures produced relatively decent results when trained on the balanced Brain MRI datasets. This empirical evaluation of brain tumor classification is focused on the Imbalance problem. However, in training such Transfer Learning models with fine tuning on the imbalance dataset did not show as good results in comparison to the Transfer models integrated with newly added CNN layers [2]. Since the later case demonstrated improved predicted accuracies on the unbalanced dataset, we experiment with 8 of these Transfer Learning models that are combined with newly added CNN layers. The ImageNet training dataset was used to train the traditional pre-train models, namely VGG16, EfficientNetB0, EfficientNetB3, ResNet50, DenseNet201, MobileNet, GoogleNet, XceptionNet. For effective training, each model's trained parameter weights were transferred to the newly added layers of the additional CNN model. The CNN model was then fine-tuned using the brain tumor augmented MRI data set for final classification into 4 classes. CNN, VGG16-CNN, EfficientNetB0-CNN, EfficientNetB3-CNN, ResNet50-CNN, DenseNet201-CNN, MobileNet-CNN, GoogleNet-CNN, XceptionNet-CNN are the resultant models we empirically evaluated the training performance on the imbalanced dataset using various approaches.

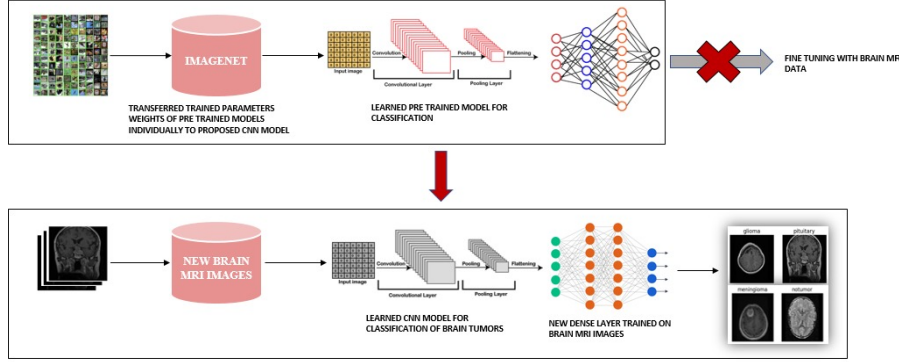


Figure 2: Difference between traditional Transfer Learning vs Transfer Learning-CNN

#### 3.2 Proposed Method

The proposed Transfer Learning-CNN (VGG-CNN) model incorporates the base Transfer learning model, i.e., VGG16, followed by the basic structure of the CNNs. It might take days to weeks to train a raw CNN completely from scratch, making it a difficult task [7]. Therefore, it would be preferable to train the suggested deep learning approach using a pretrained classifier rather than creating a new deep learning classifier from start [9]. Additionally, in terms of predictive accuracy, our Transfer Learning models combined with CNN performed significantly better on the imbalanced Brain MRI dataset than the base CNN as well as the base Transfer Learning model with Fine tuning. In order to do this, we chose the most accurate current Transfer Learning model on the balanced dataset, VGG16, as our foundation model. CNN architecture VGG16 was employed to win the 2014 ILSVR (Imagenet) competition. It is regarded as one of the best vision model architectures created to date. The most distinctive feature of VGG16 is that it prioritized having convolution layers of 3x3 filters with a stride 1 and always utilized the same padding and maxpool layer of 2x2 filters with a stride 2. Throughout the entire architecture, convolution and max pool layers are arranged in the same manner. It concludes with two fully connected layers (FC) and a softmax for output.

The pre-trained CNN models (VGG16-CNN, EfficientNetB0-CNN, EfficientNetB3-CNN, ResNet50-CNN, DenseNet201-CNN, MobileNet-CNN, GoogleNet-CNN, and XceptionNet-CNN) are trained using the Imagenet dataset, and the produced weights of these pre-trained models are individually transferred to the suggested CNN model for effective training.

Moreover, the final four layers of VGG-16 were dropped from the planned VGG-CNN architecture, and 8 new levels were added in their place. After these adjustments, there were 148 layers overall

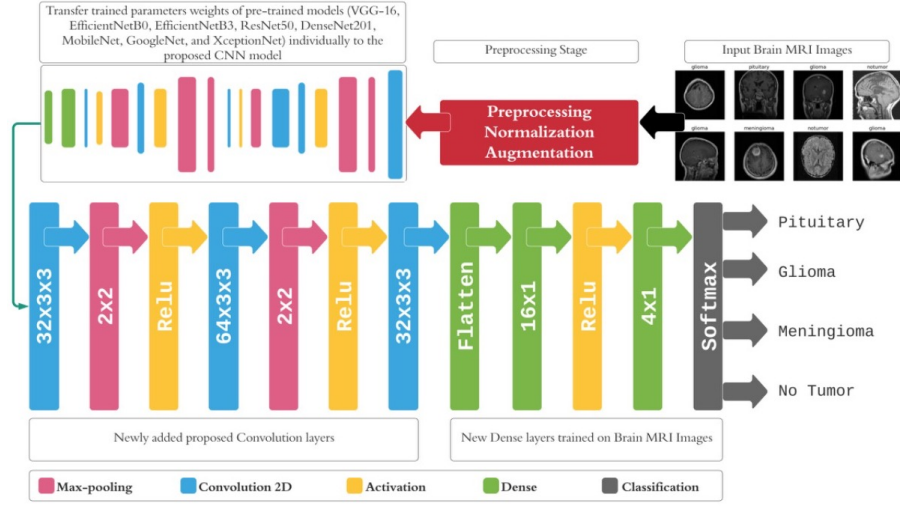


Figure 3: A flowchart showing the suggested classification scheme for brain tumors

instead of 144. Addition to the layers of transfer learning model, the first convolution layer of the CNN portion, uses a filter size of 3x3 with the depth of 32, which immediately reduces the image size followed by a max pooling layer of 2x2 filter. The second convolution layer has a depth of 64 with the same filter size of 3x3, followed by a max pooling layer of 2x2 filter. Again, a 2D convolution layer of depth 32 with filter size 3x3 is incorporated followed by a Flatten layer. Dense layer of 16 hidden units with relu activation and the final output layer of 4 units with softmax activations are used. Compared to the initial 4 layers of transfer learning models, adding the extra convolutional layers in the CNN portion gave us more detailed, accurate, and robust features. These 3 later convolutional layers extracted high-level features compared with the initial layer, which extracted low-level features. Hence, compared to the traditional Transfer Learning model, the proposed Transfer Learning-CNN model achieves higher accuracies in comparison as more intricate, exclusionary, and deep features have been acquired in the proposed methodology. The specific architecture of the proposed methodology is also shown in Figure 3.

### 3.3 Experiments

The experiments have been performed on the dataset which is a combination of three distinct datasets taken from Kaggle namely figshare, SARTAJ and Br35H. This accumulated custom imbalance dataset contains about 4200 images of human brain MRI images which are classified into 4 classes: no-tumor (1760), glioma (858), meningioma (1265) and pituitary (341) cancer, whereas no-tumor class images were taken from the Br35H dataset. The dataset images have been normalized and resized as a part of pre-processing according to the input size of the proposed model, i.e., 128x128. In addition, the accumulated dataset is divided into train and test with 90:10 ratio for validation purpose. The results mentioned in this study only apply to the test set. The testing dataset is then used to evaluate all of the predictive metric claims made about the models in the results and discussion section.

Predictive modeling is challenged by imbalanced classifications because the majority of machine learning methods for classification were devised on the premise that there should be a similar number of samples in each category. The distribution of examples among the recognized classes is skewed or biased in imbalance classes. These kind of issues with a strong to weak bias in the dataset are typical in real-world applications. As a result of such imbalance, models perform poorly in terms of prediction, particularly for the minority class [5], [10]. This is a complication since, in general, the minority class is more significant and, as a result, the issue is more susceptible to errors in classifying for the minority class than the classes with the higher samples. In the real-world scenarios of a model deployment, imbalanced classification is a huge complication [11]. Hence, in our experiments, we tried to empirically evaluate majority of the existing solutions that can be used for imbalanced or long tail classification problems.

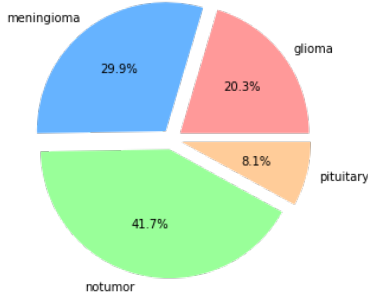


Figure 4: Data distribution of different classes for imbalanced Brain MRI dataset

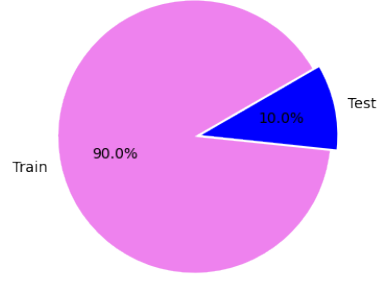


Figure 5: Train-test split of the original dataset

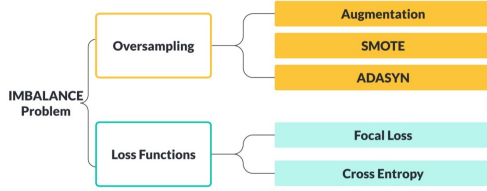


Figure 6: Distribution of incorporated approaches to resolve data imbalance problem

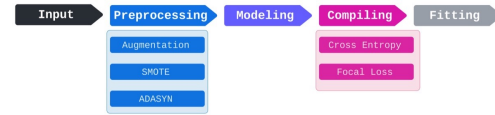


Figure 7: Implementation of data imbalance approaches throughout the phases of experimentation on models

**Focal Loss and Cross Entropy** For each of the 8 deep learning models, we comparatively examined 5 distinct approaches on each of them in order to address the imbalance scenarios that arise during the classification of brain tumors. We explore with changing the loss function in our modeling because it's one of the main reasons that simple examples will divert training in an unbalanced class scenario. We compared the results of test accuracy on our models by changing Cross Entropy with Focal loss in order to obtain better results. Focal Loss, one of the effective solutions in terms of loss functions, deals with this imbalance issue and is created in a way that allows the network to concentrate on training the challenging cases by lowering the loss (or "down-weight") for the simple examples. In other words, Focal loss reduces the importance of the simple examples and emphasizes the difficult ones; thus, smaller class counts would have heavier weights in errors than cross-entropy. The Cross-Entropy loss is multiplied by a modifying factor in focal loss [12]. When a sample is incorrectly classified, the modulating factor is close to 1,  $p$  is low, and the loss is unchanged. The modulating factor gets closer to zero as  $p$  approaches 1 and the loss for correctly categorized samples gets down-weighted.

$$FL(P_t) = -\alpha_t(1 - P_t)^\gamma \log(P_t)$$

Here,  $P_t = P$  if  $y = 1$ , else  $(1-P)$ ; where  $\gamma = 0$ ,  $\alpha_t = 1$  then FL is Cross Entropy Loss.

In addition, oversampling, which uses artificial data creation to increase the number of samples in the data set, is one of the most fundamental approaches in the state of the art to handle the imbalance problem, other than loss functions [13]. By producing synthetic observations based on the minority observations already there, oversampling aims to grow the minority class so that the data set becomes balanced. In our experimental evaluations, we used 3 oversampling approaches, namely Augmentation, SMOTE, and ADASYN methods.

**Data Augmentation** Data augmentation is used to expand the volume of data by introducing slightly altered versions of either existing data or brand-new synthetic data that is derived from available data. It also serves as a regularization term and aids in lowering overfitting when a model is being trained [8]. In our experiments, we tried to augment the samples from minority classes to achieve the number of the samples equal to the majority class. The augmentation tasks we applied to such minority class samples are brightness, contrast, and sharpness alteration each with a random intensity of 80% to 120%. This was followed by the normalization process as a part of preprocessing.

**SMOTE** Second oversampling method we implemented on our dataset was SMOTE (Synthetic Minority Oversampling Technique). By creating artificial data samples that are marginally different from the

current data points on the basis of the existing data points, SMOTE performs oversampling. Following is how the SMOTE algorithm operates:

- A random sample is picked from the minority class.
- Locate the  $k$  nearest neighbors for the observations in this minority class sample.
- The vector (line) between the existing data point and one of those neighbors will then be determined using that neighbor.
- The vector is then multiplied by a randomized range unit between 0 and 1.
- We combine this with the existing data point to get the synthetic data sample in the space.

**ADASYN** ADASYN is an improvised version of SMOTE. It accomplishes the same things as SMOTE, albeit slightly better. It then adds a random tiny value to the points to make it more realistic after creating those synthetic samples [7]. In other words, the sample’s variance, or degree of dispersion, is a little higher than its linear correlation to the parent. Moreover, ADASYN adaptively change the weights of different minority samples to compensate for the skewed distributions. Difference between SMOTE and ADASYN oversampling approaches is also shown in Figure 8.

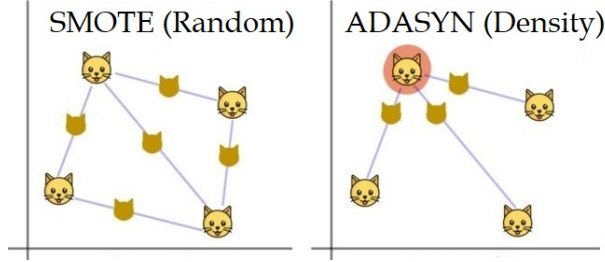


Figure 8: Working of SMOTE vs ADASYN algorithm

We examined each of our 8 TL-CNN models on each of these above-mentioned 5 approaches - Focal Loss, Cross Entropy, Data Augmentation, SMOTE, and ADASYN, with an aim to achieve higher test accuracy on the imbalance dataset. Focal loss is initially compared with the Cross Entropy loss on unbalanced dataset, whereas Data Augmentation, SMOTE, and ADASYN are compared on the unbalanced dataset too but with cross entropy as these later 3 approaches are oversampling approaches [6]. With the parameters listed in Table X, we conducted tests using a trial-and-error methodology. In order to determine the best convergence for each TL-CNN model, we continuously tracked the development of the training validation accuracy and error. With a minibatch size of 20 images and an initial learning rate ranging between 0.001 and 0.0001 depending on the model, we utilized mini-batch GD to train the TL-CNN models. To get the results that are explained in the following section, the evaluated models were trained on 5–12 epochs for brain tumor classification.

Table 1: Several parameters utilised during the training phase

Name	Values
Mini Batch Size	20
Number of Epochs	5 to 12
Learning Rate(s)	0.001 to 0.0001
Shuffle	True
Steps Per Epoch	Train Size (1600) / 20
Dense Activations	RELU
Optimizer	ADAM

### 3.4 Evaluation

For the typical assessment of a classifier, numerous performance metrics are specified, including Classification Accuracy, Precision, Recall, and F1-Score. The evaluation employs criteria besides the total classification accuracy due to the unbalanced dataset. The mean of correctly classified samples from each class is used to compute balanced accuracy. A class’s F-score is determined by taking the harmonic mean of its precision and recall values.

$$Precision = \frac{TruePositive}{TruePositive+FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive+FalseNegative}$$

$$Accuracy = \frac{TrueNegative+TruePositive}{TrueNegative+FalsePositive+TruePositive+FalseNegative}$$

$$F1Score = 2 * \frac{Precision*Recall}{Precision+Recall}$$

When the test dataset has an identical amount of samples from each category, classification accuracy is a useful metric to assess performance. Nevertheless, the dataset we aim to use for this categorization problem under discussion is adequately imbalance. This calls for a more thorough assessment of the suggested system using more evaluation metrics. To evaluate the effectiveness of our tumor categorization method, we employed other metrics including Precision, Recall, and F1-Score. Tables [X-Z] presents a summary of such metrics on each of the 8 models in terms of various approaches we employed to resolve imbalance classification. The effectiveness of the relevant implemented models that have been implemented in our categorization are empirically evaluated using these classification metrics. Due to the existence of imbalance among the 4 classes, we also showed individual precision values for each class that have been implemented using different approach.

Table 2: Evaluation of Integrated frameworks performance on imbalanced dataset when trained with Cross Entropy Loss. The space complexity of each model is the number of trainable parameters. M = Million. The space complexity increases with increasing number of trainable parameters. The time complexity is the training time (in hours) of the models

Model	Space Complexity	Time Complexity (hrs)	LR	Accuracy	Precision				Recall				F1 Score			
					P	M	G	NT	P	M	G	NT	P	M	G	NT
CNN	440k	0.091	0.001	0.8125	1.00	0.67	0.73	0.98	0.48	0.91	0.51	0.96	0.65	0.77	0.60	0.97
VGG16-CNN	15.7M	0.26	0.0001	0.83	0.73	0.66	0.94	0.99	0.73	0.66	0.41	1.00	0.60	0.78	0.57	1.00
EfficientNetB0-CNN	4.05M	0.27	0.001	0.37	0.00	0.17	0.00	0.39	0.04	0.04	0.00	0.86	0.00	0.07	0.00	0.54
EfficientNetB3-CNN	10.8M	0.47	1.6e-5	0.60	0.50	0.60	0.38	0.65	0.06	0.46	0.27	0.96	0.11	0.52	0.31	0.78
ResNet50-CNN	23.5M	0.31	0.001	0.78	1.00	0.87	0.60	0.83	0.39	0.59	0.82	0.97	0.56	0.70	0.69	0.90
DenseNet201-CNN	51.8M	0.57	0.001	0.47	0.33	0.43	0.67	0.49	0.23	0.32	0.05	0.82	0.27	0.37	0.10	0.61
MobileNet-CNN	3M	0.126	0.001	0.30	0.00	0.30	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.46	0.00	0.00
GoogleNet-CNN	6M	0.14	0.0001	0.77	0.71	0.64	0.58	0.97	0.65	0.67	0.56	0.96	0.68	0.65	0.57	0.97
XceptionNet-CNN	25M	0.15	0.0001	0.83	0.90	0.66	0.89	0.99	0.61	0.95	0.42	0.99	0.73	0.78	0.57	0.99

Table 3: Evaluation of Integrated frameworks performance on imbalanced dataset when trained with Focal Loss

Model	Space Complexity	Time Complexity (hrs)	LR	Accuracy	Precision				Recall				F1 Score			
					P	M	G	NT	P	M	G	NT	P	M	G	NT
CNN	440k	0.084	0.001	0.80	0.88	0.68	0.61	0.99	0.45	0.81	0.56	0.98	0.60	0.74	0.59	0.98
VGG16-CNN	15.7M	0.267	0.0001	0.85	0.93	0.68	0.88	1.00	0.87	0.98	0.37	0.98	0.90	0.81	0.52	0.99
EfficientNetB0-CNN	4.05M	0.27	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
EfficientNetB3-CNN	10.8M	0.47	1.6e-5	0.32	0.00	0.22	0.18	0.37	0.00	0.14	0.09	0.62	0.00	0.17	0.12	0.46
ResNet50-CNN	23.5M	0.31	0.001	0.62	0.69	0.45	1.00	0.97	0.71	0.95	0.17	0.59	0.70	0.61	0.29	0.73
DenseNet201-CNN	51.8M	0.57	0.001	0.66	0.57	0.50	0.24	0.96	0.39	0.68	0.15	0.94	0.46	0.58	0.19	0.95
MobileNet-CNN	3M	0.13	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
GoogleNet-CNN	6M	0.14	0.0001	0.78	0.73	0.62	0.74	0.97	0.61	0.88	0.33	0.97	0.67	0.72	0.46	0.97
XceptionNet-CNN	25M	0.20	0.0001	0.81	0.83	0.62	0.84	1.00	0.61	0.92	0.33	1.00	0.70	0.74	0.48	1.00

Table 4: Evaluation of Integrated frameworks performance on imbalanced dataset when trained with SMOTE oversampling

Model	Space Complexity	Time Complexity (hrs)	LR	Accuracy	Precision				Recall				F1 Score			
					P	M	G	NT	P	M	G	NT	P	M	G	NT
CNN	440k	0.11	0.001	0.82	0.86	0.71	0.71	0.93	0.77	0.83	0.45	0.99	0.81	0.76	0.55	0.96
VGG16-CNN	15.7M	0.52	0.0001	0.86	0.96	0.71	0.83	0.99	0.81	0.96	0.50	0.97	0.88	0.81	0.62	0.98
EfficientNetB0-CNN	4.05M	0.45	0.001	0.47	0.00	0.38	0.00	0.51	0.00	0.41	0.00	0.82	0.00	0.39	0.00	0.63
EfficientNetB3-CNN	10.8M	1.04	1.6e-5	0.50	0.00	0.39	0.00	0.88	0.00	0.97	0.00	0.51	0.00	0.55	0.00	0.64
ResNet50-CNN	23.5M	0.77	0.001	0.66	1.00	0.97	0.37	0.96	0.19	0.27	0.95	0.89	0.32	0.42	0.54	0.92
DenseNet201-CNN	51.8M	2.08	0.001	0.86	0.96	0.70	0.97	0.99	0.77	0.98	0.45	0.99	0.86	0.82	0.61	0.99
MobileNet-CNN	3M	0.27	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
GoogleNet-CNN	6M	0.14	0.0001	0.79	0.88	0.68	0.57	1.00	0.74	0.66	0.72	0.93	0.81	0.67	0.64	0.96
XceptionNet-CNN	25M	0.27	0.0001	0.83	0.91	0.66	0.82	1.00	0.68	0.94	0.41	0.99	0.78	0.78	0.55	1.00

Precision, recall (or sensitivity), and F1-Score are crucial indicators, and they are determined using the relations described above. For each class, the harmonic mean of recall and precision yields the F1-score, another significant statistical classification metric. If we obtain high F1-Score values across



Table 5: Evaluation of Integrated frameworks performance on imbalanced dataset when trained with ADASYN oversampling

Model	Space Complexity	Time Complexity (hrs)	LR	Accuracy	Precision				Recall				F1 Score			
					P	M	G	NT	P	M	G	NT	P	M	G	NT
CNN	440k	0.12	0.001	0.83	0.83	0.75	0.64	0.97	0.77	0.77	0.59	1.00	0.80	0.76	0.61	0.98
VGG16-CNN	15.7M	0.60	0.0001	0.87	0.91	0.75	0.81	1.00	0.94	0.92	0.54	0.99	0.92	0.82	0.65	0.99
EfficientNetB0-CNN	4.05M	0.63	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
EfficientNetB3-CNN	10.8M	1.11	1.6e-5	0.52	1.00	0.36	0.33	0.86	0.13	0.15	0.87	0.69	0.23	0.21	0.48	0.69
ResNet50-CNN	23.5M	0.83	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
DenseNet201-CNN	51.8M	1.67	0.001	0.90	1.00	0.85	0.94	0.85	0.77	0.90	0.76	1.00	0.87	0.87	0.84	0.95
MobileNet-CNN	3M	0.30	0.001	0.42	0.00	0.00	0.00	0.42	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.59
GoogleNet-CNN	6M	0.15	0.0001	0.77	0.82	0.66	0.49	1.00	0.74	0.58	0.63	0.97	0.78	0.62	0.55	0.98
XceptionNet-CNN	25M	0.31	0.0001	0.86	0.96	0.70	0.90	0.99	0.81	0.97	0.45	0.99	0.88	0.81	0.60	0.99

all classes, this suggests that we have successfully identified samples free of any class of brain tumor. Due to the existence of imbalance among the 4 classes, we also showed individual precision values for each class that have been implemented using different approach. We also intend to compare several analytical factors, such as how well each model performs with less training samples from practicality aspect and how overfitting with lower training samples affects performance of the classifier.

## 4 Results and Discussion

This study aimed to appropriately categorize four distinct kind of brain tumor MRI scans. We compared 5 different methodologies—augmentation, focal loss, SMOTE, and ADASYN — while using 8 transfer learning + CNN models. We used a variety of evaluation criteria, including accuracy, precision, recall, and F1-score, to evaluate the effectiveness of our suggested model. This unbalanced dataset was categorized using deep learning models that have already been trained as well as the suggested transfer learning + CNN model, with 90% of the dataset being used for training and 10% being utilized for testing. The performance of deep learning networks for classification can be measured using various methods. In the CNN process, classification tasks are frequently carried out using a confusion matrix. The accuracy comparison on imbalanced dataset utilized in this research is shown in the table below.

Table 6: Accuracy Comparison on Unbalanced Dataset using several imbalance solution approaches

Model	CE	Focal Loss	Augment CE	SMOTE CE	ADASYN CE
CNN	0.8125	0.80	0.91	0.82	0.83
VGG16-CNN	<b>0.83</b>	<b>0.85</b>	<b>0.96</b>	<b>0.86</b>	0.87
EfficientNetB0-CNN	0.37	0.42	0.31	0.47	0.42
EfficientNetB3-CNN	0.60	0.32	0.32	0.50	0.52
ResNet50-CNN	0.78	0.62	0.80	0.66	0.42
DenseNet201-CNN	0.47	0.66	0.94	<b>0.86</b>	<b>0.90</b>
MobileNet-CNN	0.30	0.42	0.23	0.42	0.42
GoogleNet-CNN	0.77	0.78	0.80	0.79	0.77
XceptionNet-CNN	<b>0.83</b>	0.81	0.94	0.83	0.86

The result of the above experiments showed that VGG-16 performed best overall across all techniques for the accuracy evaluation metric. With augmentation, 0.96 was the highest accuracy ever attained overall. In this instance, MobileNet and EfficientNets had the worst results with an overall accuracy of 0.23 and 0.31, respectively. For the relevant unbalanced dataset on VGG16, EfficientNets, DenseNet201, MobileNet, and GoogleNet, it was found that focal loss obtained higher accuracies than cross entropy loss when the loss functions were compared having values 0.85, 0.66, 0.42 and 0.78, respectively. With the cross-entropy loss, the remaining models fared better. Data augmentation produced the overall best accuracy of the techniques used to address the data imbalance issue.

In all classes where VGG-16 + CNN with data augmentation outperformed other approaches, we computed the average precision, recall, and F1 scores. To summarize, despite having far less layers than other models, transfer learning using the VGG-16 model performed the best overall. The performance measures employing various transfer learning + CNN models on our unbalanced dataset are compared in figure 7. Here, we note that performance on an augmented dataset with cross-entropy loss outperforms alternative methods across a range of evaluation parameters.



## 5 Conclusion and Future Work

Most computer-aided modeling techniques for the investigation of medical image data heavily rely on the CNN model for precise and reliable medical image classification. We have proposed a deep learning-based detection and identification technique for classifying brain tumors in our research study. With the help of brain tumor MRI imaging data, we have examined various transfer learning models for the classification of tumor kinds, including meningioma, glioma, and pituitary. We have added various transfer learning techniques, followed by a CNN model for each method, to improve the CNN model's predictive power. According to the experimental findings, the proposed strategy, which combines VGG-16 and CNN, has a 96% accuracy rate which is significantly higher when compared with other approaches.

The proposed method's notable prediction accuracy was mostly due to data augmentation; further contributing aspects included altering the number of layers, using optimizers, and experimenting with various activation functions. In the future, we will concentrate on optimizing the VGG16 and DenseNet models through rigorous GridSearchCV and hyperparameter tuning, which requires a lot of effort and computing power. We also want to focus on MobileNet due to its simplicity and increased use in real-world settings.

## References

- [1] A Veeramuthu, S Meenakshi, G Mathivanan, Ketan Kotecha, Jatinderkumar R Saini, V Vijayakumar, and V Subramaniaswamy. Mri brain tumor image classification using a combined feature and image-based classifier. *Frontiers in Psychology*, 13, 2022.
- [2] Jian Ping Li, Shakir Khan, Mohammed Ali Alshara, Reemiah Muneer Alotaibi, Cobbinah-Bernard Mawuli, et al. Dacbt: deep learning approach for classification of brain tumors using mri data in iot healthcare environment. *Scientific Reports*, 12(1):1–14, 2022.
- [3] Tao Wang, Changhua Lu, Mei Yang, Feng Hong, and Chun Liu. A hybrid method for heartbeat classification via convolutional neural networks, multilayer perceptrons and focal loss. *PeerJ Computer Science*, 6:e324, 2020.
- [4] Hareem Kibriya, Rashid Amin, Asma Hassan Alshehri, Momina Masood, Sultan S Alshamrani, and Abdullah Alshehri. A novel and effective brain tumor classification model using deep feature fusion and famous machine learning classifiers. *Computational Intelligence and Neuroscience*, 2022, 2022.
- [5] S Deepak and PM Ameer. Brain tumor classification using deep cnn features via transfer learning. *Computers in biology and medicine*, 111:103345, 2019.
- [6] Isselmou Abd El Kader, Guizhi Xu, Zhang Shuai, Sani Saminu, Imran Javaid, and Isah Salim Ahmad. Differential deep convolutional neural network model for brain tumor classification. *Brain Sciences*, 11(3):352, 2021.
- [7] Asaf Raza, Huma Ayub, Javed Ali Khan, Ijaz Ahmad, Ahmed S. Salama, Yousef Ibrahim Daradkeh, Danish Javeed, Ateeq Ur Rehman, and Habib Hamam. A hybrid deep learning-based approach for brain tumor classification. *Electronics*, 11(7):1146, 2022.
- [8] Emrah Irmak. Multi-classification of brain tumor mri images using deep convolutional neural network with fully optimized framework. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 45(3):1015–1036, 2021.
- [9] Georgios Petmezas, Grigorios-Aris Cheimariotis, Leandros Stefanopoulos, Bruno Rocha, Rui Pedro Paiva, Aggelos K Katsaggelos, and Nicos Maglaveras. Automated lung sound classification using a hybrid cnn-lstm network and focal loss function. *Sensors*, 22(3):1232, 2022.
- [10] S Deepak and PM Ameer. Brain tumor categorization from imbalanced mri dataset using weighted loss and deep feature fusion. *Neurocomputing*, 2022.
- [11] Milica M Badža and Marko Č Barjaktarović. Classification of brain tumors from mri images using a convolutional neural network. *Applied Sciences*, 10(6):1999, 2020.

- [12] Soumick Chatterjee, Faraz Ahmed Nizamani, Andreas Nürnberger, and Oliver Speck. Classification of brain tumours in mr images using deep spatiotemporal models. *Scientific Reports*, 12(1):1–11, 2022.
- [13] Yuting Xie, Fulvio Zaccagna, Leonardo Rundo, Claudia Testa, Raffaele Agati, Raffaele Lodi, David Neil Manners, and Caterina Tonon. Convolutional neural network techniques for brain tumor classification (from 2015 to 2022): Review, challenges, and future perspectives. *Diagnostics*, 12(8):1850, 2022.

## A Appendix

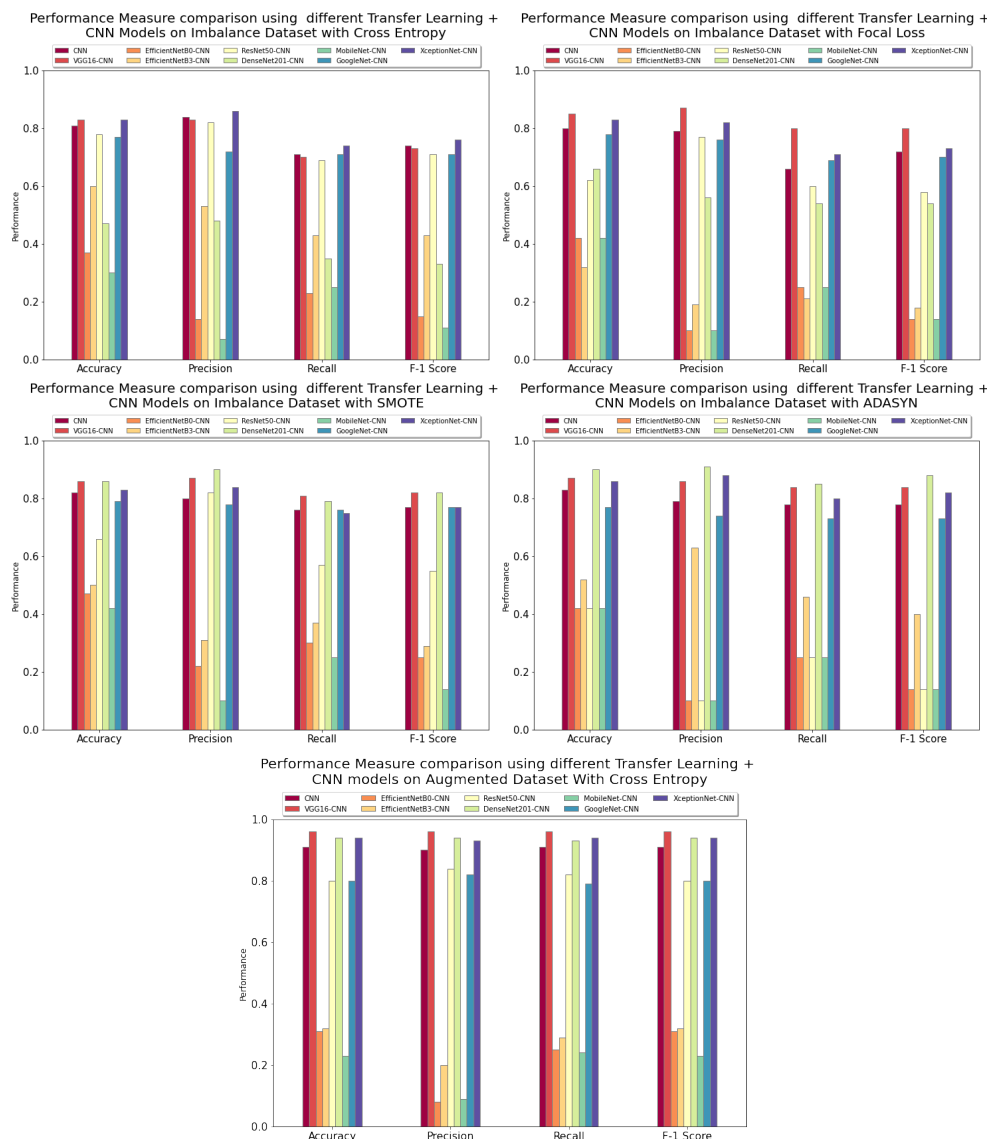


Figure 9: Performance comparison using various transfer learning models + CNN on different techniques for handling data imbalance

[Link to Project Github](#)