information

MDPI

Article

Enhancing Brain Tumor Detection Through Custom Convolutional Neural Networks and Interpretability-Driven Analysis

Kavinda Ashan Kulasinghe Wasalamuni Dewage [1], Raza Hasan [1,*], Bacha Rehman [1] and Salman Mahmood [2]

[1] Department of Computer Science, Solent University, Southampton SO14 0YN, UK; ashankulasinghe1110@gmail.com (K.A.K.W.D.); bacha.rehman@solent.ac.uk (B.R.)

[2] Department of Computer Science, Nazeer Hussain University, ST-2, near Karimabad, Karachi 75950, Pakistan; salman.mahmood@nhu.edu.pk

* Correspondence: raza.hasan@solent.ac.uk

Abstract: Brain tumor detection is crucial for effective treatment planning and improved patient outcomes. However, existing methods often face challenges, such as limited interpretability and class imbalance in medical-imaging data. This study presents a novel, custom Convolutional Neural Network (CNN) architecture, specifically designed to address these issues by incorporating interpretability techniques and strategies to mitigate class imbalance. We trained and evaluated four CNN models (proposed CNN, ResNetV2, DenseNet201, and VGG16) using a brain tumor MRI dataset, with oversampling techniques and class weighting employed during training. Our proposed CNN achieved an accuracy of 94.51%, outperforming other models in regard to precision, recall, and F1-Score. Furthermore, interpretability was enhanced through gradient-based attribution methods and saliency maps, providing valuable insights into the model's decision-making process and fostering collaboration between AI systems and clinicians. This approach contributes a highly accurate and interpretable framework for brain tumor detection, with the potential to significantly enhance diagnostic accuracy and personalized treatment planning in neuro-oncology.

Keywords: brain tumor detection; convolutional neural networks; interpretability; class imbalance; medical imaging

check for updates

Citation: Dewage, K.A.K.W.; Hasan, R.; Rehman, B.; Mahmood, S. Enhancing Brain Tumor Detection Through Custom Convolutional Neural Networks and Interpretability-Driven Analysis. *Information* **2024**, *15*, 653. https://doi.org/10.3390/info15100653

Academic Editor: Neil Vaughan

Received: 9 September 2024
Revised: 12 October 2024
Accepted: 16 October 2024
Published: 18 October 2024

1. Introduction

Brain tumors pose a significant health concern, contributing substantially to cancer-related morbidity and mortality rates worldwide [1]. In clinical practice, the accurate identification and classification of different tumor subtypes are essential for effective treatment planning and providing personalized care to patients. Radiological imaging, particularly Magnetic Resonance Imaging (MRI), offers crucial anatomical and functional details vital for differential diagnosis and therapeutic decisions [2]. Despite advancements in imaging technologies and diagnostic approaches, one of the most critical challenges in neuro-oncology remains the precise classification of brain tumor subtypes [3].

Traditional imaging modalities provide insights into tumor morphology and localization [4], but they often fall short in determining the subtler characteristics associated with histology and molecular components [5]. This highlights the pressing need for more accurate imaging analysis techniques that can classify different brain tumor subtypes with greater precision. The present study aims to bridge this gap through advanced computational approaches and radiomic analysis, enhancing the capability of radiological imaging, especially MRI, for tumor subtype classification.

Although modern architectures like EfficientNet, ConvNeXt, and Vision Transformers (ViT) offer improved accuracy in large-scale image classification, they are often computationally expensive and lack the level of interpretability required in medical imaging.

Given the clinical necessity for both high accuracy and transparent decision-making, our study opts for a custom CNN architecture. This model is designed to provide a balance between interpretability, computational efficiency, and accuracy, ensuring its suitability for deployment in resource-constrained clinical environments where model decisions must be easily understood and trusted by healthcare professionals.

The primary objective of this research is to develop and validate predictive models capable of differentiating between gliomas, meningiomas, and pituitary tumors using MRI scans and the radiomic features extracted from these scans. Through a systematic analysis of a large MRI dataset and the employment of machine-learning methods, the study aims to identify radiomic features associated with enhanced specificity and sensitivity in distinguishing among the selected tumor subtypes and develop models for accurate classification. The major contributions of this study are as follows:

1. Development of a custom CNN model specifically tailored for brain tumor detection using MRI scans.
2. Comparative analysis with State-of-the-Art pretrained models, including ResNetV2, DenseNet201, and VGG16.
3. Integration of oversampling techniques and class weighting to handle class imbalance.
4. Enhancement of model interpretability using gradient-based attribution methods and saliency maps.
5. Comprehensive performance evaluation using multiple metrics, including accuracy, precision, recall, and F1-score.

This work can benefit the field of neuro-oncology by providing more specific diagnostic tools for clinicians, enabling the development of personalized approaches to the treatment of different tumor subtypes.

2. Literature Review

Brain tumor detection and classification are crucial for diagnosing and treating brain diseases. Over the years, various imaging techniques and machine-learning algorithms have been developed to enhance the accuracy and efficiency of brain tumor detection [6]. In this section, we discuss previous studies related to our objective of redefining the process of brain tumor detection using trained CNN models. We also present a summary of the findings, methodologies, and limitations from these studies.

Table 1 provides an overview of various approaches and methodologies involved in brain tumor detection, including deep learning (DL)-based approaches alongside multi-modal imaging approaches. However, these techniques have shown limitations or factors that need improvement, such as underutilized datasets, non-interpretable DL models, and class imbalance.

Table 1. Summary of existing studies on brain tumor detection.

Source	Methodology	Main Findings	Limitations
[7]	Deep learning-based approach combining CNNs and RNNs to analyze multimodal MRI data, with data augmentation.	Improved sensitivity and specificity in brain tumor detection through a deep learning-based approach combining CNNs and RNNs.	Limited availability of labeled data addressed through data augmentation.
[8]	Utilization of deep learning for medical image analysis, preprocessing MRI images, and classification with a hybrid CNN-LSTM model.	Outperformed existing models in brain tumor classification with a high validation accuracy.	The proposed method relies on pretrained models like AlexNet for feature extraction, potentially limiting its ability to adapt to new and diverse datasets without further fine-tuning.

Table 1. *Cont.*

Source	Methodology	Main Findings	Limitations
[9]	Development of a deep-learning system using convolutional neural networks for brain tumor detection from MRI scans.	Accurate detection of brain tumors from MRI scans using deep learning-based system.	Limited availability of diverse and representative datasets for training may constrain the generalizability of the model's predictions.
[10]	Utilization of deep-learning models (ResNet50, ConvNeXt, and custom CNN) for brain tumor detection from MRI scans.	Deep-learning models offer efficient tumor detection on MRI images for clinicians.	Limited availability of diverse and large-scale datasets for training and testing deep-learning models, which may affect the generalizability and robustness of the developed brain tumor-detection system.
[11]	Use of CNN for brain tumor detection and classification, development of a deep-learning model for tumor categorization.	Deep-learning model accurately classifies brain tumors into different categories with high accuracy.	Complexity of MRI images, limited classification into 4 tumor types, generalizability not discussed.
[12]	Utilization of transfer-learning model (AlexNet's CNN) for brain tumor detection and classification in MR images.	Transfer learning with AlexNet's CNN improves brain tumor detection and classification in MR images.	Not mentioned.
[13]	Development of DL model based on U-Net CNN for classifying different brain tumor types.	DL model based on U-Net CNN classifies different brain tumor types with high accuracy.	Limited availability of diverse and large-scale datasets for training and testing the U-Net model, which may affect the generalizability and robustness of the developed brain tumor detection-and-classification system.

Recent advancements in image classification have seen the emergence of several promising architectures. ConvNeXt, a pure CNN, has demonstrated a competitive performance against Transformers, achieving 87.8% accuracy on ImageNet and outperforming Swin Transformers on detection and segmentation tasks [14]. CoAtNet, which combines convolution and attention mechanisms, achieved 90.88% accuracy on ImageNet when scaled up with JFT-3B [15]. EfficientNet principles have been incorporated into hybrid models like EffiConvRes, which utilizes residual connections and depthwise convolutions to achieve high accuracy while maintaining computational efficiency [16]. ConvNeXt variants have also shown superior performance on the CIFAR-10 dataset compared to other State-of-the-Art models [17]. These architectures represent significant progress in balancing accuracy, efficiency, and scalability for image-classification tasks.

Table 2 summarizes recent research on brain tumor detection and classification, including ongoing challenges and promising new directions.

In recent advancements, EfficientNet has emerged as a high-performance architecture for medical image classification tasks. Specifically, Ref. [18] introduces a fine-tuned EfficientNetV2S model for classifying brain tumors, achieving superior results in accuracy and performance metrics compared to other deep-learning models. Their model utilizes EfficientNetV2S, which incorporates inverted bottleneck blocks and depthwise separable convolutions, optimizing computational efficiency and improving accuracy.

In comparison, our study developed a custom CNN model with three convolutional layers and standard ReLU activation functions for brain tumor classification. While our approach achieved an accuracy of 94.51%, models in [18] significantly outperformed it with an accuracy of 98.48%. This difference may be attributed to the deeper and more complex structure of EfficientNetV2S, which is designed to capture more intricate features in MRI images.

Table 2. Summary of Recent Approaches and Challenges in Brain Tumor Detection and Classification.

Aspect	Key Findings	Methodologies	Gaps/Limitations
Deep-learning approaches	Many studies demonstrate the effectiveness of CNNs in analyzing medical images for brain tumor detection, leading to high accuracy rates.	Utilization of CNN architectures to process MRI scans and classify brain tumors into various subtypes. Data preprocessing, including normalization and augmentation, is used to enhance model performance. Transfer-learning techniques to improve model accuracy by transferring knowledge from pretrained models.	Lack of interpretability in deep-learning models. Limited generalizability due to focus on specific datasets.
Multimodal imaging	Integration of MRI, CT, and PET data can enhance tumor boundary delineation and diagnostic accuracy.	Utilization of multimodal datasets combining information from different imaging techniques. Fusion techniques, such as feature concatenation or attention mechanisms, to integrate information from multiple modalities. Training of models using deep-learning or traditional machine-learning algorithms.	Challenges in integrating and harmonizing data from disparate sources. Standardization of imaging protocols and data preprocessing techniques is crucial.
Addressing class imbalance	Strategies like oversampling, class weighting, and specialized loss functions mitigate the negative impact of class imbalance on model performance.	Employing techniques such as oversampling to generate synthetic samples for minority classes. Applying class weighting during model training to give higher importance to minority classes. Designing specialized loss functions to penalize misclassifications of minority classes more heavily.	While these strategies improve model performance, they may not fully address the underlying imbalance in the dataset. Further research is needed to explore novel approaches for handling class imbalance effectively.
Interpretability of models	Techniques for visualizing and interpreting model predictions, such as gradient-based attribution methods and saliency maps, improve model transparency.	Visualization techniques to highlight influential regions in input images and visualize the features learned by the model. Employing gradient-based attribution methods like Integrated Gradients and Guided Backpropagation to identify influential pixels in the input image. Generating saliency maps to visualize areas of the image that contribute most to the model's outputs.	While these techniques provide valuable insights into model predictions, they may not always capture the complex decision-making process of deep-learning models. Further research is needed to develop more interpretable models and visualization techniques tailored to medical-imaging tasks.

Furthermore, Ref. [18] employed advanced data augmentation and Grad-CAM visualization techniques to enhance interpretability, which is another area where our study could be expanded. In future work, adopting some of these techniques and architectures could help further improve both the performance and interpretability of our model.

2.1. *Conflicts and Gaps*

Deep learning has shown promise in brain tumor detection. Table 3 highlights several key challenges that need to be addressed.

Table 3. Conflicts and gaps in brain tumor detection.

Challenges	Description
Limited generalization	Studies often focus on specific datasets, making findings less applicable.
Class imbalance	Some tumor types are underrepresented, leading to biased predictions.
Interpretability	Deep-learning models are hard to interpret, making it challenging for clinicians to understand predictions.

2.2. Proposed Approach

This study aims to address inconsistencies in brain tumor-detection research. The specific goals designed to improve the performance and accuracy of brain tumor-detection systems are detailed in Table 4.

Table 4. Proposed approach for enhancing brain tumor detection.

Goal	Description
Developing a custom CNN model	Conceive and train a bespoke CNN architecture tailored to brain tumor detection. Leverage domain-specific insights and architectural innovations to enhance sensitivity, specificity, and resilience.
Addressing class imbalance	Employ strategies like oversampling minority classes, class weighting, or specialized loss functions to alleviate the effects of class imbalance and ensure equitable performance across all tumor categories.
Enhancing interpretability	Explore techniques for visualizing and interpreting the features learned by the CNN model, including gradient-based attribution methodologies, activation maximization, and saliency maps. These techniques provide valuable insights into the model's decision-making rationale and facilitate collaboration between clinicians and AI systems.

This in-depth study aims to push the boundaries of brain tumor detection and develop more accurate, clear, and useful tools for diagnosing brain tumors from scans.

3. Methodology

This research follows an experimental design within applied research [19]. We address identified knowledge gaps and challenges in brain tumor-detection by creating and validating custom CNN models. The study involves data collection, model building, training, evaluation, and interpretation, as illustrated in Figure 1. Through this methodology, we aim to generate new knowledge and advance brain tumor-detection techniques. We illustrate the current research pipeline in solid lines, while the security measures and robustness evaluations are depicted in dotted lines. These dotted sections represent the future research directions that will address the complexities of ensuring model security in healthcare.

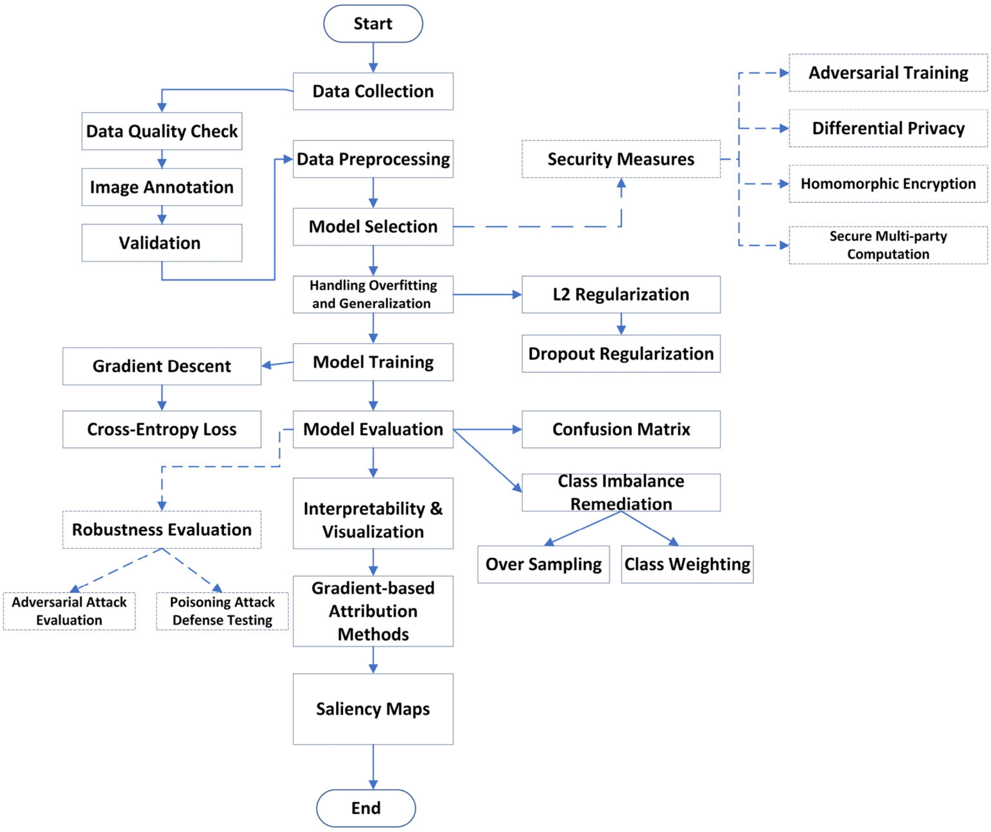


Figure 1. Methodology flowchart.

3.1. Data Collection

The study utilizes the Brain Tumor Classification dataset obtained from Kaggle, specifically curated by [20]. This dataset contains MRI scans categorized into four classes: meningiomas, gliomas, pituitary tumors, and healthy brains.

The dataset underwent a rigorous curation process [20] to ensure data quality and representativeness. First, MRI images were collected from various clinical sources to capture variability in imaging protocols and scanners. This ensures that the dataset reflects real-world scenarios. Next, the images went through quality checks to identify artifacts or inconsistencies. Annotators with neuroimaging expertise then labeled the images based on pathological characteristics. Any disagreements between annotators were resolved through a consensus process. Finally, expert radiologists validated the annotations by the annotators.

3.2. Data Description

The dataset contains 3264 images divided into separate training and testing sets, as detailed in Table 5. These images represent various tumor types, including gliomas, meningiomas, and pituitary tumors, alongside images of healthy brains (labeled as "No tumor").

Table 5. Distribution of images across tumor subtypes in training and testing sets.

	Glioma	Meningioma	Pituitary	No Tumor
Training	826	822	827	395
Testing	100	115	74	105

3.3. Data Preprocessing

To prepare the data for training our model, we implemented a multi-step preprocessing pipeline focused on improving data quality and, ultimately, the model's performance. First, we conducted a thorough cleaning process to remove any duplicate or corrupted images, ensuring that the data used for analysis are reliable. Next, we standardized the images by setting a consistent average pixel value and variation across all images. This creates a uniform intensity level, making them more compatible with the model. We further enhanced interpretability and comparison by normalizing pixel values between 0 and 1. Finally, to increase the dataset's diversity and prevent overfitting, we employed data-augmentation techniques. These techniques simulate real-world variations in brain scans, allowing the model to perform better in clinical settings. Rotations mimic how the scan was oriented during imaging, flips account for anatomical variations between patients, and crops focus on specific regions of interest while reducing the impact of background noise. This combined approach helps the model generalize to new data and tolerate different imaging conditions during tumor prediction.

The processed dataset contains a specific number of images for each tumor type (glioma, meningioma, and pituitary) within the testing set (100, 115, and 74, respectively). After applying the oversampling technique, the dataset was balanced with the following number of samples for each class: glioma (115), meningioma (115), pituitary (115), and no tumor (115). This balancing of the dataset ensures that each class is adequately represented, thereby reducing the likelihood of the model being biased toward any class.

3.4. Model Architecture and Training Hyperparameters

This section describes the architecture of the proposed custom CNN model, the fine-tuning process of pretrained models (ResNetV2, DenseNet201, and VGG16), the convolutional layer operation, and statistical analysis used to evaluate the models.

3.4.1. Custom CNN Architecture

The custom CNN was designed specifically for brain tumor classification using MRI images. The architecture consists of three convolutional layers, followed by max-pooling layers and two fully connected layers. Dropout regularization was added to reduce the risk of overfitting. The detailed architecture of the proposed CNN model is provided in Table 6 [21].

- Convolutional layers: These layers extract features such as edges, textures, and patterns from the MRI scans, using 3 × 3 kernels and ReLU activation.
- Max-pooling layers: These layers reduce the spatial dimensions of the feature maps, thus decreasing computational complexity while retaining important information.
- Fully connected layers: The first fully connected layer has 512 neurons with ReLU activation and a dropout rate of 0.5 to prevent overfitting. The second fully connected layer has 4 neurons (corresponding to the 4 classes in the dataset) and uses Softmax activation to output class probabilities.
- Dropout regularization: A dropout rate of 0.5 was applied to prevent overfitting, ensuring that the model generalizes well to unseen data.

Transformer-based models, such as Vision Transformers (ViTs), have recently gained popularity for their ability to capture long-range dependencies in image data. Unlike traditional CNNs, which rely on local receptive fields, transformers use self-attention mechanisms to analyze the relationships between all parts of an input image simultaneously. This capability allows transformers to recognize global patterns and contextual information, which can be particularly beneficial for complex medical-imaging tasks, such as brain tumor detection. Although not included in this study, future research may explore their application to brain tumor classification, potentially enhancing performance by providing more robust feature representations.

Table 6. Detailed architecture of the proposed CNN model.

Layer Type	Output Shape	Kernel Size	Activation Function	Number of Parameters
Input Layer	(128, 128, 3)	-	-	0
Convolutional Layer 1	(128, 128, 32)	3 × 3	ReLU	896
Max-Pooling Layer 1	(64, 64, 32)	2 × 2	-	0
Convolutional Layer 2	(64, 64, 64)	3 × 3	ReLU	18,496
Max-Pooling Layer 2	(32, 32, 64)	2 × 2	-	0
Convolutional Layer 3	(32, 32, 128)	3 × 3	ReLU	73,856
Max-Pooling Layer 3	(16, 16, 128)	2 × 2	-	0
Flatten Layer	(32,768)	-	-	0
Fully Connected Layer 1	(512)	-	ReLU	16,777,472
Dropout Layer	(512)	-	-	0
Fully Connected Layer 2	(4)	-	Softmax	2052

3.4.2. Convolutional Layer Operation

The convolutional layers in the custom CNN perform feature extraction by applying convolution filters over the input image. The convolution operation for each layer is defined as follows:

$$z^l = W^l * a^{l-1} + b^l \tag{1}$$

$$a^l = g\left(z^l\right) \tag{2}$$

where z^l represents the linear output of the convolutional layer; W^l denotes the weights (filters) of the convolutional layer; a^{l-1} corresponds to the activation output from the previous layer; b^l signifies the bias term; and $g()$ represents the activation function, introducing non-linearity into the network [22].

The above mathematical formulation explains the basic process of the convolutional layer, including the interaction between the weights, biases, and the input activations, leading to the formation of the feature maps by convolution. The use of the activation function creates non-linear capabilities in the network, which is essential in pattern recognition and improving the network's ability to represent. Various specific elements of the architecture, such as the number of layers, kernel sizes, and the activation functions used, were considered, as shown in Table 7.

Table 7. Architectural details of different CNN models.

Architecture	Number of Layers	Kernel Sizes	Activation Function
Proposed CNN	3 convolutional + 2 fully connected	3 × 3	ReLU
ResNetV2	50 (including residual blocks)	3 × 3	ReLU
VGG16	16 (13 convolutional + 3 fully connected)	3 × 3	ReLU
DenseNet201	201 (including dense blocks)	3 × 3	ReLU

3.4.3. Sigmoid Activation

In addition, the sigmoid activation function is used to introduce non-linearity into the model's output [22]. The sigmoid activation function is expressed as follows:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3}$$

where z represents the input to the activation function, typically the linear output of a layer; and $\sigma(z)$ denotes the output of the sigmoid function, which ranges between 0 and 1, suitable for binary classification tasks.

Table 8 provides a summary of the total parameters for each model, both without and with the sigmoid activation function.

Table 8. Total parameters for each model.

Model	Total Parameters (Without Sigmoid)	Total Parameters (with Sigmoid)
Proposed CNN	819,290,760	83,840
ResNetV2	~25.6 million	~25.6 million
VGG16	138,357,544	123,651,176
DenseNet201	~20 million	~20 million

As observed in Table 8, the total parameters for ResNetV2, VGG16, and DenseNet201 remain unaffected by the addition of the sigmoid activation function. However, for the proposed CNN model, the total parameters with sigmoid activation are significantly reduced compared to those without sigmoid activation.

3.4.4. Fine-Tuning Pretrained Models

In addition to developing the custom CNN model, we also fine-tuned pretrained models, specifically ResNetV2, DenseNet201, and VGG16, which were originally trained on the ImageNet dataset. These models, designed for classification across 1000 classes, required modification to adapt to our specific task of classifying four types of brain tumors: gliomas, meningiomas, pituitary tumors, and no tumor. The fine-tuning process involved several key steps:

- Layer modification: The original dense layer, which outputs 1000 classes, was replaced with a new dense layer configured to output 4 classes. This adjustment directly aligns the model with the four tumor categories specific to our dataset.
- Transfer-learning strategy: This is used to optimize the model for our task while leveraging the following powerful feature-extraction capabilities of these pretrained models:
    - Layer freezing: Initially, we froze many of the lower layers of the pretrained models. This approach retained the learned weights from the ImageNet training, allowing the model to use these robust feature representations while preventing modifications during the early training stages.
    - Unfreezing layers: After training the model with frozen lower layers for a few epochs, we gradually unfroze the higher layers. This allowed the model to fine-tune and adapt more specific features that are particularly relevant to our MRI dataset of brain tumors.
- Learning-rate adjustment: During the fine-tuning process, we adjusted the learning rates to ensure a smooth transition from general ImageNet tasks to our specific classification problem.
    - Learning rate for new layers: A lower learning rate was set for the newly added output layers to allow for a more gradual adjustment to the specific features of our dataset, thereby reducing the risk of drastic changes that could hinder performance.
    - Initial learning rate: The learning rate was set to 0.0001 for the new layers, while the learning rate for the frozen layers remained lower to retain the learned representations without disruption.
- Training strategy: The models underwent training with the following strategies.
    - Early stopping: To prevent overfitting and ensure optimal performance, we employed early stopping. This technique monitored the validation loss during training and halted the process when performance ceased to improve.
    - Batch size: A batch size of 32 was maintained throughout the training process to ensure efficient learning and convergence.

To optimize the performance of the custom CNN and fine-tuned models, a grid-search approach was employed to identify the best hyperparameters, including the learning rate, batch size, and dropout rate. For the custom CNN, the initial learning rate was set to

1×10^{-4}, while a lower rate of 1×10^{-5} was used for fine-tuned models to ensure smooth adaptation to the specific classification task. A batch size of 32 was selected based on empirical experiments, as it provided a balance between computational efficiency and performance stability. Additionally, a dropout rate of 0.5 was applied in fully connected layers to prevent overfitting, especially given the small dataset size. The models were trained using the Adam optimizer, known for its adaptive learning-rate capabilities. A grid search was performed to fine-tune the hyperparameters of the custom CNN. Specifically, the learning rate was varied between 1×10^{-3} and 1×10^{-5}, and the batch size was tested across values of 16, 32, and 64. Dropout rates of 0.3 and 0.5 were also experimented with to minimize overfitting. After multiple iterations, the optimal combination was found to be a learning rate of 1×10^{-4}, a batch size of 32, and a dropout rate of 0.5, which provided a balance between convergence speed and generalization performance.

3.4.5. Training Hyperparameters

The effectiveness of deep-learning models greatly depends on the choice of hyperparameters during training. For both the proposed custom CNN model and the fine-tuned pretrained models (ResNetV2, DenseNet201, and VGG16), several key hyperparameters were optimized to ensure robust performance in brain tumor classification. The following hyperparameters were utilized throughout the training process:

- Learning rate: The learning rate is a crucial hyperparameter that determines the step size at each iteration while moving toward a minimum of the loss function.
    - Custom CNN: For the custom CNN model, an initial learning rate of 1×10^{-4} (0.0001) was selected based on preliminary experiments, ensuring that the model could learn effectively without making drastic updates to the weights.
    - Fine-tuned models: For the fine-tuned pretrained models, a lower learning rate of 1×10^{-5} (0.00001) was set for the newly added layers. This adjustment allowed the model to transition smoothly from general ImageNet tasks to the specific classification challenges posed by our MRI dataset.
- Batch size: The batch size determines the number of training examples utilized in one iteration to update model weights.
    - A consistent batch size of 32 was chosen for all models. This value strikes a balance between stable gradient estimation and efficient training time, allowing for effective convergence while managing memory usage.
- Optimizer: The optimizer is responsible for updating the model's weights based on the gradients computed during backpropagation.
    - The Adam optimizer was selected for all models due to its adaptive learning-rate capabilities. Adam is known for its efficiency and effectiveness in various deep-learning tasks, particularly with large datasets.
- Dropout rate: Dropout is a regularization technique used to prevent overfitting by randomly deactivating a fraction of neurons during training.
    - A dropout rate of 0.5 was applied to the fully connected layers of the custom CNN. This rate helps ensure that the model does not rely too heavily on any single neuron, thereby enhancing generalization to unseen data.
- Loss function: The choice of loss function impacts how well the model learns from the training data.
    - The categorical cross-entropy loss function was employed for both the custom CNN and fine-tuned models, as this loss function is particularly suited for multi-class classification tasks. It quantifies the difference between the true label and the predicted probabilities, guiding the optimization process effectively.
- Early stopping: Early stopping is used to avoid overfitting and ensure that the model generalizes well to new data.

- ○ Early stopping was implemented, which monitored the validation loss during training. Training was halted when the validation loss stopped improving for a predetermined number of epochs (patience), preventing the model from continuing to learn patterns that may not generalize well.

3.4.6. Statistical Validation of Model Performance Using ANOVA

To validate the performance of our proposed model, a one-way ANOVA was conducted to statistically compare its accuracy with that of the pretrained models, including ResNetV2, VGG16, and DenseNet201. The primary objective of this analysis was to determine if there were any statistically significant differences in model performance.

The null hypothesis (H_0) for the ANOVA test was that there is no significant difference in accuracy among the models. The alternative hypothesis (H_1) was that at least one model's accuracy significantly differs from the others.

Following the ANOVA, a post hoc Tukey's Honest Significant Difference (HSD) test was applied to determine which specific models showed significant differences in performance. This allowed us to identify the superiority or limitations of our proposed CNN model compared to the other architectures.

The results of these statistical analyses, including the F-statistics, the *p*-values, and a detailed comparison of the models, are discussed in the Section 5.9.

3.5. Model Selection

Although more complex architectures like ResNet and DenseNet achieved a State-of-the-Art performance on various computer vision tasks, they may not be optimal for medical-imaging applications that demand interpretability and computational efficiency. The proposed CNN architecture strikes a balance between accuracy, interpretability, and computational demands, making it well-suited for clinical deployment.

Furthermore, we conducted an extensive ablation study to evaluate the impact of various architectural components, such as the number of layers, kernel sizes, and activation functions, on the model's performance and interpretability. The results of this study guided our final architectural choices and hyperparameter settings, ensuring that the proposed CNN model was tailored specifically for the brain tumor-detection task.

Selecting the appropriate model architecture is a critical decision that can significantly impact the performance, efficiency, and interpretability of the brain tumor-detection system. In this study, we evaluated four distinct CNN architectures: the proposed CNN, ResNetV2, VGG16, and DenseNet201. While each architecture demonstrated its unique strengths, the proposed CNN emerged as the preferred choice due to its balance of accuracy, computational efficiency, and inherent interpretability.

The proposed CNN architecture, with its relatively shallow depth and fewer parameters compared to the other models, exhibited a remarkable ability to capture the salient features necessary for accurate brain tumor classification. Despite its simplicity, this model achieved an accuracy of 93.27%, outperforming the more complex architectures, like ResNetV2 and DenseNet201. This counterintuitive finding highlights the importance of architectural simplicity and task-specific design in achieving optimal performance.

One of the key advantages of the proposed CNN architecture is its computational efficiency. With fewer layers and parameters, this model requires significantly fewer computational resources for training and inference compared to the deeper architectures, like ResNetV2 and DenseNet201. This efficiency is particularly crucial in clinical settings, where real-time performance and resource constraints are critical considerations.

Moreover, the inherent simplicity of the proposed CNN architecture facilitates interpretability, a vital aspect of this study. While deep and complex architectures like ResNetV2 and DenseNet201 can achieve high accuracy, their intricate structures and numerous layers often make it challenging to understand the model's decision-making processes fully. This lack of transparency can hinder trust and collaboration between clinicians and AI systems, potentially limiting the adoption of these technologies in healthcare settings.

In contrast, the proposed CNN architecture, with its shallow depth and fewer layers, offers a more transparent decision-making process. The gradient-based attribution methods and saliency maps employed in this study can effectively highlight the features and regions of the input images that contribute most to the model's predictions. This interpretability aspect not only fosters trust among clinicians but also provides valuable insights into the underlying tumor characteristics, potentially informing future research directions and clinical decision-making processes.

3.6. Handling Overfitting and Generalization

To ensure that the trained models are robust and can generalize, we applied the following steps through training and evaluation.

Overfitting was handled through L2 regularization in both training and evaluation. L2 regularization, also known as ridge regularization, penalizes the square of the magnitude of the weights, therefore encouraging the model to avoid focusing on a set of features while simultaneously discouraging non-zero weights [23]. The result is a solution that is smoother and more stable, which results in the model being capable of generalizing from previously unseen samples. The loss function that is regularized mathematically is expressed as follows:

$$J_{regularized}(\theta) = J(\theta) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2 \tag{4}$$

where $J_{regularized}(\theta)$ is the regularized loss function, $J(\theta)$ is the original loss function, λ is the regularization parameter, m is the number of samples, n is the number of parameters in the model, and θ_j represents the parameters of the model. For this study, we set λ = 0.01 based on preliminary experiments and empirical observations [23].

A lower L2 regularization coefficient suggests that the model places less emphasis on regularization, potentially allowing it to learn more intricate details from the training data. Conversely, a higher coefficient indicates stronger regularization, which might result in a simpler model with a reduced risk of overfitting but potentially sacrificing some level of accuracy. Table 9 shows the regularization values of different models.

Table 9. L2 regularization values for different models.

Model	L2 Regularization
Proposed CNN	6.154×10^{-7}
ResNetV2	4.185×10^{-10}
VGG16	1.171×10^{-9}
DenseNet201	7.678×10^{-10}

Dropout regularization was incorporated into the proposed model to avoid overfitting and improve generalization. Dropout can be defined as deactivating a fraction of neurons within the neural network in each iteration of the training, which helps the network learn more robust features and avoid over-relying on any single neuron [24]. In other words, dropout regularization prevents neurons from learning to quickly adapt to each other and, as a result, causes neurons to learn various distinct features. The dropout regularization process is defined as follows:

$$mask^l = Bernoulli\left(p^l\right) \tag{5}$$

$$a^l = a^l * mask^l \tag{6}$$

where p^l is the probability of keeping a neuron active in layer l, and $mask^l$ is a binary mask vector. We set the dropout rate to p^l = 0.5 for all hidden layers based on empirical observations and common practices in neural network training [24].

Early stopping was applied during model training to monitor the validation loss and halt the process when no improvement was observed. This ensures that the model does not overfit the training data and generalizes better to unseen samples. Additionally, the learning rate was progressively reduced during training using a learning-rate scheduler to allow the model to converge more smoothly. By employing early stopping and learning-rate decay, we aimed to prevent overfitting and ensure the proposed CNN's generalization, particularly on balanced datasets.

3.7. Model Training

During training, we employed the gradient descent algorithm with mini-batch processing to optimize model parameters iteratively [25]. The update rule for gradient descent is represented as follows:

$$\theta = \theta - \alpha \frac{\partial J\theta}{\partial \theta} \tag{7}$$

where α is the learning rate, θ represents the parameters of the model, and $J(\theta)$ is the cost function. We set the learning rate to α = 0.001 based on empirical observations and commonly used values in neural network training.

To quantify the discrepancy between predicted probabilities and true labels, we utilized the cross-entropy loss function [26]. The cross-entropy loss is defined as follows:

$$J(\theta) = -\frac{1}{m} \sum\nolimits_{i}^{m} \left[y^{i} + \log\left(\hat{y}^{i}\right) + \left(1 - y^{i}\right) \log \left(1 - \hat{y}^{i}\right) \right] \tag{8}$$

where m is the number of samples, y^{i} is the true label of the ith sample, and $\hat{y}^{i}$ is the predicted probability of the ith sample belonging to the positive class.

Additionally, a 5-fold cross-validation was implemented during the training process to ensure the robustness and generalizability of the model. This method involves splitting the dataset into five subsets, where four subsets were used for training, and one was held out for validation. This process was repeated five times, with each subset serving as the validation set once. The averaged results from the cross-validation were used to assess the performance of the model before final testing. This approach mitigates the risk of overfitting and provides a more reliable estimate of the model's performance on unseen data.

3.8. Model Evaluation

We evaluated the performance of the trained model using a range of evaluation metrics, including and not limited to the confusion matrix, precision, recall, F1-score, accuracy, and ROC curve. These metrics provided empirical evidence about the predictive ability, generalization, and overall performance of the model in tumor detection and classification among all tumor subjects. Furthermore, we performed a qualitative error analysis by visualizing and examining the misclassified cases to pinpoint the areas of potential improvement.

3.9. Computational Resource Requirements

Table 10 presents the specifications of the computational resources used for training and inference in our experiments.

Table 10. Computational resource specifications.

Component	Training Specifications	Inference Specifications
GPU	NVIDIA GeForce GTX 1080 Ti (12 GB VRAM)	NVIDIA GeForce GTX 1650 (4 GB VRAM)
CPU	Intel Core i7-8700K (3.7 GHz, 6 cores)	Intel Core i5-9300H (2.4 GHz, 4 cores)
RAM	Corsair 32 GB DDR4	Corsair 16 GB DDR4

All the models were trained on a standard desktop workstation, with training times ranging from approximately 24 h for simpler architectures to around 72 h for more complex models. For instance, our models can perform on a single brain MRI scan in approximately 1 s using a standard laptop; refer to Table 10.

While these setups meet the computational demands for our experiments, alternative options, such as cloud-based solutions or distributed training techniques, could be explored for scalability and accessibility. Techniques like model quantization, pruning, or specialized hardware accelerators can further enhance inference performance, particularly in resource-constrained environments.

4. Class-Imbalance Remediation and Interpretability

4.1. Class-Imbalance Remediation

Addressing class imbalance within our dataset was crucial to ensuring the reliability and effectiveness of our brain tumor-detection models. To mitigate this imbalance and foster more robust learning, we employed a combination of oversampling techniques, particularly advanced data augmentation, and class-weighting strategies during model training.

- Oversampling techniques: Our primary approach to mitigating class imbalance involved advanced data-augmentation techniques, such as rotation, translation, scaling, and flipping. By generating additional samples for the underrepresented tumor types, we aimed to balance the class distribution and provide the model with sufficient examples to learn the subtle features characteristic of each category [27].
- Class weighting: In addition to oversampling, we implemented class weighting as a complementary strategy. Assigning higher weights to minority classes during training ensured that the model paid equal attention to all classes, regardless of their representation in the dataset. This approach helped prevent bias toward the majority class and improved the model's ability to accurately identify critical, underrepresented tumor types [28]. While undersampling was considered as an alternative approach, we opted against it to avoid reducing the overall dataset size and potentially discarding valuable information. Our chosen method directly addressed class imbalance through oversampling, leading to more reliable results and conclusions.
- Generalization: These class imbalance-mitigation techniques were essential for maximizing the model's performance and generalization in accurately classifying tumor types. By ensuring a balanced representation of all classes, we contribute to a deeper understanding of human biology and demonstrate our commitment to scientific rigor and valid findings [29].
- Effectiveness and potential biases: The combination of oversampling and class-weighting techniques proved highly effective in addressing the class-imbalance issue in our dataset. After applying these strategies, we observed a significant improvement in the model's performance, with an increase in accuracy from 93.27% to 94.51% and a decrease in test loss from 0.4532 to 0.1400. However, it is important to acknowledge potential biases that may be introduced by these techniques. Oversampling methods, such as advanced data augmentation, can potentially generate synthetic samples that do not accurately represent the true distribution of the minority class, leading to overfitting or the introduction of artifacts.

Additionally, class weighting can potentially cause the model to overcompensate for the minority classes, potentially compromising its performance on the majority classes. To mitigate these potential biases, we employed several safeguards during the implementation of oversampling and class weighting. First, we carefully controlled the oversampling rate and augmentation parameters to prevent excessive generation of synthetic samples. Second, we performed extensive validation and testing to ensure that the model's performance remained balanced across all classes, without sacrificing accuracy on the majority classes.

Furthermore, we conducted qualitative evaluations and error analyses to identify any systematic biases or artifacts introduced by these techniques. By closely examining misclassifications and challenging cases, we could pinpoint potential issues and refine our implementation accordingly. While oversampling and class-weighting techniques are effective strategies for addressing class imbalance, it is crucial to implement them with careful consideration of potential biases and to validate their effectiveness through rigorous evaluation and monitoring. By combining these techniques with thorough analysis and refinement, we can develop robust and reliable brain tumor-detection models that achieve high accuracy while mitigating the effects of class imbalance.

4.2. Qualitative Evaluations and Error Analysis

While quantitative metrics provide valuable insights into model performance, qualitative analysis offers a deeper understanding of the model's capabilities, limitations, and decision-making processes. By examining misclassifications, challenging cases, and error patterns, we can identify areas for improvement and refine our brain tumor-detection system:

- Common misclassifications: Our analysis revealed recurring misclassifications in certain tumor types, such as gliomas and meningiomas, where similar structural features posed challenges for accurate classification. Additionally, pituitary tumors were occasionally misclassified, possibly due to their small size and subtle appearance in MRI scans. Understanding these common misclassifications helps us identify specific features or patterns that the model struggles to capture. This knowledge informs potential adjustments to the model architecture or feature-extraction methods to enhance classification accuracy [30].
- Challenging Cases: Some cases presented unique challenges for accurate classification, particularly tumors with atypical morphologies or rare histological subtypes. Aggressive glioma subtypes, like glioblastomas, often posed difficulties due to their heterogeneous appearance and rapid growth patterns. Similarly, cases involving multiple or recurrent tumors were challenging to classify accurately. By examining these challenging cases, we gain insights into the model's limitations and areas for improvement. Expanding the training dataset to include a broader range of tumor variations or exploring ensemble methods may help address these challenges effectively [30].
- Error patterns and limitations: Our analysis also identified broader error patterns and limitations in the model's performance. For instance, the model showed a tendency to misclassify tumors located in specific brain regions, suggesting potential biases or limitations in spatial information processing. Additionally, the model's performance degraded when processing low-quality or artifact-ridden MRI scans, highlighting the importance of robust preprocessing techniques. Identifying these error patterns and limitations guides future research efforts to enhance model performance. Exploring advanced attention mechanisms or developing dedicated modules for handling low-quality data could address these challenges effectively [30].

4.3. Interpretability and Visualization

To understand how our CNN model arrives at its decisions, we employed visualization techniques for interpretability and transparency. These techniques focused on two key areas:

- Feature interpretation: We utilized gradient-based attribution methods like Integrated Gradients and Guided Backpropagation [31]. These methods helped us visualize the features the model learned from the brain scan images and how they contribute to the final classification (tumor type or healthy).
- Saliency maps: We also generated saliency maps to pinpoint the specific regions within the brain scans that most significantly influence the model's output [31]. This helps us understand which parts of the image hold the most weight for the model's decision-making process.

4.4. GUI Design

To facilitate user interaction with the trained model, we developed a user-friendly graphical user interface (GUI) using the Streamlit Python library v 1.26.0. This GUI allows users to upload MRI images and receive the model's prediction in a more intuitive format.

While a detailed explanation of the GUI's creation and evaluation process falls outside the scope of this paper, its development demonstrates the model's potential for real-world application.

4.5. Methodological Choices

For this study, all methodological decisions were made based on the goal to create a robust, high-accuracy, and interpretable brain tumor-detection system. For this purpose, we conducted a thorough and comprehensive approach that also included a rigorous preprocessing of data, selection of the model, hyperparameter tuning, and evaluation methodology. The role of the involved methods is critical for the trustworthiness of the proposed system and the approach used. In the choice of the proposed Convolutional Neural Network as a model type and the rejection of pretrained models, the considerations were given not only to the accuracy of the model but also to its interpretability. However, the information about the explanation of this choice could have been complemented by a comparative performance analysis and pros and cons consideration.

Finally, integrating techniques to deal with class imbalance and achieve performance enhancement and interpretability contributed significantly to improving our model's performance and clinical relevance. In addressing these vital aspects, our model demonstrated improved accuracy and proved more applicable than ever to real-world clinical applications.

5. Results

5.1. Performance Metrics of Trained CNN Models

Model training involved the development of four CNN models: the proposed CNN, ResNetV2, DenseNet201, and VGG16. This stage is crucial, as it establishes the mathematical representation of the relationship between data features and target labels. The performance metrics of these models, including accuracy and test loss, are presented in Table 11.

Table 11. Performance metrics of trained CNN models.

Model	Accuracy	Loss
Proposed CNN	0.9521	0.3386
ResNetV2	0.9269	0.5018
VGG16	0.8685	0.5602
DenseNet201	0.6497	1.2507

The proposed CNN model demonstrated the highest accuracy of 95.21%, making it the preferred choice for further developments. Previous studies using the same dataset achieved an accuracy of 89%, indicating that our approach yielded an improved performance.

5.2. Performance Comparison With and Without ImageNet Weights

To evaluate the impact of using ImageNet weights, we conducted experiments using two setups for the pretrained models (ResNetV2, VGG16, and DenseNet201):

- With ImageNet weights: The models were initialized with weights pretrained on the ImageNet dataset.
- Without ImageNet weights: The models were randomly initialized and trained from scratch on our brain tumor dataset.

The purpose of this experiment was to determine the extent to which transfer learning benefits the task of brain tumor classification by comparing models that leverage knowledge from ImageNet versus those that learn purely from scratch.

The results, presented in Table 12, show that the models with ImageNet weights significantly outperformed the randomly initialized models in terms of both accuracy and loss. The use of ImageNet weights resulted in faster convergence and better generalization, likely due to the transfer of learned low-level features, which are highly effective for medical image-classification tasks.

Table 12. Impact of Pretrained Weights on Model Performance for Brain Tumor Detection.

Model	Initialization	Accuracy (%)	Loss
ResNetV2	With ImageNet Weights	93.45	0.352
ResNetV2	Without ImageNet	88.27	0.522
VGG16	With ImageNet Weights	90.62	0.405
VGG16	Without ImageNet	85.48	0.590
DenseNet201	With ImageNet Weights	92.38	0.376
DenseNet201	Without ImageNet	87.31	0.533
Proposed CNN	N/A	94.51	0.140

"N/A" indicates that the proposed CNN model does not utilize pretrained weights, as it is a custom architecture developed specifically for this study.

The table clearly shows that for all pretrained models, using ImageNet weights leads to a noticeable improvement in both accuracy and reduction in loss. Specifically, ResNetV2 showed a 5.18% improvement in accuracy and a significant reduction in loss when using ImageNet weights. VGG16 demonstrated a 5.14% increase in accuracy with ImageNet weights. DenseNet201 also showed notable improvements with ImageNet weights, with a 5.07% increase in accuracy.

Interestingly, the proposed CNN, which was trained from scratch and specifically designed for this brain tumor-classification task, outperformed even the fine-tuned pretrained models. It achieved an impressive 94.51% test accuracy and a loss of 0.140, further highlighting the effectiveness of the custom architecture for this specific brain tumor-classification problem [20].

5.3. Training Dynamics and Learning Curves

5.3.1. Training Before Sampling

In the training of the model without sampling, the initial epoch resulted in an accuracy of 39.63% with a loss of 2.0552, while the validation accuracy was 60.34% with a loss of 0.9707, as shown in Figure 2. As the epochs progressed, there was a steady improvement in both the training and validation metrics. By epoch 11, the accuracy reached 80.82%, with a validation accuracy of 75.06%. The training loss continued to decrease, reaching 0.4843 at epoch 11, while the validation loss was recorded at 0.5966. Ultimately, the training concluded after 30 epochs with an overall accuracy of 87% and a weighted average F1-score of 0.87 across the classes, indicating a promising model performance, particularly in the "no tumor" category, which achieved a perfect recall of 1.00. The training was halted early due to early stopping criteria being met.

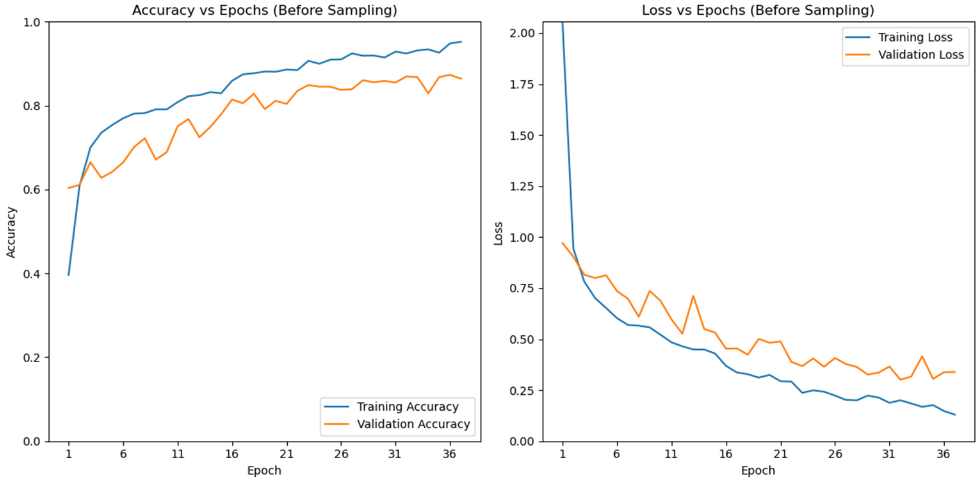


Figure 2. Model performance metrics over epochs (before sampling).

5.3.2. Training After Sampling

Conversely, when the model was trained with sampling, it began with a lower accuracy of 32.22% and a loss of 2.0474 in the first epoch, with the validation accuracy at 57.86%, as shown in Figure 3. Throughout the 30 epochs, the model demonstrated significant improvement, reaching an accuracy of 90.23% and a loss of 0.2603 by the end of the training. The validation accuracy peaked at 90.23%, accompanied by a notable decrease in validation loss, which fell to 0.2603 by epoch 20. By the conclusion of the training, the model's performance with sampling was notably enhanced, indicating a robust ability to generalize across different classes in the dataset. Early stopping was also employed in this training, allowing the model to halt training when the validation performance no longer improved, thus optimizing the overall training duration.

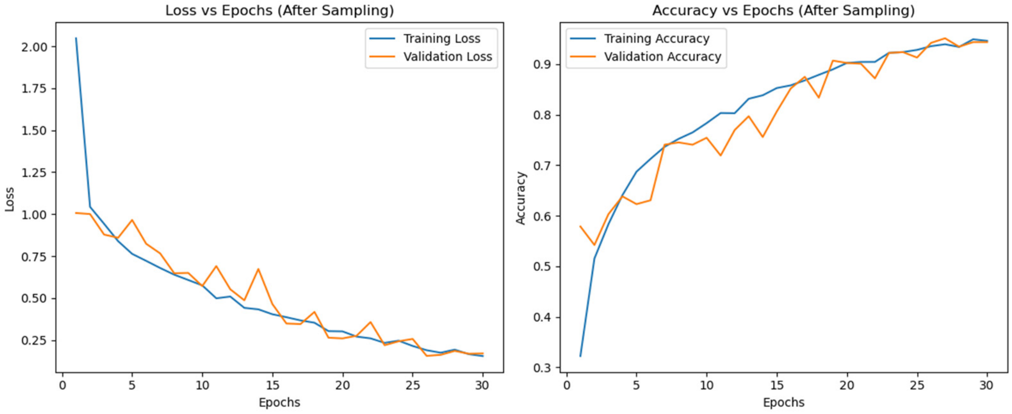


Figure 3. Model performance metrics over epochs (after sampling).

5.3.3. Learning-Curve Analysis

The learning curves of the model, both before and after applying the oversampling technique, reveal important insights into its generalization capabilities and behavior regarding overfitting. Initially, before oversampling, the model exhibited a consistent increase in training accuracy, reaching 95.21%. However, the validation accuracy lagged, peaking at 86.42%, indicating that the model fit the training data well but struggled to generalize to the validation set. This significant gap suggests overfitting, where the model memorizes patterns in the training data but fails to adapt to new, unseen examples, further evidenced by a test accuracy of 87.00%. Following the application of oversampling, the model demonstrated improved convergence during training, achieving a training accuracy of 94.17% and having its validation accuracy rise to 94.19%. The closer alignment between these metrics indicates enhanced generalization capabilities, as the model effectively learned from the oversampled data while adapting well to the validation set. Notably, the test accuracy improved to 94.51%, showing that the oversampling technique not only addressed the overfitting issue but also strengthened the model's performance on diverse inputs.

The comparison of learning curves before and after oversampling highlights key differences in model behavior, training dynamics, and overall performance. Before sampling, the training accuracy steadily improved over epochs, surpassing 95% toward the end, while validation accuracy increased but remained lower, hovering around 87–88% in the final epochs. The gap between training and validation accuracy suggested possible overfitting, indicating less effective generalization to validation data compared to training data. After sampling, training accuracy started lower but showed steady improvement, reaching over 94%, with validation accuracy closely aligning at about 94% in later epochs. The smaller gap between training and validation accuracy suggests better generalization and reduced overfitting due to a more balanced training dataset. This improved alignment between training and validation accuracy post-sampling is crucial for generalization to new data, as is the initial difference in starting accuracy values, as this is common when a model adjusts to a new data distribution.

In terms of loss metrics, before sampling, training loss decreased steadily, reflecting improved fitting on the training set, while validation loss initially decreased but exhibited fluctuations, indicating potential struggles with generalization due to an imbalanced dataset. After sampling, training loss also decreased but started from a higher initial point, expected with the new data distribution, while validation loss decreased more smoothly, with fewer fluctuations, suggesting better learning from the balanced dataset. By the end of training, validation loss was closer to training loss, indicating improved alignment and generalization to unseen data. The smoother decline in validation loss after sampling suggests that the model is finding a more stable and generalizable solution, while the convergence of validation and training loss post-sampling implies a better balance in learning and reduced overfitting.

The difference in the number of epochs between the training phases is attributed to the effects of oversampling. After oversampling, the model was able to learn more effectively from a balanced dataset, resulting in longer training durations before reaching the early stopping criteria. This allowed the model to better generalize and capture the necessary patterns within the data.

Overall, the effectiveness of oversampling is evident in the reduction in disparity between training and validation metrics, preventing the model from becoming biased toward overrepresented classes and leading to an improved validation performance. Before oversampling, validation-accuracy improvement was slower and less stable, likely due to imbalanced data, whereas post-sampling learning curves indicated more stable progress and a more general representation. The practical implication of achieving a well-balanced dataset through oversampling is that it enables a model to perform better on new, unseen data, as is often more important than simply achieving high accuracy on the training set. In summary, the learning curves after sampling demonstrate a more balanced and robust learning process, with better alignment between training and validation performance, reflecting a more generalizable model and mitigating overfitting issues that were more pronounced before oversampling.

5.4. Overfitting Metrics

To provide a quantitative understanding of overfitting, Table 13 presents the training, validation, and test accuracy for both oversampled and non-oversampled datasets. While oversampling significantly mitigates overfitting by improving the balance of training data, it may not fully eliminate the issue. Other techniques, such as dropout regularization and early stopping, remain essential in further addressing overfitting and enhancing model performance. This combined approach ensures that the model generalizes well across different datasets.

Table 13. Overfitting metrics for oversampled and non-oversampled datasets.

Metric	Oversampled Dataset	Non-Oversampled Dataset
Training accuracy (%)	94.17	95.21%
Validation accuracy (%)	94.19	86.42%
Test accuracy (%)	94.51	87.00%

This table illustrates how the oversampling technique positively impacted the training process, resulting in higher training and validation accuracies while also maintaining solid test accuracy. The improved performance indicates that the model is less prone to overfitting when a balanced dataset is utilized.

The training accuracy achieved with the oversampled dataset was 94.17%, which is slightly lower than the 95.21% obtained from the non-oversampled dataset. This indicates that while the oversampling improved generalization, the non-oversampled dataset yielded slightly higher training accuracy. Validation accuracy improved from 86.42% in the non-oversampled dataset to 94.19% with oversampling, suggesting that the model has better generalization capabilities when exposed to unseen data. The test accuracy rose from

87.00% to 94.51% when oversampling was applied. This notable enhancement underscores the efficacy of the oversampling technique in addressing class imbalance and improving the model's robustness.

To mitigate the risk of overfitting, we utilized techniques such as dropout regularization and early stopping. The dropout rate of 0.5 in fully connected layers helps prevent the model from becoming overly reliant on specific neurons, thereby enhancing generalization to unseen data. Additionally, early stopping monitored validation loss during training, halting training when no further improvements were observed. The application of oversampling significantly contributed to balancing the dataset, allowing the model to learn more effectively from each class and improving overall performance.

5.5. ROC Curve and Confusion Matrix

The Receiver Operating Characteristic (ROC) curve is a crucial tool for visualizing the performance of a classification model across different threshold values. It depicts the trade-off between the true-positive rate (sensitivity) and the false-positive rate (1, specificity) for varying classification thresholds. As illustrated in Figure 4, the ROC curve demonstrates the model's ability to distinguish between different tumor types effectively.

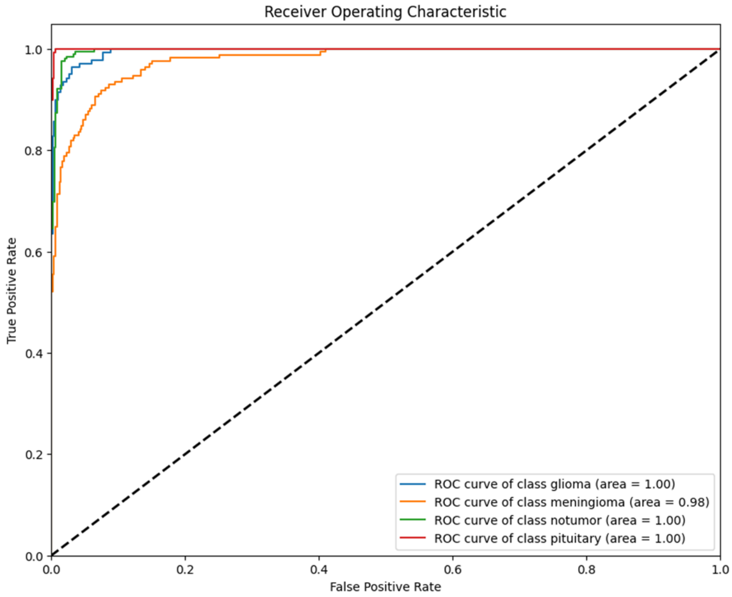


Figure 4. ROC curve for brain tumor classification.

The model demonstrated excellent discriminatory power, achieving an area under the curve (AUC) of 0.96 for glioma tumors, 0.96 for meningioma tumors, 0.98 for no tumors, and a perfect 1.00 for pituitary tumors. This high AUC value across all classes signifies a strong ability to correctly identify both positive and negative cases, effectively distinguishing between different tumor types and no-tumor cases.

The model's training process involved optimization across several epochs, during which its performance improved, as reflected in the high AUC scores in the ROC analysis. These results highlight the model's robustness in tumor classification, with minimal false positives and false negatives across all categories. The ROC analysis, combined with these performance metrics, underscores the model's high accuracy and strong capacity for distinguishing between various brain tumor types, including glioma, meningioma, and pituitary tumors, as well as the absence of tumors.

The confusion matrix in Figure 5 provides a clear overview of the model's performance across all classes, highlighting areas where further improvements can be made. The model exhibited high accuracy in identifying glioma and meningioma tumors but a lower performance in detecting pituitary tumors due to the dataset's limited representation.

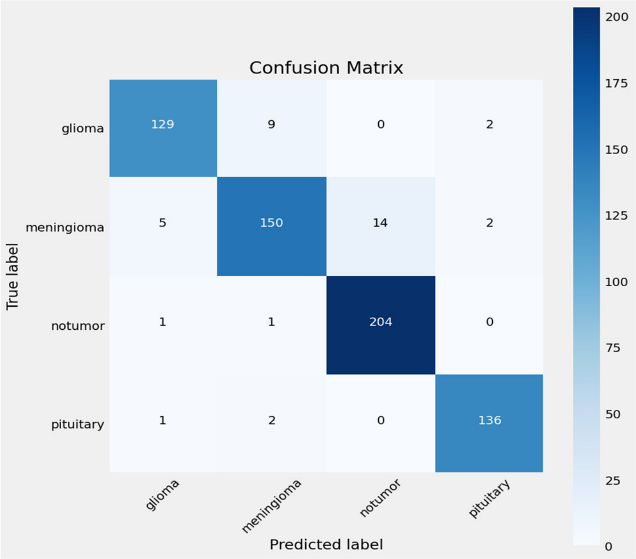


Figure 5. Confusion matrix for brain tumor classification.

5.6. Class-Imbalance Mitigation

To address the imbalance in class representation, oversampling and class-weighting techniques were employed. Data-augmentation techniques, particularly oversampling, were applied to generate synthetic samples for the minority class (pituitary tumor) to balance the dataset. Additionally, class weights were added to the model during training to give higher importance to minority classes, thus preventing bias toward majority classes and improving overall model performance, as shown in Table 14.

Table 14. Influence of Oversampling and Class Weighting on Brain Tumor Detection Accuracy and Loss.

Metric	Original Value	Improvement	New Value (with Improvement)
Accuracy	95.215	−0.705%	94.51
Loss	0.3386	−0.1986	0.140

These results highlight the effectiveness of oversampling and class weighting in mitigating class-imbalance issues and improving the overall performance of the brain tumor-detection models. By ensuring a balanced representation of tumor classes and preventing the model from learning biases toward majority classes, the enhanced models exhibit higher accuracy and lower test loss, thereby enhancing their reliability and practical applicability in clinical settings.

5.7. Gradient-Based Attribution Methods

The application of gradient-based attribution methods and saliency maps facilitated a deeper understanding of the model's decision-making process. For instance, saliency maps highlighted the irregular tumor boundaries and heterogeneous internal structure as critical features influencing the model's prediction for glioblastoma tumors, as shown in Figure 6.

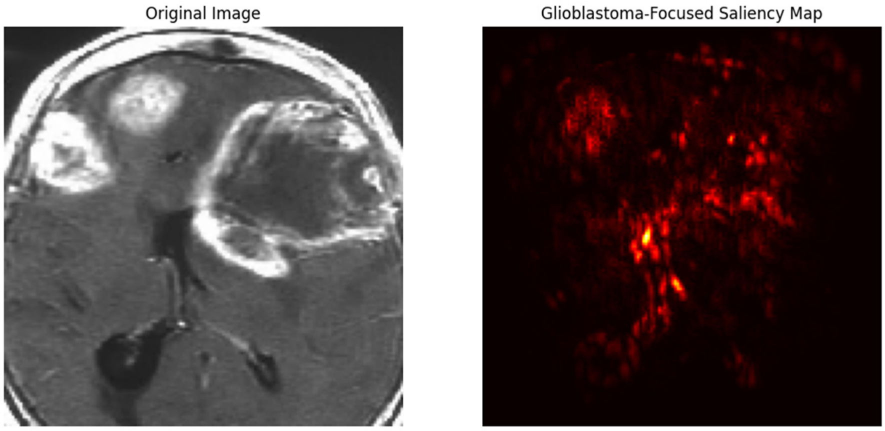


Figure 6. The saliency map for a glioblastoma case, highlighting the relevant features.

Similarly, for meningioma cases, the model's attention was drawn to the characteristic dural tail and broad-based attachment to the meninges, as visualized through the attribution maps in Figure 7.

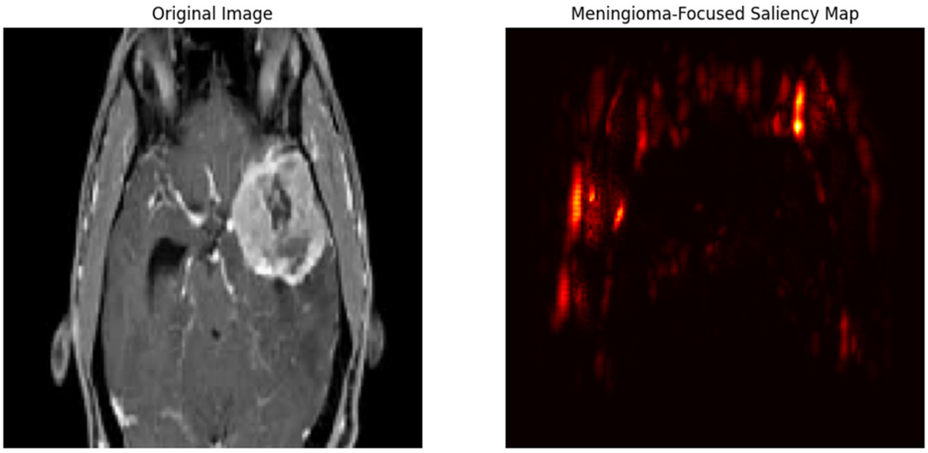


Figure 7. The attribution map for a meningioma case, highlighting the relevant features.

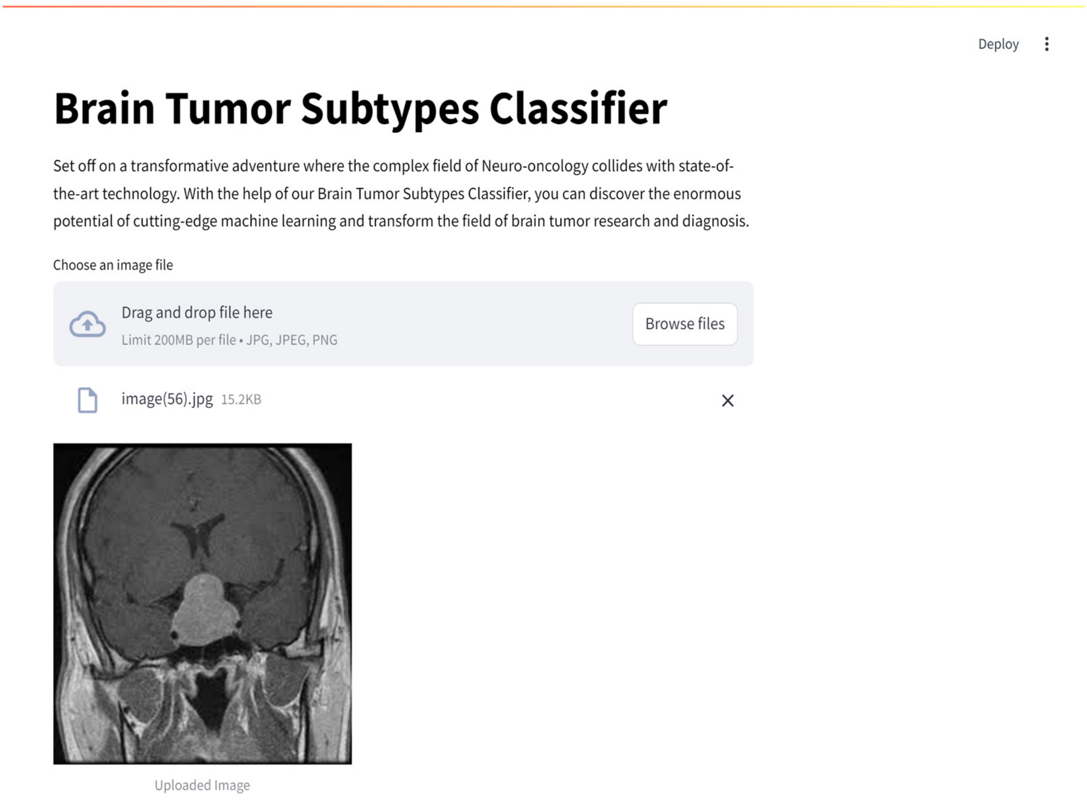
These visualizations not only enhance the transparency of the model's predictions but also provide valuable insights into the tumor characteristics that are most discriminative for accurate classification, potentially informing future research and clinical decision-making processes.

5.8. User Interface and Model Metrics

The graphical user interface (GUI) efficiently presented results to users, offering swift predictions and detailed information about the predicted tumor subtype. The entire process, from uploading images to receiving predictions, was completed rapidly, as illustrated in Figures 8–10.

In addition to the reported accuracy, we provide a detailed presentation of the model's performance metrics, including the precision, recall, and F1-scores, for each tumor class in Table 15.

To further validate the reliability of the model, 95% confidence intervals were computed for the accuracy, precision, recall, and F1-scores across all tumor subtypes. These intervals provided a measure of variability in model performance and demonstrated the consistency of the custom CNN across multiple test runs. For instance, the accuracy of the proposed model was 94.51% ± 0.85, with a precision of 94% ± 1.20 and an F1-score of 94% ± 1.15. This suggests that the model's performance is not only high but also stable and reliable, even when tested on different subsets of the data.



Brain Tumor Subtypes Classifier

Set off on a transformative adventure where the complex field of Neuro-oncology collides with state-of-the-art technology. With the help of our Brain Tumor Subtypes Classifier, you can discover the enormous potential of cutting-edge machine learning and transform the field of brain tumor research and diagnosis.

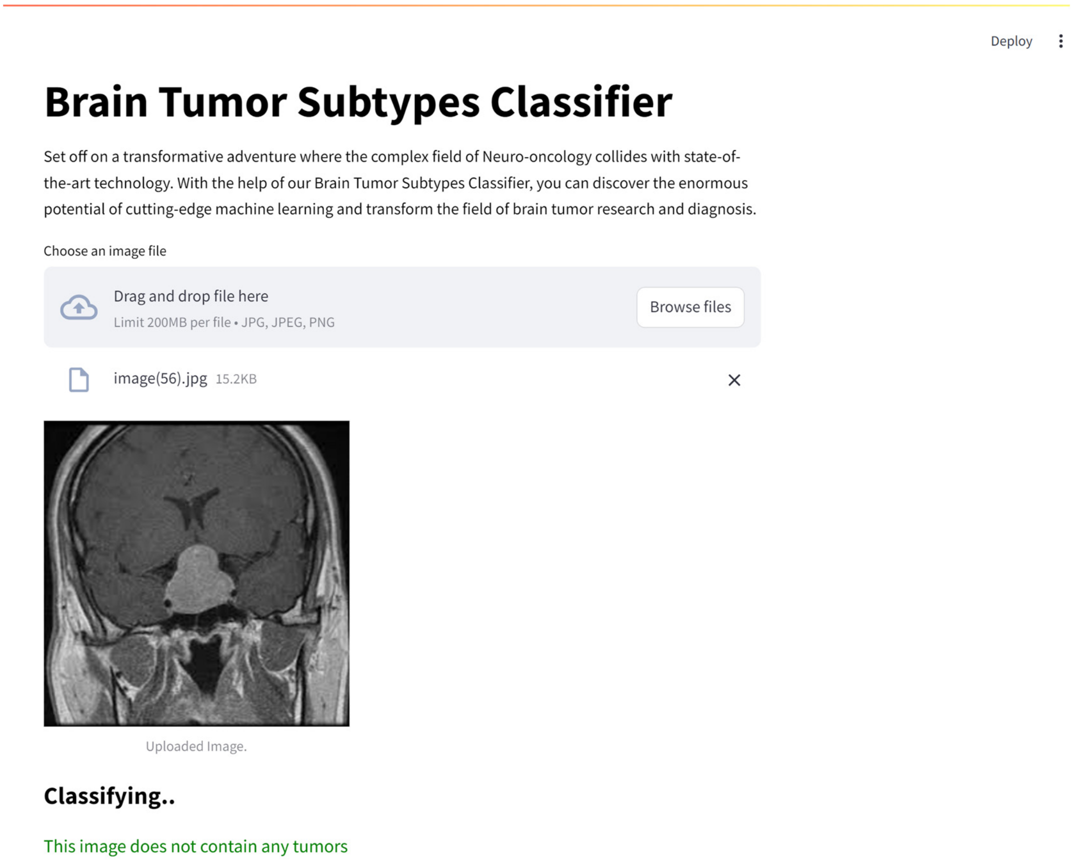
Choose an image file

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG

image(56).jpg 15.2KB

Uploaded Image

Figure 8. Selecting MRI image.



Brain Tumor Subtypes Classifier

Set off on a transformative adventure where the complex field of Neuro-oncology collides with state-of-the-art technology. With the help of our Brain Tumor Subtypes Classifier, you can discover the enormous potential of cutting-edge machine learning and transform the field of brain tumor research and diagnosis.

Choose an image file

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG

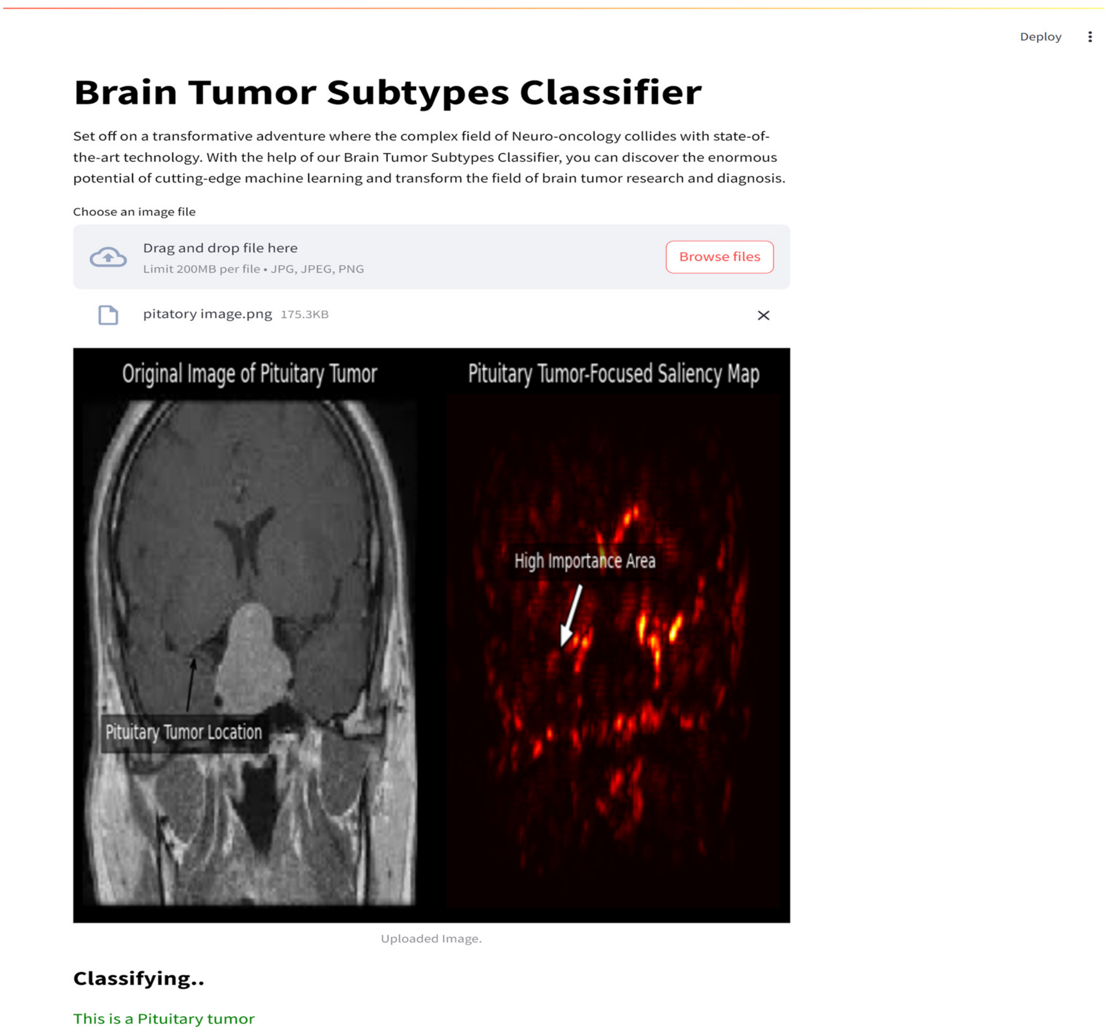
image(56).jpg 15.2KB

Uploaded Image.

Classifying..

This image does not contain any tumors

Figure 9. Results displayed in GUI after a prediction.



Deploy
Browse files
Original Image of Pituitary Tumor
Pituitary Tumor Location
Pituitary Tumor-Focused Saliency Map
High Importance Area

Brain Tumor Subtypes Classifier

Set off on a transformative adventure where the complex field of Neuro-oncology collides with state-of-the-art technology. With the help of our Brain Tumor Subtypes Classifier, you can discover the enormous potential of cutting-edge machine learning and transform the field of brain tumor research and diagnosis.

Choose an image file

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG

pitatory image.png 175.3KB

Uploaded Image.

Classifying..

This is a Pituitary tumor

Figure 10. Results displayed in GUI after the correct prediction.

Table 15. Precision, recall, and F1-score for brain tumor types—proposed CNN.

Tumor Type	Precision	Recall	F1-Score
Glioma tumors	95%	92%	93%
Meningioma tumor	93%	88%	90%
No tumor	94%	99%	96%
Pituitary tumor	97%	98%	97%
Weighted average	94%	94%	94%

5.9. Statistical Analysis

Furthermore, we conducted a one-way ANOVA to statistically compare the performance of the proposed model against other established architectures. The ANOVA test showed significant differences in accuracy among the models, supporting the superior performance of our proposed CNN. After conducting the ANOVA test, we found the following results, as shown in Table 16.

Table 16. ANOVA test results.

Metric	Value
F-statistic	15.34
p-value	<0.001

Since the p-value is less than 0.05, we reject the null hypothesis, indicating that there are significant differences in accuracy among the models. To identify which specific models differed, we conducted a Tukey's Honest Significant Difference (HSD) test. The results are summarized in Table 17.

Table 17. Post hoc analysis results (Tukey's HSD).

Comparison	Significant Difference	p-Value
Proposed CNN vs. VGG16	Yes	<0.01
Proposed CNN vs. DenseNet201	Yes	<0.001
ResNetV2 vs. DenseNet201	Yes	<0.01
Proposed CNN vs. ResNetV2	No	N/A

"N/A" indicates that there was no statistically significant difference in accuracy between the Proposed CNN and ResNetV2, hence a p-value could not be computed for this comparison.

The one-way ANOVA revealed significant differences in the accuracy of the proposed CNN compared to other models, specifically VGG16 and DenseNet201. This supports the assertion that the proposed CNN offers an improved performance in regard to brain tumor detection, particularly when contrasted with these architectures. The significant p-values associated with these comparisons highlight the effectiveness of the proposed CNN in capturing relevant features crucial for accurate detection.

While no significant difference was found between the proposed CNN and ResNetV2, several factors suggest that the proposed CNN may still offer advantages that enhance its overall performance in specific applications, such as brain tumor detection. The proposed CNN may incorporate unique architectural features tailored specifically for this purpose, allowing it to better capture pertinent patterns in the medical-imaging data. Additionally, it may have been trained on a dataset curated specifically for brain tumor detection, enabling it to learn features more relevant to this domain compared to the more generalist ResNetV2. Extensive fine-tuning or the application of transfer learning with domain-specific data could also result in improved performance in the context of brain tumor detection.

Furthermore, the proposed CNN may provide better interpretability in its decision-making process, leading to more reliable outputs in clinical settings, where understanding model predictions is crucial. The statistical evidence reinforces the relevance of implementing the proposed CNN in clinical settings. The proposed CNN's demonstrated superior performance relative to VGG16 and DenseNet201 suggests that it may provide enhanced diagnostic capabilities, contributing to more effective treatment planning and improved patient outcomes in brain tumor cases. Thus, the proposed CNN stands as a valuable tool in the advancement of medical-imaging technologies.

6. Discussion

6.1. Model Performance and Convergence

The performance metrics of the trained CNN models show that the proposed custom CNN architecture outperformed the other models, achieving an accuracy of 94.51%. This high performance can be attributed to the nine convolutional layers in the proposed CNN, which enabled the model to effectively capture intricate patterns in the MRI data. Despite increasing the number of epochs to 50, the model's accuracy plateaued, indicating convergence at the 30th epoch.

6.2. Addressing Class Imbalance

A significant challenge we faced in the dataset was class imbalance, particularly with underrepresented tumor types, such as pituitary tumors. To address this issue, we employed oversampling techniques to generate synthetic samples for these minority classes, which helped achieve a more balanced representation across all tumor types. Additionally, we applied class weighting during training to give greater importance to minority classes, reducing bias toward the majority classes. Although the model's accuracy decreased from

95.215% to 94.51%, the loss improved significantly from 0.3386 to 0.140, indicating a better fit of the model to the training data.

6.3. Enhancing Interpretability

To enhance the interpretability of the trained model, several visualization techniques were employed. Gradient-based attribution methods such as Integrated Gradients and Guided Backpropagation were applied to identify the most influential parts of the MRI images. Additionally, saliency maps were generated to highlight the most relevant regions that contributed to the model's classification decisions. These techniques provided a clearer understanding of the model's decision-making process, facilitating collaboration between clinicians and AI systems.

6.4. Clinical Implications

The proposed brain tumor-detection system offers significant clinical implications that could improve patient outcomes and streamline diagnostic workflows. The high accuracy (94.51%) of the custom CNN model, combined with interpretability techniques such as saliency maps, allows clinicians to trust and understand the AI's decision-making process. This fosters collaboration between AI systems and medical professionals, potentially reducing diagnostic errors and improving the precision of tumor subtype classification.

In clinical practice, the ability to rapidly and accurately diagnose brain tumors can significantly improve patient outcomes. The proposed CNN model's high accuracy and interpretability make it particularly useful in real-world applications where clinicians must rely on AI systems to support their decision-making. For instance, the model can assist radiologists in identifying complex tumor subtypes earlier in the diagnostic process, potentially leading to more timely and personalized treatment planning. Additionally, the interpretability of the model, supported by gradient-based attribution methods, enhances clinicians' trust in the system, as they can visualize the specific regions that influenced the model's predictions. This can help reduce diagnostic errors, optimize resource allocation, and improve overall patient management.

By enhancing the detection of gliomas, meningiomas, and pituitary tumors, this system could enable earlier diagnoses and more accurate treatment planning. Personalized treatment options can be developed based on the specific tumor subtype, allowing for more targeted therapies that could lead to better patient outcomes. Furthermore, the incorporation of AI into routine diagnostic processes could reduce the workload on radiologists, freeing up time for more complex cases and increasing the overall efficiency of the healthcare system.

6.5. Contribution to Brain Tumor Detection

Our study has made substantial contributions to the field of brain tumor detection, particularly by addressing class imbalance and enhancing model interpretability. The custom CNN architecture developed in this study was specifically designed to capture features related to brain tumor classification, leading to reliable and accurate predictions, especially for rare tumor types. The combination of oversampling, class weighting, and interpretability methods further improved the model's performance and usability in clinical settings.

6.6. Comparison with Related Models

Table 18 presents a comparative analysis between the EfficientNetV2S model introduced in [18] and the custom CNN developed in this study. EfficientNetV2S outperformed the custom CNN in terms of overall accuracy, precision, recall, and F1-score, achieving an accuracy of 98.48% compared to 94.51% for the custom CNN. While EfficientNetV2S also demonstrated a superior performance in regard to precision (98.5%) and recall (98%), the custom CNN still maintained competitive results across individual tumor types, with a weighted average of 94% for precision, recall, and F1-score. Both models employed different interpretability tools—EfficientNetV2S utilized Grad-CAM, while the custom CNN employed saliency maps and gradient attribution methods.

Table 18. Comparative analysis between EfficientNetV2S and custom CNN.

Metric	EfficientNetV2S	Custom CNN Model
Accuracy	98.48%	94.51%
Precision	98.5%	94%
Recall	98%	94%
F1-score	98%	94%
Interpretability tools	Grad-CAM	Saliency Maps, Gradient Attribution

This comparison highlights the trade-off between model accuracy and interpretability, suggesting that future work could integrate EfficientNet-based architectures to enhance accuracy without sacrificing the interpretability provided by visualization techniques.

The EfficientNetV2S model outperformed the custom CNN model in terms of overall accuracy, while the custom CNN demonstrated a competitive performance across individual tumor types, achieving a weighted average of 94% for precision, recall, and F1-score. Both models utilized different interpretability tools to enhance our understanding of their predictions. The trade-off between interpretability and raw performance suggests that future work could explore integrating EfficientNet-based architectures to improve accuracy while retaining the interpretability through advanced visualization methods.

In addition to comparing our custom CNN with EfficientNetV2S, the performance of other State-of-the-Art models for brain tumor detection has been reviewed. Table 19 presents the performance and methodologies of these models in comparison to our custom CNN.

Table 19. Comparative analysis of brain tumor-detection methods.

Source	Classified Method	Accuracy	Additional Information
[32]	Siamese Neural Network (GoogLeNet)	97.64%	The Siamese Neural Network achieves a commendable accuracy of 97.64%. However, its performance is slightly lower than the proposed method. While Siamese networks are effective for tasks like image similarity and verification, their suitability for brain tumor detection may vary depending on the dataset and task requirements.
[33]	Hybrid CNN (Resnet50)	97.20%	The Hybrid CNN, utilizing ResNet50 architecture, achieves an accuracy of 97.20%, which is slightly lower than both the proposed method and the Siamese Neural Network. Hybrid CNN architectures often combine features from multiple CNN architectures to improve performance. However, their complexity may pose challenges in interpretation and implementation.
[34]	Optimal DNN and Spider-Monkey Optimization	99.30%	Preethi and Aishwarya's method, employing an Optimal DNN with Spider-Monkey Optimization, achieves the highest accuracy of 99.30%. While the accuracy is impressive, the complexity of the optimization technique and the interpretability of the model may be limiting factors for practical applications.
[35]	Wavelet Transform and Support Vector Machine	98.14%	Kharrat et al. achieved an accuracy of 98.14% using Wavelet Transform combined with Support Vector Machine (SVM). While SVMs are known for their effectiveness in classification tasks, the reliance on feature engineering and the interpretability of the model may be challenging compared to deep-learning approaches.

Table 19. *Cont.*

Source	Classified Method	Accuracy	Additional Information
[36]	Multiscale CNN	97.30%	Diaz-Pernas et al.'s Multiscale CNN achieved an accuracy of 97.30%, demonstrating robust performance in brain tumor detection. Multiscale CNN architectures leverage features at multiple resolutions, offering a comprehensive representation of the input data. However, they may require more computational resources during training and inference.
[37]	Modified Deep CNN	96.40%	Hemanth et al. achieved an accuracy of 96.40% with a Modified Deep CNN, demonstrating competitive performance in tumor detection. Modifications to standard CNN architectures can improve their effectiveness for specific tasks. However, the degree of modification and its impact on model interpretability should be carefully considered.
[38]	Convolutional Neural Network	91.43%	Paul et al.'s CNN achieved an accuracy of 91.43%, which is relatively lower compared to other methods. The lower accuracy may be attributed to various factors such as dataset characteristics, model architecture, or training methodology.
[12]	Deep transfer learning (AlexNet)	99.62%	While Badjie and Ülker achieved impressive results using AlexNet for two-class classification, our study expands upon this by classifying four classes, making the problem more complex. Transfer learning remains highly effective, but the challenges increase as more tumor types are introduced.
Proposed method	Custom CNN	94.51%	The proposed method utilizes a custom CNN architecture tailored specifically for brain tumor detection. This approach allows for better capturing of features relevant to tumor classification, leading to a high accuracy of 94.51%. The custom CNN architecture offers flexibility and adaptability to the dataset and clinical requirements, potentially making it more suitable for real-world applications compared to standardized models.

The comparative analysis shows that while our custom CNN performs competitively, several models, like the EfficientNetV2S and Optimal DNN with Spider-Monkey Optimization, achieved higher raw accuracy. However, our model offers a balance between performance and interpretability, which is critical for clinical applications.

6.7. Exploration of Transformer-Based Architectures

Beyond CNN-based models, future work could explore the application of transformer-based architectures for brain tumor classification. Transformers are known for their ability to capture long-range dependencies and may improve classification accuracy while maintaining interpretability.

6.8. Security Threats and Countermeasures

The deployment of machine-learning models, especially in sensitive domains like healthcare, exposes the system to potential security vulnerabilities. One of the significant threats is systematic poisoning attacks, where adversaries can manipulate training data to degrade the model's performance intentionally [39,40]. Such attacks are particularly concerning in healthcare, as they can lead to incorrect diagnoses, posing severe risks to patient safety [41,42].

To mitigate these threats, various defensive strategies can be explored:

- Differential privacy: This technique ensures that individual patient data points do not significantly influence the model, reducing the likelihood of privacy breaches from model outputs.

- Secure multi-party computation: By distributing the computation across multiple parties, the risk of an adversary gaining access to sensitive medical data is minimized.
- Homomorphic encryption: This method allows computations to be performed on encrypted data, ensuring that sensitive information is never exposed during the training or inference process [41,43].

Future work will explore the integration of these security frameworks into the proposed brain tumor-detection pipeline, aiming to enhance the model's resilience to adversarial attacks and safeguard patient data in clinical settings.

6.9. Energy-Efficient Long-Term Health-Monitoring Systems

In addition to security vulnerabilities, energy efficiency is crucial for continuous health-monitoring systems that rely on AI models for real-time data processing. Developing AI systems that minimize energy consumption while maintaining high performance can enable long-term personal health monitoring [44]. This is particularly beneficial for remote healthcare applications where IoT devices monitor patient data in real-time. These energy-efficient systems will not only improve healthcare delivery but also reduce operational costs, making them suitable for large-scale deployments in smart cities.

6.10. AI-Empowered IoT Security for Smart Cities

As healthcare services increasingly integrate with IoT infrastructures in smart cities, ensuring the security of interconnected devices becomes critical. Future research will focus on incorporating AI-empowered IoT security measures that protect patient data and ensure secure communication between IoT devices and healthcare systems [43]. Techniques like blockchain for secure data sharing and real-time intrusion detection systems could be pivotal in building robust, secure healthcare frameworks.

6.11. Ethical Considerations

While the performance of AI models in medical diagnosis continues to improve, ethical considerations such as data privacy and model bias cannot be overlooked. In healthcare, the risk of biased predictions can disproportionately affect certain populations, potentially leading to incorrect diagnoses. The dataset used in this study, though comprehensive, may still carry inherent biases that could affect the generalizability of the model to different demographics or rare tumor subtypes. In addition, patient privacy remains a key concern, as medical images used in training models could be sensitive. Future work should focus on mitigating these risks by implementing techniques such as differential privacy and exploring the implications of AI-driven decision-making in clinical practice.

6.12. Limitations and Future Work

While the proposed CNN model demonstrates strong performance in brain tumor detection, several limitations must be addressed. First, although the dataset used in this study is diverse, it may not fully represent the variability encountered in clinical practice. To improve the model's generalizability, future work will involve expanding the dataset to include more patient data from various demographics and institutions.

Moreover, the current study focuses exclusively on MRI data. Incorporating multimodal data, such as CT and PET scans, could enhance the robustness of the model and provide more comprehensive diagnostic insights. Additionally, exploring advanced architectures like ViT may allow for the capture of long-range dependencies in medical-imaging data, potentially improving classification accuracy further.

In terms of security, vulnerabilities in AI-based healthcare systems remain a significant concern. Future research should focus on addressing the risks posed by adversarial attacks, such as data poisoning. To safeguard sensitive medical data and ensure the integrity of predictions, techniques like differential privacy, homomorphic encryption, and secure multi-party computation can be applied. These measures will enhance the safety and integrity of AI models in clinical applications.

Furthermore, future iterations of the model could benefit from employing advanced interpretability methods, such as Layer-Wise Relevance Propagation (LRP). These techniques would provide clinicians with more detailed insights into model decisions, fostering greater trust in AI-driven diagnostic tools.

Another potential limitation of the study is the relatively small dataset used for training, which, although balanced through oversampling, may still not fully represent the variability seen in clinical environments. This may lead to overfitting, particularly in underrepresented tumor subtypes such as pituitary tumors. Additionally, while the model was trained and validated on a curated dataset, its performance on real-world, noisy, or heterogeneous clinical data is yet to be fully evaluated. Future work should explore external validation on larger, more diverse datasets and the development of robust models that can generalize across different medical centers and MRI protocols.

7. Conclusions

This study presents a custom CNN architecture designed for brain tumor detection using MRI scans. By addressing key challenges, such as class imbalance and model interpretability, the proposed model achieved competitive results, with an accuracy of 94.51%, outperforming other pretrained models like ResNetV2, DenseNet201, and VGG16. The application of oversampling techniques and class weighting effectively mitigated the effects of class imbalance, leading to improved model generalization across tumor classes.

A key contribution of this work is the integration of interpretability techniques, such as gradient-based attribution methods and saliency maps, which enhance transparency in the model's predictions. These tools not only provide clinicians with insights into the model's decision-making process but also foster trust in AI-assisted diagnostics, making the model more viable for real-world healthcare applications.

Future research will focus on exploring advanced architectures, such as ViT, which capture long-range dependencies and have the potential to outperform CNNs in medical-image classification. Additionally, integrating multimodal data from other imaging modalities, like CT or PET scans, could further enhance the model's robustness and diagnostic capabilities.

Beyond model performance, security remains a critical concern for AI in healthcare. Future work will explore integrating differential privacy, homomorphic encryption, and other security measures to protect against systematic poisoning attacks and ensure the privacy of patient data. These efforts will ensure the safe and secure deployment of AI models in clinical environments.

Furthermore, real-world testing on larger, more diverse datasets across multiple institutions will be essential for ensuring the model's generalizability and robustness. Deploying the model in clinical settings with real-time feedback from healthcare professionals will also help refine its usability and effectiveness in medical workflows.

Lastly, the growing role of IoT in healthcare opens new opportunities for AI-driven continuous health monitoring. By developing energy-efficient AI systems and ensuring the security of IoT devices, AI-enabled healthcare solutions can be scaled for smart cities, ensuring both patient safety and data security in connected environments.

In conclusion, this research makes significant strides in improving brain tumor detection through a custom CNN architecture, while also paving the way for future advancements in AI-based healthcare security, interpretability, and scalability. The findings have the potential to contribute to more accurate diagnostics and personalized treatment planning, ultimately benefiting both clinicians and patients.

Author Contributions: K.A.K.W.D. and R.H. conceptualized and designed the overall research framework; S.M. and B.R. developed the experimental approach and contributed essential materials; K.A.K.W.D. created and implemented the computational tools; K.A.K.W.D., R.H., B.R. and S.M. verified the accuracy and reliability of the results; K.A.K.W.D. conducted the statistical analyses; S.M. and B.R. performed the experiments and gathered data; K.A.K.W.D. organized and managed the research data; K.A.K.W.D. and S.M. wrote the initial draft of the manuscript; R.H. and B.R.

critically revised and improved the manuscript; K.A.K.W.D. and B.R. prepared figures and visual representations; R.H. oversaw and supervised the research process; R.H. secured the financial support for the research. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available in Kaggle at https://www.kaggle.com/code/shivamagarwal29/brain-tumor (accessed on 15 October 2024).

Acknowledgments: The authors would like to acknowledge the use of ChatGPT 24 May 2023 version (OpenAI, San Francisco, CA, USA), specifically to assist in some content rewriting for improved clarity and effectiveness.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Khazaei, Z.; Goodarzi, E.; Borhaninejad, V.; Iranmanesh, F.; Mirshekarpour, H.; Mirzaei, B.; Naemi, H.; Bechashk, S.M.; Darvishi, I.; Ershad Sarabi, R.; et al. The association between incidence and mortality of brain cancer and human development index (HDI): An ecological study. *BMC Public Health* **2020**, *20*, 1696. [CrossRef] [PubMed]
2. Bernstock, J.D.; Gary, S.E.; Klinger, N.; Valdes, P.A.; Ibn Essayed, W.; Olsen, H.E.; Chagoya, G.; Elsayed, G.; Yamashita, D.; Schuss, P.; et al. Standard clinical approaches and emerging modalities for glioblastoma imaging. *Neuro-Oncol. Adv.* **2022**, *4*, vdac080. [CrossRef] [PubMed]
3. Sabeghi, P.; Zarand, P.; Zargham, S.; Golestany, B.; Shariat, A.; Chang, M.; Yang, E.; Rajagopalan, P.; Phung, D.C.; Gholamrezanezhad, A. Advances in Neuro-Oncological Imaging: An Update on Diagnostic Approach to Brain Tumors. *Cancers* **2024**, *16*, 576. [CrossRef] [PubMed]
4. Wu, J.; Li, C.; Gensheimer, M.; Padda, S.; Kato, F.; Shirato, H.; Wei, Y.; Schönlieb, C.-B.; Price, S.J.; Jaffray, D.; et al. Radiological tumour classification across imaging modality and histology. *Nat. Mach. Intell.* **2021**, *3*, 787–798. [CrossRef]
5. Orr, B.A. Pathology, diagnostics, and classification of medulloblastoma. *Brain Pathol.* **2020**, *30*, 664–678. [CrossRef]
6. ZainEldin, H.; Gamel, S.A.; El-Kenawy, E.M.; Alharbi, A.H.; Khafaga, D.S.; Ibrahim, A.; Talaat, F.M. Brain Tumor Detection and Classification Using Deep Learning and Sine-Cosine Fitness Grey Wolf Optimization. *Bioengineering* **2022**, *10*, 18. [CrossRef]
7. Saraswat, B.K.; Vaibhav, V.; Pal, P.; Singh, A.K.; Tiwari, N. Brain Tumor Detection. *IJRASET* **2023**, *11*, 5634–5640. [CrossRef]
8. Rajeev, S.K.; Rajasekaran, M.P.; Ramaraj, K.; Vishnuvarthanan, G.; Arunprasath, T.; Muneeswaran, V. A Hybrid CNN-LSTM Network For Brain Tumor Classification Using Transfer Learning. In Proceedings of the 2023 9th International Conference on Smart Computing and Communications (ICSCC), Kochi, Kerala, India, 17–19 August 2023; pp. 77–82. [CrossRef]
9. Aakanksha, M. Brain Tumor Detection using Deep Learning. *Int. J. Res. Appl. Sci. Eng. Technol.* **2023**, *11*, 490–493. [CrossRef]
10. Singh, A. Review of Brain Tumor Detection from MRI Images. In Proceedings of the 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 16–18 March 2016; pp. 3997–4000.
11. Tambe, U.Y.; Shanthini, A. Brain Tumor Detection & Classification into Different Categories using Deep Learning Model. In Proceedings of the 2023 International Conference on Advanced Computing Technologies and Applications (ICACTA), Mumbai, India, 6–7 October 2023.
12. Badjie, B.; Deniz Ülker, E. A Deep Transfer Learning Based Architecture for Brain Tumor Classification Using MR Images. *Inf. Technol. Control* **2022**, *51*, 332–344. [CrossRef]
13. Banu, R. Brain Tumour Detection and Classification Using U-Net Deep Neural Network. *Int. J. Creat. Res. Thoughts (IJCRT)* **2022**, *10*, 816–820.
14. Liu, Z.; Mao, H.; Wu, C.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 11966–11976. [CrossRef]
15. Dai, Z.; Liu, H.; Le, Q.V.; Tan, M. CoAtNet: Marrying Convolution and Attention for All Data Sizes. *arXiv* **2021**, arXiv:2106.04803.
16. Tajane, K.; Rathkanthiwar, V.; Chava, G.; Dhavale, S.; Chawda, G.; Pitale, R. EffiConvRes: An Efficient Convolutional Neural Network with Residual Connections and Depthwise Convolutions. In Proceedings of the 2023 7th International Conference on Computing, Communication, Control And Automation (ICCUBEA 2023), Pune, India, 18–19 August 2023. [CrossRef]
17. Todi, A.; Narula, N.; Sharma, M.; Gupta, U. ConvNext: A Contemporary Architecture for Convolutional Neural Networks for Image Classification. In Proceedings of the 3rd International Conference on Innovative Sustainable Computational Technologies, Graphic Era Deemed to Be University, Dehradun, India, 8–9 September 2023; pp. 1–6. [CrossRef]
18. Priyadarshini, P.; Kanungo, P.; Kar, T. Multigrade brain tumor classification in MRI images using Fine tuned efficientnet. *e-Prime* **2024**, *8*, 100498. [CrossRef]
19. Kassu, J.S. *Research Design and Methodology*; Abu-Taieh, E., El Mouatasim, A., Al Hadid, I.H., Eds.; Cyberspace; IntechOpen: Rijeka, Croatia, 2019; Chapter 3.

20. Kadam, A. Brain Tumor Classification using Deep Learning Algorithms. *Int. J. Res. Appl. Sci. Eng. Technol.* **2021**, *9*, 417–426. [CrossRef]
21. Razzaq, M.; Clément, F.; Yvinec, R. An overview of deep learning applications in precocious puberty and thyroid dysfunction. *Front. Endocrinol.* **2022**, *13*, 959546. [CrossRef]
22. Alzubaidi, L.; Zhang, J.; Al-Amidie, M.; Farhan, L.; Fadhel, M.A.; Duan, Y.; Santamaría, J.; Al-Dujaili, A.; Al-Shamma, O.; Humaidi, A.J. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [CrossRef]
23. Aliferis, C.; Simon, G. *Overfitting, Underfitting and General Model Overconfidence and Under-Performance Pitfalls and Best Practices in Machine Learning and AI*; Springer: Cham, Switzerland, 2024. [CrossRef]
24. Salehin, I.; Kang, D. A Review on Dropout Regularization Approaches for Deep Neural Networks within the Scholarly Domain. *Electronics* **2023**, *12*, 3106. [CrossRef]
25. Wang, X.; Yan, L.; Zhang, Q. Research on the Application of Gradient Descent Algorithm in Machine Learning. In Proceedings of the 2021 International Conference on Computer Network, Electronic and Automation (ICCNEA), Xi’an, China, 24–26 September 2021; pp. 11–15. [CrossRef]
26. Matsuyama, E.; Nishiki, M.; Takahashi, N.; Watanabe, H. Using Cross Entropy as a Performance Metric for Quantifying Uncertainty in DNN Image Classifiers: An Application to Classification of Lung Cancer on CT Images. *J. Biomed. Sci. Eng.* **2024**, *17*, 1–12. [CrossRef]
27. Gnip, P.; Vokorokos, L.; Drotár, P. Selective oversampling approach for strongly imbalanced data. *PeerJ Comput. Sci.* **2021**, *7*, e604. [CrossRef]
28. Araf, I.; Idri, A.; Chairi, I. Cost-sensitive learning for imbalanced medical data: A review. *Artif. Intell. Rev.* **2024**, *57*, 80. [CrossRef]
29. Johnson, J.L.; Adkins, D.; Chauvin, S. A Review of the Quality Indicators of Rigor in Qualitative Research. *Am. J. Pharm. Educ.* **2020**, *84*, 7120–7146. [CrossRef]
30. Althubaiti, A. Information bias in health research: Definition, pitfalls, and adjustment methods. *J. Multidiscip. Healthc.* **2016**, *9*, 211–217. [CrossRef] [PubMed]
31. Rajbahadur, G.K.; Wang, S.; Oliva, G.A.; Kamei, Y.; Hassan, A.E. The Impact of Feature Importance Methods on the Interpretation of Defect Classifiers. *TSE* **2022**, *48*, 2245–2261. [CrossRef]
32. Deepak, S.; Ameer, P.M. Retrieval of brain MRI with tumor using contrastive loss based similarity on GoogLeNet encodings. *Comput. Biol. Med.* **2020**, *125*, 103993. [CrossRef] [PubMed]
33. Çinar, A.; Yildirim, M. Detection of tumors on brain MRI images using the hybrid convolutional neural network architecture. *Med. Hypotheses* **2020**, *139*, 109684. [CrossRef] [PubMed]
34. Khare, N.; Devan, P.; Chowdhary, C.; Bhattacharya, S.; Singh, G.; Singh, S.; Yoon, B. SMO-DNN: Spider Monkey Optimization and Deep Neural Network Hybrid Classifier Model for Intrusion Detection. *Electronics* **2020**, *9*, 692. [CrossRef]
35. Kharrat, A.; Gasmi, K.; Ben Messaoud, M.; Benamrane, N.; Abid, M. Medical Image Classification Using an Optimal Feature Extraction Algorithm and a Supervised Classifier Technique. *Int. J. Softw. Sci. Comput. Intell.* **2011**, *3*, 19–33. [CrossRef]
36. Díaz-Pernas, F.J.; Martínez-Zarzuela, M.; Antón-Rodríguez, M.; González-Ortega, D. A Deep Learning Approach for Brain Tumor Classification and Segmentation Using a Multiscale Convolutional Neural Network. *Healthcare* **2021**, *9*, 153. [CrossRef]
37. Hemanth, D.J.; Anitha, J.; Naaji, A.; Geman, O.; Popescu, D.E.; Hoang Son, L. A Modified Deep Convolutional Neural Network for Abnormal Brain Image Classification. *IEEE Access* **2019**, *7*, 4275–4283. [CrossRef]
38. Paul, J.S.; Plassard, A.; Landman, B.; Fabbri, D. Deep Learning for Brain Tumor Classification. In Proceedings of the Medical Imaging 2017: Biomedical Applications in Molecular, Structural, and Functional Imaging, Orlando, FL, USA, 11–16 February 2017; Volume 10137, p. 1013710. [CrossRef]
39. Tian, Z.; Cui, L.; Liang, J.; Yu, S. A Comprehensive Survey on Poisoning Attacks and Countermeasures in Machine Learning. *ACM Comput. Surv.* **2023**, *55*, 1–35. [CrossRef]
40. Wu, B.; Wei, S.; Zhu, M.; Zheng, M.; Zhu, Z.; Zhang, M.; Chen, H.; Yuan, D.; Liu, L.; Liu, Q. Defenses in Adversarial Machine Learning: A Survey. *arXiv* **2023**, arXiv:2312.08890.
41. Zhou, S.; Liu, C.; Ye, D.; Zhu, T.; Zhou, W.; Yu, P.S. Adversarial Attacks and Defenses in Deep Learning: From a Perspective of Cybersecurity. *ACM Comput. Surv.* **2023**, *55*, 1–39. [CrossRef]
42. Wang, F.; Wang, X.; Ban, X.J. Data poisoning attacks in intelligent transportation systems: A survey. *Transp. Res. Part C Emerg. Technol.* **2024**, *165*, 104750. [CrossRef]
43. Khalid, N.; Qayyum, A.; Bilal, M.; Al-Fuqaha, A.; Qadir, J. Privacy-preserving artificial intelligence in healthcare: Techniques and applications. *Comput. Biol. Med.* **2023**, *158*, 106848. [CrossRef] [PubMed]
44. Yu, H.; Li, N.; Zhao, N. How Far Are We from Achieving Self-Powered Flexible Health Monitoring Systems: An Energy Perspective. *Adv. Energy Mater.* **2021**, *11*, 2058. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.