An Approach in Brain Tumor Classification: The Development of a New Convolutional Neural Network Model

Caio dos Santos Felipe, Thatiane Alves Pianoschi Alva, Ana Trindade Winck,
Carla Diniz Lopes Becker

Universidade Federal de Ciências da Saúde de Porto Alegre (UFCSPA),
Rua Sarmento Leite, 245, Porto Alegre, Rio Grande do Sul, Brazil
`{caio.felipe,thatiane,anatw,carladiniz}@ufcspa.edu.br`

Abstract. Brain tumor diagnosis is a complex problem that requires specialized skills and knowledge. Manual analysis is often time-consuming, and there can be high subjectivity in interpreting results. Convolutional neural networks (CNNs) have emerged as a promising solution for automatically classifying brain tumors from magnetic resonance images (MRI). CNNs are a type of neural network that can automatically learn and extract relevant features from images, making them particularly suited to this task when applied in deep learning algorithms. The use of CNNs for brain tumor diagnosis has been widely explored in the literature, with many studies reporting promising results. By leveraging datasets of labeled MRI, CNNs can learn to accurately detect and classify different types of brain tumors, including gliomas, meningiomas, and pituitary adenomas. These models have been shown to outperform traditional machine-learning algorithms and even human experts in some cases. This article presents a CNN model designed to identify and classify brain tumors from MRI. The model was trained on a large dataset of MRI, and its performance was evaluated on an independent test set. The model achieved an accuracy of 99% considering all validation steps and outperformed state-of-the-art methods for brain tumor classification. When considering individual classes, the accuracy percentages were 100%, 98%, 99%, and 99% for glioma, meningioma, notumor, and pituitary, respectively. The development of accurate and efficient methods for brain tumor diagnosis is critical for improving patient outcomes and reducing healthcare costs. This article can advance our understanding of leveraging these powerful algorithms best to solve real-world healthcare problems by contributing to the growing literature on deep learning for medical image analysis.

Keywords: Brain Tumor · Deep Learning · Convolutional Neural Network.

1 Introduction

According to the World Health Organization (WHO), a brain tumor is defined as "an intracranial neoplasm, whether primary or metastatic, which is composed of abnormal cells that grow uncontrollably and have the potential to invade or compress adjacent brain tissue" [21]. The abnormal growth of cells in the brain can lead to the formation of a mass, which can be either malignant or benign. Malignant tumors are characterized by active cells that divide rapidly and disorganizedly, resulting in a mass that can invade surrounding tissue and spread to other body parts. On the other hand, benign tumors are composed of dormant cells that grow slowly and in a more organized manner, causing less damage to surrounding tissues. Although benign tumors are less likely to be life-threatening than malignant tumors, they can still cause significant health problems if they grow large enough or press on critical brain areas [14].

Tumors of the brain and other nervous systems are a significant health concern globally, ranking as the 10th leading cause of death for both men and women. In 2020, there were an estimated 308,102 new cases of brain tumors worldwide, with approximately 251,329 deaths attributed to the disease. In the United States, it is projected that 24,810 adults will be diagnosed with brain tumors in 2023 [13].

The prognosis for individuals with brain tumors varies based on several factors, including the type of tumor, its location, and the stage at diagnosis. The five-year survival rate for individuals with brain tumors is 36%, while the 10-year survival rate is just over 30% [13].

Brain tumors are typically diagnosed through brain imaging and histopathological examination [22]. Magnetic resonance imaging (MRI) is preferred among brain imaging techniques due to its superior resolution and ability to differentiate various types of brain tissue. MRI has become an essential tool in the diagnosis, treatment selection, and follow-up care of brain tumors due to its high sensitivity and specificity. Advanced MRI techniques, such as diffusion-weighted imaging, perfusion imaging, and spectroscopy, have further improved the accuracy of brain tumor diagnosis and provided critical information for treatment planning [23].

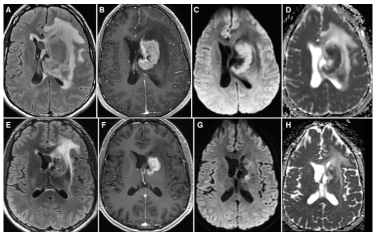


Fig. 1. Primary CNS lymphoma. A 33-yr-old woman before (A-D) and after (E-H) steroid therapy. Image and description source: Villanueva-Meyer, J.E et al. [7].

The classification of primary tumors by the WHO is based on a combination of histopathological criteria, immunohistochemical data, and the degree of malignancy exhibited by the tumor. The WHO has established a four-tiered grading system to classify tumors according to their level of aggressiveness [18]. Grade I tumors are relatively non-aggressive, while grade IV tumors are highly aggressive [8]. The grading system provides an essential tool for clinicians and researchers to assess tumors' behavior and guide treatment decisions. For instance, low-grade tumors may be treated with less aggressive therapies, while high-grade tumors may require more aggressive treatments such as surgery, radiation therapy, or chemotherapy. The WHO classification system has been widely adopted and is used in clinical practice and research worldwide.

Accurate classification is crucial for determining patients' appropriate treatment plans and prognoses. The diverse nature of tumors, with their wide-ranging appearances, sizes, locations, and shapes, presents significant challenges in accurately classifying them [12]. This complexity necessitates the involvement of highly skilled professionals, such as oncologists, radiologists, and pathologists, who have the expertise to examine and interpret the specific characteristics of tumors to determine their type, grade, and potential malignancy.

In less developed regions, access to qualified professionals may be limited due to various factors, including financial constraints, educational opportunities, and inadequate healthcare infrastructure; This can result in a need for more skilled professionals to identify and classify tumors correctly. Consequently, patients in these regions may face increased risks of misdiagnosis, delayed treatment, or inadequate care. Additionally, limited access to advanced diagnostic tools and imaging technologies can exacerbate healthcare providers' challenges in these areas.

To deal with problems like this, Computer-aided diagnosis (CAD) systems are a promising solution. CAD systems can enhance diagnostic accuracy, improve efficiency, and provide access to specialized expertise by assisting healthcare professionals in detecting, classifying, and characterizing tumors by leveraging advanced computational algorithms, machine learning, and artificial intelligence. Their scalability allows deployment in various settings, from large hospitals to rural clinics, while cloud-based systems enable remote access to advanced diagnostic tools. Furthermore, CAD systems can be integrated into telemedicine initiatives, facilitating remote consultations and collaboration between healthcare providers in different geographical locations. By incorporating CAD into the diagnostic process, less-developed regions can benefit from improved tumor classification, ultimately leading to better patient outcomes and reduced healthcare disparities. Additionally, CAD systems serve as valuable educational tools, helping healthcare professionals improve their diagnostic skills and stay current with the latest advancements in tumor classification.

CAD systems are primarily built upon machine learning methods, with a strong focus on deep learning techniques, which have shown exceptional performance in various pattern recognition tasks. Among these techniques, CNNs have emerged as an ideal choice for analyzing medical images, owing to their unique ability to automatically learn hierarchical features from large-scale data. [19].

The main objective of this study is to introduce a novel CNN architecture that can effectively analyze MRI from a publicly available database and accurately classify the type of brain tumor, if present. The study addresses the challenge of accurate and reliable diagnosis of brain tumors, which can often be complex and require expert analysis. To achieve this, the proposed CNN model will leverage advanced machine learning techniques and be trained on a large dataset of MRI images with labeled tumor classes. The ultimate goal is to contribute to the growing literature on using deep learning algorithms in medical image analysis and pave the way for more accurate and efficient diagnosis of brain tumors.

The study is structured in four parts. What is a CNN and how it works is presented in the ”Background” section; All stages of the Convolutional Neural Network development are presented in the ”Materials and Methods” section, from data preparation to the model's architecture. The next step consists of obtaining and validating the results generated from the network training. Finally, final considerations on the entire process and possibilities for the future are presented.

1.1 Related Works

In tumor classification using deep learning techniques, several notable works have been published that explore different approaches to achieve high accuracy.

Arjun Thakur et al.[9] proposed a CNN model applied to the ”Brain Tumor MRI Dataset” [15] referenced in this article. Their data pre-processing steps included resizing images, enhancing them using various techniques, and converting them to grayscale. This approach yielded an impressive accuracy of 98.93% on the test images.

Transfer learning is a widespread technique in which a model pre-trained on a large dataset is fine-tuned on a smaller dataset to obtain better results. Tariq et al.[10] leveraged this technique by combining multiple databases and training a model based on the EfficientNet architecture, explicitly using the EfficientNetB4 variant. This approach resulted in an accuracy of 98.58

The study by Rawaa Ali et al.[11] employed transfer learning with three networks: GoogleNet, ShuffleNet, and NasNet-Mobile. Among these, the ShuffleNet model achieved the best results, with an accuracy of 98.40

While these studies demonstrate high accuracy in tumor classification, it is worth noting that some validation techniques, such as cross-validation, and strategies for better consolidation of results, such as constant shuffling of images during each training iteration, received limited attention in the works developed from the database used in this research. Addressing these aspects could improve model performance and generalization, resulting in more accurate and reliable tumor classification systems.

2 Background

A CNN is an artificial neural network specifically designed for processing and analyzing grid-like data, such as images, speech signals, or any data with spatial or temporal patterns. CNNs are inspired by the organization and functionality of the visual cortex in the human brain and have been remarkably successful in tasks like image classification, object detection, and natural language processing.

The architecture of a CNN consists of several layers, including convolutional layers, pooling layers, and fully connected layers. Each layer performs a specific function and helps the network learn hierarchical features from the input data.

Convolutional Layer. This is the core building block of a CNN. It applies a set of filters (kernels) to the input data (e.g., an image) to generate feature maps. The filters are small matrices that slide over the input data and perform a convolution operation, a mathematical operation that computes the dot product between the filter values and the corresponding input data. This process helps the network to learn local features like edges, shapes, and textures in the input data [2].

Activation Function. After the convolution operation, an activation function, such as Rectified Linear Unit (ReLU), is applied to the feature maps. The activation function introduces non-linearity into the network, allowing it to learn more complex patterns and features. ReLU, for instance, replaces all negative values with zero, which has been found to work well in practice [2].

Pooling Layer. The pooling layer aims to reduce the spatial dimensions of the feature maps and improve the network's efficiency and robustness to small transformations in the input data. Standard pooling techniques include max-pooling (which selects the maximum value in a region) and average pooling (which computes the average value in a region) [2].

Fully Connected Layer. After a series of convolutional and pooling layers, the high-level features are flattened into a one-dimensional vector and passed through one or more fully connected layers. These layers combine the high-level features and make the final predictions or classifications. The last fully connected layer typically uses a softmax activation function to produce a probability distribution over the output classes [2].

A labeled dataset is used to train a CNN. The goal is to minimize the difference between the predicted output and the ground truth labels, typically measured using a loss function, such as cross-entropy loss. The learning process involves adjusting the filter values and weights in the network using an optimization algorithm, such as stochastic gradient descent (SGD) or Adam.

In summary, a Convolutional Neural Network is a specialized neural network that efficiently processes grid-like data using convolutional, pooling and fully connected layers. It learns hierarchical features from the input data, making it highly effective for image classification and object detection tasks.

Planning a CNN model is a complex and iterative process that requires careful consideration of several factors, including the dataset's size and complexity, the desired accuracy level, and the available computational resources. Due to the highly subjective nature of this process, it is often not an exact science and may vary significantly depending on the specific needs of the database used.

AlexNet is a popular and widely used CNN architecture introduced in 2012 and has been shown to achieve state-of-the-art results in several image classification tasks. However, even with a well-established architecture like AlexNet, adjustments and modifications may need to be made to optimize the model's performance for a specific dataset.

This study selected the AlexNet architecture as the initial development base for the CNN model. To this end, the AlexNet architecture was adjusted based on the emergence of better results in both the training and testing phases. This iterative process involved experimenting with different hyperparameters, such as the number of filters, filter size, and learning rate, to find the optimal configuration that achieved the highest accuracy on the dataset.

3 Materials and Methods

3.1 Development Environment and Technologies

The development environment utilized for this work was Google Colab, which offered a robust and efficient platform for creating, testing, and evaluating the model. Specifically, the TensorFlow, scikit-learn, and Keras libraries were employed to facilitate the development of the model, providing a range of tools and techniques for optimizing performance and generating accurate predictions.

The dataset used for this work was obtained through the Kaggle platform, which provided a comprehensive and diverse range of images for training and evaluation. The classification and evaluation of the Kaggle platform database and the number of images contained were the primary criterion parameters for selecting the dataset.

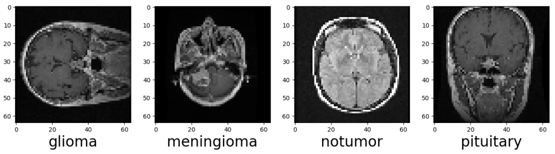
3.2 Data Description and Pre-Processing

The database utilized in this study was the "Brain Tumor MRI Dataset"[15], which comprises 7023 images. These images are divided into two directories: one for testing, containing 1311 images, and another for training, with 5712 images. Each directory is further divided into four subfolders, each named according to the classification of the images they contain: glioma, meningioma, notumor, and pituitary.

During the study, it was discovered that some images in the Glioma directory were mislabeled. The mislabeled images were replaced with correctly labeled Glioma images from the "Figshare Brain Tumor Classification" dataset [16]. This change ensures the accuracy and integrity of the dataset, thereby enhancing the reliability of the study's findings.

The current distribution of images in each class within the training set, after the necessary adjustments, is as follows: glioma has 1126 images (20.41%), meningioma has 1339 (24.27%), notumor has 1595 (28.91%), and pituitary has 1457 (26.41%). These distributions provide a comprehensive overview of the dataset used for training the model.

We defined the training and test sets, comprising 5531 and 1297 images, respectively. A validation set was created by randomly selecting 20% of the training set images. All images were resized to a 64x64 pixel resolution, ensuring consistency and computational efficiency. Furthermore, pixel values were normalized by dividing each value by 255 to prevent numerical instability and enhance model convergence. Consequently, pixel values ranged from 0 to 1. Representative images from each class, namely glioma, meningioma, notumor, and pituitary, are displayed in Figure 2.



glioma meningioma notumor pituitary

Fig. 2. Examples of images referring to each possible class. The glioma image was sourced from the "Figshare Brain Tumor Classification"[16] dataset, while the images for meningioma, notumor, and pituitary were sourced from the "Brain Tumor MRI Dataset"[15]

In order to enable the use of image classes in training, it was necessary to convert them from nominal to categorical numerical values. This transformation allows the convolutional neural network to understand the relationship between the different classes and use them as labels for supervised learning. Each image class was assigned a unique numerical value, such as 0 for glioma, 1 for meningioma, 2 for notumor, and 3 for pituitary.

Data Balancing. During this study's exploratory data analysis phase, an imbalance in the distribution of images across each class was observed. Such a data imbalance is a common challenge in developing image classification models and can impact the accuracy and reliability of the results generated. In cases like this, several methods can be employed to balance the data and ensure the model is unbiased toward any particular class.

In order to balance the training data, a set of approximate weights for each of the four classes, namely glioma, meningioma, notumor, and pituitary, was defined. These weights were defined based on the relative frequency of images in each class and aimed to ensure that the model was trained on a more balanced and representative dataset. The weights were defined as follows: 1.21 for the glioma class, 1.03 for the meningioma class, 0.86 for the notumor class, and 0.94 for the pituitary class.

Data Shuffling. In order to avoid biased results and show the network's generalization capacity, the database fragments were shuffled in all validation folds before training. This technique is often used to prevent the model from overfitting the training data by exposing it to a different order of data in each epoch.

By shuffling the training data during each epoch, the model is forced to learn more generally and robustly from the data, leading to better generalization performance. Additionally, data shuffling can help avoid the model from memorizing the training data, which can occur when the model sees the same data repeatedly in the same order during training.

Addressing Data Leakage. To avoid data leakage, the dataset was initially split into training and validation sets before undergoing normalization. Additionally, K-Fold Cross Validation was employed, further safeguarding against potential leakage.

3.3 Model Architecture and Settings

The model consists of three convolution blocks, image-processing layers, and other multidimensional data types. Convolutional layers work by applying a set of learnable filters to the input data and convolving them across the input space to extract relevant features. These features are then passed to the subsequent layers for further processing. A detailed representation of the model architecture is provided in Figure 3.

In addition to the convolutional layers, the model also includes maximum pooling layers, which reduce the dimensionality of the input data by selecting the maximum value within a defined pool size, helping to decrease the number of parameters in the model.

Furthermore, the model includes dropout layers, which are a type of regularization technique that is used to prevent overfitting in deep learning models. Dropout works by randomly dropping out a fraction of the neurons in the network during training, which forces the network to learn more robust and generalized features.

The convolutional layers are a crucial model component for extracting relevant input data features. Each convolutional layer is configured with a 3x3 kernel size, which defines the sliding window size used to convolve over the input data. The ReLu activation function is also applied to each convolutional layer, which helps to introduce non-linearity into the model and makes it more capable of learning complex features.

The first convolutional layer comprises 16 filters and has a dropout layer with a probability of 20%. The second convolutional layer comprises 32 filters and has a dropout layer with a probability of 25%. The third convolutional layer comprises 64 filters and has a dropout layer with a probability of 30%. This architecture is designed to increase the model's ability to learn and generalize to a wide range of features.

The fully connected layers are the final layers in the model and are responsible for performing the classification task. The first fully connected layer contains 32 neurons and is activated by the ReLU activation function, with an additional L2 regularizer applied to promote regularization and prevent overfitting. The second fully connected layer contains four neurons. It is activated by the softmax activation function, which is commonly used in multi-class classification problems to calculate the probabilities of each class.

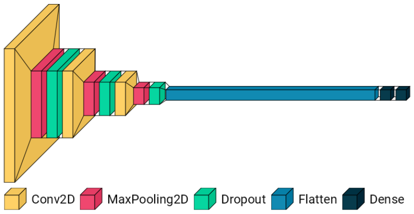


Fig. 3. Visual representation of the model's architecture.

For the compilation, the Adam function was utilized as an optimizer to enhance the efficiency of the model's weight update mechanism. The sparse categorical cross-entropy was chosen as a loss function to compute the difference between the predicted and actual labels of the dataset and to guide the model toward minimizing this difference during training. Finally, accuracy was selected as a metric to visualize the training results, enabling a more comprehensive understanding of the model's performance over time.

3.4 Training

To train a neural network model for a specific task, such as brain tumor classification, each neuron's weights and biases must be adjusted to minimize an appropriate loss function, such as binary cross-entropy, for binary classification problems. This adjustment process is achieved by feeding a subset of the data into the network and using an optimization algorithm, such as SGD, to search for the minimum of the loss function. The SGD algorithm updates the weights and biases of each neuron in each iteration based on the learning rate. The batch size, which refers to the number of training examples used to create the subset, determines the number of samples processed in each iteration. Each iteration updates the weights and biases of the neurons based on the loss computed from the current batch. When the entire dataset has been processed once, it is considered an epoch.[2].

Callback functions can be used to monitor and modify the model during the training process, if needed, to improve the model's performance and ensure that it converges to an optimal solution.

Early Stopping callbacks were used for training, which allowed the model to terminate the training process early if the loss in the validation set did not improve for a set number of epochs, referred to as patience. In this case, the patience was set to 18 epochs, meaning that the training process would terminate if the validation loss did not improve for 18 epochs.

A Learning Rate Scheduler was also implemented to optimize the training process further. This scheduler was designed to reduce the learning rate by a factor of 0.2 if the loss of the training set did not change after five epochs. This approach enabled the model to converge more efficiently and improve its performance over time.

Finally, a Checkpoint was used to store the best-performing weights of the model in the cross-validation steps. This technique ensured that the model's best performance was captured and stored, generating more reliable and accurate predictions.

K-Fold Cross Validation. In order to solidify the results and make them more reliable, the K-Fold Cross Validation method was applied, where several instances of our model are trained using fragments of the database, thus providing a better estimate of the actual accuracy achieved by the network.

The data is divided into K equal parts. The model is trained in K-1 parts, and one part is set aside for testing. For each iteration, the average accuracy of the model is calculated. The same is repeated K times, changing the test part until all are used [20].

K-Fold Cross Validation can be used to evaluate the model's performance on different subsets of the data, which can help identify potential issues with overfitting or underfitting. It can also tune hyperparameters, such as the learning or dropout rates, to improve the model's performance.

This study used five folds (K). Each fold is trained independently, where in each iteration, 20% of the training set, randomly selected, is used for training, and the remaining 80% is used for validation. During the training phase, the model is trained through 60 epochs, each representing one pass over the entire training set. A batch size of 32 is used, meaning that each iteration updates the model's parameters based on 32 training data samples. Using a smaller batch size, such as 32, has improved the training process by reducing the computational burden and allowing more frequent updates to the model's parameters.

4 Results

The neural network demonstrated a remarkable performance, achieving an overall accuracy of 99% on the test set, considering all validation steps. When examining the accuracy for individual classes, the network attained 100% for glioma, 98% for meningioma, 99% for notumor, and 99% for pituitary. These results indicate a high degree of correctness in the model's predictions for each class.

In addition to accuracy, the recall rate—representing the proportion of true positives that were correctly identified—was assessed. The recall rate was 100%

for glioma, 97% for meningioma, 100% for notumor, and 99% for pituitary. These values suggest that the model is highly effective at detecting true positive cases across all classes, with a solid performance in the notumor and pituitary categories.

The F1 score was calculated to provide a more comprehensive evaluation of the model's performance. The F1 score combines precision and recall metrics, offering a balanced perspective on the model's ability to make accurate predictions while minimizing false positives and negatives. The F1 score was 100% for glioma, 98% for meningioma, 99% for notumor, and 99% for pituitary. These results reinforce the network's impressive performance, demonstrating its capability to accurately identify and classify various types of brain tumors with high precision and recall.

Table 1 presents an array of critical metrics, including precision, recall, F1 score, and accuracy. Precision quantifies the ratio of true positives among all optimistic predictions, while recall calculates the ratio of true positives among all actual positive instances. The F1 score, a precision and recall metrics, is frequently employed to assess a model's overall performance. In contrast, accuracy evaluates the proportion of correct predictions the model makes.

Table 1. Classification report categorized by class.

	precision	recall	f1-score	support
glioma	1.00	1.00	1.00	286
meningioma	0.98	0.97	0.98	306
notumor	0.99	1.00	0.99	405
pituitary	0.99	0.99	0.99	300
accuracy	N/A	N/A	0.99	1311
macro avg	0.99	0.99	0.99	1311
weighted avg	0.99	0.99	0.99	1311

The subsequent figures offer an in-depth examination of the model's performance. Specifically, Figure 4 depicts the training and validation loss, which evaluates the model's generalization capabilities on unseen data. This figure spans 60 epochs, facilitating the analysis of temporal trends. Furthermore, the results are presented across five cross-validation folds, ensuring a more rigorous evaluation of the model's performance.

Accuracy, a metric that determines the frequency of correct predictions by the model, is essential for assessing the performance of any machine learning algorithm. The data in Figure 5 supplements the information conveyed in Figure 4, as it allows the determination of the model's accuracy improvement over time and across various validation sets. Conversely, Figure 5 showcases the training and validation accuracy throughout the same 60 epochs and across five identical cross-validation folds. By scrutinizing both figures, we can better understand the model's performance and capacity to generalize to novel data.

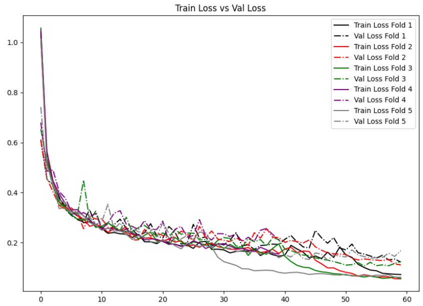


Fig. 4. Loss graph of the 5 cross validation folds.

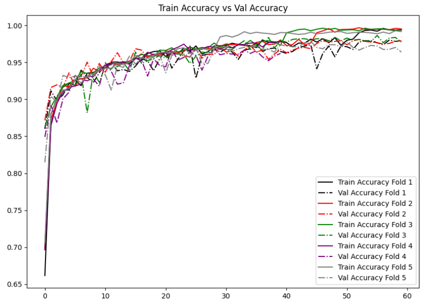


Fig. 5. Accuracy graph of the 5 cross validation folds.

The Receiver Operating Characteristic (ROC) curve is a valuable evaluation tool used in CNNs to assess the performance of binary classification tasks. The ROC curve illustrates the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity) at various classification thresholds. The area under the ROC curve (AUC-ROC) is often used as a summary metric to quantify the model's overall performance, with a higher AUC-ROC indicating better classification performance. In our study, we present the ROC curve analysis in Figure 6, which shows a curve nearly hugging the upper left corner, indicating an almost perfect balance between true positives and false positives.

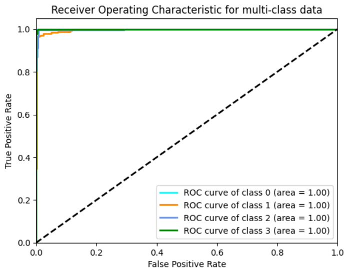


Fig. 6. ROC curve generated from the trained model.

Deep Learning models, such as CNNs, are often criticized for their black-box nature, particularly in critical fields like medical imaging, where the interpretability of the model is crucial. As our work involves using a CNN for brain tumor classification, it is essential to address this aspect. The complexity and opaqueness of CNNs prompt the implementation of Explainable Artificial Intelligence (XAI), an approach advocated by François Chollet, among other leading researchers in the field. XAI's fundamental goal is to unravel these models' intricate decision-making processes, making their predictions transparent and comprehensible.

For our CNN, we employ techniques inspired by Chollet's work to generate 'heatmaps' of class activation, highlighting the regions in the input image most influential for the model's prediction. This approach, known as Gradient-weighted Class Activation Mapping (Grad-CAM), helps us understand what the model 'sees' as significant within an MRI scan, thus offering a visual explanation. The heatmaps can be seen in Figure 7.

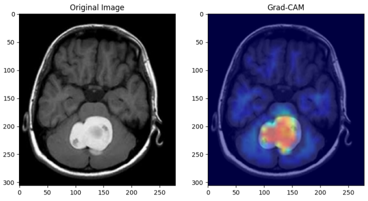


Fig. 7. GRAD-CAM heatmaps generated from Meningioma images of the "Brain Tumor MRI Dataset"[15] test set.

Validation. Random images were selected from the "Brain Tumor Classification (MRI)"[17] dataset to thoroughly test the trained model's performance. These images were not part of the model's training dataset, ensuring a fair evaluation of its generalization capabilities. We generated heatmaps by utilizing different convolutional layers of the CNN model to gain further insights into the model's decision-making process. The successful performance of the model on the random images reinforces its potential as a powerful tool for accurate brain tumor classification. The validation can be seen in Figure 8.

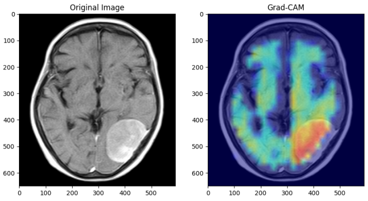


Fig. 8. GRAD-CAM heatmaps generated from Meningioma images of the "Brain Tumor Classification (MRI)"[17].

5 Conclusions

This study showcased encouraging results in accuracy and robustness, outperforming several other architectures within the same domain. The findings contribute significantly to the expanding body of research investigating deep learning techniques, particularly CNNs, for medical image analysis and their potential for clinical applications in real-world scenarios.

However, it is crucial to recognize the limitations and areas for improvement within the proposed model. For instance, incorporating more diverse and extensive datasets could further validate and enhance the model's performance. Errors that led to precision and recall reductions must be thoroughly understood to further improve the model. Additionally, exploring techniques such as Data Augmentation may improve the model's robustness and generalization capabilities.

In conclusion, this research highlights the potential of deep learning techniques, particularly CNNs, to enhance brain tumor classification, and supports ongoing efforts to develop more accurate and efficient diagnostic tools within the medical field. Future research should continue exploring, optimizing, and refining deep learning models for medical image analysis to fully harness their capabilities and maximize their impact in clinical applications.

References

1. An Approach in Brain Tumor Classification: The Development of a New Convolutional Neural Network Model. (2023). Retrieved from https://anonymous.4open.science/r/brain-tumor-cnn-C44F/.
2. Pooch, E.H.P., Alva, T.A.P., Becker, C.D.L.: A Deep Learning Approach for Pulmonary Lesion Identification in Chest Radiographs. BRACIS 2020: Intelligent Systems, vol. 12319, pp. 197–211 (2020).
3. Ribeiro, A.M., Junior, F.P.S.A.: Um Estudo Comparativo Entre Cinco Métodos de Otimização Aplicados Em Uma RNC Voltada ao Diagnóstico do Glaucoma. Revista de Sistemas e Computação, vol. 10, pp. 122-130 (2020).
4. Ebermam, E., Krohling, R.A.: Uma Introdução Compreensiva às Redes Neurais Convolucionais: Um Estudo de Caso para Reconhecimento de Caracteres Alfabéticos. Revista de Sistemas de Informação da FSMA, vol. 22, pp. 49-59 (2018).
5. Liimatainen, K., Huttunen, R., Latonen, L., Ruusuvuori, P.: Convolutional Neural Network-Based Artificial Intelligence for Classification of Protein Localization Patterns. Biomolecules, vol. 11, 264 (2021).
6. Wong, T. -T., Yeh, P. -Y.: Reliable Accuracy Estimates from k-Fold Cross Validation. IEEE Transactions on Knowledge and Data Engineering, vol. 32, pp. 1586-1594 (2020).
7. Villanueva-Meyer, J.E., Mabray, M.C., Cha, S.: Current Clinical Brain Tumor Imaging. Neurosurgery, vol. 81,3, pp. 397-415 (2017).
8. Xie, Y., Zaccagna, F., Rundo, L., Testa, C., Agati, R., Lodi, R., Manners, D.N., Tonon, C.: Convolutional Neural Network Techniques for Brain Tumor Classification (from 2015 to 2022): Review, Challenges, and Future Perspectives. Diagnostics (Basel), vol. 12,8 1850 (2022).

9. Arjun, T., Rishab, K.,Ganesh, B., Rajat, H., Sankalp, C.: AI Technique for Classification of Brain Tumor MRI Images for General Physicians. YMER, vol. 21, n. 10 (2022).
10. Tariq, R., Naqvi, N.: Classification of Medical Images using Deep Learning. SSRN (2022). https://doi.org/10.2139/ssrn.4121811
11. Ali, R., Al-jumaili, S., Duru, A.D., Uçan, O.N., Boyaci, A., Duru, D.G.: Classification of Brain Tumors using MRI images based on Convolutional Neural Network and Supervised Machine Learning Algorithms. International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), pp. 822-827 (2022). https://doi.org/10.1109/ISMSIT56059.2022.9932690
12. Wadhwa, A., Bhardwaj, A., Verma, V.S.: A review on brain tumor segmentation of MRI images. Magnetic Resonance Imaging, vol. 61, pp. 247-259 (2019).
13. Cancer.Net, https://www.cancer.net/cancer-types/brain-tumor. Last accessed 12 Mar 2023
14. National Institute of Neurological Disorders and Stroke. Brain Tumors Fact Sheet, https://www.ninds.nih.gov/Disorders/Patient-Caregiver-Education/Fact-Sheets/Brain-Tumors-Fact-Sheet. Last accessed 31 Mar 2023
15. Msoud Nickparvar. (2021). Brain Tumor MRI Dataset [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/2645886
16. Figshare Brain Tumor Classification [Data set]. kaggle.com/datasets/rahimanshu/figshare-brain-tumor-classification. Last accessed 17 May 2023
17. Sartaj Bhuvaji, Ankita Kadam, Prajakta Bhumkar, Sameer Dedge, Swati Kanchan. (2020). Brain Tumor Classification (MRI) [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/1183165
18. Perkins, A., Gerald L.: Primary Brain Tumors in Adults: Diagnosis and Treatment. American family physician, vol. 93, 3 (2016).
19. Chan, Heang-Ping et al.: Deep Learning in Medical Image Analysis. Advances in experimental medicine and biology, vol. 1213, pp. 3-21 (2020).
20. Yadav, S., Shukla, S.: Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification. 2016 IEEE 6th International Conference on Advanced Computing (IACC), (2016). https://doi.org/10.1109/iacc.2016.25
21. Louis, D.N., Perry, A., Reifenberger, G., von Deimling, A., Figarella-Branger, D., Cavenee W.K., et al.: The 2016 World Health Organization Classification of Tumors of the Central Nervous System: a summary. Acta Neuropathol. vol. 131(6), pp. 803 (2016). https://doi.org/10.1007/s00401-016-1545-1.
22. Louis, D. N., et al.: The 2021 WHO Classification of Tumors of the Central Nervous System: a summary. Neuro-oncology. vol. 23(8), pp. 1231-1251 (2021). https://doi.org/10.1093/neuonc/noab106
23. Nandu, H., et al.: Imaging in neuro-oncology. Therapeutic advances in neurological disorders. vol. 11 (2018). https://doi.org/10.1177/1756286418759865