Jurnal Ilmiah KURSOR
Menuju Solusi Teknologi Informasi
Vol. 13, No. 2, Desember 2025
ISSN 0216 – 0544
e-ISSN 2301– 6914

Convolutional layer exertion on few-shot learning for brain tumor classification

Victor Immanuel Sunarko[a], Eva Yulia Puspaningrum[b], Riana Retno Widiastuty[c], Surjo Hadi[d], Mohd Khalid Awang[e], I Gede Susrama Mas Diyasa[f]

[a,b] *Department of Informatics, Faculty of Computer Science, University of Pembangunan Nasional Veteran Jawa Timur, Surabaya, Indonesia*
[c] *Department of Medical Education, Faculty of Medicine, University of Pembangunan Nasional Veteran Jawa Timur Indonesia*
[d] *Department of Industrial Engineering, Faculty of Engineering, University of Yos Soedarso Surabaya, Indonesia*
[e] *Faculty of Informatics and Computing, University Sultan Zainal Abidin Besut Campus, 22200 Besut, Terengganu, Malaysia*
E-mail: 21081010145@student.upnjatim.ac.id, evapuspaningrum.if@upnjatim.ac.id, riana.retno.fk@upnjatim.ac.id, bestsuryo@yahoo.com, khalid@unisza.edu.my, igsusrama.if@upnjatim.ac.id

Abstract

Brain tumors, though relatively rare, pose a significant threat due to their critical location within the brain, impacting essential bodily functions. Accurate and timely diagnosis is vital, but traditional diagnostic methods are time-intensive and rely heavily on large labeled datasets. This study addresses these challenges by proposing a Few-Shot Learning (FSL) framework enhanced with Convolutional Neural Networks (CNNs) to classify brain tumors using MRI images. By employing the Matching Network architecture, the model leverages limited training data through an N-way-K-shot setup. Training results demonstrated accuracy levels of 71.58% (1-shot) and 82.89% (5-shot) for 1-layer CNNs, 66.65% (1-shot) and 84.03% (5-shot) for 3-layer CNNs, and 63.43% (1-shot) and 84.94% (5-shot) for 5-layer CNNs. However, validation accuracy revealed overfitting concerns, with the highest performance at 51.56% (1-layer, 1-shot). These results underscore the potential of FSL in medical imaging while highlighting the need for advanced augmentation and feature representation techniques to improve generalization.

Key words: Brain Tumor, Convolutional Neural Networks, data augmentation, Few-Shot Learning, MRI images.

INTRODUCTION

Brain tumors occur when cells or tissues grow abnormally in the brain area. In Indonesia, brain tumor cases are considered rare compared to other types of tumors, such as breast cancer. According to GCO data from 2022, there are 2 brain tumor cases per 100,000 inhabitants, a relatively low figure compared to breast cancer cases, which stand at 41.8 per 100,000 inhabitants. However, the location of these abnormal cells makes brain tumors particularly dangerous. The brain functions as the control center of the human body, so any abnormality in this area can lead to severe and extensive complications, such as swelling and pressure on blood vessels. Given the life-threatening nature of brain tumors, if not addressed promptly, it is crucial for medical experts to make swift and accurate decisions.

The diagnosis of brain tumors typically relies on imaging from Magnetic Resonance Imaging (MRI) equipment. This is due to MRI's ability to provide detailed images that highlight brain structures more effectively than CT scans or ultrasounds [1]. MRI imaging creates contrast in brain fibers, making it easier to distinguish between solid structures and tumors. Additionally, brain tumor MRI scans can be viewed in three different planes: axial, coronal, and sagittal. The axial plane divides the brain into top and bottom sections [2], the coronal plane separates it from front to back, and the sagittal plane divides it parallel to the body's midline, from left to right [3]. This variety of perspectives enhances the visualization of important features in both normal and tumorous brain structures.

The brain's complex structure and the variety of potential brain conditions make accurate diagnosis a time-consuming task for medical teams [4]. As technology advances, improving the speed and accuracy of diagnoses becomes increasingly important, especially since traditional methods for diagnosing brain tumors are predominantly manual. One promising approach to enhance both the speed and accuracy of diagnosis is the use of artificial intelligence (AI) [5].

AI has become an integral part of daily life, particularly in healthcare. In the medical field, it is commonly used to support decision-making and image analysis. AI helps provide valuable information, including treatment recommendations, options, and additional insights [6].

Recent studies indicate that artificial intelligence (AI) systems and AI-based applications are increasingly being used to enhance the professional medical environment [7]. It is crucial to apply AI techniques, particularly in the field of computer vision, to improve the speed and accuracy of brain tumor diagnosis [8]. Computer vision plays a vital role in assisting with brain tumor diagnosis by identifying objects and even individuals within images [9]. By utilizing methods from computer vision, computers can interpret contextual information from image data based on specific patterns extracted directly from the images.

One of the most used methods in image recognition is the Convolutional Neural Network (CNN) [10] [11] [12], which is a popular architecture for image classification. The convolutional layers effectively highlight essential features, such as the edges and textures of an object, allowing machines to recognize the object's unique identity more accurately. However, a primary limitation of the CNN architecture is its complexity. When developers try to increase the model's complexity, they may encounter the vanishing gradient problem, where key image features can be lost due to the pooling layers in the CNN structure.

In Emrah Irmak's 2021 study [13], CNNs were used as the foundation for multi-class classification of brain tumors. The study optimized CNN architecture by integrating three classification processes within a single AI model. It utilized data from four datasets, totaling approximately 11,500 brain MRI images, and achieved exceptional performance: 99.33% accuracy for the first layer classification, 92.66% for the second layer, and 98.14% for the third. This demonstrates the potential of CNNs to handle high complexity with substantial training data. However, the study was limited to axial brain MRI images, which resulted in uniformity in the extracted tumor features.

Another one-sectioned brain tumors classification study by Suganya Athisayamani et al. [14] in 2023 utilized ResNet-152 with the Chimp Optimizer for classifying brain tumors, focusing specifically on axial plane images. This approach achieved a training accuracy of 98.85% and a validation accuracy of 97.64%. By concentrating on just one brain plane for data selection, the researchers effectively reduced the gap between training and validation accuracy.

In a 2020 study, Bakhtyar Ahmed Mohammed and Muzhir Shaban Al-Ani [15] utilized Convolutional Neural Network (CNN) architecture to classify brain tumors, demonstrating its effectiveness in diagnosing brain tumors through MRI analysis. They achieved a classification accuracy of 98.677% in identifying tumor types such as ependymoma, meningioma, and medulloblastoma. This research emphasized the importance of optimizing the number of training epochs, as increasing the epochs from 1 to 15 significantly improved the system's accuracy and sensitivity. However, challenges such as overfitting and dependence on dataset quality persisted, highlighting the need for a

careful approach in developing classification models.

In 2021, Steinar Laenen and Luca Bertinetto conducted a study comparing Few-Shot Learning approaches using Matching Networks and Prototypical Networks. The authors utilized three datasets commonly employed in Few-Shot Learning training: MiniImageNet, CIFAR-FS, and tieredImageNet. The testing scenarios involved using ResNet-12 as the feature extraction layer, followed by training with both Matching and Prototypical Networks approaches. Their findings concluded that episodic training significantly impacts the performance of Few-Shot Learning models, regardless of whether the Matching or Prototypical Networks approach is used [16]. The authors also referenced earlier research from 2016 by Oriol Vinyals et al., which highlighted the continued relevance of the Matching Network method, provided that the similarity metric is modified from cosine similarity to Euclidean distance.

These studies collectively indicate that relying solely on CNN architecture may not yield significant results [17] [18] in brain tumor classification. The demand substantial amounts of data, MRI data undergoes various preprocessing techniques, such as separating sectional planes, research focusing on this segmentation shows robust and high accuracy [1]. Moreover, Few-Shot Learning is designed to tackle the lack of data availability problem by utilizing cosine similarity function [19]. It allows extracted feature to be compared and calculated its distances, later predicted using softmax function.

Data separation based on sectional planes is crucial for improving the performance of the proposed model [13] [14]. By keeping the sectional planes distinct, the model preserves specific information and unique patterns associated with each plane, preventing them from merging [20] [21]. This separation will be explained in the next discussion that enables the model to identify brain tumors more accurately by recognizing distinct, unmixed patterns.

Considering the background provided, the main objective of this study is to design and exert a Convolutional based Few-Shot Learning model that can classify brain tumors, specified in axial sectional plane obtained from MRI scans while ensuring robustness. This research will outline the data preprocessing steps, which include cleaning, separation, and formatting. The prepared data will be processed using the proposed model, and comparisons will be made across different Convolutional composition scenarios.

MATERIAL AND METHODS

As the research progresses, a block diagram is developed to keep the study focused on its primary objective. This diagram provides a functional overview of the research workflow, illustrating how each element of the study is interconnected. This structured approach helps to maintain clarity and alignment with the research goals.

The block diagram in Figure 1 illustrates the initial step of data acquisition, specifically focusing on MRI images of brain tumors. After acquiring the data, it is categorized according to the perspective of each MRI image. Once processed, this data is input into a classification model under various scenarios, and its accuracy and robustness are evaluated.

Data Acquisition

The data collection process started with a review of previous studies to identify the appropriate type of data for analysis. We selected brain tumor classification data that utilizes MRI images. This data was sourced from Kaggle, specifically from the dataset titled "MRI for Brain Tumor with Bounding Boxes."

Table 1. Brain tumors data

No	Category	Train	Val
1	Glioma	1,153	136
2	Meningioma	1,449	140
3	Pituitary	1,424	136
4	No Tumor	711	100
	Total	4,737	512

The collected data consists of secondary data, comprising 5,249 training images in ".jpg" format, categorized as shown in Table 1.

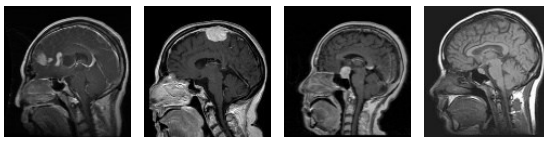

Fig. 2. Brain tumor sample data

Fig. 2 shows a sample of the data used in this study. From left to right, the images are labeled as Glioma, Meningioma, Pituitary, and No

Tumor. This data will be preprocessed for the subsequent stages.

Data Acquisition

The collected data will then undergo a manual cleaning process. This cleaning involves sorting the data based on axial, coronal, and sagittal perspectives. The results of this categorization into the three perspectives are shown in Fig. 3. below.

Due to the data imbalance observed in the no tumor-coronal and no tumor-sagittal categories, it is not feasible to include these label groups in the study. Therefore, this research focuses on training and evaluation using data from the axial perspective. The selected data includes 362 Glioma images, 391 Meningioma images, 617 No Tumor images, and 431 Pituitary images. Meanwhile, the validation set consists of 61 Glioma images, 62 Meningioma images, 80 No Tumor images, and 53 Pituitary images.

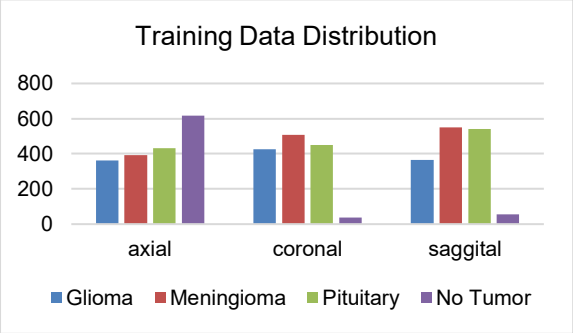

Fig. 3. Brain tumors training distribution

Data Augmentation

Referring to Fig. 3, which was discussed in the previous chapter, we can gain further insights that the selected axial data still exhibits significant imbalance, particularly between the no tumor-axial class and the other classes. To address this, data augmentation was applied to each class to reduce the disparity in data quantities [22] [23]. The augmentation technique used involves brightness adjustment, utilizing the following formula.

$$f(x,y)' = f(x,y) \pm b \quad (1)$$

As shown in Equation 1, brightness adjustment involves manipulating pixel values by either increasing or decreasing a scalar value for each pixel. Increasing the value of each pixel in the image enhances its brightness. Conversely, decreasing the value of each pixel reduces the image's brightness.

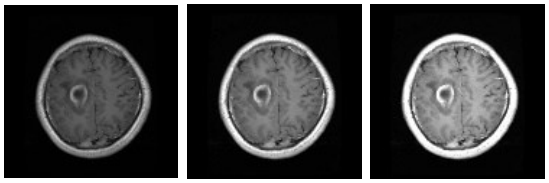

Fig. 4. Augmented data visualization

Illustrated in Fig. 4, the images from left to right show the results of darkening, a normal image, and brightness enhancement, respectively. This aligns with the pixel scale, where higher pixel values approach white, while lower pixel values tend toward black.

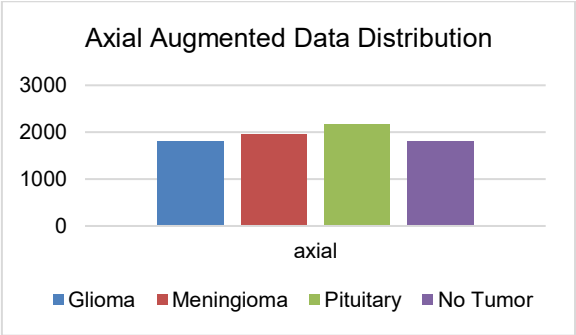

Fig. 5. Axial augmented data distribution

The augmented data from the four classes, specifically from the axial perspective, is presented in Fig. 5 above. The resulting augmented data includes 1,810 Glioma images, 1,945 Meningioma images, 1,791 No Tumor images, and 2,155 Pituitary images. The final dataset contains approximately 2,000 images per class, significantly reducing the data imbalance compared to the initial distribution

Data Formatting

In this research, the training method adopted is similar to the training strategy used in Matching Networks. The data will be formatted in an N-way-K-shot setup, where N represents the number of classes in the data and K denotes the number of example data per class. The data will first be split into a support set and a query set, where K data points are randomly selected from each of the N classes to be included in the support set [19]. Thus, there will be N x K pairs and N unpaired instances for a single training episode. Since the data used consists of 4 classes, the support set will include 4 different

tumor types, with the number of samples for each type varying.

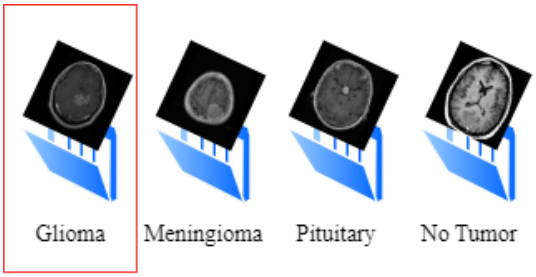

Fig. 6. Query set visualization

In the pairing process, the 4-class array is transformed into training-ready data, starting by selecting an image to be used as the query. For example, in Fig. 6, one Glioma image is selected as the query, which will later serve as the reference for data similarity. This process results in a 1-dimensional array called the query set, which consists of all the images in the dataset.

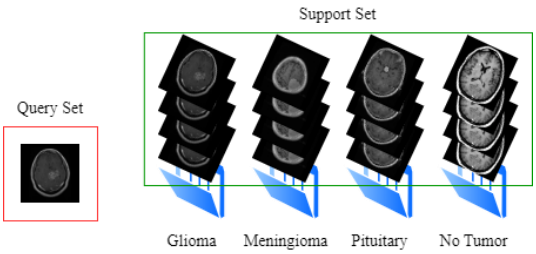

Fig. 7. Support set visualization

Next, K images are randomly selected from each class, ensuring that at least 4 images are chosen as support. For example, in Fig. 7, 4 images are randomly selected from each label, without being constrained by the previously chosen query or label. Although the selection includes 4 different classes, all selected images will be paired with the previously chosen query. These selected images are referred to as the support set. The 16 selected images will be stored initially in 4 separate 1-dimensional arrays, each corresponding to the directory from which the images were taken. These four arrays will then be combined into a single 1-dimensional array, resulting in a 2-dimensional array, with the first dimension representing the length of the query set and the second dimension representing the predetermined value of K.

This data pairing process will repeat the same procedure for each image from the four classes, incorporating them into the query set. The pairing process will be complete once all images in the training data have been assigned to the query set. The outcome of this process will produce at least 1 query and 4 support images within a single prediction episode. Furthermore, the paired data will be divided into several batches with adjusted sizes to reduce the data loading burden during training.

Proposed Model

Few-Shot Learning is a visual recognition technique that involves many categories with insufficient training data for conventional classification [16]. Unlike conventional supervised classification techniques, which require hundreds or even thousands of labeled data to train deep learning models to recognize data classes, Few-Shot Learning addresses scenarios where acquiring such large datasets is difficult or costly. With seemingly limited data, Few-Shot Learning excels in classifying data categories, making it a practical approach in situations where traditional supervised learning is not feasible.

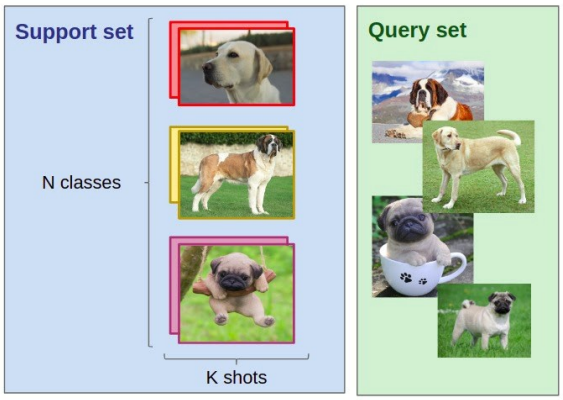

Fig. 8. Few-Shot learning data structure

The training or prediction structure of Few-Shot Learning, as shown in Fig. 8, typically applies an N-way-K-shot framework, which will henceforth be referred to as an episode. Each episode consists of N labels and K support images per label, resulting in a total of NxK support images [24]. Each set of K support images per label is called the support set, while the data to be predicted is referred to as the query set.

In developing the Few-Shot Learning framework, the Matching Network is utilized as a processing method for both query sets and support sets. The Matching Network is a neural network architecture specifically designed for

One-Shot Learning and Few-Shot Learning tasks. This model operates by mapping a set of support instances to generate an output represented as a probability distribution over potential class. Instead of training the model to recognize each class in isolation, the Matching Network capitalizes on the information available in the support set, enabling it to make informed predictions based on the similarity to the instances within that set [16].

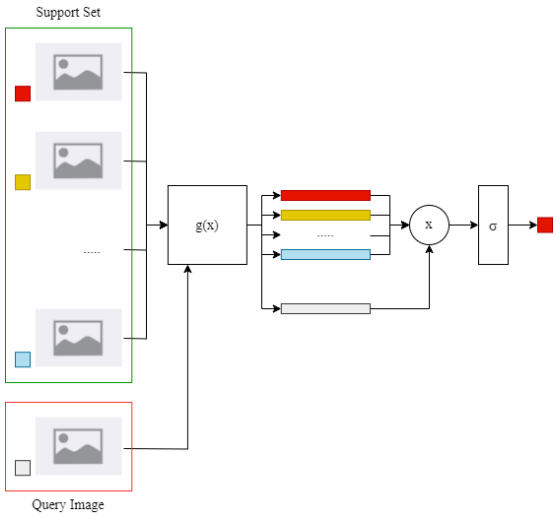

Fig. 9. Matching network architecture

In Fig. 9, the architecture of the Matching Network is visualized with an example input in the form of image data. The input to the Matching Network consists of the query image and the support set, which are processed by the encoder function $g(x)$ to generate feature map vectors using convolutional layers. Convolution is an ordered operation that involves two or more related pieces of information. It processes data in matrix form regionally, where each data point is processed along with its neighboring points using a dot product operation with the existing filters [15].

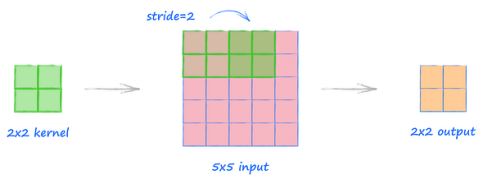

Fig. 10. Convolutional layer illustration

The training condition of deep learning, a filter is a matrix, as shown in Fig. 10, which automatically adjusts itself. The expectation is that when the data passes through the convolution operation, it enhances specific values in a data matrix or image, resulting in a distinctive identity for each data point. This adjustment allows the model to learn the most relevant features from the data, helping to improve the model's ability to recognize patterns and make accurate predictions [25].

$$a \cdot b = a_1b_1 + a_2b_2 + a_3b_3 + \cdots + a_nb_n \quad (2)$$

The matrix will undergo a series of matrix operations, including the dot product, to generate a feature map. The dot product operation, as shown in Equation 2, produces a 2x2 feature map matrix. Overall, the data will exhibit a specific pattern generated from the previous identity search, commonly referred to as the feature map. This feature map captures the most relevant characteristics of the input data, which are essential for the model's understanding and subsequent classification or prediction tasks.

$$cosine\ similarity(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad (3)$$

Cosine similarity is a metric used to measure the similarity between the flattened results of the test and training data [26]. As shown in Equation 3, this metric calculates similarity based on the angle between two vectors, ignoring their magnitudes, and produces a value ranging from -1 to 1. A value of 1 indicates the two vectors are highly similar, 0 indicates no similarity, and -1 indicates the two vectors have opposite directions. Using the cosine similarity layer, the query image, which is already represented as a feature map, is compared to the feature maps of each support set to calculate their similarity

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad (4)$$

Through the matching network structure, the Few-Shot Learning model is able to compute the similarity between the support set and the query set using the softmax activation function. The softmax function can be formulated as shown in Equation 4. The input to the softmax function consists of a set of K elements from the vector $\vec{z}$. The numerator of the softmax formula represents the exponential function calculation for one element of the vector $\vec{z}$. Meanwhile, the denominator of the softmax formula represents the sum of the exponential

functions of all elements of the vector $\vec{z}$. Softmax is an activation function commonly used in the final layer of multi-class classification models [27]. This function converts the logit scores into probabilities by mapping each input value to a number between 0 and 1, with the total summing to 1 [8]. Each class receives a probability corresponding to the logit value produced. The softmax function is often used to interpret classification results, especially in neural networks used for tasks such as image or text classification.

$$L_{CE} = -\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (5)$$

Categorical Crossentropy calculates how well the model predicts the correct class by measuring the logarithmic distance between the predicted probabilities and the actual labels [28]. As shown in Equation 5, y_i represents the probability assigned to the predicted label by the model. The goal is to minimize this loss, allowing the model to become more accurate in predicting the correct class.

There are four general approaches Few-Shot learning architecture: data-level, parameter-level, metric-level, and gradient-based Meta-learning. This study focuses on the metric-level approach, utilizing cosine similarity as a similarity measure and Matching Network as the main architecture.

The optimization method applied to Few-Shot Learning focuses on the Matching Network workflow by adding complexity to the convolutional layers before the embedding process. Considering the advantages of convolutional layers in image feature extraction and their widespread use, the hypothesis formed is that increasing the complexity of the convolutional layers will correspond to improved accuracy in image recognition. In the process of calculating similarity in Few-Shot Learning, a distinctive feature map is necessary to facilitate pattern discovery. By adding complexity to the feature extraction layers of the convolutional network, it is expected that the performance and flexibility of the Few-Shot Learning method will be enhanced.

Evaluation Metrics

The trained model's performance is evaluated using accuracy derived from the confusion matrix, a table that provides a comprehensive assessment by comparing model predictions to actual labels. This matrix includes four main components: True Positives (TP), which represent instances correctly identified as positive; True Negatives (TN), indicating instances accurately classified as negative; False Positives (FP), where instances are mistakenly classified as positive; and False Negatives (FN), where instances are incorrectly classified as negative. Together, these elements offer a detailed view of the model's strengths and weaknesses, allowing for an analysis that goes beyond simple accuracy and reveals patterns in prediction errors, helping identify any tendencies the model may have toward overpredicting or underpredicting specific classes.

$$Accuracy = \frac{TP+TN}{(TF+TN+FP+FN)} \quad (6)$$

The specific metric extracted from the confusion matrix for evaluation is overall accuracy, calculated as shown in Equation 6. This involves summing the total number of correct predictions—both true positives (TP) and true negatives (TN)—and dividing this by the total number of predictions made. This ratio provides a straightforward measure of the model's general classification accuracy, offering insight into how effectively the model is distinguishing between classes across the dataset.

RESULT AND DISCUSSION

The research results were derived from training each model under various scenarios. Each training session used a single data source: secondary data collected from the Kaggle website. This standardized data source ensured consistent input quality across all model scenarios, facilitating a clear comparison of each model's performance in different experimental settings.

Table 2. Training time of proposed few-shot learning with augmented data

Convolutional Layer	Support Set	
	1 shot	5 shot
1 Layer	± 6m 30s	± 1m 40s
3 Layers	± 55s	± 1m 15s
5 Layers	± 45s	± 1m 5s

Table 2 highlights a trend of decreasing training time as the number of CNN layers increases. The model with 1 layer recorded the longest training time, taking approximately 6

minutes and 30 seconds for the 1-shot scenario and 1 minute and 40 seconds for the 5-shot scenario. In contrast, the 3-layer model significantly reduced training time, requiring only 55 seconds for 1-shot and 1 minute and 15 seconds for 5-shot. The 5-layer model achieved the shortest training times, at 45 seconds for 1-shot and 1 minute and 5 seconds for 5-shot. This trend suggests that deeper architectures enable more efficient feature extraction, with MaxPooling layers effectively reducing data dimensions and deeper layers facilitating faster convergence by generating more optimal representations.

The comparison between 1-shot and 5-shot scenarios reveals an interesting pattern. In the 1-layer model, the 1-shot scenario required nearly four times the training time of the 5-shot scenario, highlighting the difficulties simpler models face when processing small support sets. However, this time difference decreased significantly in the 3-layer model and became minimal in the 5-layer model. This suggests that deeper CNN architectures are more efficient at managing larger support sets, resulting in more consistent training times across different scenarios.

Table 3. Training accuracy of proposed few-shot learning with augmented data

Convolutional Layer	Support Set	
	1 shot	5 shot
1 Layer	71.58%	82.89%
3 Layers	66.65%	84.03%
5 Layers	63.43%	84.94%

The training accuracy results in Table 3 demonstrate contrasting trends between 1-shot and 5-shot scenarios as the number of CNN layers increases. In the 1-shot scenario, there is a decline in training accuracy with deeper architectures, dropping from 71.58% for the 1-layer model to 66.65% for 3 layers and 63.43% for 5 layers. Conversely, in the 5-shot scenario, deeper models achieve higher training accuracy, with the 5-layer model performing the best at 84.94%. These trends indicate that while deeper models are better suited for processing larger support sets, simpler architectures may perform better when the support set is limited.

Validation accuracy results in Table 4 reveal significant overfitting across all scenarios, as evidenced by the substantial gap between training and validation accuracy. Validation accuracy decreases as the number of layers increases for both 1-shot and 5-shot scenarios, with the best performance achieved by the 1-layer model in the 1-shot scenario (51.56%). However, validation accuracy drops sharply with deeper models, particularly in the 5-shot scenario, where the 5-layer model achieves only 42.58%. These findings suggest that adding more layers diminishes the model's ability to generalize, likely due to an overcomplex representation of features.

Table 4. Validation accuracy of proposed few-shot learning with augmented data

Convolutional Layer	Support Set	
	1 shot	5 shot
1 Layer	51.56%	47.66%
3 Layers	45.31%	43.36%
5 Layers	44.14%	42.58%

The gap between training and validation accuracy is particularly pronounced in deeper architectures, with the largest gap observed in the 5-layer model for the 5-shot scenario (84.94% vs. 42.58%). This highlights the challenge of overfitting when using deeper models in few-shot learning. Additionally, the smallest gap, observed in the 1-layer model with 1-shot (71.58% vs. 51.56%), underscores the superior generalization of simpler architectures. These results suggest that the random brightness augmentation alone may not be sufficient to combat overfitting and incorporating more diverse augmentation strategies or regularization techniques might be necessary to improve generalization.

Fig. 11. illustrates the different outcomes for each training scenario. The matrix showing how many data points were correctly and incorrectly classified using augmented data in various convolutional layers combination. From left to right, there are One-Shot Learning with 1 convolutional layer, 3 convolutional layer, 5 convolutional. These variations in training histories demonstrate how each data-handling approach affects model performance across the different scenarios.

Based on the confusion matrix for the 1-layer CNN scenario with 4 classes and 1-shot, the model shows excellent performance on the Meningioma class (class 0) with 100% accuracy. However, it performs poorly in

recognizing the Pituitary class (class 2) and No Tumor class (class 3). Most classification errors occur by predicting other classes as Meningioma, indicating a potential dominance of this class's representation in the feature space. The Pituitary class is frequently misclassified as Meningioma, while the No Tumor class is almost entirely unrecognized, with most predictions falling under Meningioma.

In the 3-layer CNN scenario with 1-shot, the model shows improved recognition capabilities compared to the 1-layer CNN, though significant weaknesses remain. The Meningioma class (class 0) still performs very well, with 79 correct predictions and only one misclassification to the Pituitary class (class 2). The Glioma class (class 1) shows 34 correct predictions but still has many misclassifications to Pituitary (16) and a few others. The Pituitary class (class 2) remains difficult to recognize, with only 2 correct predictions, and the majority are misclassified as Meningioma (47). The No Tumor class (class 3) shows slight improvement, with one correct prediction but still most misclassifications to Meningioma (47). Overall, the model demonstrates better robustness for Meningioma and Glioma classes but still struggles with Pituitary and No Tumor, likely due to overlapping feature distributions.

In the 5-layer CNN scenario with 1-shot, the model shows improved feature representation compared to models with fewer layers. The Meningioma class (class 0) continues to perform well, with 77 correct predictions and only a few errors spread across other classes (total of 3). The Glioma class (class 1) shows moderate improvement with 30 correct predictions, but there are still significant misclassifications to Pituitary (21) and other classes. The Pituitary class (class 2) remains difficult to recognize, with only 6 correct predictions and the majority misclassified as Meningioma (38). The No Tumor class (class 3) shows slight improvement with 1 correct prediction, but still predominantly misclassified as Meningioma (49). Overall, the model is more robust for Meningioma, but Pituitary and No Tumor remain challenging, indicating that the feature space distributions for these classes still overlap.

Moving on to the next scenarios, from left to right, Fig. 12. illustrates the different outcomes for Few-Shot (5-shot) Learning with 1 convolutional layer, 3 convolutional layer, 5 convolutional. These variations in training histories demonstrate how each data-handling approach affects model performance across the different scenarios.

In the 1-layer CNN scenario with 5-shot, the model shows slight improvement for the Glioma class (class 1) compared to the 1-shot scenario, with 40 correct predictions, though significant errors still occur, particularly misclassifying 11 samples as Meningioma (class 0). The Meningioma class (class 0) continues to maintain high accuracy with 80 correct predictions and no misclassifications. The Pituitary class (class 2) remains very weak, with most predictions misclassified as Meningioma (55). The No Tumor class (class 3) shows slight improvement over the 1-shot scenario, with one correct prediction but still predominantly misclassified as Meningioma (52). Overall, the model exhibits a similar error pattern to the 1-shot scenario, where samples are more likely to be predicted as Meningioma, suggesting that the feature representation for this class is more dominant. Despite small improvements in some classes, the model's robustness remains limited, indicating the need for richer representations, such as adding more CNN layers and more diverse data augmentation, to better handle the dataset's complexity.

In the 3-layer CNN scenario with 5-shot, the model shows moderate improvement compared to the 1-shot scenario, but significant misclassification patterns across classes remain. The Meningioma class (class 0) continues to perform perfectly with 80 correct predictions and no errors. The Glioma class (class 1) shows improvement with 27 correct predictions, but errors are still present, particularly misclassifying 13 samples as Pituitary and 14 as No Tumor. The Pituitary class (class 2) remains challenging to recognize, with only 1 correct prediction and the majority misclassified as Meningioma (52). The No Tumor class (class 3) shows slight improvement, with 3 correct predictions, but most predictions still go to Meningioma (46). Overall, the model performs excellently for Meningioma, but its performance for other classes, especially Pituitary, remains weak. This indicates that although the addition of data (5-shot) and more CNN layers provide some improvement, the feature representation is still ineffective for classes with complex or overlapping distributions.

In the 5-layer CNN with 5-shot scenario, the model shows better feature representation compared to previous scenarios, particularly for the Meningioma class (class 0), which achieves perfect accuracy with 80 correct predictions and no errors. However, other classes still exhibit significant weaknesses. The Glioma class (class 1) records 15 correct predictions but continues to make many errors, particularly misclassifying 33 samples as Pituitary and 8 as No Tumor. The Pituitary class (class 2) shows slight improvement with 14 correct predictions, though most of its predictions are still misclassified as Meningioma (37). The No Tumor class (class 3) remains difficult to recognize, with all predictions incorrectly classified as Meningioma (52). Overall, the model demonstrates excellent robustness for the Meningioma class, but performance on the other classes remains far from optimal. This suggests that while the model benefits from additional layers and data (5-shot), feature representation for the more challenging classes is still insufficient.

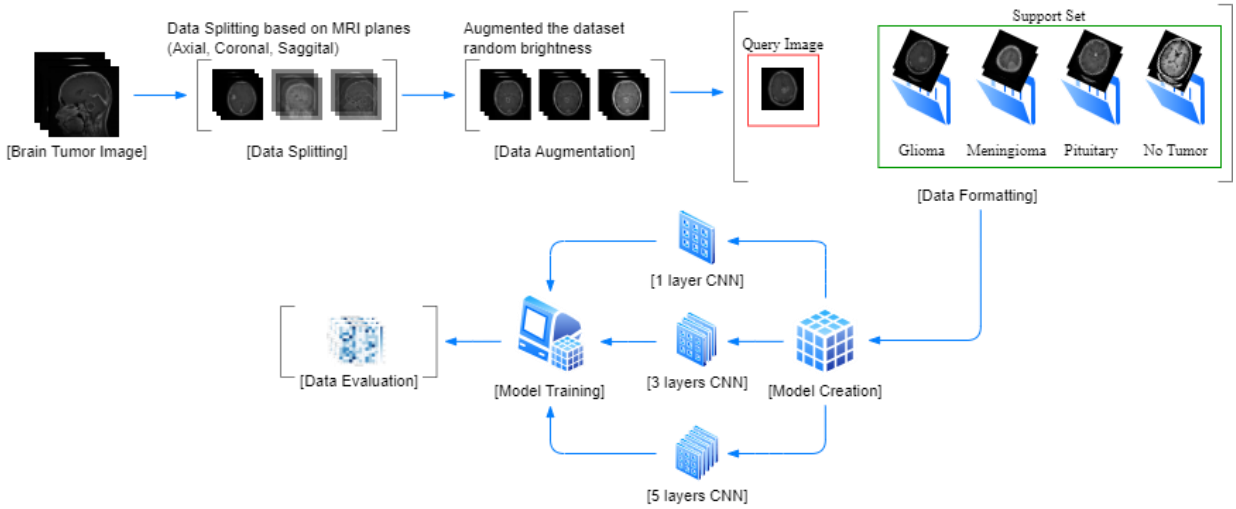

Fig. 1. Block diagram

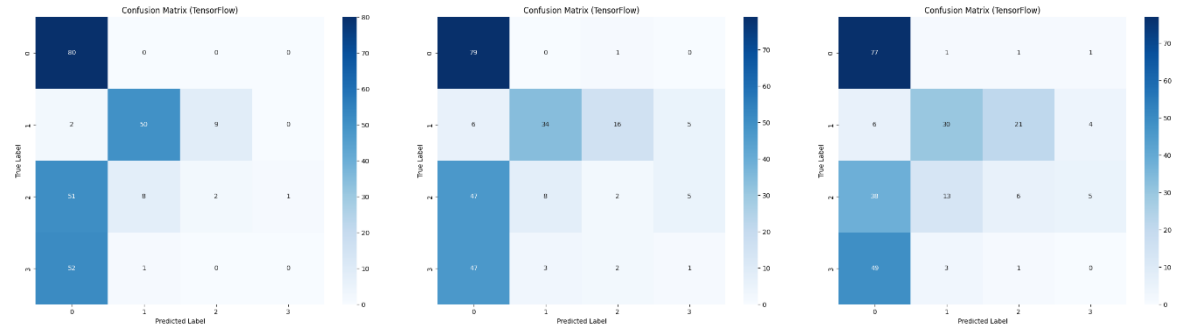

Fig. 11. One-shot with various convolutional layers combination

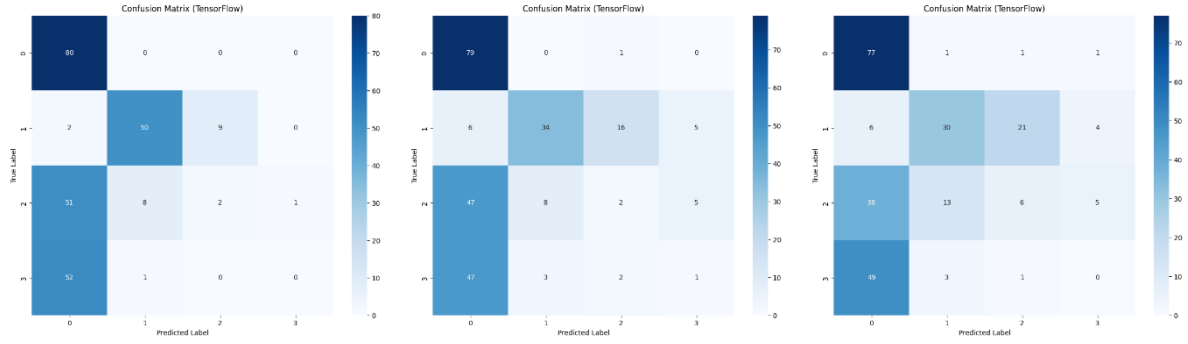

Fig. 12. Few-shot with various convolutional layers combination

CONCLUSION

This study explored the application of Few-Shot Learning (FSL) enhanced with Convolutional Neural Networks (CNNs) for brain tumor classification using MRI data, addressing the challenge of limited labeled datasets in medical imaging. The findings revealed that deeper CNN architectures, while reducing training time and improving accuracy in 5-shot scenarios, were prone to overfitting, particularly in validation phases. Specifically, the 1-layer CNN achieved the highest validation accuracy in the 1-shot scenario (51.56%), but deeper models struggled to generalize effectively.

The results highlighted a consistent performance advantage for the Meningioma class, while other classes, such as Pituitary and No Tumor, suffered from overlapping feature spaces and misclassification. This suggests that the representation of distinct patterns for challenging classes remains insufficient, even with enhanced convolutional layers and data augmentation.

Addressing these challenges, future research should focus on incorporating diverse data augmentation strategies, such as rotation, scaling, and flipping, to create a more balanced feature representation. Additionally, hybrid approaches that combine Few-Shot Learning with transfer learning or advanced regularization techniques could mitigate overfitting and improve model generalization. Expanding the dataset to include all three MRI planes (axial, coronal, and sagittal) could further enhance the model's ability to identify nuanced features across different perspectives. These improvements will strengthen the robustness of FSL models in medical diagnostics, enabling their practical application in real-world healthcare scenarios.

REFERENCES

[1] T. N. Papadomanolakis, E. S. Sergaki, A. A. Polydorou, A. G. Krasoudakis, G. N. Makris-Tsalikis, A. A. Polydorou, N. M. Afentakis, S. A. Athanasiou, I. O. Vardiambasis dan M. E. Zervakis, "Tumor Diagnosis against Other Brain Diseases Using T2 MRI Brain Images and CNN Binary Classifier and DWT," *Brain Sciences,* vol. 13, no. 2, p. 348, 2023, https://doi.org/10.3390/brainsci13020348 .

[2] W. Tian, D. Li, M. Lv dan P. Huang, "Axial Attention Convolutional Neural Network for Brain Tumor Segmentation with Multi-Modality MRI Scan," *Brain Sciences,* vol. 13, no. 1, p. 12, 2023, https://doi.org/10.3390/brainsci13010012 .

[3] S. S. Sabila dan H. P. A. Tjahyaningtyasa, "MULTIPLE BRAIN TUMORWITHMODIFIEDDENSENET121ARCHITECTURE USING BRAIN MRI IMAGES," *Jurnal Ilmiah KURSOR,* vol. 12, no. 8, pp. 147-158, 2024, https://doi.org/10.21107/kursor.v12i3.379 .

[4] M. Arabahmadi, R. Farahbakhsh dan J. Rezazadeh, "Deep Learning for Smart Healthcare—A Survey on Brain Tumor Detection from Medical Imaging," *Sensors,* vol. 22, no. 5, 2022, https://doi.org/10.3390/s22051960 .

[5] H. S. Basavegowda dan G. Dagnew, "Deep learning approach for microarray cancer data classification," *CAAI Transactions on Intelligence Technology,* vol. 5, no. 1, pp. 22-23, 2020, https://doi.org/10.1049/trit.2019.0028 .

[6] H. Ayeshaa, S. Iqbal, M. Tariqa, M. Abrar, M. Sanaullaha, I. Abbas, A. Rehmanc, M. F. K. Niazi dan S. Hussaine, "Automatic medical image interpretation: State of the art and future directions," *Pattern Recognition,* vol. 114, 2021, https://doi.org/10.1016/j.patcog.2021.107856 .

[7] I. G. S. M. Diyasa, W. S. J. Saputra, A. A. N. Gunawan, D. Herawati, S. Munir dan S. Humairah, "Abnormality

Determination of Spermatozoa Motility Using Gaussian Mixture Model and Matching-based Algorithm," *Journal of Robotics and Control,* vol. 5, no. 1, 2024, https://doi.org/10.18196/jrc.v5i1.20686 .

[8] R. Thangaraj, S. Anandamurugan dan V. K. Kaliappan, "Automated tomato leaf disease classification using transfer learning-based deep convolution neural network," *Journal of Plant Diseases and Protection,* vol. 128, pp. 73-86, 2021, https://doi.org/10.1007/s41348-020-00403-0 .

[9] C. Janiesch, P. Zschech dan K. Heinrich, "Machine learning and deep learning," *Electronics Markets,* 2021, https://doi.org/10.1007/s12525-021-00475-2 .

[10] W. Ayadi, W. Elhamzi, I. Charfi dan M. Atri, "Deep CNN for Brain Tumor Classification," *Neural Processing Letters,* vol. 53, pp. 671-700, 2021, https://doi.org/10.1007/s11063-020-10398-2 .

[11] I. G. S. M. Diyasa, A. Fauzi, M. Idhom dan A. Setiawan, "Multi-face Recognition for the Detection of Prisoners in Jail using a Modified Cascade Classifier and CNN," dalam *Journal of Physics: Conference Series*, 2021, https://doi.org/10.1088/1742-6596/1844/1/012005 .

[12] F. Badri, M. T. Alawiy dan E. M. Yuniarnoa, "Deep learning architecture based on convolutional neural network (cnn) on animal image classification," *Jurnal Ilmiah KURSOR,* vol. 12, no. 2, pp. 83-92, 2023, https://doi.org/10.21107/kursor.v12i2.349.

[13] E. Irmak, "Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering,* vol. 45, pp. 1015-1036, 2021, https://doi.org/10.1007/s40998-021-00426-9 .

[14] S. Athisayamani, R. S. Antonyswamy, V. Sarveshwaran, M. Almeshari, Y. Alzamil dan V. Ravi, "Feature Extraction Using a Residual Deep Convolutional Neural Network (ResNet-152) and Optimized Feature Dimension Reduction for MRI Brain Tumor Classification," *Diagnostics,* vol. 13, no. 4, p. 668, 2023, https://doi.org/10.3390/diagnostics13040668 .

[15] B. A. Mohammed dan M. S. Al-Ani, "An efficient approach to diagnose brain tumors through deep CNN," *Mathematical Biosciences and Engineering,* vol. 18, no. 1, pp. 851-867, 2020, https://doi.org/10.3934/mbe.2021045 .

[16] S. Laenen dan L. Bertinetto, "On Episodes, Prototypical Networks, and Few-Shot Learning," dalam *NeurIPS Proceedings*, 2021, .

[17] D. R. Nayak, N. Padhy, P. K. Mallick, M. Zymbler dan S. Kumar, "Brain Tumor Classification Using Dense Efficient-Net," *Axioms,* vol. 11, no. 34, 2022, https://doi.org/10.3390/axioms11010034 .

[18] R. Anantama, "Application Of Cost-Sensitive Convolutional Neural Network For Pneumonia Detection," *KURSOR,* vol. 11, no. 3, 2022, https://doi.org/10.21107/kursor.v11i3.264 .

[19] S. Mai, H. Hu dan J. Xu, "Attentive matching network for few-shot learning," *Computer Vision and Image Understanding,* vol. 187, 2019, https://doi.org/10.1016/j.cviu.2019.07.001 .

[20] A. G. Sooai, S. D. B. Mau, D. J. Manehat, Y. C. H. Siki, S. C. Sianturi dan A. H. Mondolang, "Optimizing Lantana Classification: High-Accuracy Model Utilizing Feature Extraction," *Jurnal Ilmiah KURSOR,* vol. 12, no. 2,

pp. 49-58, 2023, https://doi.org/10.21107/kursor.v12i2.347 .

[21] F. R. Kusumajati, B. Rahmat dan A. Junaidi, "Implementation Of Balancing Data Method Using Smotetomek In Diabetes Classification Using Xgboost," *KURSOR,* vol. 12, no. 4, 2024. https://doi.org/10.21107/kursor.v12i4.410 .

[22] G. Algana dan I. Ulusoyb, "Image classification with deep learning in the presence of noisy labels: A Survey," *Knowledge-Based Systems,* vol. 215, 2021, https://doi.org/10.1016/j.knosys.2021.106771 .

[23] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo dan C.-W. Lin, "Deep learning on image denoising : Anoverview," *Neural Networks,* vol. 131, pp. 251-275, 2020, https://doi.org/10.1016/j.neunet.2020.07.025 .

[24] D. Chen, Y. Chen, Y. Li, F. Mao, Y. He dan H. Xue, "Self-Supervised Learning for Few-Shot Image Classification," 2021, https://doi.org/10.1109/ICASSP39728.2021.9413783 .

[25] Y. Anagun, "Smart brain tumor diagnosis system utilizing deep convolutional neural networks," *Multimedia Tools and Applications,* vol. 82, p. 44527–44553, 2023, https://doi.org/10.1007/s11042-023-15422-w .

[26] C. Zhang, Y. Cai, C. Shen dan G. Lin, "DeepEMD: Few-Shot Image Classification with Differentiable Earth Mover's Distance and Structured Classifiers," dalam *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, https://doi.org/10.1109/CVPR42600.2020.01222 .

[27] F. Gao, B. Li, L. Chen, Z. Shang, X. Wei dan C. He, "A softmax classifier for high-precision classification of ultrasonic similar signals," *Ultrasonics,* vol. 112, 2021, https://doi.org/10.1016/j.ultras.2020.106344.

[28] M. R. M. Ariefwan, I. G. S. M. Diyasa dan K. M. Hindrayani, "InceptionV3, ResNet50, ResNet18 and MobileNetV2 Performance Comparison on Face Recognition Classification," *Literasi Nusantara,* vol. 4, no. 1, 2023.