

Two-step majority voting of convolutional neural networks for brain tumor classification

Irwan Budi Santoso, Shoffin Nahwa Utama, Supriyono

Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

Article Info

Article history:

Received Aug 15, 2024

Revised Mar 25, 2025

Accepted May 23, 2025

Keywords:

Brain tumor

Convolutional neural network

Image

Magnetic resonance imaging

Majority voting

ABSTRACT

Brain tumor type classification is essential for determining further examinations. Convolutional neural network (CNN) model with magnetic resonance imaging (MRI) image input can improve brain tumor classification performance. However, due to the highly variable shape, size, and location of brain tumors, increasing the performance of tumor classification requires consideration of the results of several different CNN models. Therefore, we proposed a two-step majority voting (MV) on the results of several CNN models for tumor classification. The CNN models included InceptionV3, Xception, DensNet201, EfficientNetB3, and ResNet50; each was customized at the classification layer. The initial step of the method is transfer-learning for each CNN model. The next step is to carry out two steps of MV, namely MV on the three CNN model classification results at different training epochs and MV on the results of the first step. The performance evaluation of the proposed method used the Nickparvar dataset, which included MRI images of glioma, pituitary, no tumor, and meningioma. The test results showed that the proposed method obtained an accuracy of 99.69% with a precision and sensitivity average of 99.67% and a specificity of 99.90%. With these results, the proposed method is better than several other methods.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Irwan Budi Santoso

Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim

Jalan Gajayana 50 Malang 65144, Malang, Indonesia

Email: irwan@ti.uin-malang.ac.id

1. INTRODUCTION

Tissue that grows due to abnormal cells in the brain or its surroundings can cause brain tumors [1]. Inaccuracy in classifying the type of brain tumor in a person can cause errors in subsequent medical actions, which can lead to death [2], [3]. To classify the type of tumor, radiologists usually look at the results of brain scans produced by magnetic resonance imaging (MRI). MRI images can map the internal structures of the human body [4], thus providing better visualization and spatial information [5]. However, manually classifying brain tumor types against MRI images can cause errors. Therefore, developing an automatic method is one solution to improve the performance of tumor type classification.

The method often used to classify brain tumors based on brain MRI images is convolutional neural network (CNN), as reported in [6]–[8]. Their hybrid schemes proposed involved CNN and conventional machine learning (ML) to obtain better brain tumor classification performance. The use of several CNN models in the scheme is to extract the best brain MRI image features as input to the classification stage using ML. The ML is also used before the classification stage to evaluate features and take some of the best features produced by several CNN models [6]. The best feature results are input into several ML models to

classify brain tumors. The combination of features produced by each model with several MLs at the classification stage is to choose the best classification performance results among the combinations, as shown in [7]. Ensuring that only one CNN model is used to extract brain MRI image features and continue at the classification stage with other models also improves classification performance [8]. The effort to suffice only using one CNN model is because their study emphasizes classifier optimization. Besides, using several CNN models for MRI image feature extraction in several studies aims to obtain representative features by the classifier. However, this effort will only be possible if the feature extraction and classifier methods are not separated. Then, their work was to try several models for feature extraction and combine several models (by trial and error) at the classification stage to choose the best one [9].

The proposed pre-processing and CNN methods in some previous studies also contributed to the performance of tumor classification. Pre-processing using a Gaussian filter for denoising brain MRI images improved CNN performance in tumor classification [10], [11]. However, due to the highly variable shape, size, and position of tumors in each type of brain tumor and the complicated structure of the human brain [12], only involving a CNN model would make it difficult to obtain better classification performance. To overcome this problem, several previous researchers combined convolutions from different models [13]; models with other architectures [2], [14] and some combined features from different modules or blocks [15]. Chatterjee *et al.* [13] proposed ResNet(1+2)D, which combines 1D and 2D convolutions, and proposed ResNet mixed convolution, which combines 2D and 3D convolutions. Their tests showed that proposed models were superior to ResNet3D with an accuracy of 96.98%. Khan *et al.* [2] combined the VGG16 with the 23-layer CNN architecture to avoid overfitting. The evaluation showed that their combined model produced classification accuracy of up to 97.8% and 100% for the first and second data sets. Younis *et al.* [14] conducted almost the same research by combining CNN with VGG16 and obtaining an accuracy of 98.14%. Different from that undertaken by Noreen *et al.* [15], combined features extracted from the pre-trained InceptionV3 module and entered into the SoftMax. They also combined the features of the pre-trained DensNet201 blocks and forwarded them to SoftMax. The test results showed that their proposed methods yielded accuracies of 99.34% and 99.51%, respectively. Combining convolutional [13] or combining features or architecture [2], [14], [15] has the same principle to obtain more representative features to the classifier or top layer in CNN, as well as the pre-processing method reported in [10], [11], [15]. However, involving only one or two CNN models in brain tumor classification is not enough to obtain the best classification results because brain tumors have high characteristics variation in shape, size, and position in MRI images. Combining the classification results of several CNN models through an ensemble combination of their classification results can be a solution to improve classification performance.

Based on the results, potential performance improvements and solutions are obtained by several previous studies. Combining the classification results of several CNN models is one solution to improve tumor classification performance. On the other hand, considering the classification results of many different CNN models can overcome high variations in shape and size, and the position of the tumor is difficult to determine [16]. Several well-known CNN models have different architectures with different numbers of convolution layers and convolution filter sizes and have the potential to obtain different MRI brain image features and classification results. Thus, a classification that considers the classification results of several CNN models indirectly considers the best brain MRI image features of each CNN model so that it can strengthen the final classification performance [9], [17], [18]. In addition, considering the results of CNN models at several different epochs in learning is also essential to reduce uncertainty in classification. Thus, considering the classification results using majority voting against the classification results of several CNN models at different epochs is a solution to overcome this problem. Therefore, this study proposes two-step majority voting (MV) based on several well-known CNN models for brain tumor classification. CNN models involved in classification include InceptionV3 [19], Xception [20], DensNet201 [21], EfficientNetB3 [22], and ResNet50 [23]. Each CNN model can provide the best results through transfer learning, accordingly, considering the classification results of these CNN models through majority voting can strengthen the final classification results. Two-step MV will strengthen the classification of each CNN by considering the classification results at different epochs, followed by the classification results among other CNN models. The experimental results showed that the two-step MV could improve the performance of brain tumor classification. Hence, there are several contributions to this study, which include: a) involving several CNN models (InceptionV3, Xception, DensNet201, EfficientNetB3, and ResNet50) by adjusting the top layer of each model for brain tumor classification, and b) proposing a two-step majority voting scheme to combine the classification results of CNN models at different learning epochs.

In this study, Section 2 describes the dataset for evaluation, the steps of the proposed method, and the hyperparameters of the proposed method. Section 3 contains the results of the test and discusses the results. Section 4 explains the comparison of the proposed method with several previous methods. Finally, section 5 contains conclusions based on the results and discussions.

2. MATERIAL AND METHOD

The classification of brain tumors in this study is to classify the types of brain tumors, including glioma, meningioma, and pituitary and no tumor. Measuring the classification performance of the proposed method in the classification of brain tumors is one of the objectives of this study. Figure 1 shows the steps of brain tumor classification with the proposed method. The steps include dataset acquisition, pre-processing, training of base CNN models, and two-step majority voting. Dataset acquisition involves acquiring MRI images of brain tumors (glioma, meningioma, and pituitary) and no tumors for training and testing. The process also involves splitting the training dataset into datasets for learning and model validation. The next step is pre-processing to resize each MRI brain image to the same size for training and testing. All MRI image data of the same size were forwarded into the training process. This stage required several hyperparameters in the learning process of several CNN models to obtain the weights of each trained model. The last step, brain tumor classification, uses the proposed method, namely two-step MV, to develop an ensemble combination of the first prediction results of each CNN model used.

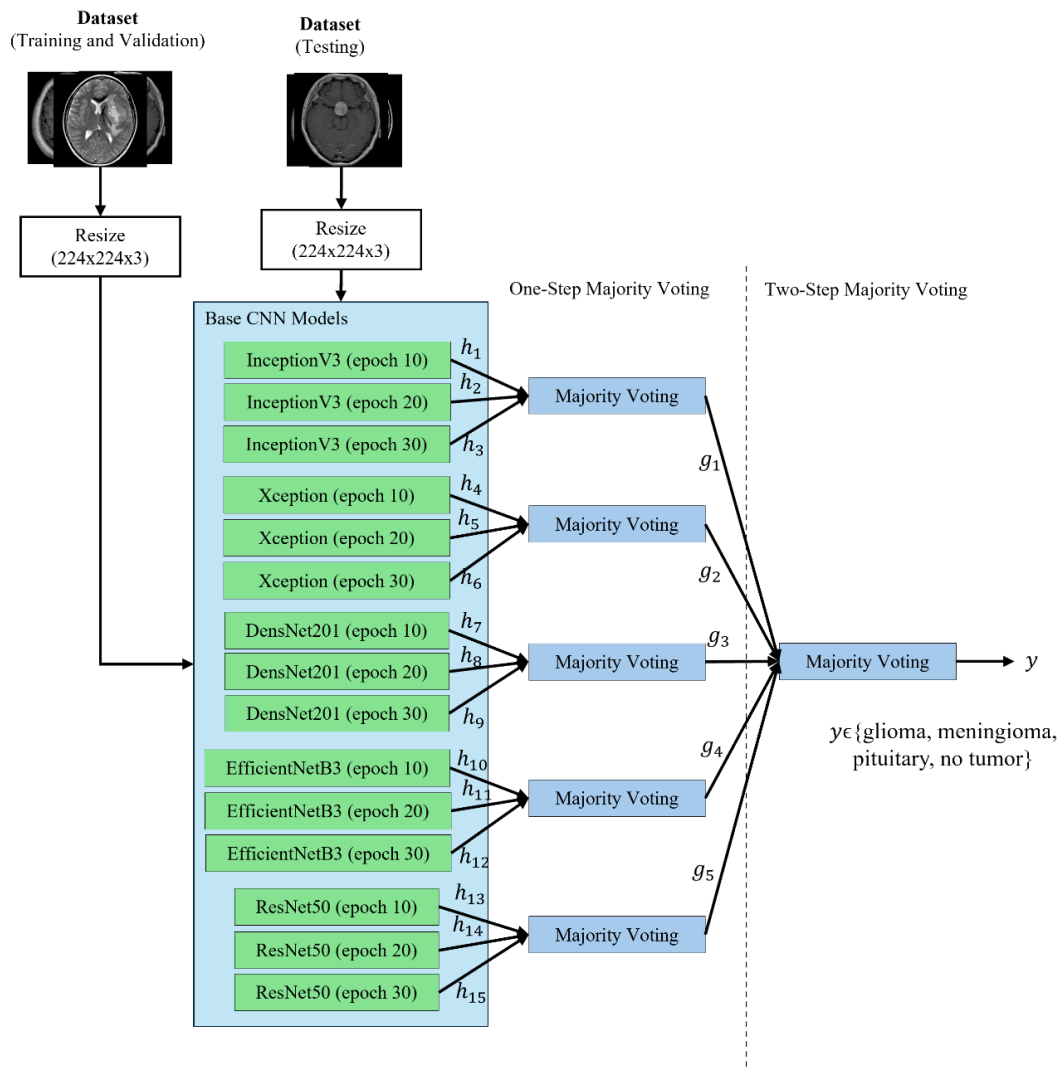


Figure 1. The proposed scheme: tumor classification using two-step majority voting of CNNs

2.1. Dataset acquisition

Dataset acquisition involves acquiring MRI images of brain tumors (glioma, meningioma, and pituitary) and no tumors for training and testing. The process also involves splitting the dataset for training into datasets for learning and validation. In this study, brain MRI images (axial, sagittal, and coronal planes) for testing were obtained from the Nickparvar brain tumor MRI dataset [24]. The dataset was obtained from the Kaggle website, which contains a combination of the Sartaj [25], BR35H [26], and Figshare [27] datasets.

The dataset consists of datasets for training and testing to evaluate the classification method. The total number of brain MRI images in the training dataset was 5,712, including 1,321 glioma images, 1,339 meningioma images, 1,457 pituitary images, and 1,595 no-tumor images. Meanwhile, the total number of brain MRI images for testing is 1,311, including 300 glioma images, 306 meningioma images, 300 pituitary images, and 405 no-tumor images. Figure 2 shows glioma, meningioma, pituitary, and no-tumor brain MRI images. To evaluate each base CNN model, 10% (572 brain MRI images) were randomly taken from the original training dataset for validation. Thus, there were 572 MRI images for validation and the remaining 5,140 MRI images for the learning process of each base CNN model. Table 1 shows the composition of the dataset used for the training, testing, and validation processes.

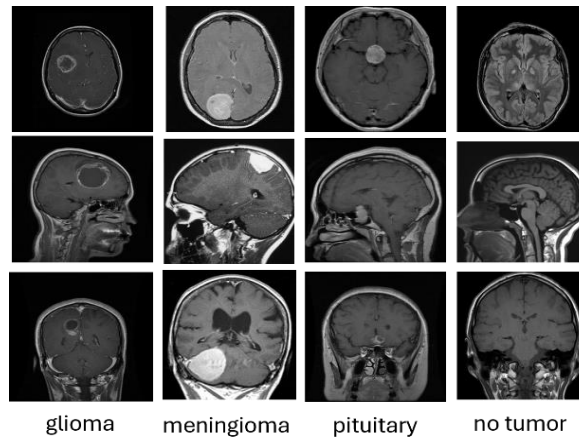


Figure 2. Examples of glioma, meningioma, pituitary, and no tumor in MRI images

Table 1. The composition of tumor (glioma, meningioma, pituitary) and no tumor for evaluation on the Nickparvar dataset

Label	Training	Validation	Testing	MRI Plane
glioma	1,189	132	300	Axial, Sagittal, and Coronal
meningioma	1,205	134	306	Axial, Sagittal, and Coronal
pituitary	1,311	146	300	Axial, Sagittal, and Coronal
no tumor	1,435	160	405	Axial, Sagittal, and Coronal
Total	5,140	572	1311	

2.2. Pre-processing of brain MRI image

In this study, the pre-processing stage is used to resize each MRI brain image to the same size for training, validation, and testing. In the proposed scheme in Figure 1, all MRI images involved as input to the base CNN models are sized $224 \times 224 \times 3$. Each MRI brain image is an RGB image with three channels, namely red, green, and blue, with each channel measuring 224×224 pixels. The input shape of the MRI brain image is the recommendation of each base CNN model, as shown in Table 2. Furthermore, all resized brain MRI images in the training and validation datasets can be forwarded to the learning process. The same thing applies to the testing dataset; after being resized, each MRI brain testing image can be entered into the classification process.

2.3. Training of base CNN models

This section covers the learning and hyperparameters of each base CNN model for ensemble combination in the next stage. The base CNN models in this study include Xception, InceptionV3, DensNet201, EfficientNetB3, and ResNet50. Before learning, each CNN model on the top layer needs customization to classify four classes: glioma, meningioma, pituitary, and no tumor. To obtain good learning results, the learning process of each base model requires hyperparameter tuning. Table 2 provides an overview of the hyperparameter settings for each base model, including input shape, customization on the top layer, epoch, batch size, loss function, optimizer, learning rate, and model weight initialization. In training and testing, the input shape of each base model is $224 \times 224 \times 3$. The input shape is the recommendation of the model in general. Adjustments to the top layer for tumor classification involve all base CNN models. The

InceptionV3, DensNet201, and EfficientNetB3 models use global average pooling and dense size 4 (many classes/labels) with a SoftMax activation function on the top layer. Meanwhile, adjusting the Xception model on the top layer also involves global average pooling, but the output layer uses a sigmoid activation function [28]. The use of the model's global average pooling and activation function has been proven to be robust in practice and is recommended by the model [29]–[31]. In ResNet50, the top layer of the model involves average pooling and flatten layers with a SoftMax activation function on the output layer. The use of the activation function is also recommended by the CNN model in [9] and [32].

Table 2. Hyperparameter tuning of base CNN models

Hyperparameter	Base CNN models				
	InceptionV3	Xception	DensNet201	EfficientNetB3	ResNet50
Input shape	224x224x3	224x224x3	224x224x3	224x224x3	224x224x3
Top layer	Global Average Pooling Dense (4, activation='SoftMax')	Global Average Pooling Dense (4, activation='sigmoid')	Global Average Pooling Dense (4, activation='SoftMax')	Global Average Pooling Dense (4, activation='SoftMax')	Average Pooling Flatten Dense (4, activation='SoftMax')
Epoch	10, 20, 30	10, 20, 30	10, 20, 30	10, 20, 30	10, 20, 30
Batch size	16	16	16	16	16
Learning rate	0.001	0.001	0.001	0.001	0.001
Loss function	Categorical Crossentropy	Categorical Crossentropy	Categorical Crossentropy	Categorical Crossentropy	Categorical Crossentropy
Optimizer	Adamax	Adamax	Adamax	Adamax	Adamax
Weight initialization	ImageNet	ImageNet	ImageNet	ImageNet	ImageNet

Each base CNN model in this study involved learning at epochs 10, 20, and 30. Learning at different epochs accommodates the possibility of varying classification results in each model. Each base CNN model's batch size and learning rate are 16 and 0.001. The learning involving batch size and learning rate sizes has shown the best performance results [17], [18], [33]. The optimizer engaged in the learning process is Adamax, which shows more stable results [34], [35]. This study's classification of brain tumors is multi-class, hence, the loss function used in the learning is categorical cross-entropy [30], [36]. In addition, the learning of each CNN model applies transfer learning. In this study, the initial weights were taken from the results of learning base CNN models using the ImageNet dataset [37], thus helping the learning process of CNN models on small datasets [38].

InceptionV3 is a CNN architecture from the Inception family with several improvements [19]. The model architecture has a steam block and inception block (A, B, and C), reduction block (A and B), and auxiliary classifier block. In this study, the InceptionV3 learning process involved three learning epochs, namely 10, 20, and 30. As a result, there are 3 InceptionV3 classification results, namely at epoch 10 (h_1), epoch 20 (h_2), and epoch 30 (h_3) in Figure 1.

Xception is a CNN architecture that continues the Inception architecture [20]. Xception has 36 convolutional layers divided into 14 modules. The architecture has a convolution operation (filter size 1×1) to obtain cross-channel correlation as a $2D + 1D$ mapping. Xception in this study also involved three learning epochs, namely 10, 20, and 30. The classification results using Xception at epoch ten are shown with h_4 , epoch 20 with h_5 , and epoch 30 with h_6 in Figure 1.

DensNet201 is a CNN architecture with each layer to each next layer connected using feedforward [21]. Thus, the architecture has direct connections of $(L \times (L+1))/2$, with each layer producing a feature map as input to the next layer. The DenseNet-201 architecture has four dense blocks, each with 6, 12, 24, and 16 convolution blocks. In this study, DenseNet-201 also applied three learning epochs, namely 10, 20, and 30, with the classification results of each epoch indicated with h_7 , h_8 , and h_9 (see Figure 1).

EfficientNetB3 is one of the CNN architectures part of the EfficientNet variant that scales all dimensions uniformly using pooling coefficients [22]. The CNN architecture has two convolution layers and seven mobile bottleneck convolution (MBConv) layer blocks. Each layer in the MBConv block, except the first layer, involves an inverted residual connection. EfficientNetB3 also applied three epochs in learning with the classification results shown by h_{10} at epoch 10, h_{11} at epoch 20, and h_{12} at epoch 30 in Figure 1.

ResNet50 is a CNN architecture that uses residual connections so that the network can learn a series of residual functions that map inputs to the desired outputs [23]. The architecture has a convolutional layer, an identity block, and a convolutional block. The convolutional layer is implemented to extract features, while the identity and convolutional blocks transform features. ResNet50 also involved three epochs in learning from which classification results are shown as h_{13} in epoch 10, h_{14} in epoch 20, and h_{15} in epoch 30 in Figure 1

2.4. Two-step majority voting

Majority voting (MV) in implementation has an impact on improving classification or detection performance [9], [17], [18]. Two-step majority voting is a proposed method that involves two steps of the majority voting process. The first step, MV (one-step MV), is performed on the classification results of the same CNN model with different learning epochs. (1), (2), (3), (4), and (5) are mathematical equations of the first step MV for the InceptionV3, Xception, DensNet201, EfficientNetB3, and ResNet50 models. The first step, MV, aims to strengthen the classification results of each CNN model. $g_1, g_2, g_3, g_4,$ and g_5 in the equation are, respectively, the voting results for the classification results of InceptionV3, Xception, DensNet201, EfficientNetB3, and ResNet50 at epochs 10, 20, and 30. Meanwhile, mode is a function that obtains the majority vote for the classification results of each CNN model.

$$g_1 = \text{mode}(h_1, h_2, h_3) \quad (1)$$

$$g_2 = \text{mode}(h_4, h_5, h_6) \quad (2)$$

$$g_3 = \text{mode}(h_7, h_8, h_9) \quad (3)$$

$$g_4 = \text{mode}(h_{10}, h_{11}, h_{12}) \quad (4)$$

$$g_5 = \text{mode}(h_{13}, h_{14}, h_{15}) \quad (5)$$

Each base CNN model's architecture can obtain different tumor classification results. The potential for different classification results is due to the other layers or convolution blocks built by each model. The involvement of several CNN models as base models with their respective advantages is a solution to classify tumor types with varied shapes, sizes, or positions. Consequently, the second step MV is applied to the results of the first step MV to obtain the final classification results that consider all the classification results of base CNN models. Mathematically, the two-step MV can be written as (6).

$$y = \text{mode}(g_1, g_2, g_3, g_4, g_5) \quad (6)$$

The steps in Algorithm 3 are used to obtain the final classification results with the proposed method (two-step MV). The algorithm's input is all base CNN models that have undergone a learning process based on parameter in Table 2. In the algorithm, the input of the base CNN models includes M_{11}, M_{12}, M_{13} which show the learning results of InceptionV3, M_{21}, M_{22}, M_{23} are the learning results of Xception, M_{31}, M_{32}, M_{33} are the learning results of DensNet201, M_{41}, M_{42}, M_{43} are the learning results of EfficientNetB3, and M_{51}, M_{52}, M_{53} are the learning results of ResNet50. These learning results are successively carried out at epochs 10, 20, and 30 for each model. The other algorithm input is X , which shows the brain MRI image of the testing dataset. The first step of the algorithm is the pre-processing stage of the image to obtain the same image size as the training image size, which is $224 \times 224 \times 3$. The next step is the first classification of tumors using each base CNN model and epoch with the input image (Z). In testing the two-step MV method, three classifications of brain tumors are carried out in stages. The first classification result is shown by h_p , which is determined by the argmax operation on the softmax value of each base CNN model, with $p = 1, 2, \dots, 15$. The h_p value is a class label (k), $k=0$ if classified as a glioma tumor, 1 for meningioma, 2 for no tumor, and 3 for pituitary. h_p includes $h_1, h_2,$ and h_3 which are the first classification results of InceptionV3. $h_4, h_5,$ and h_6 are the first classification results of Xception. $h_7, h_8,$ and h_9 are the first classification results of DensNet201. $h_{10}, h_{11},$ and h_{12} are the first classification results of EfficientNetB3. $h_{13}, h_{14},$ and h_{15} are the first classification results of ResNet50. All classification results of each model are respectively at epochs 10, 20, and 30. The initial classification results at each epoch and the CNN model are then forwarded to the second classification stage using the first MV (one-step MV) with the results indicated by g_i with $i = 1, 2, \dots, 5$. g_i includes $g_1, g_2, g_3, g_4,$ and g_5 , which are the second classification results with one-step MV using (1), (2), (3), (4), and (5). The results of the second classification are then forwarded to the final (third) classification stage with the second MV (two-step MV) using (6), whose results are indicated by y . From the results of the third classification, it can then be decided whether the MRI image of the testing brain is classified as glioma, meningioma, pituitary, or no tumor.

Algorithm 3, in its implementation, involves the classification results of base CNN models shown in Algorithm 1 and one-step MV in Algorithm 2. The comparison of the three algorithms can be seen from the classification results. Algorithm 3 is a continuation of Algorithm 2 results, and Algorithm 2 is a continuation of Algorithm 1 results. Algorithm 1 contains the steps for testing MRI image classification using the training

results of each CNN model in the base models at each epoch. At the same time, the hyperparameter tuning in Table 2 is the treatment of CNN model parameters intended for the learning process to provide good model training results.

Algorithm 1. Base CNN models classification

Input: M_{11}, M_{12}, M_{13} {InceptionV3}, M_{21}, M_{22}, M_{23} {Xception}, M_{31}, M_{32}, M_{33} {DensNet201}, M_{41}, M_{42}, M_{43} {EfficientNetB3}, M_{51}, M_{51}, M_{51} {ResNet50} {The training results of each model are consecutively at epochs 10, 20, and 30, respectively}, X {Brain MRI image of testing}
Output: h {Classification of each CNN model and epoch include h_1, h_2, \dots, h_{15} }

```

1   $Z \leftarrow \text{resize}(X, 224, 224, 3)$  {pre-processing}
2   $p \leftarrow 0$ 
3  for  $i \leftarrow 1$  to 5 {number of models}
4    for  $i \leftarrow 1$  to 3 {number of epochs {10,20,30}}
5       $p \leftarrow p+1$  {index of classification}
6       $h_p \leftarrow \text{argmax}_{0 \leq k \leq 3} (M_{ij}(Z))$  {classification of each base CNN model}
7  return  $h$ 

```

Algorithm 2. One-step majority voting

Input: M_{11}, M_{12}, M_{13} {InceptionV3}, M_{21}, M_{22}, M_{23} {Xception}, M_{31}, M_{32}, M_{33} {DensNet201}, M_{41}, M_{42}, M_{43} {EfficientNetB3}, M_{51}, M_{51}, M_{51} {ResNet50} {The training results of each model are consecutively at epochs 10, 20, and 30, respectively}, X {Brain MRI image of testing}
Output: g {Classification of one-step majority voting for each CNN model: g_1, g_2, \dots, g_5 }

```

1   $Z \leftarrow \text{resize}(X, 224, 224, 3)$  {pre-processing}
2   $p \leftarrow 0$ 
3  for  $i \leftarrow 1$  to 5 {number of models}
4    for  $i \leftarrow 1$  to 3 {number of epochs={10,20,30}}
5       $p \leftarrow p+1$  {index of classification}
6       $h_p \leftarrow \text{argmax}_{0 \leq k \leq 3} (M_{ij}(Z))$  {classification of each base CNN model}
7  for  $i \leftarrow 1$  to 5 {number of models}
8     $r \leftarrow 1+3(i-1)$ 
9     $g_i \leftarrow \text{mode}(h_r, h_{r+1}, h_{r+2})$  {the first step of majority voting}
10 return  $g$ 

```

Algorithm 3. Two-step majority voting

Input: M_{11}, M_{12}, M_{13} {InceptionV3}, M_{21}, M_{22}, M_{23} {Xception}, M_{31}, M_{32}, M_{33} {DensNet201}, M_{41}, M_{42}, M_{43} {EfficientNetB3}, M_{51}, M_{51}, M_{51} {ResNet50} {the training results of each model are consecutively at epochs 10, 20, and 30, respectively}, X {Brain MRI image of testing}

Output: y {Classification result }

```

1   $Z \leftarrow \text{resize}(X, 224, 224, 3)$  {pre-processing}
2   $p \leftarrow 0$ 
3  for  $i \leftarrow 1$  to 5 {number of models}
4    for  $i \leftarrow 1$  to 3 {number of epochs={10,20,30}}
5       $p \leftarrow p+1$  {index of classification}
6       $h_p \leftarrow \text{argmax}_{0 \leq k \leq 3} (M_{ij}(Z))$  {classification of base CNN model,  $k=0$ (glioma),  $k=1$ (meningioma),  $k=2$ (no tumor),  $k=3$ (pituitary)}
7  for  $i \leftarrow 1$  to 5 {number of models}
8     $r \leftarrow 1+3(i-1)$  {index of classification}
9     $g_i \leftarrow \text{mode}(h_r, h_{r+1}, h_{r+2})$  {the first step of majority voting}
10  $y \leftarrow \text{mode}(g)$  {the two step of majority voting}
11 return  $y$ 

```

2.5. Performance evaluation of methods

In this study, to evaluate the method's performance in brain tumor classification, several indicator measures were used, they were accuracy (ac), precision (pr), sensitivity (se), specificity (sp), and F-score (fs) [39]. All of these indicators are obtained based on the values of true positive (trp), false negative (fln), true negative (trn), and false positive (flp). These performance indicators are defined for each dataset label, namely glioma, meningioma, pituitary, and no tumor. For glioma, trp is the number of times the glioma brain MRI image, based on the classification results, is labeled as glioma, fln is the number of times the glioma brain MRI image, based on the classification results, is labeled as other than glioma, trn is the number of times the brain MRI image other than glioma is labeled as other than glioma, and flp is the number of times the brain MRI image other than glioma is labeled as glioma in the same way. Meanwhile, ac , pr , se , sp , and fs are determined for each label and are respectively defined in (7), (8), (9), (10), and (11).

$$ac = (trp + trn) / (trp + flp + trn + fln) \quad (7)$$

$$pr = trp/(flp + trp) \quad (8)$$

$$se = trp/(fln + trp) \quad (9)$$

$$sp = trn/(flp + trn) \quad (10)$$

$$fs = 2(pr)(se)/(pr + se) \quad (11)$$

3. RESULTS AND DISCUSSION

The implementation of all the methods used Google Colab in the learning process of each base CNN model, the performance testing of base CNN models, and the performance testing of the proposed method (two-step MV) in brain tumor classification. Table 3 shows the results of testing the performance of base CNN models and two-step MV in brain tumor classification on the nickparvar dataset. The results of testing the base CNN models on the testing dataset for the InceptionV3 model produced a classification accuracy of 98.78% at epoch, 99.24% at epoch 20, and 99.08% at epoch 30. For Xception, it obtained a classification accuracy of 99.24% at epoch 10, 99.47% at epoch 20, and 99.39% at epoch 30. DensNet201 obtained a tumor classification accuracy of 99.16% at epoch 10, 99.36% at epoch 20, and 99.39% at epoch 30. On the other hand, EfficientNetB3, at epochs 10, 20, and 30, produced a classification accuracy of 99.47%, 99.08%, and 99.31%. Meanwhile, ResNet50 at epochs 10, 20, and 30 obtained classification accuracy of 98.25%, 98.86%, and 98.70%. From these results, Xception at epoch 20 and EfficientNetB3 at epoch 10 yielded the highest accuracy among the base CNN models and the best average sensitivity, precision, specificity, and F-score compared to other models. The lowest accuracy of the base CNN model is ResNet50 at epoch 10, as well as the lowest average sensitivity, precision, specificity, and F-Score among the other base CNN models.

Table 3. Performance of base CNN models and two-step majority voting

Model	Accuracy	Average performance				
		Precision	Sensitivity	Specificity	F-score	
Base CNN models (epoch)	(1) InceptionV3 (10)	0.9878	0.9868	0.9868	0.9960	0.9868
	(2) InceptionV3 (20)	0.9924	0.9918	0.9917	0.9975	0.9917
	(3) InceptionV3 (30)	0.9908	0.9902	0.9907	0.9970	0.9904
	(4) Xception (10)	0.9924	0.9920	0.9917	0.9975	0.9919
	(5) Xception (20)	0.9947	0.9943	0.9942	0.9983	0.9942
	(6) Xception (30)	0.9939	0.9935	0.9938	0.9980	0.9936
	(7) DensNet201 (10)	0.9916	0.9910	0.9913	0.9973	0.9912
	(8) DensNet201 (20)	0.9939	0.9936	0.9934	0.9980	0.9935
	(9) DensNet201 (30)	0.9939	0.9934	0.9934	0.9980	0.9934
	(10) EfficientNetB3 (10)	0.9947	0.9942	0.9942	0.9983	0.9942
	(11) EfficientNetB3 (20)	0.9908	0.9906	0.9905	0.9970	0.9905
	(12) EfficientNetB3 (30)	0.9931	0.9930	0.9926	0.9977	0.9928
	(13) ResNet50 (10)	0.9825	0.9817	0.9809	0.9943	0.9811
	(14) ResNet50 (20)	0.9886	0.9880	0.9876	0.9962	0.9878
	(15) ResNet50 (30)	0.9870	0.9864	0.9859	0.9958	0.9861
One-step MV	(16) MV (1,2,3)	0.9947	0.9943	0.9942	0.9983	0.9942
	(17) MV (4,5,6)	0.9939	0.9935	0.9938	0.9980	0.9936
	(18) MV (7,8,9)	0.9954	0.9951	0.9950	0.9985	0.9950
	(19) MV (10,11,12)	0.9962	0.9959	0.9958	0.9988	0.9959
	(20) MV (13,14,15)	0.9886	0.9879	0.9876	0.9962	0.9877
Two-step MV	(21) MV (16,17,18,19,20)	0.9969	0.9967	0.9967	0.9990	0.9967

The test results for the first step MV on each CNN model with different epochs, the majority voting on the classification results of the EfficientNetB3 model (MV (10,11,12)), gave an accuracy of 99.62% and were the best among the first step MVs on other models. The lowest MV accuracy among the other models was the MV on the ResNet50 classification results at different epochs (MV (13,14,15)), with an accuracy of 98.86%. Meanwhile, the second step MV (two-step MV) gave better classification performance than the first step MV and on all base CNN models. From these results, two-step MV (MV (16,17,18,19,20)) increased classification accuracy by 0.22% to 1.44% on base CNN models and 0.07% to 0.83% majority voting on the first step. Two-step MV improves classification performance on base CNN models through two steps of MV. The first step of MV will strengthen the classification performance of each base CNN model. The MV classification results of each CNN model implemented by learning with different hyperparameters and epochs can reduce the uncertainty (coincidence factor) of the classification results of each model. Two-step

MV applied to one-step MV will strengthen the classification of specific tumor types and leave the weak CNN model in the classification of that tumor type. Table 4 shows the classification performance of the proposed method against the base CNN model and the best one-step MV in more depth. One-step MV yielded the best performance to the EfficientNetB3 model at different epochs, and the two-step MV improved the precision of glioma and meningioma classification for the one-step MV by 0.01% and 0.32%, respectively. The two-step MV also provided improvements in the sensitivity of glioma classification by 0.33%, improvements in the specificity of meningioma classification by 0.10%, and automatically provided improvements in the F-score. In addition, two-step MV also improved performance against the best base CNN model (Xception at epoch 20 and EfficientNetB3 at epoch 10), namely glioma classification precision of 0.34% to 0.67%, meningioma classification precision of 0.33% to 0.65%, glioma classification sensitivity of 0.33% to 0.66%, meningioma classification sensitivity of 0.32% to 0.65%, glioma classification specificity of 0.1% to 0.2%, meningioma classification specificity of 0.10% to 0.2%, glioma classification F-score of 0.5%, and meningioma classification F-score of 0.49%.

Table 4. Deep performance of the best base CNN models, one-step MV, and two-step MV

Model	Label	Average Performance				
		Precision	Sensitivity	Specificity	F-Score	
Base CNN models (epoch)	Xception (20)	glioma	0.9933	0.9867	0.9980	0.9900
		meningioma	0.9838	0.9935	0.9950	0.9886
		no tumor	1.0000	1.0000	1.0000	1.0000
		pituitary	1.0000	0.9967	1.0000	0.9983
		glioma	0.9900	0.9900	0.9970	0.9900
	EfficientNetB3 (10)	meningioma	0.9870	0.9902	0.9960	0.9886
		no tumor	1.0000	1.0000	1.0000	1.0000
		pituitary	1.0000	0.99.67	1.0000	0.9983
		glioma	0.9966	0.9900	0.9990	0.9933
		meningioma	0.9871	0.9967	0.9960	0.9919
One-step MV	MV (10,11,12)	no tumor	1.0000	1.0000	1.0000	1.0000
		pituitary	1.0000	0.9967	1.0000	0.99.83
		glioma	0.99.67	0.9933	0.9990	0.9950
Two-step MV	MV (16,17,18,19,20)	meningioma	0.99.03	0.9967	0.9970	0.9935
		no tumor	1.0000	1.0000	1.0000	1.0000
		pituitary	1.0000	0.9967	1.0000	0.9983
		glioma	1.0000	0.9967	1.0000	0.9983

4. COMPARISON WITH EXISTING METHODS

The comparative results of evaluating the proposed method with several previous methods on the same testing dataset are investigation evidence of the methods in tumor classification. Applying two-step majority voting to the base CNN models (the two-step MV) produces better performance than the existing methods. Table 5 shows that the proposed method is better than the existing methods, with a difference of 1.85% to 7.67%. The involvement of several base CNN models in two-step MV strengthens tumor classification by abandoning weak CNN models. The proposed method can improve the classification performance of brain tumor types, especially glioma and meningioma tumors that are almost the same shape and size.

Table 5. Performance comparison of two-step majority voting with the existing methods

References	Methods	Accuracy
Rasheed <i>et al.</i> [40]	Image enhancement and CNN	0.9784
Shilaskar <i>et al.</i> [41]	HOG and XG Boost	0.9202
Atha and Chaki [42]	SSBTCNet	0.965
Proposed Methods	Two-step majority voting	0.9969

5. CONCLUSION

Two-step majority voting is the proposed method in this study that applies two-step majority voting to the results of the classification of base CNN models based on brain MRI images. The first step is majority voting on the results of tumor classification on each base CNN model with different epochs (10, 20, and 30 epochs) (one-step majority voting). The base CNN models included InceptionV3, Xception, DensNet201, EfficientNetB3, and ResNet50. The second step is majority voting on the results of all the majority voting in the first step. The test used the Nickparvar dataset. Our proposed method produced accuracy, sensitivity, precision, specificity, and F-score in tumor classification of 99.69%, 99.67%, 99.67%, 99.90%, and 99.67%, respectively. Two-step majority voting increased accuracy over base CNN models and the one-step majority

voting. The detailed performance improvement occurred in classifying glioma and meningioma tumors, which were almost similar in shape and size. In addition, the proposed method was better than several existing methods in the testing using the same dataset. Therefore, our proposed method can help neurologists in tumor classification. Additionally, for clinical implementation, there is a potential for improving the performance of tumor classification based on brain MRI images by adding training or testing MRI image data.

FUNDING INFORMATION

This study is entirely funded by Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia, through the National Development Applied Research (Penelitian Terapan Pengembangan Nasional) scheme with contract number 1143A/LP2M/TL.00/02/2024.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Irwan Budi Santoso	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓
Shoffin Nahwa Utama		✓			✓			✓		✓	✓		✓	
Supriyono			✓	✓	✓	✓	✓			✓	✓	✓	✓	

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY




The data that support the findings of this study are openly available at <https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset>, reference number [24].

REFERENCES




- [1] S. Pereira, A. Pinto, V. Alves, and C. A. Silva, "Brain tumor segmentation using convolutional neural networks in MRI images," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1240–1251, May 2016, doi: 10.1109/TMI.2016.2538465.
- [2] M. S. I. Khan *et al.*, "Accurate brain tumor detection using deep convolutional neural network," *Computational and Structural Biotechnology Journal*, vol. 20, pp. 4733–4745, 2022, doi: 10.1016/j.csbj.2022.08.039.
- [3] A. R. Loughan *et al.*, "Death-related distress in adult primary brain tumor patients," *Neuro-Oncology Practice*, vol. 7, no. 5, pp. 498–506, 2020, doi: 10.1093/nop/npaa015.
- [4] P. Y. Wen *et al.*, "Updated response assessment criteria for high-grade gliomas: Response assessment in neuro-oncology working group," *Journal of Clinical Oncology*, vol. 28, no. 11, pp. 1963–1972, 2010, doi: 10.1200/JCO.2009.26.3541.
- [5] E. A. S. El-Dahshan, H. M. Mohsen, K. Revett, and A. B. M. Salem, "Computer-aided diagnosis of human brain tumor through MRI: A survey and a new algorithm," *Expert Systems with Applications*, vol. 41, no. 11, pp. 5526–5545, 2014, doi: 10.1016/j.eswa.2014.01.021.
- [6] J. Kang, Z. Ullah, and J. Gwak, "Mri-based brain tumor classification using ensemble of deep features and machine learning classifiers," *Sensors*, vol. 21, no. 6, pp. 1–21, 2021, doi: 10.3390/s21062222.
- [7] S. Ahmad and P. K. Choudhury, "On the performance of deep transfer learning networks for brain tumor detection using MR images," *IEEE Access*, vol. 10, pp. 59099–59114, 2022, doi: 10.1109/ACCESS.2022.3179376.
- [8] S. Shanthi, S. Saradha, J. A. Smitha, N. Prasath, and H. Anandakumar, "An efficient automatic brain tumor classification using optimized hybrid deep neural network," *International Journal of Intelligent Networks*, vol. 3, pp. 188–196, 2022, doi: 10.1016/j.ijin.2022.11.003.
- [9] I. B. Santoso, Y. Adrianto, A. Sensusiaty, D. Wulandari, and I. Purnama, "Epileptic EEG Signal Classification Using Convolutional Neural Network Based on Multi-Segment of EEG Signal," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 3, pp. 160–176, Jun. 2021, doi: 10.22266/ijies2021.0630.15.
- [10] M. Rizwan, A. Shabbir, A. R. Javed, M. Shabbir, T. Baker, and D. Al-Jumeily Obe, "Brain tumor and glioma grade classification using Gaussian convolutional neural network," *IEEE Access*, vol. 10, pp. 29731–29740, 2022, doi: 10.1109/ACCESS.2022.3153108.

- [11] A. S. Musallam, A. S. Sherif, and M. K. Hussein, "A new convolutional neural network architecture for automatic detection of brain tumors in magnetic resonance imaging images," *IEEE Access*, vol. 10, pp. 2775–2782, 2022, doi: 10.1109/ACCESS.2022.3140289.
- [12] T. A. Jenimma and Y. Jacob Vetharaj, "A survey on brain tumor segmentation and classification," *International Journal of Software Innovation*, vol. 10, no. 1, 2022, doi: 10.4018/IJSI.309721.
- [13] S. Chatterjee, F. A. Nizamani, A. Nürnberger, and O. Speck, "Classification of brain tumours in MR images using deep spatiotemporal models," *Scientific Reports*, vol. 12, no. 1, pp. 1–11, 2022, doi: 10.1038/s41598-022-05572-6.
- [14] A. Younis, L. Qiang, C. O. Nyatega, M. J. Adamu, and H. B. Kawuwa, "Brain tumor analysis using deep learning and VGG-16 ensembling learning approaches," *Applied Sciences (Switzerland)*, vol. 12, no. 14, 2022, doi: 10.3390/app12147282.
- [15] N. Noreen, S. Palaniappan, A. Qayyum, I. Ahmad, M. Imran, and M. Shoaib, "A deep learning model based on concatenation approach for the diagnosis of brain tumor," *IEEE Access*, vol. 8, pp. 55135–55144, 2020, doi: 10.1109/ACCESS.2020.2978629.
- [16] S. Asif, W. Yi, Q. U. Ain, J. Hou, T. Yi, and J. Si, "Improving effectiveness of different deep transfer learning-based models for detecting brain tumors from MR images," *IEEE Access*, vol. 10, pp. 34716–34730, 2022, doi: 10.1109/ACCESS.2022.3153306.
- [17] I. B. Santoso, Y. Adrianto, A. D. Sensusiaty, D. P. Wulandari, and I. K. E. Purnama, "Ensemble convolutional neural networks with support vector machine for epilepsy classification based on multi-sequence of magnetic resonance images," *IEEE Access*, vol. 10, pp. 32034–32048, 2022, doi: 10.1109/ACCESS.2022.3159923.
- [18] I. B. Santoso, S. N. Utama, and Supriyono, "A new voting of convolutional neural networks for brain tumor detection based on MRI images," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 1, pp. 212–227, 2024, doi: 10.22266/ijies2024.0229.21.
- [19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
- [20] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *Prepr. arXiv.1610.02357*, Oct. 2016.
- [21] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, vol. 2017-Janua, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.
- [22] M. Tan and Q. V. Le, "EfficientNet: rethinking model scaling for convolutional neural networks," in *36th International Conference on Machine Learning, ICML 2019*, 2019, vol. 2019-June, pp. 10691–10700.
- [23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Jun. 2016, doi: 10.1109/cvpr.2016.90.
- [24] M. Nickparvar, "Brain tumor MRI dataset," *Kaggle*, 2021. <https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset> (accessed Jul. 24, 2023).
- [25] S. Bhuvaji, A. Kadam, P. Bhumkar, S. Dedge, and S. Kanchan, "Brain tumor classification (MRI)," 2020. <https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri> (accessed Jul. 24, 2023).
- [26] A. Hamada, "Br35H: brain tumor detection 2020," *Kaggle*, 2020. <https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection> (accessed Jul. 24, 2023).
- [27] J. Cheng, "Brain tumor dataset figshare," *Kaggle*, 2017. https://figshare.com/articles/dataset/brain_tumor_dataset/1512427 (accessed Jul. 24, 2023).
- [28] Mp. Lakshmi, "Image classification using machine learning techniques," *International Journal of Advanced Research*, vol. 7, no. 5, pp. 1238–1245, 2019, doi: 10.21474/ijar01/9157.
- [29] L. Min, C. Qiang, and Y. Shuicheng, "Network in network," in *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, Dec. 2014, pp. 1–10.
- [30] H. Alhichri, A. S. Alswayed, Y. Bazi, N. Ammour, and N. A. Alajlan, "Classification of remote sensing images using EfficientNet-B3 CNN model with attention," *IEEE Access*, vol. 9, pp. 14078–14094, 2021, doi: 10.1109/ACCESS.2021.3051085.
- [31] P. P. Malla, S. Sahu, and A. I. Alutaibi, "Classification of tumor in brain MR images using deep convolutional neural network and global average pooling," *Processes*, vol. 11, no. 3, pp. 1–17, 2023, doi: 10.3390/pr11030679.
- [32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.
- [33] S. Taheri, Z. Golrizkhatami, A. A. Basabrain, and M. S. Hazzazi, "A comprehensive study on classification of breast cancer histopathological images: binary versus multi-category and magnification-specific versus magnification-independent," *IEEE Access*, vol. 12, pp. 50431–50443, 2024, doi: 10.1109/ACCESS.2024.3386355.
- [34] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.
- [35] B. Xiao, Y. Liu, and B. Xiao, "Accurate state-of-charge estimation approach for lithium-ion batteries by gated recurrent unit with ensemble optimizer," *IEEE Access*, vol. 7, pp. 54192–54202, 2019, doi: 10.1109/ACCESS.2019.2913078.
- [36] D. Reyes and J. Sánchez, "Performance of convolutional neural networks for the classification of brain tumors using magnetic resonance imaging," *Heliyon*, vol. 10, no. 3, p. e25468, 2024, doi: 10.1016/j.heliyon.2024.e25468.
- [37] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.
- [38] A. Patil and M. Rane, "Convolutional neural networks: an overview and its applications in pattern recognition," *Smart Innovation, Systems and Technologies*, vol. 195, pp. 21–30, 2021, doi: 10.1007/978-981-15-7078-0_3.
- [39] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 569–575, Mar. 2010, doi: 10.1109/TPAMI.2009.187.
- [40] Z. Rasheed *et al.*, "Brain tumor classification from MRI using image enhancement and convolutional neural network techniques," *Brain Sciences*, vol. 13, no. 9, 2023, doi: 10.3390/brainsci13091320.
- [41] S. Shilaskar, T. Mahajan, S. Bhatlawande, S. Chaudhari, R. Mahajan, and K. Junnare, "Machine learning based brain tumor detection and classification using HOG feature descriptor," in *In 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, Jun. 2023, pp. 67–75, doi: 10.1109/ICSCSS57650.2023.10169700.
- [42] Z. Atha and J. Chaki, "SSBTCNet: Semi-supervised brain tumor classification network," *IEEE Access*, vol. 11, pp. 141485–141499, 2023, doi: 10.1109/ACCESS.2023.3343126.




BIOGRAPHIES OF AUTHORS

Irwan Budi Santoso    received their bachelor's degree from the Department of Statistics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 2000, their master's degree from the Department of Informatics Engineering, Institut Teknologi Sepuluh Nopember, in 2007, their doctoral degree from the Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Indonesia, in 2022. He is also a lecturer in informatics engineering at Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia. His research interests include artificial intelligence, machine learning, deep learning, and medical informatics. He can be contacted at email: irwan@ti.uin-malang.ac.id.



Shoffin Nahwa Utama    is a lecturer in Department of Informatics Engineering at UIN Maulana Malik Ibrahim Malang since 2021. He earned a bachelor's in informatics engineering from UIN Maulana Malik Ibrahim, Malang, Indonesia, in 2009. and complete their master's degree in electrical engineering from Universitas Brawijaya, Indonesia, in 2013. His research interests focus primarily on the internet of things, artificial intelligence, and machine learning, contributing significantly to audio signal processing and artificial intelligence. He can be contacted at email: shoffin@uin-malang.ac.id.



Supriyono    has 14 years of experience as a lecturer in informatics engineering at UIN Maulana Malik Ibrahim Malang, where he leverages his proficiency in English and adaptability to new environments. Currently pursuing a doctoral degree in electrical engineering and informatics at Universitas Negeri Malang, Indonesia. Supriyono holds a Master of Computer degree from Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, and a Bachelor of Computer Science degree from Universitas Brawijaya, Malang, Indonesia. His research interests encompass software engineering, natural language processing, artificial intelligence, machine learning, and deep learning. He can be contacted at email: priyono@ti.uin-malang.ac.id.