Available *online* at : http://jnte.ft.unand.ac.id/

Jurnal Nasional Teknik Elektro

| ISSN (Print) 2302-2949 | ISSN (Online) 2407-7267 |

A Transfer Learning-Based Model for Brain Tumor Detection in MRI Images

Faiz Rofi Hencya [1,2], Satria Mandala [1,2], Tong Boon Tang [3], Mohd Soperi Mohd Zahid [4]

[1] *Department of Informatics, School of Computing, Telkom University, Bandung, Indonesia*
[2] *Human Centric (HUMIC) Engineering, School of Computing, Telkom University, Bandung, Indonesia*
[3] *Centre for Intelligent Signal and Imaging Research, Universiti Teknologi Petronas, 32610 Seri Iskandar, Perak, Malaysia*
[4] *Department of Computer and Information Sciences, Universiti Teknologi Petronas, 32610 Seri Iskandar, Perak, Malaysia*

ARTICLE INFORMATION

Received: June 27, 2023
Revised: July 25, 2023
Accepted: July 25, 2023
Available online: July 31, 2023

KEYWORDS

brain tumor, MRI, transfer learning, Xception, prototype

CORRESPONDENCE

E-mail: satriamandala@telkomuniversity.ac.id

A B S T R A C T

Brain tumors are life-threatening medical conditions characterized by abnormal cell proliferation in or near the brain. Early detection is crucial for successful treatment. However, the scarcity of labelled brain tumor datasets and the tendency of convolutional neural networks (CNNs) to overfit on small datasets have made it challenging to train accurate deep learning models for brain tumor detection. Transfer learning is a machine learning technique that allows a model trained on one task to be reused for a different task. This approach is effective in brain tumor detection as it allows CNNs to be trained on larger datasets and generalize better to new data. In this research, we propose a transfer learning approach using the Xception model to detect four types of brain tumors: meningioma, pituitary, glioma, and no tumor (healthy brain). The performance of our model was evaluated on two datasets, demonstrating a sensitivity of 98.07%, specificity of 97.83%, accuracy of 98.15%, precision of 98.07%, and f1-score of 98.07%. Additionally, we developed a user-friendly prototype application for easy access to the Xception model for brain tumor detection. The prototype was evaluated on a separate dataset, and the results showed a sensitivity of 95.30%, specificity of 96.07%, accuracy of 95.30%, precision of 95.31%, and f1-score of 95.27%. These results suggest that the Xception model is a promising approach for brain tumor detection. The prototype application provides a convenient and easy-to-use way for clinical practitioners and radiologists to access the model. We believe the model and prototype generated from this research will be valuable tools for diagnosing, quantifying, and monitoring brain tumors.

INTRODUCTION

In today's modern era, technological developments that are increasingly developing to become more efficient, more accessible, and more sophisticated have become something natural. The changing times significantly influence all areas of human life, including the health sector. Medical imaging is one of the most widely recognized use cases of machine learning that is most widely researched in healthcare [1]. Medical imaging plays a crucial part in detecting certain diseases [2]. Many commonly used medical imaging technologies exist, such as X-rays, ultrasound, Magnetic Resonance Imaging (MRI), Computerized Tomography (CT), Single-Photon Emission Computed Tomography (SPECT), and Positron Emission Tomography (PET) [3], [4]. Of the various medical imaging technologies, MRI stands out as the supreme method due to its unprecedented resolution when measured against other medical imaging technologies [2], [5], [6]. Brain tumor diagnosis is one such disease detection that utilizes MRI technology. A brain tumor is abnormal and uncontrolled cell proliferation in or near the brain [2], [3]. Most common brain tumors include meningioma, pituitary, and glioma [2]. Meningioma, glioma, and pituitary tumors pose a distinct challenge in their classification due to their varying size, shape, and intensity. Where meningioma affects the thin membrane that protects the spinal cord and brain, glioma occurs in the brain's glial cells, and pituitary tumors result from abnormal cell growth in the pituitary gland near the brain [2]. According to the National Brain Tumor Association, 787.000 Americans have had brain tumors [7], [8]. The patient was reported to have about a 36% survival rate [8]. In 2021, the population of patients with a brain tumor diagnosed was expected to reach 84.170 [7]. Brain tumors have become the ninth most prevalent cause of mortality worldwide and domestically due to their high incidence rate [8].

Considering the severity of the problem, it is necessary to have a technology that can help the radiology team detect brain tumors automatically and accurately to save many lives worldwide. However, many radiologists still rely on manual brain tumor detection, which leads to the possibility of human error, time-consuming, and laborious work [9]–[11]. Through MRI technology, computer-aided diagnosis (CAD) may assist radiologists in the clinical context by quickly identifying brain tumors [8]. Convolutional Neural Networks (CNN) [12], transfer learning [8], [13]–[15], and supervised and unsupervised learning

https://doi.org/10.25077/jnte.v12n2.1123.2023

techniques [4], [16], among others, have all been developed by researchers as automated systems for the detection of brain tumors. Most brain tumor detection research focuses on deep learning approaches with CNNs, such as the one conducted in [12], [17], [18]. The reason is that CNN offers several advantages, such as eliminating the need for manual feature extraction and is faster than traditional machine learning algorithms [2]. Nevertheless, the CNN-based detection model training process takes a long execution time, requires high system specifications, and increases user cost [2], [11], [19], [20], primarily when implemented in the area of medical imaging [2]. This is due to several factors; firstly, the medical image dataset has limited data because it takes radiologists to label it [2]. Secondly, training CNNs on small datasets is challenging due to the potential for overfitting [2]. Thirdly, it requires expertise to adjust the hyperparameter of the CNN classifier to reach higher performance [2]. To overcome this problem, transfer learning-based approaches have recently received a significant amount of attention for detecting brain tumors from MRI images, as they provide accurate and precise detection results [2], [4], [21]. Furthermore, it is a solution to the limitations found in CNN approaches [2], allowing the knowledge obtained from one task to be reused for related tasks, resulting in better performance when the detection of brain tumors in a defined dataset [4], [12], [22]. Despite these issues, the initial development of the brain tumor detection prototype is still ongoing. Some researchers that have created MRI prototypes for identifying brain tumors mostly employ the CNN approach [23], [24]. However, these researchers faced limitations that mainly stemmed from the limited availability of datasets.

Some researches are relevant to this research. The research was conducted by V. Kasala et al. [24] in 2018. They used a CNN model based on T1-weighted MRI images to categorize brain tumours. F. J. P. Montalbo [21] conducted the research in 2020. They have applied transfer learning and fine-tuning approaches to the YOLO model to identify three types of brain tumors from MRI images. S. M. Kulkarni et al. [25] conducted the research in 2020. They constructed a work set to separate and classify brain tumors using a deep learning approach involving CNN models and AlexNet architecture coupled with transfer learning based on GoogLeNet architecture. G. Habib et al. [26] conducted the research in 2020. They applied deep domain transfer learning to CNN for biomedical image classification. B. V. Isunuri et al. [27] conducted the research in 2021. They used a separable CNN approach to classify brain MRI images into three classes. D. Tree et al. [28] conducted the research in 2021. They adopted a transfer learning method to investigate the malignancy classification of brain tumors detected through MRI images. N. Ullah et al. [2] conducted the research in 2022. They investigated the performance of nine pre-trained transfer learning classifications. N. Kesav et al. [3] conducted the research in 2022. They suggested a new structure that utilizes the RCNN method to categorize brain tumors. M. I. Sharif et al. [7] did the most recent research in 2022. They devised an efficient approach using the transfer learning method with the Xception model and ADAM optimization algorithm.

This research discusses the design of a model to accurately detect brain tumors into four classes, namely meningioma, pituitary, glioma, and no tumor from MRI images based on transfer learning and fine-tuning methods so that the designed model can adapt to the unique features of brain tumor detection and ultimately its performance can be evaluated. Specifically, the model used is Xception. The use of this model is supported by S. Asif et al.'s prior research, which has demonstrated successful results [8]. Additionally, Xception, one of the most sophisticated transfer learning models employing completely separable convolution layers, is well known [29]. Many shreds of evidence have shown its superiority compared to inceptionV3, VGG, and ResNet in accurately classifying ImageNet datasets [8]. The research intends to develop a recommended prototype model and assess the effectiveness of the Xception model in identifying brain tumors. This research reports the overall sensitivity, specificity, accuracy, precision, and f1-score of the model and the prototype created using the proposed model.

MATERIALS AND METHODS

This section will describe the resources and procedures utilized in this research. The materials include data sources, hyperparameters, environment, and metrics. The methods relate to the procedures used in constructing a transfer learning model capable of detecting brain tumors.

Materials

Data Sources

The dataset utilized in this research is divided into three distinct portions: the validation set, the train set, and the test set. The validation set is used to check the accuracy of the model that has been trained using the train set, and if the accuracy is still thought to be not qualified, then the search for the right hyperparameter will be carried out. Then the model training is carried out again, this scenario is done to prevent overfitting in the model if the model is tested using only the test set. The train set is used to train the model to learn to detect brain tumors. The test set is used for model testing and prototypes that have been made.

The data splitting algorithm using the StratifiedKFold library produced this research's train and validation set. The splitting ratio was 20% for the validation and 80% for the train set, performed during k-fold cross-validation on the Kaggle dataset [30]. It is important to note that O. Ozkaraca et al. [31] also used this dataset in their research. This dataset contains 7023 brain MRI images of 2000 no tumor MRI images (normal brain), 1621 brain MRI images with glioma, 1645 brain MRI images with meningioma, and 1757 brain MRI images with the pituitary. Initially, this dataset consists of two directories: training and testing. However, in this research, the two folders are merged into one. For example, MRI images from this dataset are shown in Figure 1.

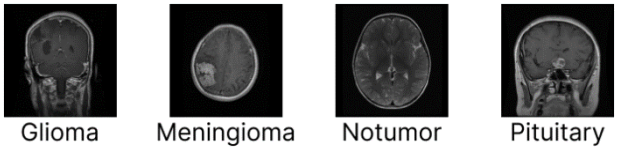

Figure 1. The example of an MRI scan used for training and validation purposes

https://doi.org/10.25077/jnte.v12n2.1123.2023

The test set, acquired through Kaggle [32], corresponded to the dataset used in the research by N. Ullah et al. [2]. The dataset has training and testing folders, but testing is only done on the training folder's data, which includes 395 MRI images of a healthy brain with no tumor, 826 images of a brain with a glioma brain tumor, 822 images of a brain with a meningioma brain tumor, and 827 images of a brain with a pituitary brain tumor. The folder names of the four classes were also changed from no_tumor to notumor, glioma_tumor to glioma, meningioma_tumor to meningioma, and pituitary_tumor to the pituitary. Figure 2 displays an example MRI image from this dataset.

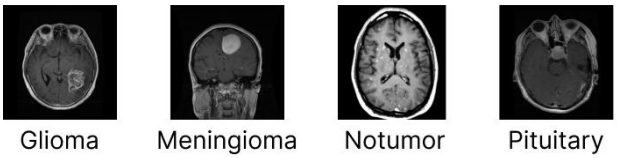

Figure 2. The example of an MRI scan used for the test set

Hyperparameter For Experiments

Hyperparameters encompass a set of parameters influencing the model training process. It comprises epochs, batch size, activation functions, learning rate, dropouts, and other parameters. Table 1 shows the hyperparameter used to train the model for this research.

Table 1. The hyperparameter used in this research

Parameter	Value
Epoch	50
Epoch (Fine Tuning)	60
Batch Size	64
Image Size	224x224
Optimizer	Adam
Activation Function	Softmax
Learning Rate	0.0001
Learning Rate (Fine Tuning)	0.00001
Dropout	0.2
K-Fold	5

Environments

This research utilizes software and hardware requirements to execute various experiments such as:

a. The experimental hardware setup involved using a personal laptop converted into a server equipped with an Intel Core i7-6700HQ processor, 16GB RAM, 250GB SSD, and 1TB HDD.
b. The research employs Ubuntu 22.04.1 as the software, including a Python program designed for conducting experiments. Furthermore, Visual Studio Code is installed to facilitate the editing of Python programs.

Metrics

The indicators from previous research were utilized to evaluate the brain tumor detection model's performance [2], [4], as well as following the recommendations provided by research experts [33]. These metrics include sensitivity, specificity, accuracy, precision, and f1-score. In addition, this research also added a confusion matrix as a metric. Sensitivity is the equation used to correctly identify patients afflicted with brain tumor disease [8]. Specificity is the equation used to identify patients without brain tumor disease correctly [8]. Accuracy is an equation of correctly predicted images' proportions to the total number of images [8]. Precision is an equation that represents the model's reliability in classifying MRI brain tumors as positive [8]. F1-score is an equation that incorporates precision and sensitivity by taking the harmonic mean [8]. Confusion matrix is a metric used for predictive analysis in machine learning. Classification-based models can have their effectiveness evaluated using a confusion matrix [8]. A confusion matrix can also be a table summary of the classification model's counts of right and wrong predictions [8]. Details of sensitivity, specificity, accuracy, precision, and f1-score equations can be seen in (1)-(5), as below:

$$sensitivity = \frac{TP}{TP + FN} \quad (1)$$

$$specificity = \frac{TN}{TN + FP} \quad (2)$$

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

$$precision = \frac{TP}{TP + FP} \quad (4)$$

$$f1 - score = 2 \times \frac{sensitivity \times sensitivity}{sensitivity + precision} \quad (5)$$

TP (True Positive) denotes the images accurately diagnosed as brain tumor patients using N samples. TN (True Negative) also describes the count of healthy images accurately diagnosed as brain tumor patients in those N samples. On the other hand, FP (False Positive) describes the count of images that were improperly diagnosed as brain tumor patients in the N samples. At the same time, FN (False Negative) describes the count of healthy images that were improperly diagnosed, also in those N samples.

Methods

The primary goal of the research is to design a model which can automatically detect brain tumors in MRI images. The tumors were divided into four groups: meningioma, pituitary, glioma, and situations with no tumors. To accomplish this, transfer learning and fine-tuning techniques are used, which allow models to be customized and adapted to the unique characteristics of brain tumor detection. The model used in this research is Xception.

To make sure optimum results, some techniques are proposed. Firstly, a k-fold cross-validation principle is adopted to enhance the detection performance [34]. This involved splitting the dataset into train, validation, and test sets. This separation works to avoid overfitting and ensure model generalization [34]. Secondly, an early-stopping approach has been implemented to stop model training if there is no improvement in validation accuracy [8].

This technique protects against overfitting and ensures model stability [8].

Further, additional layers are added after the base model from the pre-trained model. This enables further customization and adaptation specific to the brain tumor detection task. The additional layer added can be seen in Table 2, which is the step after feature extraction. Furthermore, L1 and L2 regularizers are inserted into the first dense layer, as seen in Table 2. These regularization techniques help prevent overfitting by restraining the coefficients significantly when the complexity of the model increases [8].

Table 2. Additional layer

Layer	Activation Function	Regularizers
GlobalAveragePooling2D	-	-
Dense (512)	ReLU	L1 L2
BatchNormalization	-	-
Dropout (0.2)	-	-
Dense (256)	ReLU	L1 L2
BatchNormalization	-	-
Dropout (0.2)	-	-
Dense (4)	softmax	-

Figure 3 shows the workflow of our proposed approach for predicting brain tumors. The suggested methodology in this research is generally broken down into five stages: Image preprocessing, feature extraction, classification, fine-tuning, and prototyping.

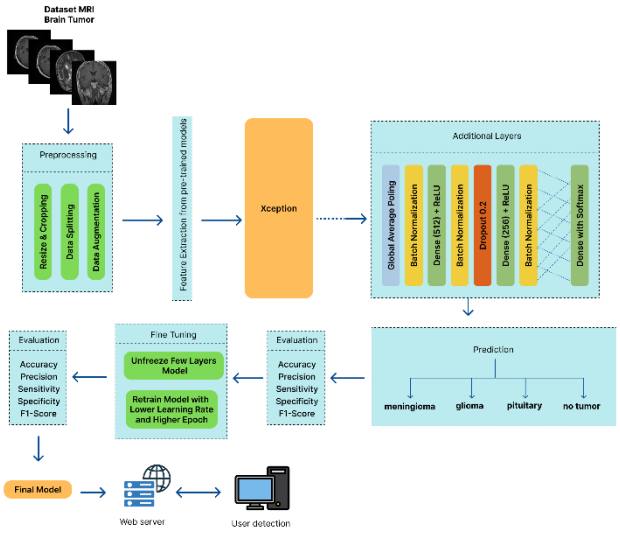

Figure 3. Workflow of the proposed method in this paper

Image Preprocessing

Image processing is a very crucial step in brain MRI image analysis as the datasets used are often unstructured and come from many different sources [4], [8], [35]. Furthermore, MRI images may contain irrelevant spaces and areas, noise, and missing values, reducing classification performance [4], [8]. Therefore, standardization and processing of the dataset before training the model is required [4], [8]. This research divides image processing into three stages: image cropping and resizing, data splitting, and data augmentation.

a. Image cropping and resizing

Multiple procedures occur throughout this stage, from acquiring input images for preprocessing. Subsequently, all RGB MRIs are converted to greyscale, and noise is removed using the Gaussian blur technique. Threshold processing then converts the greyed-out employee image into a binary format [36]. The next step is to locate contours in the threshold picture by choosing the biggest contour that is present. After that, extreme points are determined based on these largest contours. Next, the image is cropped according to the contours and extreme points. Finally, resizing is performed to adjust the image's dimensions to fit the model's needs. The image preprocessing process is shown in Figure 4.

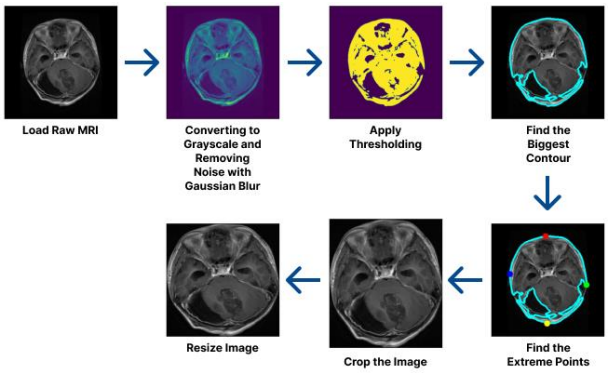

Figure 4. Image preprocessing at this stage

b. Data splitting

The data is partitioned into a train and validation set at this stage. This data separation is carried out using the StratifiedKFold library with a ratio of 80% and 20%, which is applied in the k-fold cross-validation process.

c. Data Augmentation

Data augmentation is performed to avoid overfitting and improve model performance [4], [8]. Data augmentation adds copies of data to increase the variety of datasets used in the training process [4]. This research involves the data augmentation process, including rescaling, rotation, zoom, horizontal flip, and vertical flip using the ImageDataGenerator library provided by Keras. Figure 5 shows the original images that will be augmented, and Figure 6 shows the results of the data augmentation.

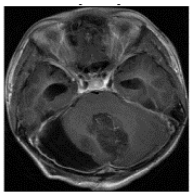

Figure 5. Original image

https://doi.org/10.25077/jnte.v12n2.1123.2023

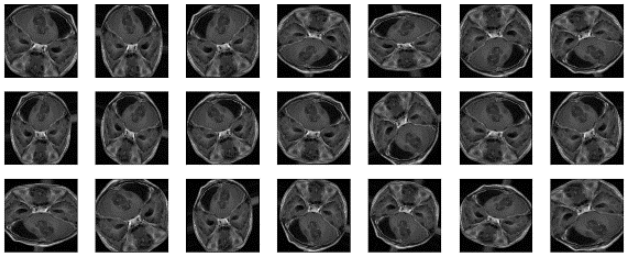

Figure 6. Augmentation images

Feature Extraction

In the feature extraction step of this research, Xception was used. It was trained on the ImageNet dataset. Xception, as developed by Chollet [29], is one of the most sophisticated deep-learning architecture models [29]. This is based on deeply separable convolution layers and enhances the previous Inception network [29]. In Xception, there are 36 separable convolution layers, and the model could be segmented into 14 separate modules [29]. All modules contain linear residual links surrounding them, except for the initial and final modules [29]. Xception consists of a convolutional layer stack that can be separated by depth [29]. This process involves a depth-based convolution on all input channels to map the spatial relationship [29].

The Xception model can be utilized because it uses a transfer learning approach to transfer knowledge from previously completed tasks to new ones. This allows this research to focus on more specific tasks, such as brain tumor detection, to overcome the constraints of limited training data. The output of the feature extraction process is the resulting feature set of the brain tumor image after passing through the transfer learning model. The features represent valuable information about the image that can be used to distinguish and classify types of brain tumors, such as meningioma, pituitary, glioma, and no tumor. After the feature extraction step is completed, the inner features obtained are forwarded to the three dense or fully connected layers. The spatial average value of the features is computed using the global average pooling layer. The base model architecture of Xception and its adjustments, which are utilized in this research, are shown in Figure 7.

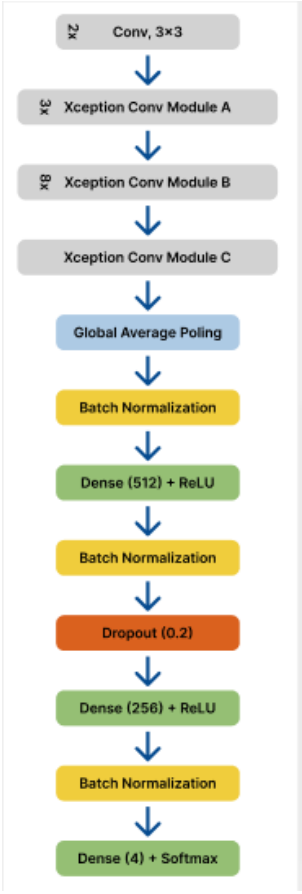

Figure 7. Base and customization architecture

Classification

In the classification stage, the previously extracted features from the brain tumor images will be the input for this stage. In the context of this research, the softmax layer is used to classify a brain tumor image into the desired categories, such as meningioma, pituitary, glioma, and no tumor. The probabilities generated by the softmax layer will provide information regarding the confidence level of the model for each brain tumor class. Softmax transforms the final output of the last layer in the neural network into an underlying probability distribution, which is one of the benefits of utilizing it [23]. By dividing the total exponential value of all potential outputs by the exponential value of the final output, the softmax formula (6) can be calculated [23].

$$Softmax(z_i) = \frac{\exp(z_i)}{\sum_i \exp(z_i)} \quad (6)$$

Fine-Tuning

Fine-tuning is the customization of model parameters based on the specific dataset used in research. The aim is to make the model focus and more effectively adapt to the brain tumor features in the specific dataset. Fine-tuning in this research is applied at layer 100 onwards. It means that the parameters in the initial layer retain the knowledge learned from the pre-trained model. In contrast, the adjustment of layer parameters after layer 100 is customized to the specific dataset used in the research. In this research, the learning rate was reduced from 0.0001 to 0.00001. Decreasing the learning rate made the learning process of the model slower but more stable and precise in adapting parameter changes to specific datasets. In addition, the epoch was increased

from 50 to 60. Epoch represents the number of iterations the model undergoes during self-training with the dataset. By increasing the epoch, the model has more opportunities to refine and improve its ability to detect brain tumors.

Prototyping

After fine-tuning, a final model was obtained that will be used to develop a brain tumor detection prototype with four classes (meningioma, pituitary, glioma, and no tumor) from brain MRI images. The Flask framework was used to build the prototype's backend, and the user interface was developed using HTML and JavaScript. Two routes are constructed in the Flask application: the prediction route to call predictions from the best model built and the index page route to upload brain MRI images. This prototype is hosted locally. With this prototype, users can automatically upload brain MRI images in JPEG/JPG format and perform tumor prediction.

RESULTS AND DISCUSSION

This section discusses the Xception model's pre-trained performance using the ImageNet dataset to accurately detect brain tumors in four categories, namely meningioma, pituitary, glioma, and no tumor. Transfer learning and fine-tuning on a series of MRI images are used. Transfer learning and fine-tuning significantly minimize the overfitting issue that frequently arises in algorithms when testing algorithms on train and validation sets. In the training and validation stage, several additional techniques exist, namely the principle of k-fold cross-validation, early stopping techniques, adding additional layers, and using L1 and L2 regularizers. According to Table 1, the model is both validated and trained using the same hyperparameter. We analyzed and evaluated the Xception model using sensitivity, specificity, accuracy, precision, and f1-score evaluation metrics, whose formulas can be seen in (1)-(5).

Training and Validation

This stage shows the training and validation process's results on the Xception model.

Figure 8. The graph of accuracy and loss in training and validation at fold 3

Figure 8 shows the results of the accuracy and loss graph generated by fold 3, which is the best accuracy and loss graph in the five-fold cross-validation process. We can see that fold 3 reaches epoch 50 without being impacted by early stopping, achieving 92.88% accuracy with 55.22% loss, as shown in the accuracy and loss graphs from training and validation results on other folds in the section APPENDICES Figure 16 to Figure 19.

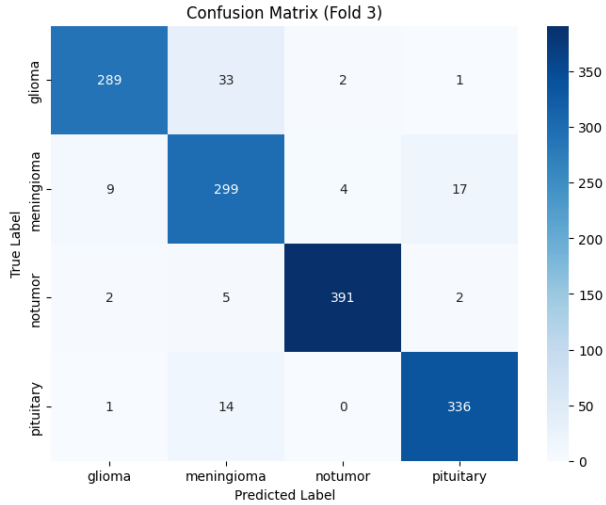

Figure 9. The resulting confusion matrix at fold 3

Figure 9 shows the best confusion matrix produced during the five-fold training and validation process. This confusion matrix was generated by fold 3. Only a few mispredictions occurred: 90 images, while all other MRIs were correctly predicted, for images of confusion matrix results from training and validation on other folds can be seen in APPENDICES Figure 20 to Figure 23.

Table 3. The performance rate (%) of a five-fold cross-validation

Fold	Accuracy	Precision	Sensitivity	Specificity	F1-score
1	91.89%	91.70%	91.55%	90.00%	91.60%
2	89.40%	89.03%	88.88%	**90.06%**	88.92%
3	**93.59%**	**93.52%**	**93.32%**	89.75%	**93.36%**
4	91.17%	90.97%	90.77%	88.96%	90.84%

https://doi.org/10.25077/jnte.v12n2.1123.2023

5	90.95%	90.60%	90.52%	89.10%	90.48%

The metrics obtained after the five-fold cross-validation procedure are displayed in Table 3. These findings indicate that, except for the specificity generated by fold 2, fold three produces the highest results across all matrices.

Fine-Tuning

The fold selected for fine-tuning is fold 3 due to its highest overall performance metrics, the lowest misprediction error, and the highest sensitivity among all folds. Figure 10 shows the accuracy and loss rates during fine-tuning training and validation. The outcomes of fine-tuning training and validation are also shown in Figure 11 as a confusion matrix. Only a few mispredictions occurred: 26 images, while all other MRIs were correctly predicted.

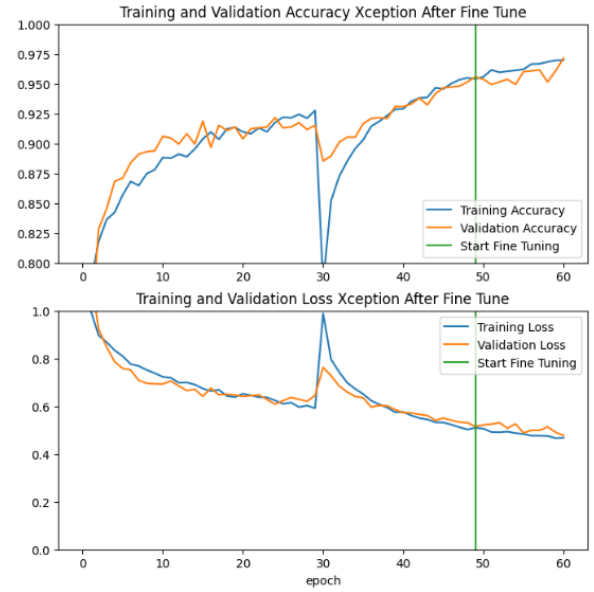

Figure 10. The graph of accuracy and loss in training and validation after fine-tuning process

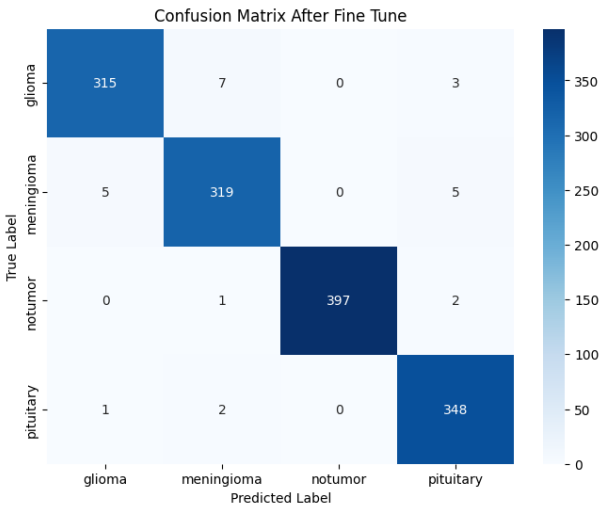

Figure 11. The resulting confusion matrix in the fine-tuning process

Table 4 shows the overall Xception model's final performance after fine-tuning, and Table 5 shows the performance results for each class.

Table 4. The performance rate after fine-tuning (%)

Accuracy	Precision	Sensitivity	Specificity	F1-score
98.14%	98.07%	98.07%	97.83%	98.07%

Table 5. The performance rate for each class fine-tuning (%)

Class	Accuracy	Precision	Sensitivity	specificity	F1-score
Glioma	98.15%	98.13%	96.92%	97.83%	97.52%
Meningioma	98.15%	96.96%	96.96%	97.83%	96.96%
Notumor	98.15%	100%	99.25%	97.83%	99.62%
Pituitary	98.15%	97.20%	99.14%	97.83%	98.16%

Prototype Development and Testing

This research used the Xception model in a brain tumor detection prototype to predict MRI images into four classes: meningioma, pituitary, glioma, and no tumor. The prototype results can be observed in Figure 12, which displays the main page of the prototype. Figure 13 shows the interface when the user has uploaded an MRI for brain tumor detection. Figure 14 shows the prototype interface when displaying the detection results. When the detection results detect a tumor, it will display red text; if it is healthy, it will show green text.

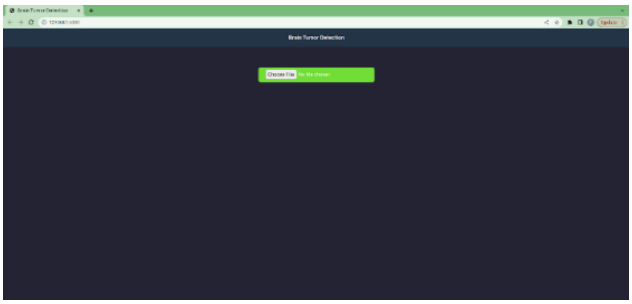

Figure 12. The prototype main page

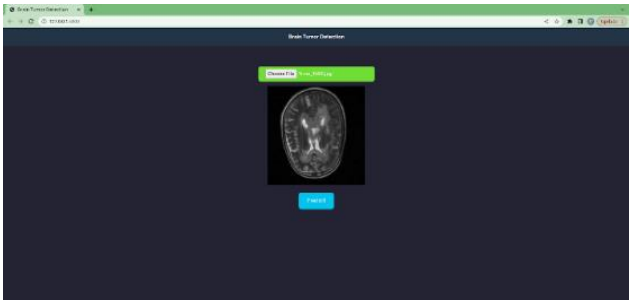

Figure 13. The prototype interface when the user has uploaded an MRI

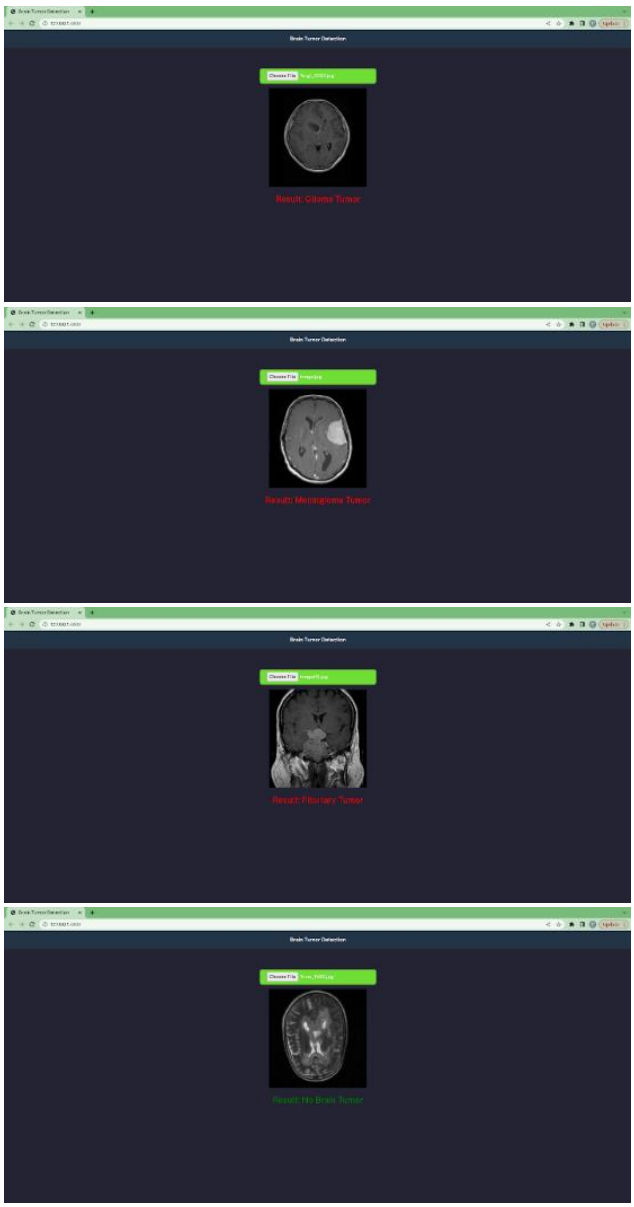

Figure 14. The prototype interface when showing detection results

Figure 15 shows the confusion matrix of the prototype testing process using the test set. Only a few mispredictions occurred: 125 images, while all other MRIs were predicted correctly. Table 6 shows the performance of the prototype testing using the test set.

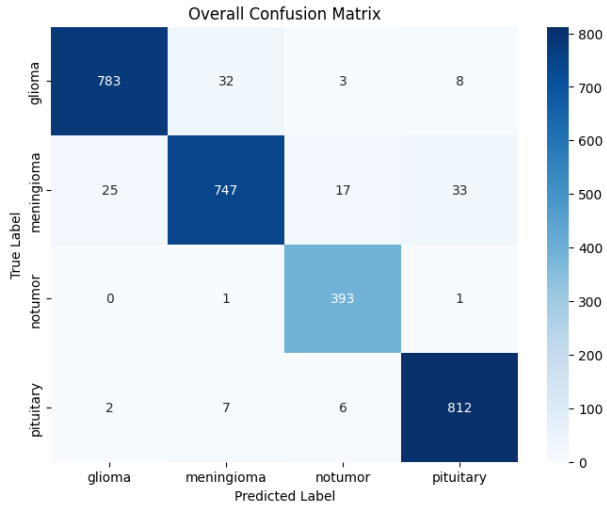

Figure 15. Confusion matrix on prototype testing

Table 6. Prototype performance rate (%)

Accuracy	Precision	Sensitivity	Specificity	F1-score
95.30%	95.31%	95.30%	96.07%	95.27%

Discussion

This section examines the model design experiments that went into creating the Xception model, which is used to detect brain tumors from MRI scans, as well as the prototype tests conducted using the final model developed from the model design experiments.

The effectiveness of the models is assessed during the cross-validation procedure's training and validation phases. Figure 8 shows no discernible change in the accuracy and loss trends between the training and validation sets. This observation concludes that the trained models perform commendably and can generalize well. Therefore, the model's outcome does not suffer from overfitting as their performance on the validation data is comparable to that of the training data. However, from the illustration, it can also be observed that the training and validation losses are high, which indicates that the models have difficulty making accurate predictions or capturing patterns in the data set.

Observing the brain tumor mispredictions results is also possible during the training and validation process in the cross-validation step. It can be seen through the confusion matrix shown in Figure 9 that this model has a relatively low prediction error rate of only 1.28%. Despite a relatively small error, prediction errors can have severe consequences in detecting brain tumors. For example, false negative errors (not detecting a tumor present) can lead to delays in proper diagnosis and treatment, which can negatively impact the patient. False positive errors (detecting a tumor that is not present) can also lead to anxiety and unnecessary additional procedures. Therefore, this research performs fine-tuning to reduce these prediction errors.

During the fine-tuning process, the fold selection for fine-tuning is based on several factors. The selected fold exhibited the highest overall performance metrics and minimum misprediction error, accentuating achieving the highest sensitivity among all folds. This emphasis on sensitivity is significant in healthcare research

https://doi.org/10.25077/jnte.v12n2.1123.2023

as it calculates the probability of obtaining a positive test result in subjects with brain tumor disease [37].

It can be noticed from Figure 10 that there is a decrease in model performance in the early stage of fine-tuning due to rapid adaptation while the learning rate is still high. In contrast, the drastic improvement occurs later because the model has adapted the weights of the unfreeze layer better after several longer training stages. However, with time and further training, the model will better adapt to the unfreeze layer's weights. This can result in a drastic improvement in model performance, where the accuracy graph suddenly rises, and the loss graph suddenly drops significantly.

The confusion matrix in Figure 11 shows that the model has a prediction error rate of 0.37% after fine-tuning. In addition, Table 3 and Table 4 show that the improved performance and decreased prediction error indicate that the model adapts well to the target dataset.

By referring to Table 4 and Table 6, it can be noticed that there is a performance variation of less than 3% in the prototype. This slight performance difference indicates that the model that has been created is good enough. Despite this difference, since the difference is less than 3% and there is no strong indication of overfitting or underfitting, the difference is acceptable, and the model can predict well on the new test data. The difference in performance is likely due to specific characteristics in the test dataset that are not well represented in the training dataset, so the model has difficulty generalizing to the test data.

Comparison with State-of-the-Art Related Works

Table 7 shows the comparative differences between the approaches based on sensitivity (recall). Only sensitivity (recall) is considered in Table 7 as a performance parameter. Sensitivity is crucial in health research as it calculates the percentage of people with brain tumor disease who test positive [37]. Our research shows sensitivity in detecting brain tumors, which is one of the best compared to previous research. The difference in our results is only about 0.26% compared to research conducted by D. Tree et al. [28].

Table 7. Comparison with existing approaches

Work	Method	Sensitivity
G. Habib et al [26]	Deep domain transfer learning	79.85%
F. J. P. Montalbo et al [21]	YOLO	88.58%
K. N. Deeksha et al [23]	CNN	91%
V. Kasala et al [24]	CNN	93%
B. V. Isunuri et al [27]	Separable CNN	97.19%
D. Tree et al [28]	EfficientNet-B3	98.33%
Our Proposed Model	**Xception**	**98.07%**

CONCLUSIONS

This research demonstrates the effectiveness of the Xception model combined with fine-tuning and MRI analysis for detecting brain tumors of various types (meningioma, pituitary, glioma, and no tumor). We achieved exceptional performance by automating the detection process and building a prototype based on the final model. The Xception model exhibited a sensitivity of 98.07%, specificity of 97.83%, accuracy rate of 98.15%, precision of 98.07%, and f1-score of 98.07%. Implementing the Xception model, the prototype also showed impressive performance on the test set, with a sensitivity of 96.07%, specificity of 95.30%, accuracy of 95.31%, precision of 95.30%, and f1-score of 95.27%. The model and prototype can assist clinical practitioners and radiologists diagnose, measure, and monitor brain tumors. Although there is a slight performance difference between the trained model and the prototype due to limited training dataset diversity and characteristics, this research has achieved satisfactory results. Future improvements should focus on using more significant and diverse datasets for training and conducting comprehensive testing with datasets from multiple hospitals.

ACKNOWLEDGMENT

The Department of Informatics, School of Computing, Telkom University, Indonesia, and the Human Centric (HUMIC) Engineering, School of Computing, Telkom University, Bandung, Indonesia, funded this research.

REFERENCES

[1] J. C. Gore, "Artificial intelligence in medical imaging," *Magn. Reson. Imaging*, vol. 68, pp. A1–A4, 2020, doi: 10.1016/j.mri.2019.12.006.

[2] N. Ullah *et al.*, "An Effective Approach to Detect and Identify Brain Tumors Using Transfer Learning," *Appl. Sci.*, vol. 12, no. 11, 2022, doi: 10.3390/app12115645.

[3] N. Kesav and M. G. Jibukumar, "Efficient and low complex architecture for detection and classification of Brain Tumor using RCNN with Two Channel CNN," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 6229–6242, 2022, doi: 10.1016/j.jksuci.2021.05.008.

[4] 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, no. Ml, pp. 59099–59114, 2022, doi: 10.1109/ACCESS.2022.3179376.

[5] S. Amiri, M. Ali Mahjoub, and I. Rekik, "Tree-based Ensemble Classifier Learning for Automatic Brain Glioma Segmentation," *Neurocomputing*, vol. 313, pp. 135–142, 2018, doi: 10.1016/j.neucom.2018.05.112.

[6] S. A. Abdelaziz Ismael, A. Mohammed, and H. Hefny, "An enhanced deep learning approach for brain cancer MRI images classification using residual networks," *Artif. Intell. Med.*, vol. 102, p. 101779, 2020, doi: 10.1016/j.artmed.2019.101779.

[7] M. I. Sharif, J. P. Li, M. A. Khan, and M. A. Saleem, "Active deep neural network features selection for segmentation and recognition of brain tumors using MRI images," *Pattern Recognit. Lett.*, vol. 129, pp. 181–189, 2020, doi: 10.1016/j.patrec.2019.11.019.

[8] 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.

[9] M. Malathi and P. Sinthia, "Brain tumour segmentation using convolutional neural network with tensor flow," *Asian Pacific J. Cancer Prev.*, vol. 20, no. 7, pp. 2095–2101, 2019, doi: 10.31557/APJCP.2019.20.7.2095.

[10] A. A. Malibari *et al.*, "Arithmetic Optimization with RetinaNet Model for Motor Imagery Classification on Brain Computer Interface," *J. Healthc. Eng.*, vol. 2022, no. Mi, 2022, doi: 10.1155/2022/3987494.

[11] J. Amin, M. A. Anjum, M. Sharif, S. Jabeen, S. Kadry, and P. Moreno Ger, "A New Model for Brain Tumor Detection Using Ensemble Transfer Learning and Quantum Variational Classifier," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/3236305.

[12] K. Muhammad, S. Khan, J. Del Ser, and V. H. C. D. Albuquerque, "Deep Learning for Multigrade Brain Tumor Classification in Smart Healthcare Systems: A Prospective Survey," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 2, pp. 507–522, 2021, doi: 10.1109/TNNLS.2020.2995800.

[13] Z. N. K. Swati *et al.*, "Brain tumor classification for MR images using transfer learning and fine-tuning," *Comput. Med. Imaging Graph.*, vol. 75, pp. 34–46, 2019, doi: 10.1016/j.compmedimag.2019.05.001.

[14] J. Amin, M. Sharif, M. Yasmin, and S. L. Fernandes, "A distinctive approach in brain tumor detection and classification using MRI," *Pattern Recognit. Lett.*, vol. 139, pp. 118–127, 2020, doi: 10.1016/j.patrec.2017.10.036.

[15] Ö. Polat and C. Güngen, "Classification of brain tumors from MR images using deep transfer learning," *J. Supercomput.*, vol. 77, no. 7, pp. 7236–7252, 2021, doi: 10.1007/s11227-020-03572-9.

[16] P. M. Siva Raja and A. V. rani, "Brain tumor classification using a hybrid deep autoencoder with Bayesian fuzzy clustering-based segmentation approach," *Biocybern. Biomed. Eng.*, vol. 40, no. 1, pp. 440–453, 2020, doi: 10.1016/j.bbe.2020.01.006.

[17] M. Ahmed Hamza *et al.*, "Optimal and Efficient Deep Learning Model for Brain Tumor Magnetic Resonance Imaging Classification and Analysis," *Appl. Sci.*, vol. 12, no. 15, 2022, doi: 10.3390/app12157953.

[18] M. Toğaçar, B. Ergen, and Z. Cömert, "BrainMRNet: Brain tumor detection using magnetic resonance images with a novel convolutional neural network model," *Med. Hypotheses*, vol. 134, p. 109531, 2020, doi: 10.1016/j.mehy.2019.109531.

[19] J. Amin, M. Sharif, M. Yasmin, T. Saba, and M. Raza, "Use of machine intelligence to conduct analysis of human brain data for detection of abnormalities in its cognitive functions," *Multimed. Tools Appl.*, vol. 79, no. 15–16, pp. 10955–10973, 2020, doi: 10.1007/s11042-019-7324-y.

[20] J. Amin *et al.*, "Brain Tumor Detection by Using Stacked Autoencoders in Deep Learning," *J. Med. Syst.*, vol. 44, no. 2, 2020, doi: 10.1007/s10916-019-1483-2.

[21] F. J. P. Montalbo, "A computer-aided diagnosis of brain tumors using a fine-tuned yolo-based model with transfer learning," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 12, pp. 4816–4834, 2020, doi: 10.3837/tiis.2020.12.011.

[22] P. K. Chahal, S. Pandey, and S. Goel, "A survey on brain tumor detection techniques for MR images," *Multimed. Tools Appl.*, vol. 79, no. 29–30, pp. 21771–21814, 2020, doi: 10.1007/s11042-020-08898-3.

[23] K. N. Deeksha, M. Deeksha, A. V. Girish, A. S. Bhat, and H. Lakshmi, "Classification of Brain Tumor and its types using Convolutional Neural Network," *2020 IEEE Int. Conf. Innov. Technol. INOCON 2020*, pp. 1–6, 2020, doi: 10.1109/INOCON50539.2020.9298306.

[24] V. Kasala and L. Baton Rouge, "DETECTION OF TYPES OF BRAIN TUMORS USING SCRATCHED," no. May, 2021.

[25] S. M. Kulkarni and G. Sundari, "A framework for brain tumor segmentation and classification using deep learning algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 8, pp. 374–382, 2020, doi: 10.14569/IJACSA.2020.0110848.

[26] G. Habib and S. Qureshi, "Biomedical Image Classification using CNN by Exploiting Deep Domain Transfer Learning," *Int. J. Comput. Digit. Syst.*, vol. 10, no. 1, pp. 1075–1083, 2021, doi: 10.12785/ijcds/100197.

[27] B. V. Isunuri and J. Kakarla, "Three-class brain tumor classification from magnetic resonance images using separable convolution based neural network," *Concurr. Comput. Pract. Exp.*, vol. 34, no. 1, pp. 1–9, 2022, doi: 10.1002/cpe.6541.

[28] D. Tree and K. Neighbor, "Multi-Modal Case Study on MRI Brain Tumor," vol. 20, no. 3, pp. 107–117, 2021, doi: https://doi.org/10.53799/ajse.v20i3.175.

[29] F. Chollet, "Fatigue Behavior of Stainless Steel Sheet Specimens at Extremely High Temperatures," *SAE Int. J. Mater. Manuf.*, vol. 7, no. 3, pp. 560–566, 2014, doi: 10.4271/2014-01-0975.

[30] M. NICKPARVAR, "Brain tumor mri dataset," 2021. [Online]. Available: https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset. [Accessed: 11-May-2023].

[31] O. Özkaraca *et al.*, "Multiple Brain Tumor Classification with Dense CNN Architecture Using Brain MRI Images," *Life*, vol. 13, no. 2, 2023, doi: 10.3390/life13020349.

[32] K. Sartaj, B. and Ankita, K. and Prajakta, B. and Sameer, D. and Swati, "Brain Tumor Classification (MRI)," 2020. [Online]. Available: https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri. [Accessed: 11-May-2023].

[33] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation," *AAAI Work. - Tech. Rep.*, vol. WS-06-06, pp. 24–29, 2006, doi: 10.1007/11941439_114.

[34] D. Berrar, "Cross-validation," *Encycl. Bioinforma. Comput. Biol. ABC Bioinforma.*, vol. 1–3, no. January 2018, pp. 542–545, 2018, doi: 10.1016/B978-0-12-809633-8.20349-X.

[35] I. S. Dwira Kurnia Larasati, "Implementation of Template Matching on Detection of Stop Line Violations." doi: https://doi.org/10.25077/jnte.v10n3.898.2021.

[36] E. R. J. Geminiesty Lathifasari Djavendra, Siti Aisyah, "Desain Sistem Pengatur Lampu Lalu Lintas dengan Identifikasi Kepadatan Kendaraan Menggunakan Metode Subtraction." doi: https://doi.org/10.25077/jnte.v7n2.541.2018.

[37] A. M. Simundic, "Measures of Diagnostic Accuracy: Basic Definitions," *Ejifcc*, vol. 19, no. 4, pp. 203–211, 2009.

AUTHOR(S) BIOGRAPHY

Faiz Rofi Hencya

Currently a final year student at the Department of Informatics, School of Computing, Telkom University, Bandung, Indonesia.

Satria Mandala
PhD degree in Computer Science, Wireless Network Security at Universiti Teknologi Malaysia (UTM). Currently a Director in Human Centric (HUMIC) Engineering, School of Computing, Telkom University, Bandung, Indonesia. The area of research concentrates on Computer Networks, Wireless Ad hoc Networks, Network Security, and Biomedical Engineering.

Tong Boon Tang
PhD degree in Intelligent Sensor Fusion at The University of Edinburgh, U.K. Currently a Professor at the Centre for Intelligent Signal and Imaging Research, Universiti Teknologi Petronas, Perak, Malaysia. The area of research concentrates on NIRS, EEG, Mental Health, and Neurotechnology.

Mohd Soperi Mohd Zahid
PhD in Computer Science by University of Wisconsin - Milwaukee, USA. Currently an Associate Professor in the Department of Computer and Information Sciences, Universiti Teknologi Petronas, Perak, Malaysia. The area of research concentrates on Computer Networks and Machine Learning for Healthcare.

APPENDICES

Figure 16. The graph of accuracy and loss in training and validation at fold 1

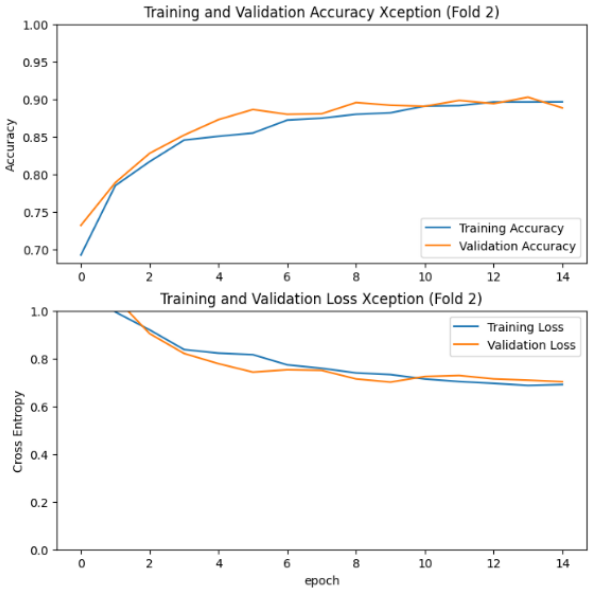

Figure 17. The graph of accuracy and loss in training and validation at fold 2

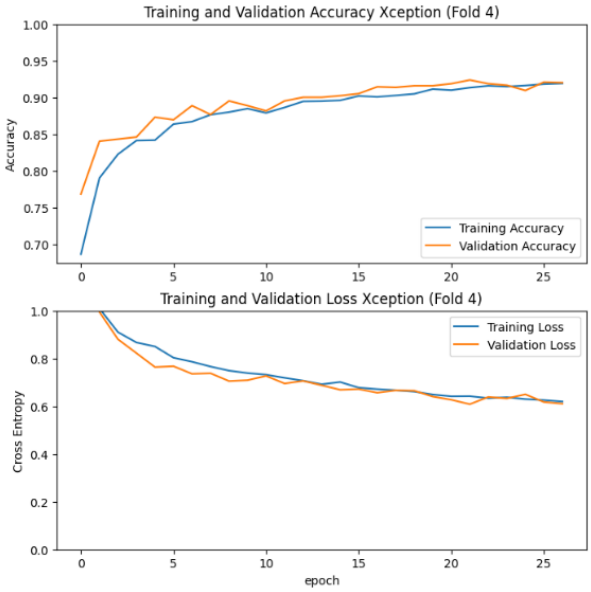

Figure 18. The graph of accuracy and loss in training and validation at fold 4

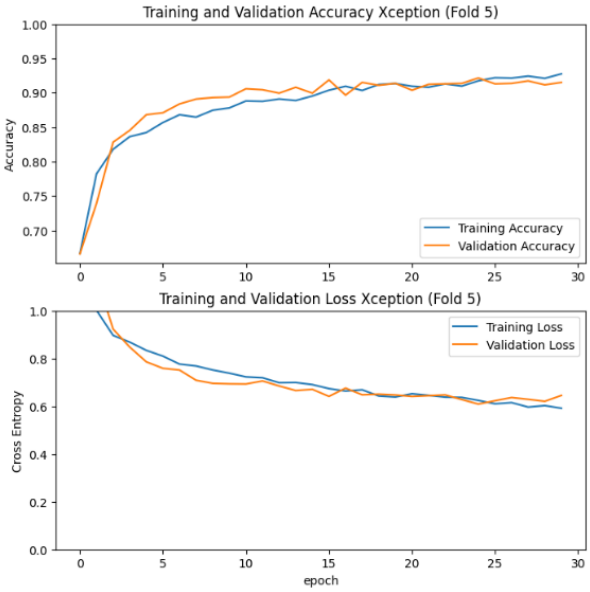

Figure 19. The graph of accuracy and loss in training and validation at fold 5

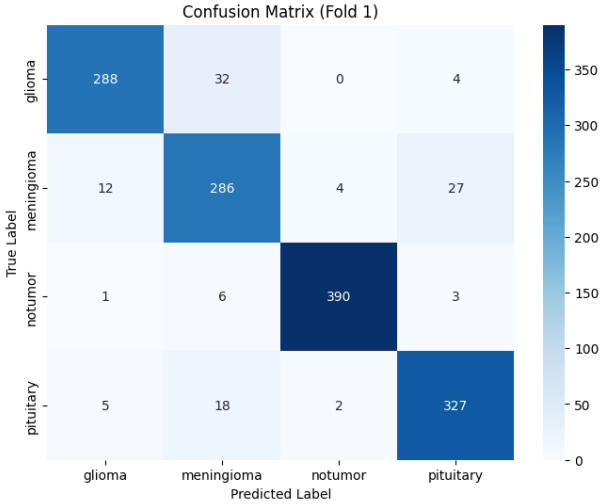

Figure 20. The resulting confusion matrix at fold 1

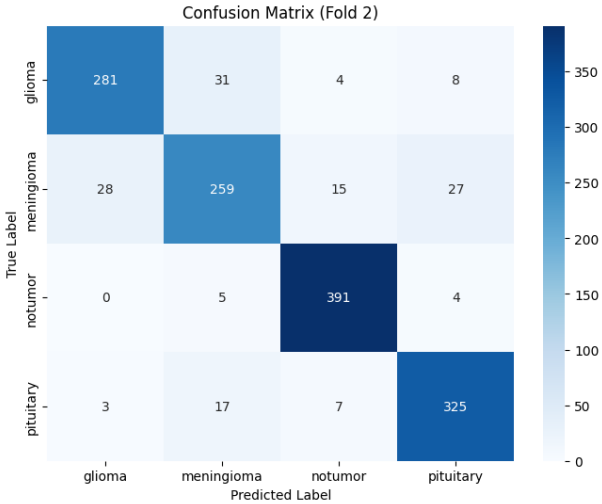

Figure 21. The resulting confusion matrix at fold 2

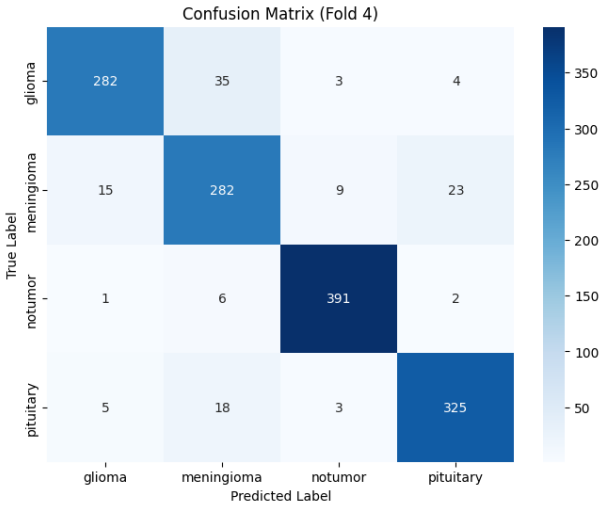

Figure 22. The resulting confusion matrix at fold 4

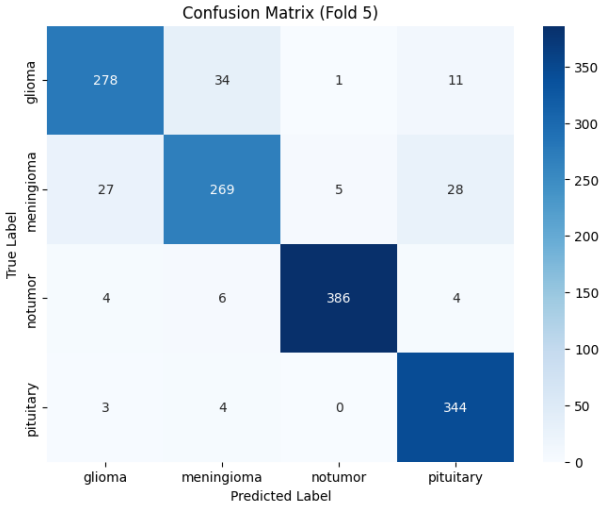

Figure 23. The resulting confusion matrix at fold 5

https://doi.org/10.25077/jnte.v12n2.1123.2023