JUSIKOM PRIMA (Jurnal Sistem Informasi dan Ilmu Komputer Prima)
Vol. 8 No. 1, August 2024 E-ISSN:2580-2879

Optimization Of Node Size Configuration In Cnn-Elm Model For Brain Tumor Mri Image Classification

*Sulthan Ahmad Ferdiansyah[1], Basuki Rahmat[*2], Fetty Tri Anggraeny[3]*
[1]Informatika, Fakultas Ilmu Komputer, UPN "Veteran Jawa Timur"
[2,3]Magister Informatika, Fakultas Ilmu Komputer, UPN "Veteran Jawa Timur"

ABSTRACT

This study proposed a method to classify four types of brain tumors—Glioma, Meningioma, Pituitary, and Non-Tumor—using the Kaggle Brain Tumor MRI Dataset. The research involved stages of data collection, preprocessing, model design, model training, and evaluation. A hybrid Convolutional Neural Network - Extreme Learning Machine (CNN-ELM) algorithm was employed, demonstrating the importance of selecting the optimal number of hidden nodes for achieving high accuracy. The test results revealed that with 2000 hidden nodes, the CNN-ELM model achieved an overall accuracy of 98.86%, with F1-scores of 97% for Glioma, 98% for Meningioma, 100% for Non-Tumor, and 100% for Pituitary tumors. In comparison, the model with 1000 hidden nodes achieved an accuracy of 96.96%, while models with 3000 and 4000 hidden nodes achieved 98.10% and 96.58% accuracy, respectively. These findings highlight the critical role of hidden node selection in optimizing model performance. The CNN-ELM algorithm proves to be a viable alternative for classifying brain tumor MRI images, contributing to advancements in medical technology.

Keywords: CNN-ELM, Brain Tumor, Classification, Optimization, Node Size.

INTRODUCTION

Brain tumor is one of the cancers that must be taken seriously. Brain tumors can affect children to adults (Noreen et al., 2020). Brain tumors are the most dangerous cancer because they attack the brain. All organs of the human body are controlled by the nervous system of the human brain. Brain tumors with varying types, locations, and shapes make them difficult to detect. Diagnosis of brain tumors depends on the type and location of the tumor so that doctors can determine the appropriate treatment between surgery, radiotherapy, or chemotherapy. This is important to increase the chances of survival for brain tumor patients (Chaki & Wozniak, 2023).

Previous research conducted by (Sheergojri & Iqbal, 2022) introduced a hybrid method in brain tumor identification using Magnetic Resonance Imaging (MRI) images to assist doctors in diagnosing and characterizing tumors with high accuracy and consistency. The proposed hybrid method is based on the CNN-ELM hybrid structure. The CNN-ELM method can surpass existing methods such as C-SVM and ELM in identifying tumors based on MRI images. With precision, f1-score, and recall rates of 91.5% and 97.14% accuracy, respectively.

The next previous research conducted by (Wahid et al., 2021) proposed a CNN-ELM hybrid method. ELM was chosen for its advantage in a fast training process, while CNN was chosen to perform feature extraction. CNN is one method of deep learning that has a special layer which allows it to conduct feature extraction and direct extraction training from raw input data (Purbasari et al., 2021). As a result, CNN-ELM with 8 filters in the convolution layer and 6000 nodes in the hidden layer showed good performance compared to other models with different number of filters and hidden nodes. With an average precision, recall, and f1-score of 91.5% and testing accuracy of 91.4%.

Therefore, the role of technology in the medical field is needed to assist medical personnel in diagnosing brain tumor diseases more efficiently. This research proposes a hybrid CNN-ELM (Convolutional Neural Network - Extreme Learning Machine) method to classify brain tumor diseases. CNN is useful for extracting features that will later be classified by ELM.

METHODS

This study uses the CNN-ELM method to classify brain tumor MRI images. Based on the literature study discussed earlier, the author can arrange the research methods carried out in this study. The research method is presented in Figure 1.

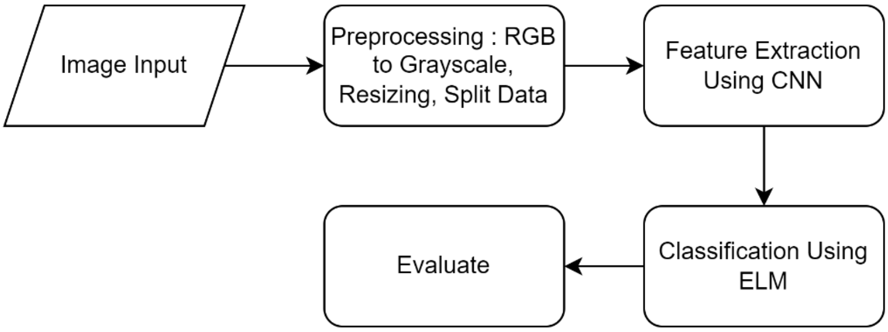

Figure 1. Methods

Data Collection

The dataset used in this study comes from the Kaggle platform. The dataset contains MRI scan images of brain tumors with several classes. It has 4 classes namely Glioma, Meningioma, Pituitary, and Non Tumor. There are 2 types of data, 5712 training data images and 1311 testing data images.

Image Preprocessing

All data will be preprocessed first before proceeding to the next step. There are several preprocessing steps such as grayscaling, resizing, and data splitting. In the grayscaling step, the image with RGB format is converted into grayscale format. This step helps to reduce the color channels of an image. RGB images with 3 color channels, namely Red (red), Green (green), and Blue (blue) will be converted into grayscale images with 1 color channel. The grayscaling process in this study uses the grayscale formula from scikit-image, namely $Y = 0.2125 * R + 0.7154 * G + 0.0721 * B$. The images in the dataset may have different sizes. Therefore, resizing is required so that the dataset images are of the same size. The difference in the image size of the dataset will affect the model training to be not optimal. In this study, the size used is 128x128 pixels. Then, split the testing data into two parts, validation data and testing data with a ratio of 80% validation data and 20% testing data.

CNN Feature Extraction

A CNN model was built to perform feature extraction on the dataset. This study uses a CNN model with five convolution layers. The first convolution layer has 16 filters with a size of 3x3 and accepts an input image of 128x128 and one color channel. There are four other convolution layers after the first layer with 32, 64, 128, and 256 filters and 3x3 filter sizes. There are five pooling layers after each convolution layer with a size of 2x2. Then there is a flatten layer which is useful for converting the output of the convolution layer into a one-dimensional vector. The vector will be used as input for the ELM model to perform classification. The CNN model is trained with 10 epochs and a learning rate of 0,001 to get optimal feature extraction results. The structure of the CNN model can be seen in Figure 2.

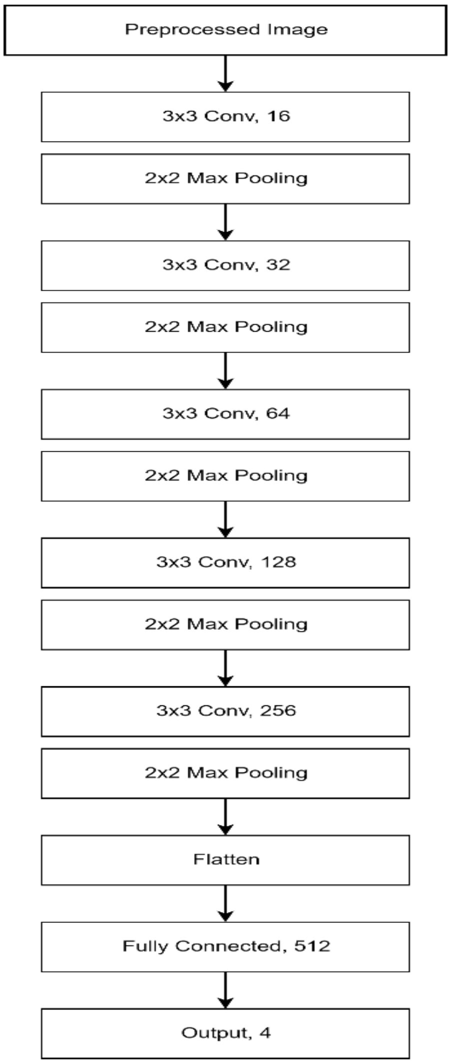

Figure 2. CNN Architecture

ELM Classification

The ELM model is used for final classification with input features that have been extracted by CNN. There are three layers in the ELM model: input layer, hidden layer, and output layer. The input layer receives a one-dimensional vector generated by the flatten layer in the CNN model. The vector generated by the flatten layer is a representation of the features extracted by the convolution and pooling layers. After that, the hidden layer randomly initializes the weights and biases. The weight and bias values will not change during training. In addition, the configuration of the number of nodes is manually initialized as needed. The activation function used in the hidden layer is a sigmoid function. The sigmoid function is applied after the product of the input features and weights is added with a bias. The pseudoinverse value is calculated after the sigmoid activation function is applied. The output layer has the same number of nodes as the number of classified categories. Thus, the ELM model can utilize the features extracted by CNN for classification. The algorithm of the ELM model can be seen in Figure 3.

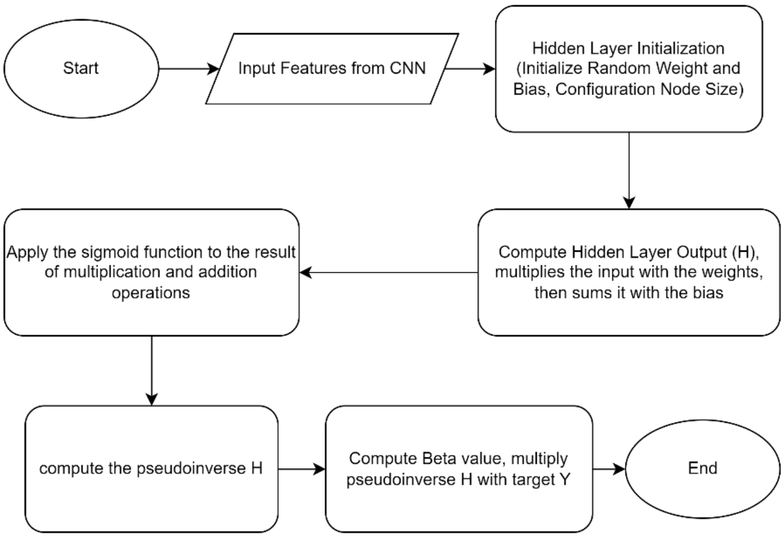

Figure 3. ELM Algorithm

Evaluation

Evaluation of the model is carried out to determine the extent to which the model can perform classification. The evaluation is done by referring to the precision, recall, F1-score, and accuracy values. There are 4 study scenarios, with node sizes of 1000, 2000, 3000, and 4000.

RESULTS AND DISCUSSION

Preprocessing

The first preprocessing is grayscaling, using the formula from scikit-image. Although the image already looks grayscale, the color channel in the image is still RGB. Therefore, the grayscale process is needed to reduce the color channel from RGB to grayscale. After that, the image is resized to 128x128 to have the same size during training. Then, the testing data is divided into two parts, namely validation and testing data with a ratio of 80% validation and 20% testing. Figure 4 is the result of the preprocessing that has been done.

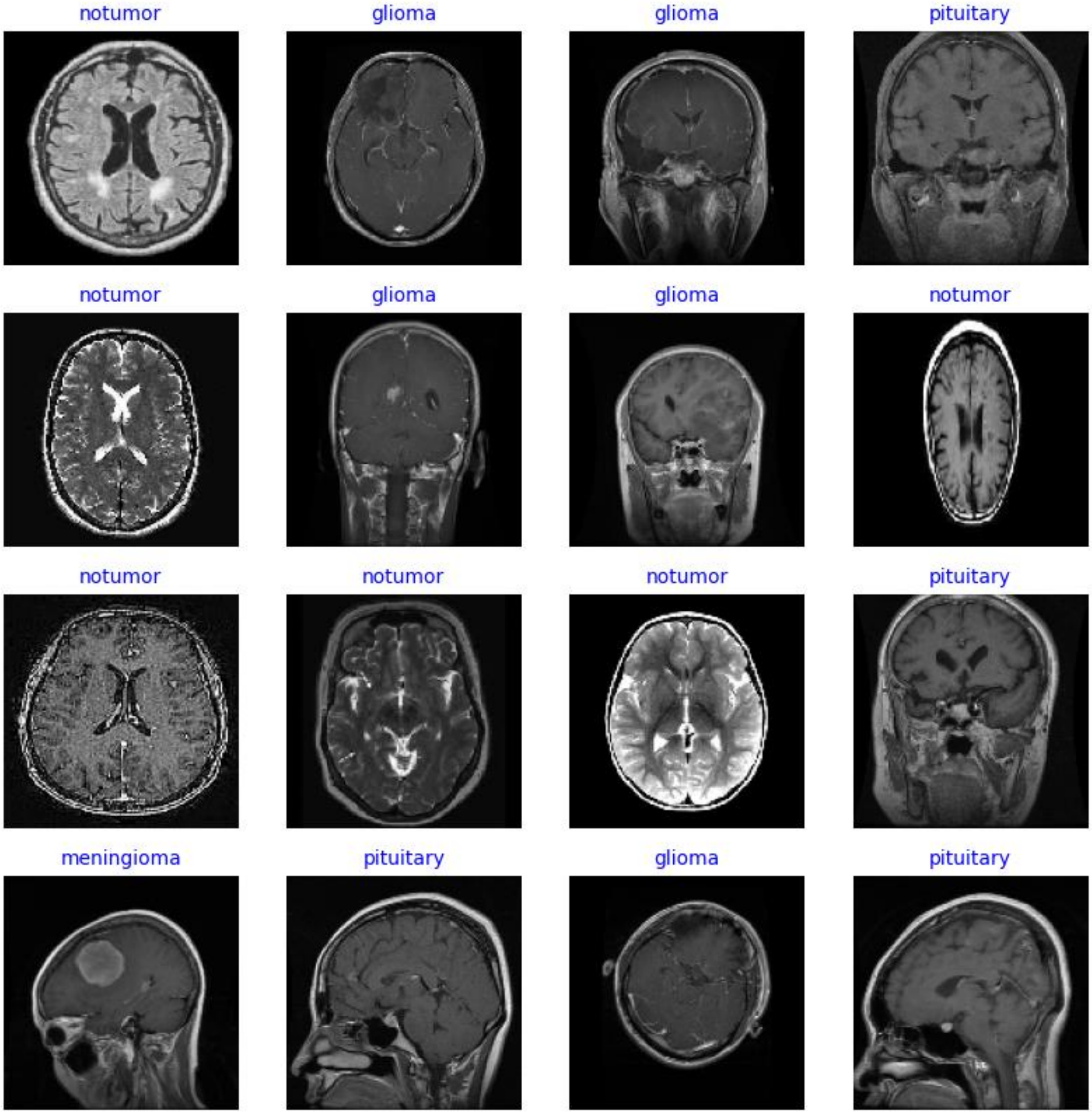

Figure 4. Preprocess Result

Training CNN-ELM Model

In this study, the CNN-ELM method is used to classify brain tumor MRI images. The CNN model with five convolution layers is trained first with 10 epochs to get optimal feature extraction results. After that, the features that have been obtained by the flatten layer in the CNN model will be processed by ELM for classification. Each image has 1024 features obtained after passing through the flatten layer. The input features are multiplied by the weights and then added with a bias. Then the sigmoid activation function is applied to the result. After sigmoid activation, a pseudoinverse calculation is performed to overcome the problem where the hidden layer output matrix has no solution. After that, the result of the pseudoinverse calculation is multiplied by the label to get the beta value used for prediction. Figure 5 is a visualization of feature extraction on a data sample.

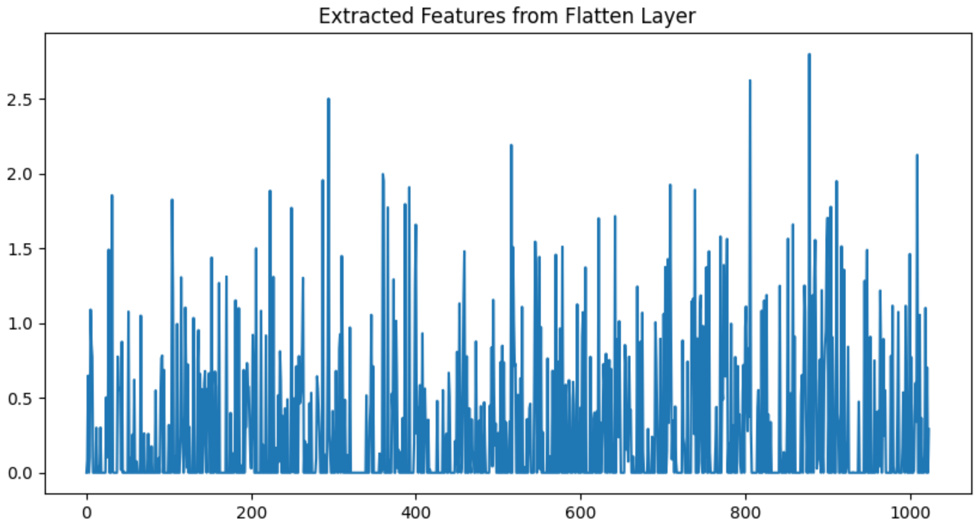

Figure 5. Extracted Features Flatten Layer CNN

Model Evaluation

Table 1 shows the model evaluation results. The evaluation used precision, recall, F1-score, and accuracy metrics to measure the reliability of the model. In the model with 1000 hidden nodes, the Glioma, Non-Tumor, and Pituitary categories scored high F1-score metric values ranging from 97% to 98%. While the Meningioma category has a lower F1-score metric value of 94%. This is due to the lower value of the recall metric in the Meningioma category of 92%, indicating that the model with 1000 hidden nodes is less effective in detecting all Meningioma cases in the data. Overall, the accuracy of the 1000 hidden nodes model gets a value of 96.96%.

Table 1. Model Evaluation

Model	Kategori	Presisi	Recall	F1-score	Akurasi
CNN-ELM 1000 Hidden Node	Glioma	97%	98%	98%	96,96%
	Meningioma	97%	92%	94%	
	Non Tumor	96%	100%	98%	
	Pituitary	98%	97%	97%	
CNN-ELM 2000 Hidden Node	Glioma	98%	97%	97%	98,86%
	Meningioma	97%	98%	98%	
	Non Tumor	100%	100%	100%	
	Pituitary	100%	100%	100%	
CNN-ELM 3000 Hidden Node	Glioma	98%	95%	97%	98,10%
	Meningioma	98%	97%	98%	
	Non Tumor	99%	100%	99%	
	Pituitary	97%	100%	98%	
CNN-ELM 4000 Hidden Node	Glioma	92%	95%	93%	96,58%
	Meningioma	94%	95%	94%	
	Non Tumor	100%	100%	100%	
	Pituitary	100%	95%	97%	

In the model with 2000 hidden nodes, the Non-Tumor and Pituitary categories scored a perfect F1-score metric of 100%. While the Glioma F1-score metric value gets 97% and Meningioma gets 98%. Overall, the accuracy of the model with 2000 hidden nodes increased from the 1000 hidden nodes model with an accuracy metric value of 98.86%. In the model with 3000 hidden nodes, the F1-score metric value in the Non Tumor and Pituitary categories decreased slightly to 99% in the Non Tumor category and 98% in the Pituitary category. Overall, the accuracy of the 3000 hidden node model has decreased with an accuracy metric value of 98.10%.

In the model with 4000 hidden nodes, the F1-score metric value in some categories dropped more. The F1-score metric in the Glioma category is 93%, the Meningioma category is 94%, and the Pituitary is 97%, while Non Tumor increases to get a perfect value of 100%. But overall, the model with 4000 nodes scored the lowest accuracy metric of 96.58%.

CONCLUSION

In this study, it can be concluded that increasing the number of hidden nodes in the CNN-ELM model has a varying impact on model performance. The model with 1000 hidden nodes showed good performance with an overall accuracy of 96.96%, however, it had difficulty in detecting the Meningioma category, which was reflected in the lower recall value (92%). Increasing the number of hidden nodes to 2000 increased the accuracy of the model to 98.86%, with significant improvement in the F1-score value for the Meningioma category, which rose to 98%, and perfect F1-score for the Non-Tumor and Pituitary categories.

However, in the model with 3000 hidden nodes, there was a decrease in F1-score values in some categories, such as Non Tumor and Pituitary, although the overall accuracy remained high at 98.10%. In the model with 4000 hidden nodes, the performance degradation became more obvious, with a decrease in F1-score values in the Glioma, Meningioma, and Pituitary categories, and the overall accuracy of the model dropped to 96.58%, which was the lowest value among the evaluated models.

In conclusion, while increasing the number of hidden nodes can improve the performance of the model up to a certain point, too many hidden nodes can actually cause a decrease in performance. The model with 2000 hidden nodes showed the most consistent and optimal performance in detecting various tumor categories, including Meningioma, compared to the other models.

REFERENCES

Chaki, J., & Wozniak, M. (2023). Brain Tumor Categorization and Retrieval Using Deep Brain Incep Res Architecture Based Reinforcement Learning Network. *IEEE Access*, *11*, 130584–130600. https://doi.org/10.1109/ACCESS.2023.3334434

Noreen, N., Palaniappan, S., Qayyum, A., Ahmad, I., Imran, M., & Shoaib, M. (2020). A Deep Learning Model Based on Concatenation Approach for the Diagnosis of Brain Tumor. *IEEE Access*, *8*, 55135–55144. https://doi.org/10.1109/ACCESS.2020.2978629

Purbasari, I. Y., Rahmat, B., & Putra PN, C. S. (2021). Detection of Rice Plant Diseases using Convolutional Neural Network. *IOP Conference Series: Materials Science and Engineering*, *1125*(1), 012021. https://doi.org/10.1088/1757-899x/1125/1/012021

Sheergojri, A. R., & Iqbal, P. (2022). MRI Brain Tumor Classification and Uncertainty Estimation using Hybrid CNN-ELM Model. *3rd International Conference on Smart Electronics and Communication, ICOSEC 2022 - Proceedings*, 978–983. https://doi.org/10.1109/ICOSEC54921.2022.9952006

Wahid, R. R., Anggraeny, F. T., & Nugroho, B. (2021). Brain Tumor Classification with Hybrid Algorithm Convolutional Neural Network-Extreme Learning Machine. *IJCONSIST JOURNALS*, *3*(1), 29–33. https://doi.org/10.33005/IJCONSIST.V3I1.53