A Lightweight Multi-Scale ResNet34 Framework for Brain Tumor MRI Image Classification

Yufeng Li
University of Southampton
Southampton, UK

Wenchao Zhao
University of Science and Technology of China
Anhui, China

Weimin Wang
The Hong Kong University of Science and Technology
Hong Kong, China

Bo Dang
Computer Science, San Francisco Bay University
Fremont, CA, US

ABSTRACT

Accurate and efficient image classification is a critical requirement for intelligent medical systems, especially under performance and resource constraints. Traditional manual interpretation and shallow learning models are often inefficient and lack sufficient classification accuracy. To address these challenges, this paper proposes a multi-scale deep learning framework based on an improved ResNet architecture for medical image classification, achieving high performance. In the proposed framework, By introducing a multi-scale feature extraction module at the network input and combining it with a residual downsampling structure based on Inception-v2, the ResNet34 backbone network's ability to represent and extract multi-scale features is effectively improved. Furthermore, a channel attention mechanism is introduced to adaptively reweight feature channels, enabling the model to emphasize more informative feature representations while suppressing redundant features. Experimental evaluation employs five-fold cross-validation using a brain tumor image dataset. Experimental results show that the framework achieves an average accuracy of approximately 98.8% while maintaining high classification performance, which is about 1% higher than the original ResNet34, and the number of model parameters is reduced to about 80% of the baseline model. These results demonstrate that the proposed method effectively improves classification performance while enhancing computational efficiency, making it suitable for performance-constrained intelligent image analysis systems.

General Terms

Algorithms, Performance, Experimentation, Medical Image Classification.

Keywords

Performance-efficient deep learning; Medical image classification; Lightweight neural networks; Multi-scale feature extraction; Intelligent systems; Channel attention mechanism

1. INTRODUCTION

With the accelerating pace of social development and changes in lifestyles, health-related issues are becoming increasingly prominent. Among them, brain tumors, as a serious disease, are gradually becoming a health hazard troubling society. Brain tumors refer to a type of tumor that forms in the human brain tissue. They can affect a person's cognition, behavior, perception, and motor abilities, and their mortality rate accounts for 2.4% of the human tumor incidence rate[1]. There are many types of brain tumors, the most common of which are gliomas, meningiomas, and pituitary tumors[2]. The most commonly used method for diagnosing brain tumors in clinical practice is magnetic resonance imaging (MRI)[3]. MRI has multi-sequence and multi-type imaging capabilities, and can clearly display the soft tissue structure of the human body, providing richer imaging information for clarifying the nature of the lesion. Therefore, how to efficiently and accurately determine whether an MRI image contains a tumor and what type of tumor it contains is an urgent clinical problem to be solved. From an engineering perspective, the increasing volume and complexity of medical imaging data also pose significant challenges to the performance, efficiency, and reliability of intelligent image analysis systems.

Early MRI imaging relied on manual visual examination of brain MRI images. Due to the large data volume and the varying types of brain tumors, this method was time-consuming and prone to human error, making computer-aided diagnosis necessary[4]. In recent years, convolutional neural networks (CNNs) have achieved good results in tasks such as image processing, speech recognition, natural language processing, object tracking, and medical diagnosis due to their powerful feature representation capabilities[4]. Along with this research wave, more and more scholars have introduced CNNs into the field of medical imaging in recent years. Hossain et al.[4] studied brain tumor detection using both traditional classifiers and deep learning-based CNNs. Their results showed that the detection performance using CNNs was significantly better than that of traditional classifiers. Abiwinanda[7] also applied CNNs to brain tumor classification, achieving a three-classification task of glioma, meningioma, and pituitary adenoma using a simple convolutional network. Although the accuracy was slightly lower, it was still comparable to the accuracy of traditional algorithms based on region preprocessing. Vankdothu et al.[8] combined CNNs with Long Short-Term Memory (LSTM) to construct a novel convolutional network, achieving a 92% accuracy rate in classifying gliomas, meningiomas, pituitary adenomas, and tumor-free images on the Kaggle brain tumor dataset. Ayadi et al.[9] constructed an 11-layer convolutional CNN to classify brain tumor images from the Figure dataset into three categories, achieving 94.74% accuracy. These models are shallow CNNs and do not yet reach the level of deep CNNs. Khan et al.[10] used a simple CNN model to classify benign and malignant brain tumors with 100% accuracy, but due to the small sample size (only 253 images), it was difficult to overcome the impact of the small sample size on the model's generalization performance.

As CNNs are applied to various fields, researchers have found that with increasing convolutional layer depth, gradient vanishing or exploding phenomena occur, leading to a significant drop in accuracy. Residual networks can mitigate this phenomenon. Kumar et al.[11] used residual networks and global average pooling to perform three-class classification on a brain tumor dataset, replacing fully connected layers with global average pooling to alleviate overfitting and achieving an accuracy of 97.48%. ResNet[12] residual networks can train very deep neural networks, effectively mitigating gradient vanishing and exploding during training, accelerating network convergence, and improving the model's expressive power and performance. The residual connections preserve original features, making network learning smoother and more stable, further improving the model's accuracy and generalization ability. However, ResNet extracts features at a single scale, without integrating features extracted from convolutional kernels at multiple scales. In CNNs, large convolutional kernels have a larger effective receptive field and higher shape bias, while small convolutional kernels focus more on texture bias[13]. Therefore, combining convolutional kernels of different sizes to process input data at multiple scales can appropriately improve model performance. Diaz et al.[14] applied a multi-scale approach to deep CNNs, performing a three-class classification on 3064 MRI images from a publicly available brain tumor dataset, achieving a classification accuracy of 97.3%. Multi-scale residual networks further extend standard residual blocks by splitting feature channels into several groups and connecting them hierarchically, so that features at different receptive-field scales can be learned within a single block[14]. Furthermore, when processing input data, attention mechanisms can filter high-value information from a large amount of data, giving more attention to key information, enabling the model to better represent key features, while also reducing sensitivity to noise and redundant features, and improving the model's generalization ability.

To address the problems of time-consuming and labor-intensive traditional manual classification of brain tumor medical images, low classification accuracy of shallow CNN models, and gradient vanishing issues in stacked deep networks, this paper combines the advantages of multi-scale concepts and attention mechanisms to study a deep residual network that can alleviate the gradient vanishing problem. The main research aspects are: 1) Using a ResNet34 residual network as the backbone network, where the residual modules can effectively alleviate the gradient vanishing problem in deep networks; 2) By introducing a multi-scale feature extraction module into the backbone network, the features of the first convolutional layer and the downsampling layer are extracted and stitched together at multiple scales to obtain richer feature information and improve the feature utilization rate in the downsampling stage; 3) Finally, adding an SE channel attention mechanism module to the network to improve the representation ability of important features. Through these studies, a new network is proposed to improve the accuracy of brain tumor classification.

2. NETWORK STRUCTURE

2.1 ResNet34 Residual Structure

ResNet34 is one of the implementations of the ResNet series of networks. The network architecture of this model mainly includes convolutional layers, pooling layers, residual blocks, and fully connected layers. Residual blocks are a key component of ResNet, and their fast connection structure allows shallow features to be directly mapped to deeper layers of the network. This shortcut connection design helps alleviate the accuracy decline problem that occurs in deep neural networks as the number of layers increases. The structure of the residual block is shown in Figure 1. Its core idea is to achieve residual learning through cross-layer connections, which makes it easier for the network to learn residual information during training, thereby improving model performance and convergence speed. The innovation of this architecture has greatly promoted the development of deep convolutional neural networks, enabling us to build deeper neural networks without being constrained by performance degradation.

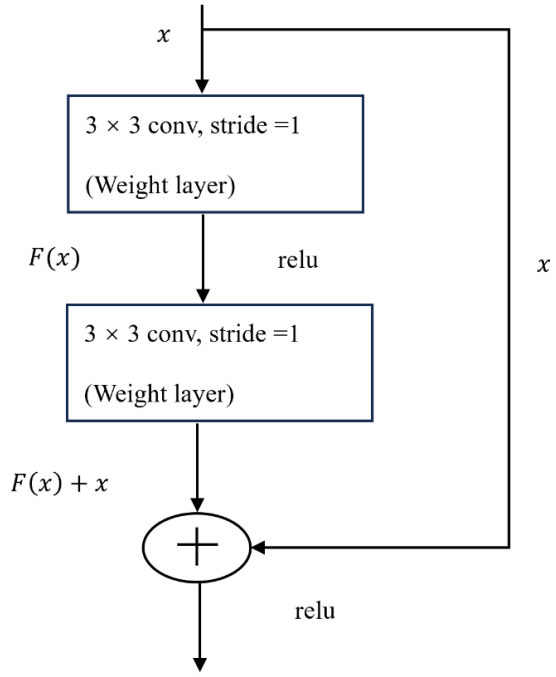


Figure 1 Residual block structure diagram

In Figure 1, the right branch x represents the identity mapping of the residual structure, indicating that pixels are added one by one. The formula is as follows:

$$H(x) = F(x) + x \quad (1)$$

In the formula, $H(x)$ is the bottom layer mapping, F represents the computation process through two convolutional layers (including ReLU activation operation), and x is the input.

The residual mapping $F(x) = H(x) - x$, when the network reaches relative saturation, that is, when F(α) is small enough, the output $H(x)$ will approximately become α, which becomes the identity mapping function $H(x)= x$. The identity mapping ensures that the network performance will not decrease, so that the network can learn new features based on the input features. Moreover, the identity mapping will not add extra parameters and computation to the network, but can speed up the training speed of the model and optimize the training effect[15]. Therefore, as the network depth increases, the gradient will not disappear in backpropagation, which makes the residual network widely used in many fields.

2.2 Improved ResNet34 Network Model

To improve the ResNet34 network and overcome some of its limitations, this study optimized ResNet34. The improved network model still uses ResNet34 as the backbone network structure, but the architecture has been adjusted. First, the first 7×7 convolutional layer of ResNet34 is replaced with a multi-scale input module. This module can use convolutional kernels of various sizes to extract features from the image, thereby

improving the richness and diversity of feature information. Second, the residual downsampling module in ResNet34 is replaced with the downsampling module proposed in this paper. This novel downsampling module not only introduces the multi-scale concept but also makes full use of all input information, further enhancing the model's performance. Finally, to further improve the network's performance, an SE attention mechanism is introduced in each residual block, which helps the network to learn and utilize key features more effectively. The network structure is shown in Figure 2.

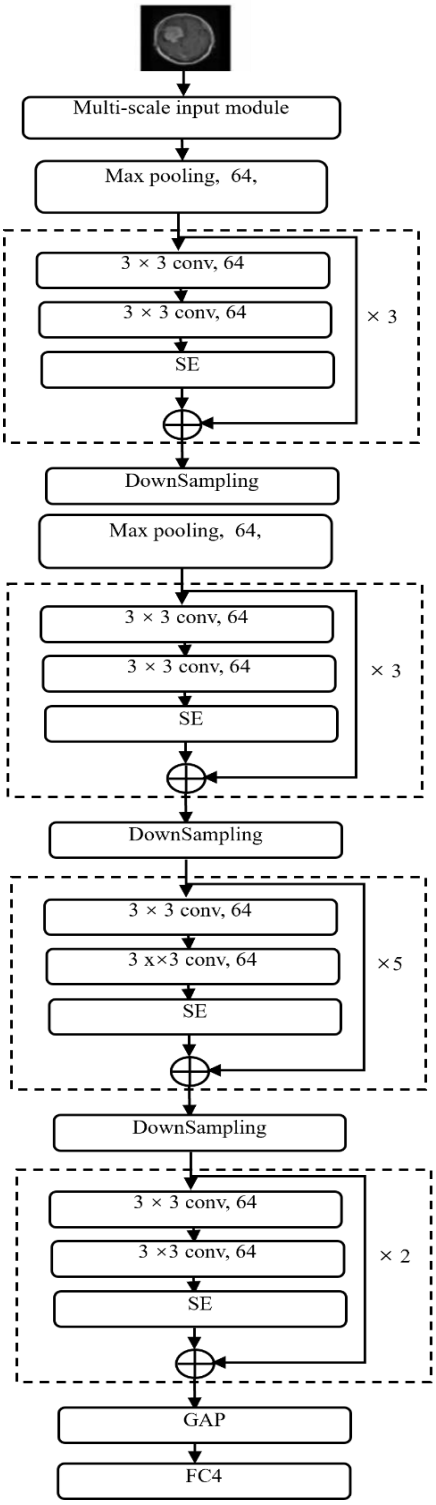


Figure 2 Improved ResNet34 network architecture diagram

2.3 Downsampling Module

Downsampling is an operation used to reduce the size or resolution of an image or signal, thereby reducing the computational cost of the model. In the ResNet34 network, in addition to using a max-pooling downsampling layer with a kernel size of 3×3 and a stride of 2, a residual-based downsampling module is also included, as shown in Figure 3. It is important to note that the shortcut connection of the downsampling module in Figure 3 is a convolutional layer with a kernel size of 1×1 and a stride of 2. This means that the input features passing through this convolutional layer will experience information loss, which may include some important information. To address this issue, combining the ideas of multi-scale and the Inception module, this paper proposes using the Inception v2 module as the downsampling module, changing the original 1×1 convolutional kernel branch to a 3×3 convolutional kernel. The network structure is shown in Figure 4. As shown in Figure 4, "channels" represent the number of channels, "H" and "W" represent the spatial size of the input feature map, and "stride" is the convolution stride. When the "stride" is set to 2, a downsampling operation is performed to reduce the feature map size to half of its original size. This downsampling module has four branches. After each branch completes its operation, channels are concatenated. The final output channel count of each branch is the same, and the concatenated channel count is twice the input channel count.

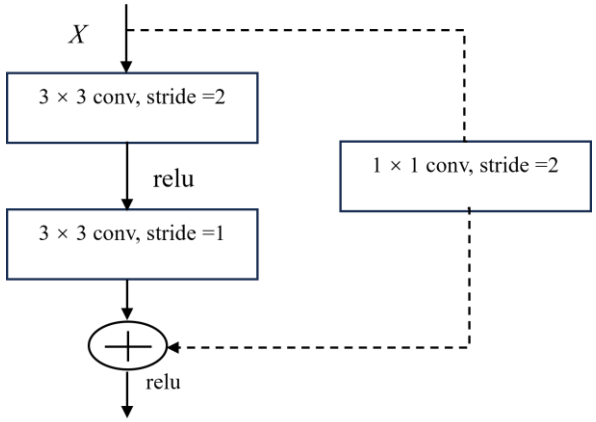


Figure 3 Differential block downsampling module

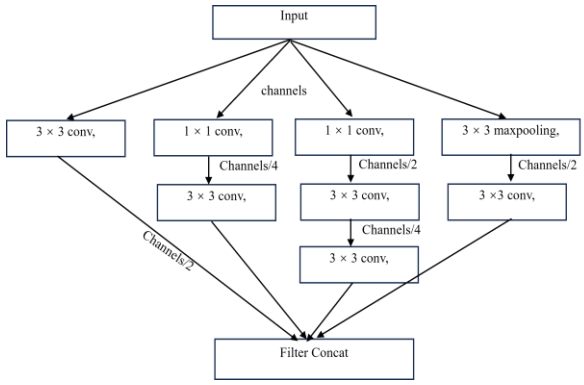


Figure 4 Inception v2 downsampling module

2.4 Multi-Scale Input Feature Extraction Module

Different sizes of convolutional kernels exhibit differences in feature extraction. Large convolutional kernels primarily focus on shape information, while small convolutional kernels emphasize capturing texture details[13]. Multi-scale feature extraction utilizes convolutional kernels of different scales to

extract features, and obtains richer feature representations through feature fusion. This paper employs a multi-scale input feature extraction module, which extracts features from the input data through a multi-branch convolutional structure. Each branch uses convolutional kernels of different scales (3×3, 5×5, 7×7, and 11×11) to enhance the model's ability to perceive features at different spatial scales. Then, these four outputs are added pairwise to generate two intermediate outputs. Next, these two intermediate outputs are passed through two convolutional layers with kernel sizes of 3×3 and 5×5, respectively, and finally merged by channel concatenation. This multi-scale feature extraction strategy helps to capture various feature information in the image more comprehensively. The specific structure is shown in Figure 5, where 32 and 64 represent the number of output channels, respectively.

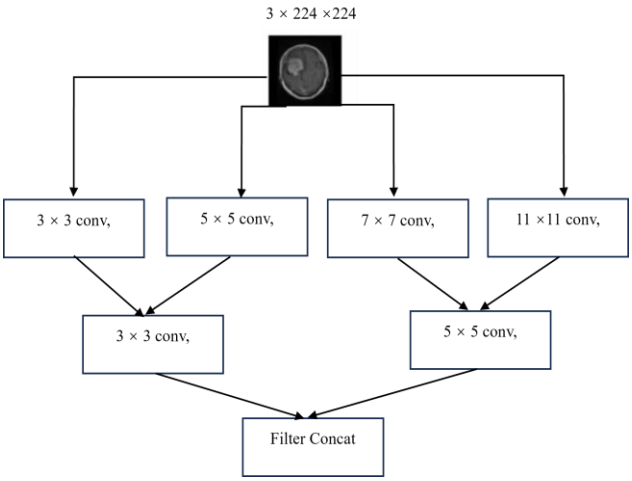


Figure 5 Multi-scale input module

2.5 Channel Attention Mechanism

During convolution and pooling, it is assumed that each channel of the feature map has the same importance. However, in real-world problems, the importance of features in different channels may differ. This paper introduces the SE channel attention mechanism to adaptively weight each channel of the feature map, thereby more effectively capturing the difference information between channels [16]. It uses a weight matrix to assign different weights to different channels of the image from a channel domain perspective, highlighting more important feature information. SE mainly consists of three parts: squeezing, excitation, and scaling.

The squeezing part uses global average pooling (GAP) to compress the size of the input feature map from H×W×C to 1×1×C. This involves averaging all pixels in each channel of the feature map to obtain a unique value, as shown in the following formula:

$$Z_c = F_{sq}(u_c) = \frac{1}{H\ X\ W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c\ (i,j) \qquad (2)$$

During the compression stage, the spatial dimensions of the input feature map are compressed using global average pooling to obtain a global description of each channel. In the formula, represents the global compressed feature of the c-th channel of the input feature map, represents the compression process, represents the feature map of the c-th channel of the input feature map, and and represent the height and width of the input feature map, respectively.

During the activation phase, the feature set obtained in the compression phase is input into a nonlinear mapping structure consisting of two fully connected layers and an activation function to learn the dependencies between channels and generate a channel weight vector . Different values in correspond to different importance weights for each channel, used to adaptively weight each channel of the original feature map, thereby highlighting key information and suppressing redundant features. The corresponding calculation process is shown in the formula.

$$s = F_{ex}(z, W) = \sigma[W_2 \delta(W_1\, z)] \qquad (3)$$

In the formula, represents the activation process; represents the output obtained from the compression part; represents the fully connected layer, and represent the first and second fully connected layers respectively, represents the ReLU activation layer, and represents the sigmoid activation function.

The reset part multiplies the weight value s of each channel obtained from the activation part with the input before the compression step for the corresponding channel, so as to assign different weights to different channels. The formula is as follows:

$$\tilde{X}_c = F_{scale}(u_c\ , S_c) = S_c u_c \qquad (4)$$

In the formula, $\tilde{X}_c = [\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \ldots \tilde{x}_c]$, $\tilde{x} = s_1 u_1$, F_{scale} represents the reset process, u_c is the input feature, and S_c represents the output weights obtained after the input feature u_c undergoes compression and activation processes.

3. EXPERIMENT AND ANALYSIS

3.1 Experimental environment and model parameter settings

The PyTorch deep learning model framework was used in this experiment. The programming language was Python 3.14, and the compiler was PyCharm. The operating system is Windows 10, the CPU is an Intel i7-9750H, and the GPU is an NVIDIA RTX 2080 Ti to utilize the powerful graphics processing unit for computation. During network training, the number of epochs was set to 300, and the batch size for each epoch was set to 64. The Adam optimization algorithm was used, with an initial learning rate of 0.001. A linear decay strategy for the learning rate was adopted, meaning the learning rate was reduced to 0.1 times its original value every 100 epochs to accelerate model convergence and alleviate oscillations and overfitting.

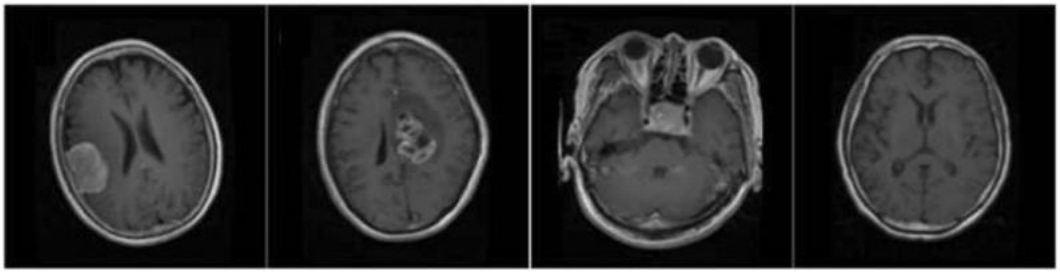
3.2 Dataset and Preprocessing

The brain tumor MRI dataset used in this experiment consists of 3064 brain tumor MRI images[17] from the FigShare website and 1436 tumor-free MRI images[18] from a publicly available no-tumor MRI dataset. The use of public datasets in research also highlights the importance of data accessibility, standardization, and interoperability, which are crucial for supporting reproducible and scalable intelligent data analysis[19]. The FigShare dataset contains 3,064 brain tumor images from 257 patients, covering three types of brain tumors: meningioma (708 images), glioma (1,426 images), and pituitary adenoma (930 images). The tumor type in each slice was labeled by the radiologist. This dataset is one of the most widely used public datasets for current brain tumor classification tasks. It is provided in Matlab format, with each file storing a structure containing image information, such as the relationship between tumor type and ground truth labels (where labels 1, 2, and 3 correspond to meningioma, glioma, and pituitary adenoma, respectively). Figure 6 shows the different types of brain tumors and tumor-free MRI images after converting the Matlab format files to image files.

After converting the entire dataset into image files, one-fifth of them were randomly selected as the test set. Subsequently, the remaining data was randomly divided into training and validation sets in a 4:1 ratio, as shown in Table 1. Before feeding the dataset into the training network, preprocessing was required. First, each image was randomly cropped and resized to a fixed 224×224 size for easy input into the network. Next, the cropped images were standardized to eliminate scale differences and improve the stability of gradient descent. Given the small size of the dataset, data augmentation strategies are introduced to increase data diversity in order to alleviate the problem of insufficient samples[19]. The input images were randomly flipped horizontally, vertically, and brightened to ensure that the data input to the network was not identical in each training cycle. This mitigated potential overfitting during model training and improved the model's generalization performance.

Table 1 Dataset distribution

Tumor types	Training set	Validation set	Test set	Total
Meningioma	454	113	141	708
Glioma	913	228	285	1426
Pituitary	596	148	186	930
No tumor	920	229	287	1436



(a) Meningioma (b) Glioma (c) Pituitary tumor (d) No tumor

Figure 6 Images of tumors and tumor-free areas

3.3 Evaluation Metrics

Evaluation metrics are used to measure the quality of model training. This paper adopts the following evaluation metrics as the standard for experimental results:

1) Accuracy: Represents the proportion of samples in the test results where the predicted value is the same as the true value, out of the total sample. The formula is as follows:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (5)$$

2) Precision: Represents the percentage of true positive samples out of all samples predicted to be positive in the test results. The formula is as follows:

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

3) Recall: Represents the percentage of samples where the true value is positive, and also the percentage of samples where the predicted value is positive. The formula is as follows:

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

4) Specificity: Represents the proportion of samples with negative predicted values (also negative) among samples with negative true values in the test results. The formula is as follows:

$$Specificity = \frac{TN}{FP+TN} \quad (8)$$

5) $F_{1\,score}$: Represents a weighted average of precision and recall. A higher $F_{1\,score}$ indicates a more robust model. The formula is as follows:

$$F_{1\,score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (9)$$

In the above formulas, TP represents the number of positive class samples correctly classified as positive, i.e., the number of positive class samples successfully identified by the model; TN represents the number of negative class samples correctly classified as negative, i.e., the number of negative class samples successfully identified by the model; FP represents the number of negative class samples incorrectly classified as positive, i.e., the number of negative class samples incorrectly labeled as positive; and FN represents the number of positive class samples incorrectly classified as negative, i.e., the number of positive class samples incorrectly labeled as negative.

3.4 Experimental Results

To further investigate the effectiveness of the proposed method and the impact of the improved modules on the accuracy of the original ResNet34 network, this paper combines the ResNet34 network with the Inception v2 module, the multi-scale input module, and the attention mechanism module, respectively. An ablation study was used to verify the integration of these modules into the ResNet34 network, and the network was trained on the training dataset. The performance of these five models was evaluated on the test dataset, and the average test results are shown in Table 2. Table 3 lists other evaluation metrics for each model. This experimental design aims to comprehensively evaluate the impact of different modules on model performance, providing profound insights for model optimization.

Table 2 Average test accuracy of the ablation models

Model	ResNet34	A	B	C	Accuracy/%
1	√	×	×	×	97.71
2	√	√	×	×	98.33
3	√	×	√	×	98.22
4	√	×	×	√	98.28
5	√	√	√	√	98.82

In Table 2, A represents the Inception v2 module, B represents the multi-scale input module, and C represents the SE attention mechanism module. Model 1 represents a ResNet34 network only; Model 2 represents a new network formed by replacing the downsampling layer in ResNet34 with the Inception v2

module; Model 3 represents a new network formed by combining ResNet34 with the multi-scale input module; Model 4 represents a new network formed by adding the SE attention mechanism module to ResNet34; and Model 5 represents a new network formed by incorporating modules A, B, and C (a total of three modules) into ResNet34. The average accuracy results of these five different network architectures on the test set show that the improved networks with added modules all have higher accuracy in brain tumor image classification than the original networks, indicating that these modules improve the performance of the original model in brain tumor image classification. Among them, the network model integrating three modules has the best accuracy, reaching 98.82%, which is about 1.1 percentage points higher than the ResNet34 network.

Table 3 Average evaluation index of each model (%)

Model	Type	Precision	Recall	Specificity	F1-score
1	Glioma	98.18	98.25	99.15	98.21
1	Meningioma	94.06	93.9	98.89	93.97
1	No tumor	100	99.3	100	99.65
1	Pituitary	96.28	97.31	99.02	96.79
1	Average	97.13	97.19	99.27	97.16
2	Glioma	97.92	99.12	99.02	98.52
2	Meningioma	96.08	95.74	99.27	95.91
2	No tumor	100	99.48	100	99.74
2	Pituitary	98.1	97.31	99.51	97.71
2	Average	98.03	97.91	99.45	97.97
3	Glioma	98.94	98.42	99.51	98.68
3	Meningioma	94.41	95.75	98.95	95.07
3	No tumor	100	99.83	100	99.91
3	Pituitary	97.31	97.31	99.3	97.31
3	Average	97.67	97.83	99.44	97.74
4	Glioma	98.44	99.12	99.1	98.6
4	Meningioma	96.09	95.75	99.27	95.92
4	No tumor	100	98.95	100	99.47
4	Pituitary	97.59	97.85	99.37	97.72
4	Average	98.03	97.92	99.44	97.93
5	Glioma	97.93	98.88	99.74	99.16
5	Meningioma	96.12	97.87	99.26	96.98
5	No tumor	100	99.79	100	99.9
5	Pituitary	97.95	97.74	99.47	97.85
5	Average	98.33	98.57	99.62	98.47

In Table 3, under the category of "type," glioma, meningioma, no tumor, and pituitary adenoma represent glioma, meningioma, no tumor, and pituitary adenoma, respectively, and "average" represents the average value of the three. Based on the detailed data analysis in Tables 2 and 3, it can be clearly observed that models 2, 3, 4, and 5 outperform model 1 in multiple performance metrics, including precision, recall, specificity, and F1 score. This result demonstrates the performance improvement of the improved network models; compared to the original network, the improved models exhibit higher levels of performance across all evaluation metrics.

To ensure the reliability and robustness of the model, a five-fold cross-validation method was used for extensive experimental verification. Figure 7 compares the training loss and validation accuracy of one of the improved models with the original ResNet34 model. The improved network model has a lower training loss and a higher validation accuracy than the original model, indicating that the training results of the improved model are closer to the true values. The confusion matrix of the test results helps to understand the classification results and performance of the model in more detail, so the confusion matrix after each validation was calculated, as shown in Figure 8. The horizontal axis in the graph represents the true class, and the vertical axis represents the class predicted by the model. The confusion matrix indicates that only a very small number of brain tumor MRI images in each class were incorrectly predicted as other classes, and the vast majority of brain tumor MRI images were correctly classified, with the prediction of no tumors being almost 100%. The confusion matrix can also be used to calculate the model's performance on multiple evaluation indicators, as detailed in Table 4.

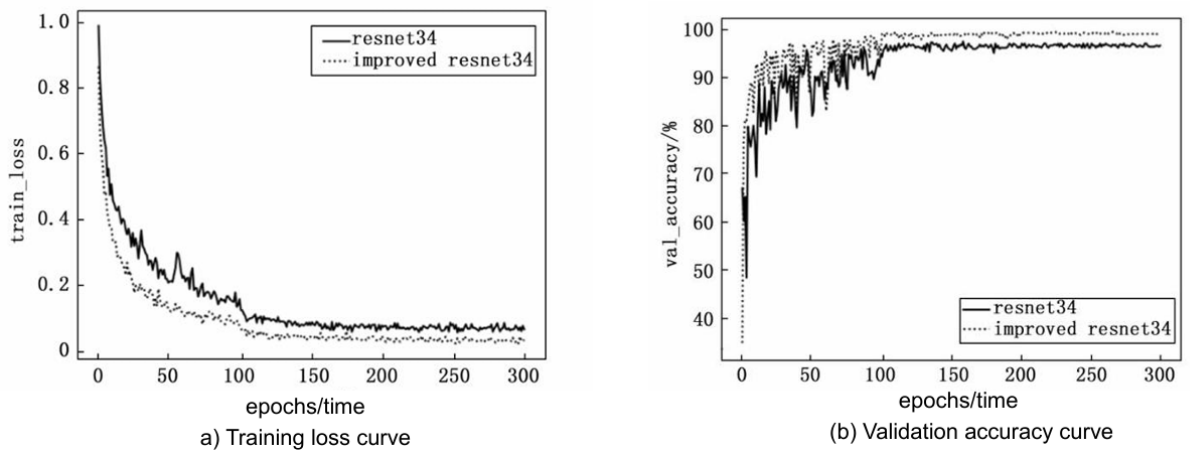


Figure 7 Training loss and validation accuracy curves of the proposed model and the ResNet34 network

Table 4 The average evaluation metric of the five-fold cross-validation of the improved network model (%)

K-Fold	Accuracy	Precision	Recall	Specificity	F1-score
1	99	98.64	98.84	99.67	98.74
2	98.67	98.15	98.58	99.59	98.35
3	98.89	98.58	98.62	99.65	98.6
4	98.67	98.17	98.44	99.59	98.3
5	98.89	98.54	98.62	99.65	98.58
Average	98.82	98.42	98.62	99.63	98.51

A thorough analysis of the data in Table 4 revealed that the improved model exhibited similar test accuracy across five cross-validations, demonstrating its performance stability. In addition to accuracy, key performance indicators such as precision, recall, specificity, and F-score were comprehensively evaluated. These indicators also showed similar results across the five validations, further confirming the reliability and robustness of the improved model. To more comprehensively evaluate the model's performance, a detailed comparison was made with several widely used classification algorithms in current academic and practical fields. To ensure fairness, all models were evaluated on the same test set. The specific experimental results are listed in Table 5.

Furthermore, this paper provides a detailed statistical analysis of the parameter counts of the new model and the ResNet34 model. The ResNet34 network model has approximately 21.3M parameters, while the proposed new model has approximately 17.3M parameters, about 80% of the ResNet34 parameter count. This indicates that the new model maintains high performance while reducing space complexity, providing feasibility for resource efficiency and model lightweighting. Table 6 compares the performance of the new model proposed in this paper with representative research methods in the field of brain tumor classification in recent years; our model exhibits relatively higher accuracy.

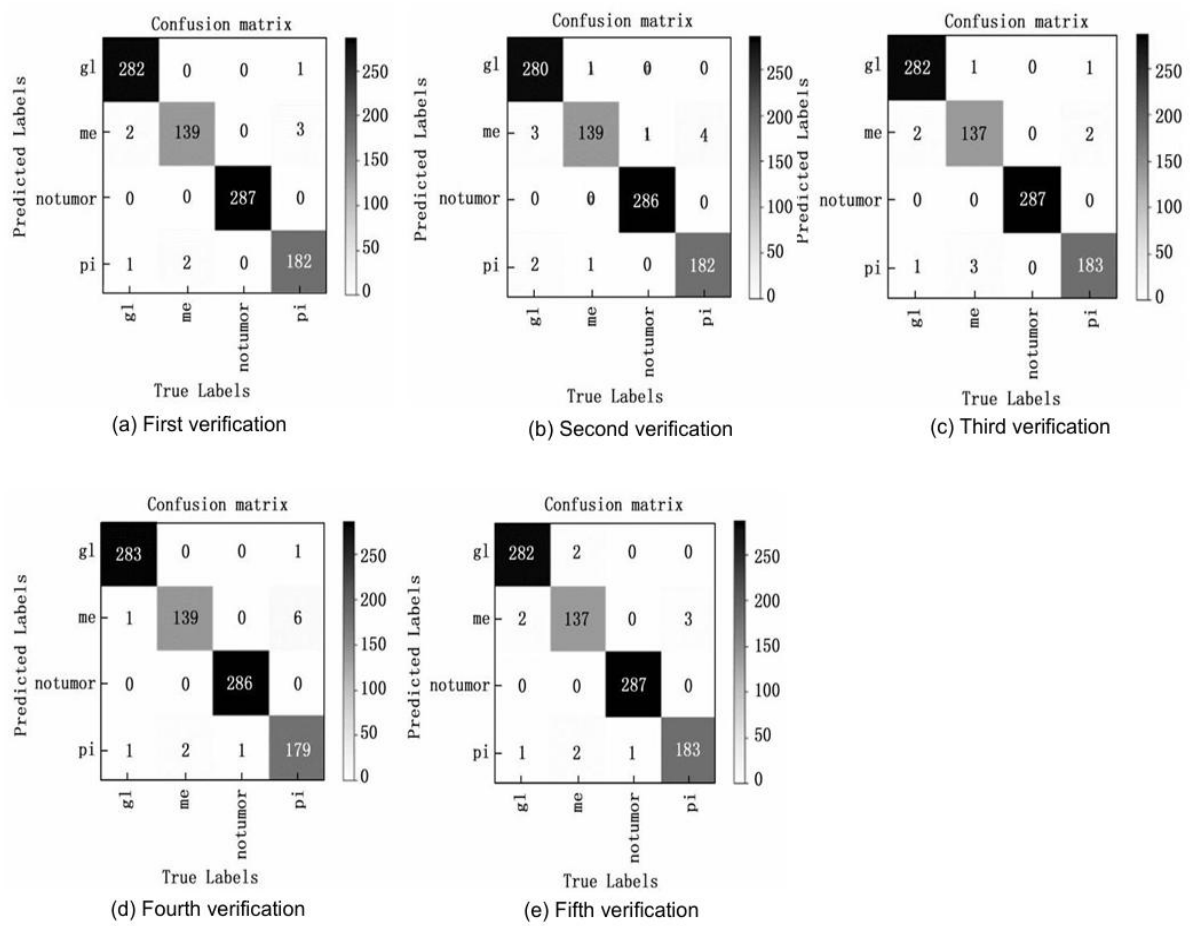


Figure 8 The prediction confusion matrix of the proposed model with 5-fold cross-validation

Table 5 Performance comparison with other classification networks (%)

Model	Accuracy	Precision	Recall	Specificity	F1-score
Vgg16	95.77	95.59	94.41	98.55	94.93
ResNet50	97.88	97.96	96.99	99.27	97.4
ResNext101	98.22	97.79	97.83	99.43	97.8
GoogleNet	97.66	97.24	96.66	99.23	97.1
EfficientNet V2	98.22	97.9	97.56	99.41	97.72
DenseNet	98.22	98.06	97.69	99.4	97.87
Ours	98.82	98.42	98.62	99.63	98.51

Table 6 Accuracy comparison of relevant studies in recent years

Author	Year	Approach	Accuracy/ %
Swati et al[21]	2019	VGG19 + Fine Tuning	94. 82
Kaplan et al[22]	2020	LBP	95.56
Kumar et al[11]	2021	ResNet-50 + Global Average Pooling	97.48
Huang et al[23]	2023	Multiscale Residual Network	98.58
ours	—	ResNet34+Multiscale Inputs + Inception v2+ SE	98.82

4. CONCLUSION

This paper proposes an efficient multi-scale deep learning framework based on an improved ResNet architecture for medical image classification. This framework enhances feature representation while effectively controlling model complexity by integrating multi-scale input processing, an Inception-v2-based residual downsampling strategy, and a channel attention mechanism.

Experimental results using five-fold cross-validation demonstrate that, compared to the standard ResNet34 model, the proposed method achieves higher classification accuracy while significantly reducing the number of network parameters. This indicates that the framework can balance classification performance and computational efficiency, which is crucial for performance-constrained intelligent systems.

This method does not rely on excessive model scaling or complex post-processing strategies, making it suitable for practical deployment scenarios where resource efficiency and system reliability are critical. Future work will focus on further evaluating the framework's generalization ability on broader datasets and exploring its application in other intelligent image analysis tasks.

5. REFERENCES

[1] XUllah, N., Khan, J. A., Khan, M. S., Khan, W., Hassan, I., Obayya, M., ... & Salama, A. S. (2022). An effective approach to detect and identify brain tumors using transfer learning. *Applied Sciences*, *12*(11), 5645.

[2] Asif, S., Zhao, M., Chen, X., & Zhu, Y. (2023). BMRI-NET: A deep stacked ensemble model for multi-class brain tumor classification from MRI images. *Interdisciplinary Sciences: Computational Life Sciences*, *15*(3), 499-514.

[3] Abd El Kader, I., Xu, G., Shuai, Z., Saminu, S., Javaid, I., & Salim Ahmad, I. (2021). Differential deep convolutional neural network model for brain tumor classification. *Brain Sciences*, *11*(3), 352.

[4] Xie, Y., Zaccagna, F., Rundo, L., Testa, C., Agati, R., Lodi, R., ... & Tonon, C. (2022). Convolutional neural network techniques for brain tumor classification (from 2015 to 2022): Review, challenges, and future perspectives. *Diagnostics*, *12*(8), 1850.

[5] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, *33*(12), 6999-7019.

[6] Hossain, T., Shishir, F. S., Ashraf, M., Al Nasim, M. A., & Shah, F. M. (2019, May). Brain tumor detection using convolutional neural network. In *2019 1st international conference on advances in science, engineering and robotics technology (ICASERT)* (pp. 1-6). IEEE.

[7] Abiwinanda, N., Hanif, M., Hesaputra, S. T., Handayani, A., & Mengko, T. R. (2018, May). Brain tumor classification using convolutional neural network. In *World Congress on Medical Physics and Biomedical Engineering 2018: June 3–8, 2018, Prague, Czech Republic (Vol. 1)* (pp. 183-189). Singapore: Springer Nature Singapore.

[8] Vankdothu, R., Hameed, M. A., & Fatima, H. (2022). A brain tumor identification and classification using deep learning based on CNN-LSTM method. *Computers and Electrical Engineering*, *101*, 107960.

[9] Ayadi, W., Elhamzi, W., Charfi, I., & Atri, M. (2021). Deep CNN for brain tumor classification. *Neural processing letters*, *53*(1), 671-700.

[10] Khan, H. A., Jue, W., Mushtaq, M., & Mushtaq, M. U. (2021). Brain tumor classification in MRI image using convolutional neural network. *Mathematical Biosciences and Engineering*.

[11] Kumar, R. L., Kakarla, J., Isunuri, B. V., & Singh, M. (2021). Multi-class brain tumor classification using residual network and global average pooling. *Multimedia Tools and Applications*, *80*(9), 13429-13438.

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[13] Ding, X., Zhang, X., Han, J., & Ding, G. (2022). Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11963-11975).

[14] Díaz-Pernas, F. J., Martínez-Zarzuela, M., Antón-Rodríguez, M., & González-Ortega, D. (2021, February). A deep learning approach for brain tumor classification and segmentation using a multiscale convolutional neural network. In *Healthcare* (Vol. 9, No. 2, p. 153). MDPI.

[15] Gao, S. H., Cheng, M. M., Zhao, K., Zhang, X. Y., Yang, M. H., & Torr, P. (2019). Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, *43*(2), 652-662.

[16] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7132-7141).

[17] Deepak, S., & Ameer, P. M. (2019). Brain tumor classification using deep CNN features via transfer learning. *Computers in biology and medicine*, *111*, 103345.

[18] Singh, P., Sizikova, E., & Cirrone, J. (2022). CASS: cross architectural self-supervision for medical image analysis. *arXiv preprint arXiv:2206.04170*.

[19] Li, Y., Yan, X., Xiao, M., Wang, W., & Zhang, F. (2023, December). Investigation of creating accessibility linked data based on publicly available accessibility datasets. In *Proceedings of the 2023 13th International Conference on Communication and Network Security* (pp. 77-81).

[20] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, *6*(1), 1-48.

[21] Swati, Z. N. K., Zhao, Q., Kabir, M., Ali, F., Ali, Z., Ahmed, S., & Lu, J. (2019). Brain tumor classification for MR images using transfer learning and fine-tuning. *Computerized Medical Imaging and Graphics*, *75*, 34-46.

[22] Kaplan, K., Kaya, Y., Kuncan, M., & Ertunç, H. M. (2020). Brain tumor classification using modified local binary patterns (LBP) feature extraction methods. *Medical hypotheses*, *139*, 109696.

[23] Huang, M., Xiong, Z. Y., & Zhu, J. L. (2023). MRI brain tumor classification based on multiscale residual network. Chinese Journal of Magnetic Resonance Imaging, 14(1), 124-129.