Journal of Digital Imaging (2023) 36:973–987
https://doi.org/10.1007/s10278-023-00789-x

PatchResNet: Multiple Patch Division–Based Deep Feature Fusion Framework for Brain Tumor Classification Using MRI Images

Taha Muezzinoglu[1] · Nursena Baygin[2] · Ilknur Tuncer[3] · Prabal Datta Barua[4,5] · Mehmet Baygin[6] · Sengul Dogan[7] · Turker Tuncer[7] · Elizabeth Emma Palmer[8,9] · Kang Hao Cheong[10] · U. Rajendra Acharya[11,12,13]

Received: 27 September 2022 / Revised: 30 January 2023 / Accepted: 31 January 2023 / Published online: 16 February 2023

Abstract

Modern computer vision algorithms are based on convolutional neural networks (CNNs), and both end-to-end learning and transfer learning modes have been used with CNN for image classification. Thus, automated brain tumor classification models have been proposed by deploying CNNs to help medical professionals. Our primary objective is to increase the classification performance using CNN. Therefore, a patch-based deep feature engineering model has been proposed in this work. Nowadays, patch division techniques have been used to attain high classification performance, and variable-sized patches have achieved good results. In this work, we have used *three* types of patches of different sizes (32 × 32, 56 × 56, 112 × 112). Six feature vectors have been obtained using these patches and two layers of the pretrained ResNet50 (global average pooling and fully connected layers). In the feature selection phase, three selectors—neighborhood component analysis (NCA), Chi2, and ReliefF—have been used, and 18 final feature vectors have been obtained. By deploying k nearest neighbors (kNN), 18 results have been calculated. Iterative hard majority voting (IHMV) has been applied to compute the general classification accuracy of this framework. This model uses different patches, feature extractors (two layers of the ResNet50 have been utilized as feature extractors), and selectors, making this a framework that we have named PatchResNet. A public brain image dataset containing four classes (glioblastoma multiforme (GBM), meningioma, pituitary tumor, healthy) has been used to develop the proposed PatchResNet model. Our proposed PatchResNet attained 98.10% classification accuracy using the public brain tumor image dataset. The developed PatchResNet model obtained high classification accuracy and has the advantage of being a self-organized framework. Therefore, the proposed method can choose the best result validation prediction vectors and achieve high image classification performance.

Keywords PatchResNet · Transfer learning · Brain image classification · Tumor classification · Biomedical engineering

Introduction

The central nervous system (CNS) consists of the brain and spinal cord [1, 2]. Primary CNS tumors stem from cells within the brain and spinal cord. They constitute malignant tumors (cancer), where cells grow uncontrolled and can invade nearby tissues and spread to other parts of the brain, as well as benign (non-malignant) tumors, which may grow larger but not spread to other parts of the body [3, 4]. The brain can also be affected by secondary tumors, which spread (metastasize) from other body sites such as the lungs [5]. It is estimated that secondary brain tumors will develop in 30% of adults with a primary tumor elsewhere in the body [6].

The health burden of brain tumors is significant [7]. Survival for many malignant primary brain tumors remains very poor [6]. Moreover, brain tumors are the leading cause of cancer-related deaths in children [8, 9].

Early diagnosis of both primary and secondary brain tumors is critical to optimizing health outcomes [10]. Various medical imaging methods, such as computerized tomography (CT), positron emission tomography (PET), and magnetic resonance imaging (MRI) are currently used in the diagnosis of brain tumors [11]. These techniques are non-invasive methods and provide important information to medical professionals for the diagnosis of the disease [12]. However, due to the brain's complex structure, making a

✉ Sengul Dogan
sdogan@firat.edu.tr

Extended author information available on the last page of the article

robust diagnosis is difficult. Making a fast, reliable, and accurate brain tumor diagnosis is time-consuming.

Computer-aided diagnosis systems (CAD) have become actively used in medicine [13–15]. The application of these systems can improve the rapidity and accuracy of diagnoses and reduce the workload of clinicians, especially in regions where access to highly trained radiologists is limited [16]. Therefore, CAD is highly suitable for automated and rapid preliminary diagnosis.

Literature Review

Nowadays, many studies have been conducted on the accurate classification of brain tumors using artificial intelligence (AI) techniques [10, 17, 18]. A summary of studies conducted on brain tumor classification using AI techniques is provided in Table 1.

It can be noted from the table that the majority of proposed methods have used deep learning methods. These methods need big data to train deep networks, and data augmentation techniques are to be applied to such datasets [36]. Nevertheless, such methods generally achieve high classification accuracy.

Motivation and Our Framework

The main motivation of this research is to increase the classification ability of ResNet [39] in transfer learning mode. Today, patch-based models have attained high classification ability for computer vision. Notable models include vision transformers (ViT) [40], multilayer perceptron mixers (MLP-mixer) [41], and convolutional mixers (ConvMixer) [42], all of which have attained high classification ability. In ConvMixer, the authors show that the high classification capability has been attained using the mixer layer with CNN. In ViT, they used fixed-sized patches and transformers to classify an image. In the experiments with ViT, they used 14×14, 16×16, and 32×32 sized patches to show experiments. We have used three types of patches together in this framework to attain different results.

We proposed a new framework called PatchResNet, which uses two feature extractors and three feature selectors for patches. This architecture produces 18 results. In addition, IHMV was used in this study. A total of 34 (=18+16) results were generated, and best among them were chosen using IHMV. Hence, our developed architecture is self-organizing image classification.

Novelties and Contributions

The novelties of the proposed work are given below:

- In this work, three different types of fixed-size patch divisions have been applied.
- In our framework, features have been generated using pretrained ResNet50. By deploying the last pooling and fully connected layer, two feature extractors have been created, and these feature extractors have been applied to these patches to get different features.
- Three feature selectors have been used in our framework together.
- By applying IHMV, voted results have been created.

The Proposed PatchResNet

As stated in the literature, ResNets are useful deep learning networks that can be used for computer vision. Newly developed computer vision models compared to ResNets to evaluate their classification performances [43]. This indicates the ResNet architecture's high classification potential [44]. A new framework is proposed using this hypothesis and has four main phases. These phases are patch-based deep feature extraction, selection of top features by deploying three feature selectors, classification, and iterative hard majority voting (IHMV) [45].

A graphical outline of PatchResNet is given in Fig. 1.

In this framework, each brain MRI images are resized to 224×224. In the multiple patch division, 32×32, 56×56, and 112×112 sized patches have been used, and three types of patches have been used to extract features. Six feature vectors have been created using two feature extractors (the last pooling layer and the fully connected layer of the pretrained ResNet50) and the computed patches. Using NCA [46], Chi2 [47], and ReliefF [48] selectors, 18 feature vectors are generated from the extracted six feature vectors. Herein, we used a statistical feature selector and two weight-based feature selectors. By using Chi2 statistical moment, the most informative features have been selected using the Chi2 selector. NCA and ReliefF are weight-based feature selectors, and they used L1-norm distance. NCA generates non-negative feature weights and ReliefF generates both negative and positive weights to choose features. By using the generated weights by NCA and ReliefF, the most meaningful features have been selected from the generated feature vector. kNN (it is a well-known distance-based classifier) has been applied to these selected feature vectors, and 18 results (validation prediction vectors) have been developed. IHMV generated 16 voted validation prediction vectors. In the last step, the best resulting validation prediction vector among the generated 34 results is obtained.

More details of the PatchResNet are given below, phase by phase.

Table 1 Summary of works done on automated brain tumor classification

Author(s) and year	Dataset	Method	Result(s) (%)	Key point(s) and limitation(s)
Gudigar et al. [19] 2019	Brain datasets [20]	Shearlet transform, particle swarm optimization, support vector machine	Acc. = 97.38	• tenfold CV • Small data • Only 2 class
Talo et al. [2]2019	Brain datasets [20]	ResNet34	Acc. = 100.0	• fivefold CV • Small data • Data augmentation • Only 2 class • High complexity
Talo et al. [21]2019	Brain datasets [20]	ResNet-50	Acc. = 95.23	• fivefold CV • Small data • High complexity

Khan et al. [22], 2020	BraTS 2015 [23], BraTS 2017 [24] and BraTS 2018 [25]	Edge-based histogram equalization, discrete cosine transform (DCT), VGG16- and VGG19-based feature extraction, extreme machine learning	BraTS 2015
			Acc. = 98.16
			BraTS 2017
			Acc. = 97.26
			BraTS 2018
			Acc. = 93.40
				• 60:40 split ratio
				• tenfold CV
				• High complexity

Ghassemi et al. [26], 2020	Brain tumor dataset [27]	Generative adversarial network (GAN), convolutional neural networks (CNN)	Acc. = 95.60
			Sen. = 94.91
			Spe. = 97.69
			Pre. = 95.29
			FScr. = 95.10
				• Preprocessing and data augmentation
				• fivefold CV
				• High complexity

Raghavendra et al. [28], 2021	TCIA [29]	Feature extraction with VGG16 and k nearest neighbor (kNN)	Acc. = 94.25
			Sen. = 94.33
			Spe. = 94.20
				• tenfold CV
				• Only 2 class

Ahmad et al. [30], 2022	Brain tumor dataset [27]	Variational autoencoders (VAEs), GAN, and classification with ResNet50	Acc. = 96.25
			Rec. = 76.9
			Spe. = 83.7
			Pre. = 83.3
			FScr. = 80.0
				• 60:20:20 split ratio
				• 3 class
				• High complexity

Nayak et al. [31], 2022	Brain tumor dataset [27]	Min–max normalization, data augmentation, EfficientNet-based custom CNN	Acc. = 98.78
			Pre. = 98.75
			FScr. = 98.75
				• 80:10:10 split ratio
				• End-to-end training
				• High complexity

Zahoor et al. [32], 2022	Two dataset, normal class from Kaggle [33], tumor class from brain tumor dataset [27]	Data augmentation, deep feature extraction, HOG features	Acc. = 99.20
			Rec. = 99.13
			Pre. = 99.06
			FScr. = 99.09
				• 80:20 split ratio
				• Data augmentation
				• High complexity

Table 1 (continued)

Author(s) and year	Dataset	Method	Result(s) (%)	Key point(s) and limitation(s)

Shaik and T. K. Cherukuri [34], 2022	Brain tumor dataset [27] and BraTS 2018 [25]	Xception-based multilevel attention network	Brain tumor dataset
			Acc. = 96.51
			BraTS 2018
			Acc. = 94.91
				• fivefold CV
				• Tumor type classification (not healthy images)
				• High complexity

Raza et al. [35], 2022	Brain tumor dataset [27]	GoogleNet-based custom CNN model-DeepTumorNet	Acc. = 99.67
			Pre. = 99.6
			Rec. = 100
			FScr. = 99.66
				• 70:30 split ratio
				• End-to-end training
				• High complexity

Neelima et al. [36], 2022	Brain tumor dataset [27] and BraTS 2018 [25]	Segmentation with sailfish political optimizer, feature extraction with CNN (DeepMRSeg), data augmentation, GAN	Brain tumor dataset
			Acc. = 91.7
			Sen. = 92.8
			Spe. = 92.5
			BraTS 2018
			Acc. = 91.4
			Sen. = 91.1
			Spe. = 91.4
				• High complexity
				• Relatively low accuracy

Oksuz et al. [37], 2022	Brain tumor dataset [25]	Deep feature extraction with AlexNet, ResNet18, GoogleNet, and ShuffleNet, shallow feature extraction (ShallowNet), feature merging, kNN	Acc. = 97.25
			FScr. = 95.26
			Spe. = 97.9
				• fivefold CV
				• High complexity

Ahuja et al. [38], 2022	Brain tumor dataset [25]	Preprocessing, data augmentation, DarkNet19, and DarkNet53	Acc. = 99.43
			Sen. = 98.84
			Spe. = 99.60
				• 80:10:10 split ratio
				• Low performance without data augmentation

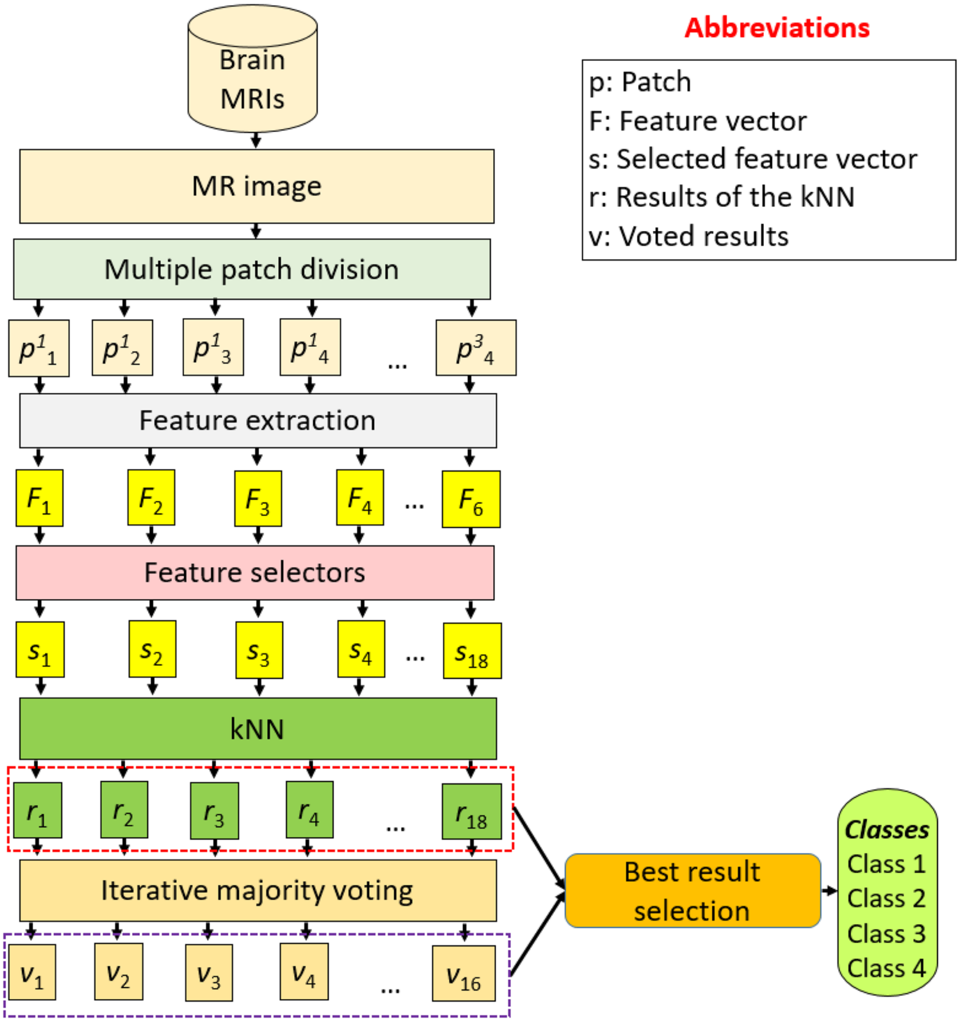

Fig. 1 Overview of the proposed PatchResNet

Multiple Patch-Based Deep Feature Extraction

The primary novelty of this layer is multiple patch divisions. 32 × 32, 56 × 56, and 112 × 112 sized fixed-size patches have extracted features from local areas. Using these sizes of the patches (32 × 32, 56 × 56, and 112 × 112), 49, 16, and 4 patches have been created. Six feature vectors have been generated by extracting features from each group by deploying the last pooling and fully connected layers of the pretrained ResNet50. A graphical explanation has been given to explain the proposed feature extraction layer.

In this figure (see Fig. 2), the abbreviations used are given as follows: FC: fully connected layer, Ff: features of the fully connected layer, Pf: features of the pooling layer, F: final feature vector. Herein, 32 × 32, 56 × 56, and 112 × 112 sized patches have been applied to the image to create patches. Using 32 × 32, 56 × 56, and 112 × 112 sized non-overlapping blocks, 49, 16, and 4 patches have been created from 224 × 224 sized image, and these are named p^1, p^2, and p^3 in this image. Using each patch group, FC (fully connected), and pooling layer of the pretrained ResNet50, 138 feature vectors are generated. Sixty-nine (49, 16, and 4 of them belong to first, second, and third patch groups) of them are generated using the FC layer, and 69 out of them

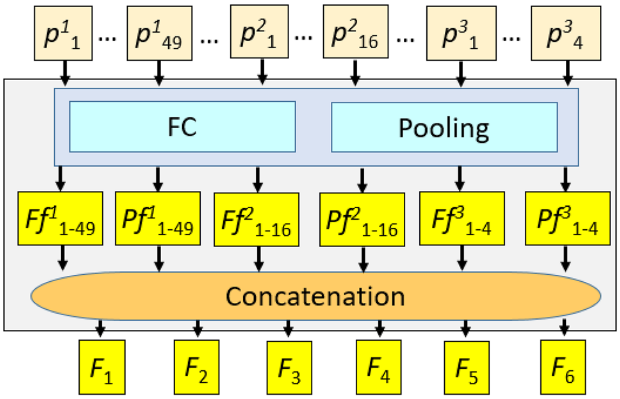

Fig. 2 Feature extraction of the proposed PatchResNet

are generated by deploying the pooling layer. In this layer, the generated 138 features are divided into six groups. These groups are named Ff^1 (this group contains 49 feature vectors, and they are generated using 32×32 sized patches and FC layer), Pf^1 (this group contains 49 feature vectors, and they are generated using 32×32 sized patches and pooling layer), Ff^2 (this group contains 16 feature vectors and they generated using 56×56 sized patches and FC layer), Pf^2 (this group contains 16 feature vectors and they generated using 56×56 sized patches and pooling layer), Ff^3 (this group contains 4 feature vectors and they generated using 112×112 sized patches and FC layer), and Pf^3 (this group contains 4 feature vectors, and they generated using 112×112 sized patches and pooling layer).

By merging these groups, six feature vectors have been generated.

The steps of the proposed multiple patch-based feature generation layers are:

Step 0: Load the image and resize the image to 224×224 sized images.

Step 1: Divide the image into patches with sizes of 32×32, 56×56, and 112×112.

Step 2: Generate features deploying the last pooling layer (global average pooling layer – avg_pool) and fully connected layer (fc1000). The used ResNet50 was trained on the ImageNet1K dataset.

$$Ff_t^h = R50\left(p^t, fc\right), t \in \{1, 2, \dots, N\}, h \in \{1, 2, 3\}, N \in \{49, 16, 4\} \tag{1}$$

$$Pf_t^h = R50\left(p^t, avg\_pool\right) \tag{2}$$

Herein, *Ff* and *Pf* are fully connected and pooling features. These features are generated from pretrained ResNet50 (*R*50(.)).

Step 3: Concatenate the feature vectors generated.

$$F_h = concat\left(Pf_1^h, Pf_2^h, \dots Pf_t^h\right) \tag{3}$$

$$F_{2h} = concat\left(Ff_1^h, Ff_2^h, \dots, Ff_t^h\right) \tag{4}$$

Herein, F_h is *h*th created the final feature vector, and we generated six feature vectors.

In this layer, six feature vectors have been calculated, and presented in Table 2.

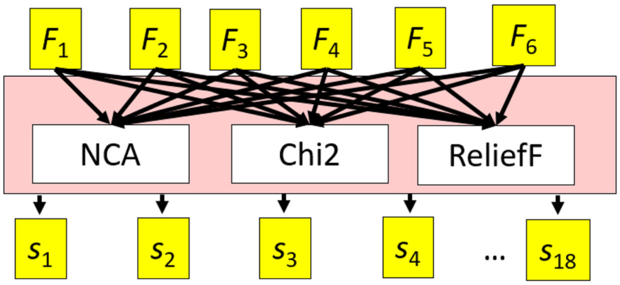

Fig. 3 Feature selection layer of the proposed framework

Feature Selection Layer

This layer is needed to decrease the number of features and increase the number of feature vectors. Three commonly known feature selectors are used in this layer. These feature selectors are NCA [46], Chi2 [47], and ReliefF [48]. NCA and ReliefF are distance base feature selectors that calculate weights for each feature. NCA only generates positive weights, but ReliefF can generate positive and negative weights to qualify features. Chi2 is one of the fastest feature selection functions since it uses a simple statistical moment.

This paper proposes multiple selectors based on the most informative features selection layer. The model developed in this paper uses the pooling and fully connected layers of the ResNet50 architecture to extract *six* different feature vectors (two feature vectors for each patch size). Then, these six feature vectors are fed to the NCA, Chi2, and ReliefF methods for feature selection. This way, three qualified index values are calculated for each feature vector. This phase generates 18 feature vectors containing qualified index information. The graphical outline of this layer is demonstrated in Fig. 3.

The steps of the proposed feature selection model are:

Step 4: Calculate qualified indexes of each feature vector by deploying NCA, Chi2, and ReliefF.

$$idx_j^1 = \mu\left(F_j, y\right), j \in \{1, 2, \dots, 6\} \tag{5}$$

$$idx_j^2 = \chi\left(F_j, y\right) \tag{6}$$

$$idx_j^3 = \varpi\left(F_j, y\right) \tag{7}$$

Table 2 Details of lengths of feature vectors

Layer	Patch size	Total patches	Feature size	Length of the feature vector
Pooling	32×32	49 (=224×224/32×32)	2048	100,352 (=49×2048)
Fully connected	32×32	49 (=224×224/32×32)	1000	49,000 (=49×1000)
Pooling	56×56	16 (=224×224/56×56)	2048	32,768 (=16×2048)
Fully connected	56×56	16 (=224×224/56×56)	1000	16,000 (=16×1000)
Pooling	112×112	4 (=224×224/112×112)	2048	8192 (=4×2048)
Fully connected	112×112	4 (=224×224/112×112)	1000	4000 (=4×1000)

Herein, $\mu(.,.)$ is NCA, $\chi(.,.)$ represents Chi2, and $\varpi(.,.)$ defines ReliefF functions. The input parameters of these feature selectors are feature vectors and actual output (y). Three qualified indexes (idx) have been generated using these three functions,.

Step 5: Select the top 512 features by deploying the indexes generated.

$$s_c(q,i) = F_j\left(q, idx_j^p(i)\right), q \in \{1,2,\dots,dim\}, p \in \{1,2,3\},$$

$$c \in \{1,2,\dots,18\}, i \in \{1,2,\dots,512\} \tag{8}$$

where s_c is the selected cth feature vector with a length of 512, and dim represents the number of images/observations.

Classification

A simple/shallow classifier (kNN) has been used in the classification layer [49]. We used kNN to demonstrate the classification capabilities of the 18 feature vectors. A MATLAB classification learner was used to select the most appropriate classifier, and Fine kNN (1NN) was selected. We only changed the distance parameter of the Fine kNN. We changed the distance parameter to L1-norm (City block) instead of L2-norm (Euclidean) since NCA and ReliefF use L1-norm to calculate distances. Tenfold cross-validation (tenfold CV) has been used to get robust results.

Step 8: Classify the generated each feature by deploying kNN.

$$r_c = kNN(s_c, y) \tag{9}$$

Herein, r_c is cth validation prediction vector with a length of dim.

Majority Voting Layer

The primary goal of this layer is to increase the calculated classification performance in the classification layer. Therefore, the IHMV algorithm was used. IHMV is a loop-based majority voting model and uses a mode function. The steps of this layer are:

Step 9: Sort the calculated results (r) in accordance with their accuracy.

$$ind = \varsigma(acc^r) \tag{11}$$

where ind defines sorted/qualified indexes by descending, $\varsigma(.)$ is the sorting function, and acc^r is an accuracy vector with a.

Step 10: Create an array using a loop.

$$arr_{k-2}^i = \left[r_{ind(1)}(i), r_{ind(2)}(i), \dots r_{ind(k)}(i)\right], k \in \{3,4,\dots,18\}, i \in \{1,2,\dots,dim\} \tag{12}$$

Herein, arr is an array.

Step 11: Calculate the voted results by deploying the mode function.

$$v_{k-2}(i) = \psi\left(arr_{k-2}^i\right) \tag{12}$$

Herein, v_k is kth voted result (validation prediction vector) and $\psi(.)$ is the mode function. In this step, 16 voted results are generated from 18 validation prediction vectors.

In the last step, the best accurate validation prediction vector was chosen as a result.

Step 12: Select the best accurate vector among the 34 (18 kNN results + 16 voted results) results.

Experimental Results

Experimental Setup

We used the MATLAB (2022a) programming tool to implement PatchResNet. The pretrained ResNet50 was also imported to MATLAB. We used a simple configured laptop for implementation. This laptop has Intel Core i7 10870H processor, 16 GB main memory, and 512 GB hard disk. We did not use any graphical processing units since we used ResNet50 in transfer learning mode. The transition of the proposed PatchResNet is tabulated in Table 3.

The parameters of the proposed PatchResNet are tabulated in Table 3. The calculated results have been generated using these parameters. Using different sizes of patches, feature extractors, feature selectors, classifiers, and voted algorithms, variable classification models can be proposed.

Dataset

We used an open-access MRI dataset that is popular for computer vision applications and is publicly available on the Kaggle website (https://www.kaggle.com/). This dataset has four categories with 3264 MR images. The distribution of this dataset is given as follows. There are 926 scans of brains with glioblastoma multiforme (GBM), 937 meningioma images, 901 pituitary tumor images, and 500 control scans of healthy individuals [50, 51]. The sample images of this dataset have been demonstrated in Fig. 4.

Performance Evaluation Metrics

Standard performance evaluation metrics—accuracy, F1-score, precision, and recall—were used to evaluate classification results. Accuracy is the oldest classification evaluation performance metric and is calculated using the number of true predicted observations. Recall defines the ratio of

Table 3 Details of the presented PatchResNet

Layer	Operator	Size/explanation
Feature extraction	Image resizing	224 × 224
	Patch division	49 patches of size 32 × 32 are deployed (first patch type). 16 patches of size 56 × 56 are deployed (second patch type). 4 patches of size 112 × 112 are deployed (third patch type).
	Feature extractors	FC and global average pooling of the pretrained ResNet50
	Feature vectors creation	F_1: 100,352 (first patch type + pooling layer) F_2: 49,000 (first patch type + FC layer) F_3: 32,768 (second patch type + pooling layer) F_4: 16,000 (second patch type + FC layer) F_5: 8192 (third patch type + pooling layer) F_6: 4000 (third patch type + FC layer)
Feature selection	Multiple feature selectors applying and generating	18 selected feature vectors are created with a length of 512 s_1: First patch type + pooling layer + NCA s_2: First patch type + pooling layer + Chi2 s_3: First patch type + pooling layer + ReliefF s_4: First patch type + FC layer + NCA s_5: First patch type + FC layer + Chi2 s_6: First patch type + FC layer + ReliefF s_7: Second patch type + pooling layer + NCA s_8: Second patch type + pooling layer + Chi2 s_9: Second patch type + pooling layer + ReliefF s_{10}: Second patch type + FC layer + NCA s_{11}: Second patch type + FC layer + Chi2 s_{12}: Second patch type + FC layer + ReliefF s_{13}: Third patch type + pooling layer + NCA s_{14}: Third patch type + pooling layer + Chi2 s_{15}: Third patch type + pooling layer + ReliefF s_{16}: Third patch type + FC layer + NCA s_{17}: Third patch type + FC layer + Chi2 s_{18}: Third patch type + FC layer + ReliefF
Classification	Applying kNN	Generating 18 prediction vectors using the selected 18 chosen feature vectors. Attributes: k: 1, Distance: L1-Norm, Voting: None, Validation: tenfold CV
Majority voting	IHMV	Creating 16 voted feature vectors from 18 prediction vectors Selection of the best results from 34 (= 18 + 16) predicted vectors.

the number of true positives to the sum of true positives and false negatives which is a useful performance measure to evaluate the unbalanced datasets. Finally, precision is used to calculate the ratio of the true positives with all positives and is very important to show the diagnosis rate. To express precision and recall using a parameter, the F1-score (it is the harmonic mean of the precision and recall) has been used.

Results

Precision, recall, accuracy, and F1-score have been calculated to compute results. The obtained confusion matrix is presented in Fig. 5.

The results obtained from deploying the confusion matrix (see Fig. 5) are presented in Table 4.

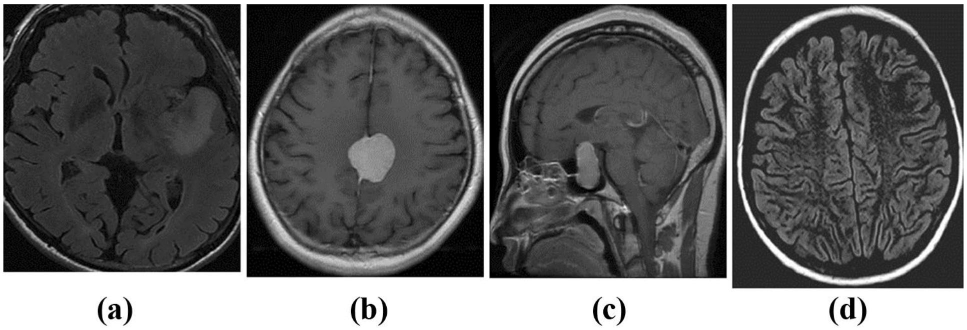

Fig. 4 Sample brain MR images in four classes of the used dataset: **a** glioblastoma multiforme (GBM), **b** meningioma, **c** pituitary tumor, and **d** healthy control

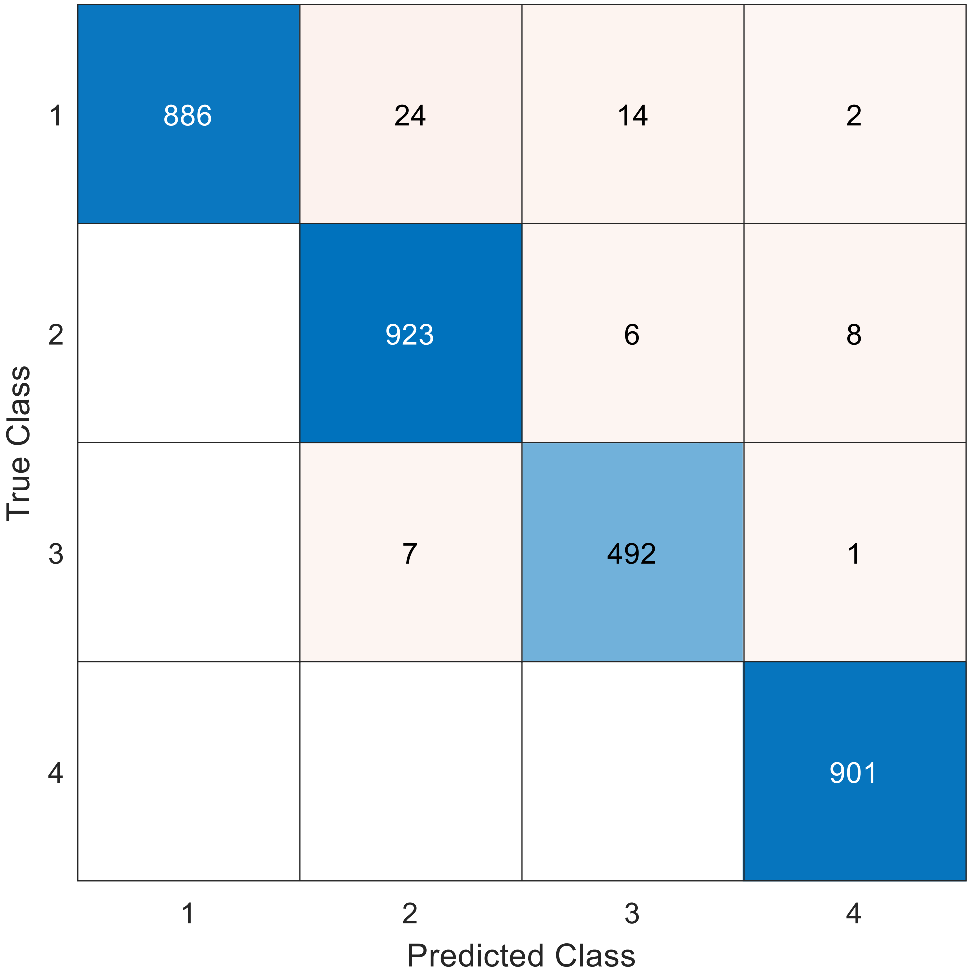

Fig. 5 Results of the confusion matrix using the proposed PatchResNet. 1 is GBM, 2 is meningioma, 3 is healthy controls, and 4 is pituitary

Table 4 demonstrates that the proposed PatchResNet attained 98.10% classification accuracy, 98.15% unweighted average recall, 97.91% average precision, and 98.01% overall F1-score. Moreover, the best results class is Pituitary since the recall of this class is equal to 100%. GBM also attained 100% precision.

We proposed a new framework named PatchResNet because three types of patch divisions were used in this work. We selected pretrained ResNet50 to extract features as the fully connected layer of ResNet50 has generally been used to extract deep features in the literature. This research used both global average pooling and fully connected layers of the pretrained ResNet50 to obtain two deep feature extractors. Variable-sized feature vectors have been obtained by using different patch divisions and two feature extractors. In the feature selection phase, the most informative 512 features were selected, three feature selectors were used, and 18 (=3 × 2 × 3) feature vectors were calculated. kNN was applied to these 18 selected feature vectors to calculate classification results, and the calculated accuracies of these 18 selected feature vectors are depicted in Fig. 6.

Figure 6 demonstrates that the best accurate feature vector is the 7th feature vector and the 7th feature vector yielded 96.54% classification accuracy. This vector is created using fixed-size patches of 56 × 56, feature extraction using the

Table 4 Summary of overall and category-wise classification results (%)

Class	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Glioblastoma multiforme (GBM)	95.68	95.68	100	97.79
Meningioma	98.51	98.51	96.75	97.62
Healthy control	98.40	98.40	96.09	97.23
Pituitary tumor	100	100	98.79	99.39
Overall	98.10	98.15	97.91	98.01

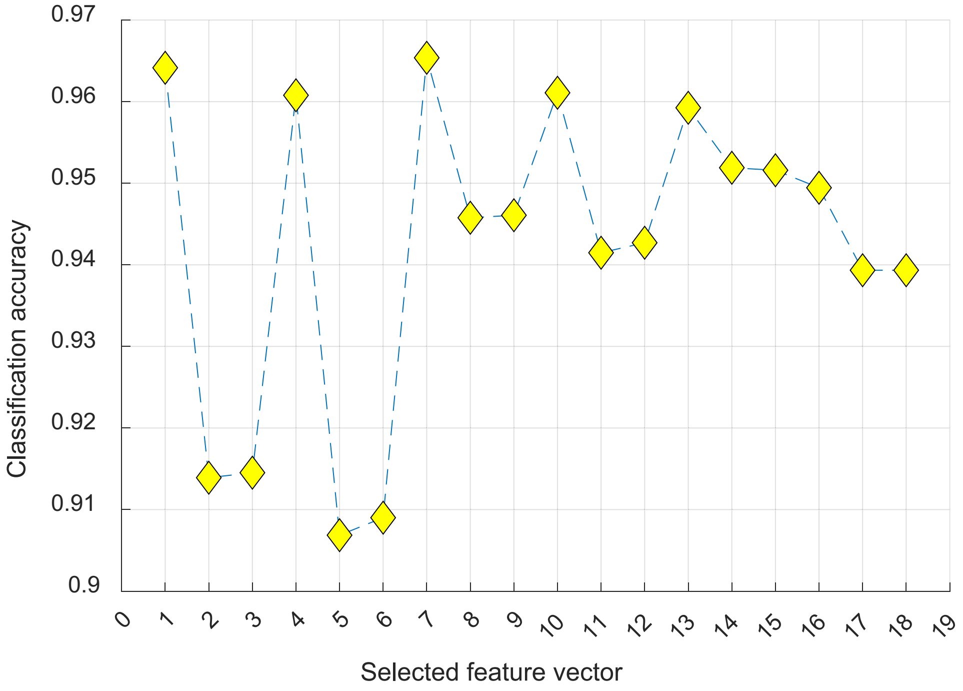

Fig. 6 Plot of classification accuracy versus selected feature vectors using kNN classifier with tenfold cross-validation

global average pooling layer of the ResNet50 and NCA feature selector. Using the outputs of the proposed PatchResNet, comparative results have been calculated in accordance with the size of the patch, feature extractor, and feature selectors. These comparative results are demonstrated in Fig. 7.

Figure 7 demonstrates the average classification accuracies of the methods used. We have used three types of patch divisions, and these are 32×32, 56×56, and 112×112. Our calculated average classification accuracies are 92.82%, 95.04%, and 94.85% for 32×32, 56×56, and 112×112 sized patch divisions, respectively. Two feature selectors were used. The average classification accuracy of the pooling layer–based feature extractor is 94.58%, and the average classification accuracy of the FC-based feature extractor is 93.89%. According to the feature selectors performance evaluation, the best selector is NCA since the average classification of NCA is 96%. Average classification accuracies ReliefF and Chi2 are 93.39% and 93.32%, respectively. According to Fig. 7, the best size for patch division is 56×56, the best feature extraction model pooling layer of the ResNet50, and the most suitable selector is NCA. Moreover, the 7th selected feature vector used these components and achieved the best classification accuracy among the 18 generated classification results by deploying a kNN classifier with a tenfold CV (see Fig. 6). Moreover, the statistical analysis of these components is given in Table 5.

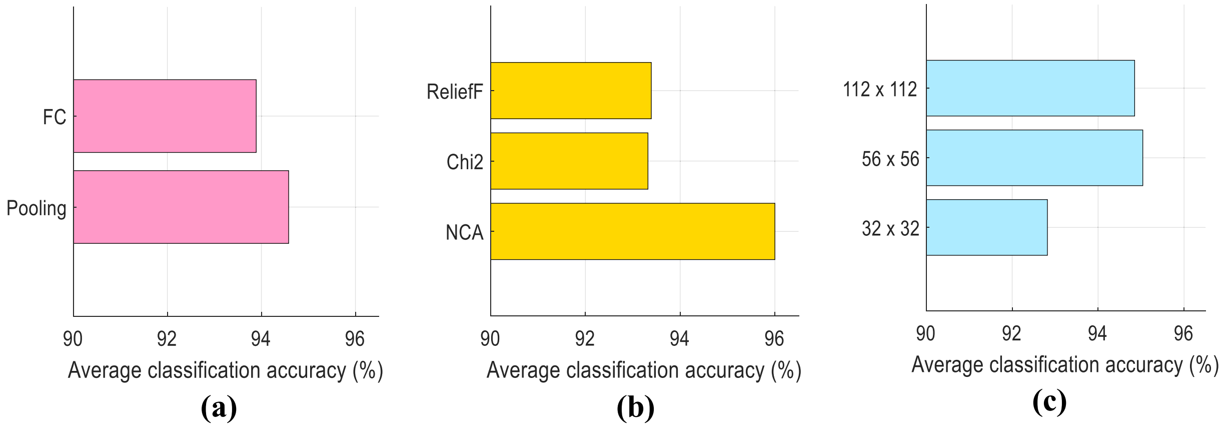

Fig. 7 Performance comparison of the used components: **a** feature extractors, **b** feature selectors, and **c** patch division model

Table 5 General classification (mean ± standard deviation) of the used components

Component	Parameters	General accuracy (%)
Feature extractors	FC	93.90 ± 1.95
	Pooling	94.63 ± 1.92
Feature selectors	ReliefF	93.38 ± 1.72
	Chi2	93.33 ± 1.76
	NCA	96.08 ± 0.66
Patch size	112 × 112	94.83 ± 0.79
	56 × 56	95.06 ± 1.15
	32 × 32	92.92 ± 2.68

We applied the statistical *t*-test to the generated 18 feature vectors to obtain clinically significant features. Herein, our reference point is *p*-value since features with a *p*-value less than 0.005 are considered distinct features. In each feature vector, there are 512 features, and this dataset has four classes. Therefore, *p*-values of $6 = \binom{4}{2}$ couples have been calculated. Using *p*-values, the number of distinct features has been calculated and is shown in Fig. 8.

Figure 8 demonstrates that our generated features are distinctive based on *p*-value analysis and yielded high classification performance.

IHMV was used to calculate voted classification accuracies, and 16 voted results have been calculated. These classification accuracies of the voted vectors are demonstrated in Fig. 9.

Figure 9 demonstrates classification accuracies via the number of predicted vectors used to calculate voted vectors. According to Fig. 9, the best classification accuracy was 98.10%, attained by voting the best 11 results. Moreover, all voted results are higher than 97%.

We used 18 pretrained CNNs to get comparative results. The used CNNs are (1) ResNet18, (2) ResNet50, (3) ResNet101, (4) DarkNet19, (5) MobileNetV2, (6) DarkNet53, (7) Xception, (8) ShuffleNet, (9) NasNetMobile, (10) NasNetLarge, (11) DenseNet201, (12) InceptionV3, (13) InceptionResNetV2, (14) GoogLeNet, (15) AlexNet, (16) VGG16, (17) VGG19, and (18) SqueezeNet. Pooling/fully connected layers of these networks have been used to extract features. By deploying NCA, the top 512 features were selected, and classification was performed by deploying kNN. The calculated classification accuracies of these pretrained CNNs are shown in Table 6.

Table 6 demonstrates that the best feature extractor among the used 18 CNNs is pretrained DenseNet201, and it attained 95.83% classification accuracy. Moreover, ResNet50 attained 93.90% classification accuracy without using patch division. By deploying patch division, the accuracy rate of the ResNet50 was increased from 93.90 to 96.45% (see Fig. 6). Moreover, PatchResNet increased

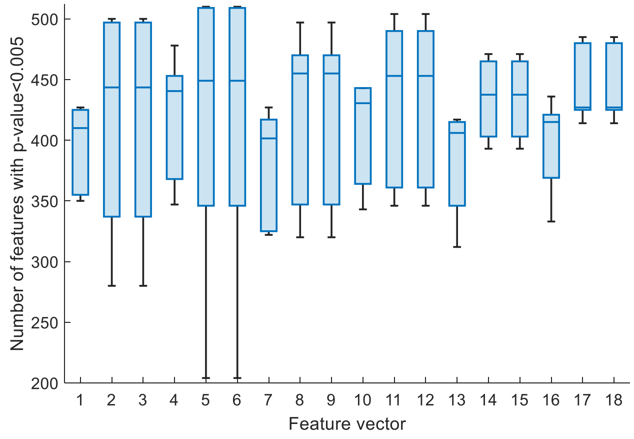

Fig. 8 Boxplot of clinically significant feature vectors

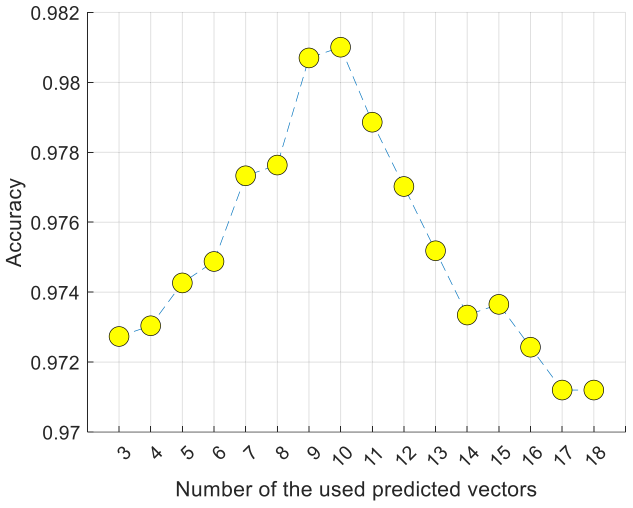

Fig. 9 Plot of classification accuracies versus the number of the used predicted vectors

Table 6 Comparison results (%) of the proposed PatchResNet with other pretrained models

Method	Accuracy (%)	Method	Accuracy (%)
ResNet18	92.62	DenseNet201	95.83
ResNet50	93.90	InceptionV3	91.85
ResNet101	93.66	InceptionResNetV2	91.36
DarkNet19	91.33	GoogLeNet	92
MobileNetV2	92.62	AlexNet	93.78
DarkNet53	93.35	VGG16	91.15
Xception	92.65	VGG19	91.61
ShuffleNet	93.38	SqueezeNet	93.72
NasNetMobile	90.41	PatchResNet	98.10
NasNetLarge	90.72		

Table 7 Comparison of accuracy (%) with ViT method

Classes	Our method	ViT method used in [52]
GBM	95.68	**98.01**
Meningioma	**98.51**	94.8
Pituitary	**100**	99.4

The best results are highlighted in bold

classification performance to 98.10%. Table 6 tabulates that our proposed PatchResNet increased the classification performance of the pretrained ResNet50. To show the superiority of the developed framework, the class-wise comparison results with ViT are given in Table 7.

Tummala et al. [52] used the ViT method with the same dataset in their study. As can be seen in Table 7, our method has achieved better classification performance for 2 classes (Meningioma and Pituitary). For GBM, the ViT method achieved higher classification accuracy. However, they applied a 70:30 hold-out validation strategy in their study. But, we have used a tenfold CV to obtain more generalized robust results.

In this study, we used the second dataset to evaluate the proposed method's performance. The dataset presented by Cheng et al. [27] contains 3064 images belonging to three classes: GBM (1426), Meningioma (708), and Pituitary (930) [53]. The test results obtained on this dataset are given in Table 8.

As seen in Table 8, the proposed framework has achieved more than 95% classification accuracy using the second dataset. Hence, our proposed method illustrates high classification performance using both datasets.

Discussion

To better imply the success of the proposed PatchResNet on the used brain image dataset, comparative results are tabulated in Table 9.

The research given in Table 9 uses the same dataset as this study. According to Table 9, Musallam et al. [54] achieved an accuracy of 98.22%. However, end-to-end training was carried out in this study. Rasool et al. [55] achieved an accuracy of 98.1%. This value is the same as our result, but 80:20 hold-out validation was used as a validation technique. Aurna et al. [56] applied data augmentation to the dataset. In this way, it increased the amount of data and provided 98.16% classification accuracy. Ullah et al. [57] used the data augmentation method. In addition, a control class was not used in the study, and only the types of brain tumors were classified. Although the method proposed by Kang et al. [58] was complex, it could reach 93.72% accuracy. Senan et al. [59] proposed an approach similar to our method. However, data augmentation was also used, and an accuracy value of 95.1% was achieved in this study. Gupta et al. [60] proposed a two-level method. In this method, firstly, detects whether there is a tumor; secondly, if there is a tumor, its type is classified. In addition, data augmentation was performed using the GAN method, and 98% accuracy was achieved in this paper. A similar situation exists in the method proposed by Alanazi et al. [61]. In this study, three classes were used, and 95.75% accuracy was achieved with the custom-designed CNN. The proposed method by Kibriya et al. [62] classifies the types of brain tumors and uses only three classes for this. In this study, a CNN was designed, and 97.2% classification accuracy was obtained. Considering the studies given in Table 9, the proposed method in this paper has low computational complexity and still shows high classification success.

The important points of this research are discussed below.

The advantages of the proposed method are given below:

- A new multiple patch-based transfer learning framework was proposed in this work to efficiently utilize patch-based image classification models.
- We have proposed a parametric image classification architecture (see Table 3). New-generation patch-based image classification models can be proposed.
- kNN (shallow machine learning algorithm) was used to demonstrate the high classification ability of the selected feature vectors.
- IHMV was used to increase classification capability.
- The proposed PatchResNet attained 98.10% classification accuracy.
- Our proposed architecture increased the classification ability of the proposed ResNet50.
- Performances of the methods used were compared. The most appropriate size of the patch is 56×56, the best

Table 8 Overall and category-wise results (%) obtained using the second dataset

Class	Accuracy (%)	Recall (%)	Precision (%)	F1-score (%)
Glioblastoma multiforme (GBM)	–	97.62	95.28	96.43
Meningioma	–	89.83	94.64	92.17
Pituitary tumor	–	99.25	99.14	99.19
Overall	96.31	95.56	96.35	95.93

Table 9 Comparison of performance with state-of-the-art models

Author(s)	Year	Method	Validation	Result(s) (%)
Musallam et al. [54]	2022	Custom-designed CNN	–	Acc. = 98.22

Rasool et al. [55]	2022	GoogleNet-based feature extraction, SVM classifier	80:20 hold-out validation	Acc. = 98.1
				Pre. = 98.2
				Rec. = 98.1

Aurna et al. [56]	2022	Data augmentation, principal component analysis, two stage ensemble of CNN models	eightfold cross-validation	Acc. = 98.16
				Pre. = 98.0
				Rec. = 98.0
				FScr. = 98.0

Ullah et al. [57]	2022	Data augmentation, InceptionResNetV2	80:20 hold-out validation	Acc. = 98.91
				Pre. = 98.28
				Rec. = 99.75
				FScr. = 99.0

Kang et al. [58]	2021	Image preprocessing, DenseNet169, ShuffleNet, and MnasNet-based feature extraction and SVM with the radial basis function	80:20 hold-out validation	Acc. = 93.72

Senan et al. [59]	2022	Data augmentation, deep feature extraction with AlexNet, SVM	80:20 hold-out validation	Acc. = 95.1
				Sen. = 95.25
				Spe. = 98.5

Gupta et al. [60]	2022	Image preprocessing, data augmentation with GAN, feature extraction with InceptionResNetV2, and Random Forest Tree	–	Acc. = 98.0
Alanazi et al. [61]	2022	Custom-designed CNN	80:20 hold-out validation	Acc. = 95.75

Kibriya et al. [62]	2022	Custom-designed CNN	70:30 hold-out validation	Acc. = 97.2
				Pre. = 97.0
				Rec. = 96.0

Our method		ResNet50-based deep feature extraction, multiple feature selector (NCA, Chi2, ReliefF) and kNN	tenfold cross-validation	Acc. = 98.10
				Rec. = 98.15
				Pre. = 97.91
				FScr. = 98.01

layer for feature extraction is pooling, and the most suitable feature selector is NCA for this dataset.

The drawbacks of our method are given below:

- More extensive datasets need to be used.
- We used a shallow classifier and did not use any optimization methods to get higher classification accuracy. Moreover, this framework uses feature extraction and feature selection phases. We did not use any fine-tuning operator in these phases.

Conclusions

A new image classification framework called PatchResNet was proposed. The primary aim of the PatchResNet is to increase the classification ability of the transfer learning–based ResNet50 model. The presented PatchResNet was developed using a brain tumor dataset with four categories. Our framework attained best accuracy of 96.54% using kNN classifier with a tenfold CV. This performance was increased to 98.10% using IHMV methodology.

Our developed model is a self-organized framework involving patches, feature extractors, and feature selectors. The limitation of this work is that we have used fewer patients in each. In the future, we plan to validate our work with a huge database. Also, we plan to employ explainable artificial intelligence (XAI) techniques in the developed model to visualize the region of brain tumors and build trust in the clinicians on our diagnosis [63].

Data Availability The data used in this study were downloaded from https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection.

Declarations

Conflict of Interest The authors declare no competing interests.

References

1. Thau L, Reddy V, Singh P: Anatomy, Central Nervous System, 2019
2. Talo M, Baloglu UB, Yıldırım Ö, Acharya UR: Application of deep transfer learning for automated brain abnormality classification using MR images. Cognitive Systems Research 54:176-188, 2019
3. Shoeibi A, et al.: Diagnosis of brain diseases in fusion of neuroimaging modalities using deep learning: A review. Information Fusion, 2022

4. Nayak DR, Dash R, Majhi B, Acharya UR: Application of fast curvelet Tsallis entropy and kernel random vector functional link network for automated detection of multiclass brain abnormalities. Computerized Medical Imaging and Graphics 77:101656, 2019
5. Arvanitis CD, Ferraro GB, Jain RK: The blood–brain barrier and blood–tumour barrier in brain tumours and metastases. Nature Reviews Cancer 20:26-41, 2020
6. Lapointe S, Perry A, Butowski NA: Primary brain tumours in adults. The Lancet 392:432-446, 2018
7. Raghavendra U, Acharya UR, Adeli H: Artificial intelligence techniques for automated diagnosis of neurological disorders. European neurology 82:41-64, 2019
8. Liu K-W, Pajtler KW, Worst BC, Pfister SM, Wechsler-Reya RJ: Molecular mechanisms and therapeutic targets in pediatric brain tumors. Science signaling 10:eaaf7593, 2017
9. Jones DT, et al.: Molecular characteristics and therapeutic vulnerabilities across paediatric solid tumours. Nature Reviews Cancer 19:420-438, 2019
10. Abd-Ellah MK, Awad AI, Khalaf AAM, Hamed HFA: A review on brain tumor diagnosis from MRI images: Practical implications, key achievements, and lessons learned. Magnetic resonance imaging 61:300-318, 2019
11. Herholz K, Langen K-J, Schiepers C, Mountz JM: Brain tumors: City, 2012 Year
12. Pellico J, Gawne PJ, de Rosales RTM: Radiolabelling of nanomaterials for medical imaging and therapy. Chemical Society Reviews 50:3355-3423, 2021
13. Qayyum A, Qadir J, Bilal M, Al-Fuqaha A: Secure and robust machine learning for healthcare: A survey. IEEE Reviews in Biomedical Engineering 14:156-180, 2020
14. Fujita H: AI-based computer-aided diagnosis (AI-CAD): the latest review to read first. Radiological physics and technology 13:6-19, 2020
15. Gudigar A, Raghavendra U, Hegde A, Kalyani M, Ciaccio EJ, Acharya UR: Brain pathology identification using computer aided diagnostic tool: A systematic review. Computer Methods and Programs in Biomedicine 187:105205, 2020
16. Chan HP, Hadjiiski LM, Samala RK: Computer-aided diagnosis in the era of deep learning. Medical physics 47:e218-e227, 2020
17. Ullah Z, Usman M, Jeon M, Gwak J: Cascade multiscale residual attention cnns with adaptive roi for automatic brain tumor segmentation. Information sciences 608:1541-1556, 2022
18. Tiwari A, Srivastava S, Pant M: Brain tumor segmentation and classification from magnetic resonance images: Review of selected methods from 2014 to 2019. Pattern Recognition Letters 131:244-260, 2020
19. Gudigar A, Raghavendra U, San TR, Ciaccio EJ, Acharya UR: Application of multiresolution analysis for automated detection of brain abnormality using MR images: A comparative study. Future Generation Computer Systems 90:359-367, 2019
20. Harvard Medical School Data, http://www.med.harvard.edu/AANLIB/.
21. Talo M, Yildirim O, Baloglu UB, Aydin G, Acharya UR: Convolutional neural networks for multi-class brain disease detection using MRI images. Computerized Medical Imaging and Graphics 78:101673, 2019
22. Khan MA, et al.: Multimodal brain tumor classification using deep learning and robust feature selection: A machine learning application for radiologists. Diagnostics 10:565, 2020
23. Menze BH, et al.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE transactions on medical imaging 34:1993-2024, 2014
24. Bakas S, et al.: Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. Scientific data 4:1-13, 2017
25. Bakas S, et al.: Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge. arXiv preprint arXiv:181102629, 2018
26. Ghassemi N, Shoeibi A, Rouhani M: Deep neural network with generative adversarial networks pre-training for brain tumor classification based on MR images. Biomedical Signal Processing and Control 57:101678, 2020
27. Cheng J, et al.: Enhanced performance of brain tumor classification via tumor region augmentation and partition. PloS one 10:e0140381, 2015
28. Raghavendra U, et al.: Feature-versus deep learning-based approaches for the automated detection of brain tumor with magnetic resonance images: A comparative study. International Journal of Imaging Systems and Technology 32:501-516, 2022
29. Clark K, et al.: The Cancer Imaging Archive (TCIA): maintaining and operating a public information repository. Journal of digital imaging 26:1045-1057, 2013
30. Ahmad B, Sun J, You Q, Palade V, Mao Z: Brain Tumor Classification Using a Combination of Variational Autoencoders and Generative Adversarial Networks. Biomedicines 10:223, 2022
31. Nayak DR, Padhy N, Mallick PK, Zymbler M, Kumar S: Brain Tumor Classification Using Dense Efficient-Net. Axioms 11:34, 2022
32. Zahoor MM, et al.: A New Deep Hybrid Boosted and Ensemble Learning-Based Brain Tumor Analysis Using MRI. Sensors 22:2726, 2022
33. Brain MRI Images for Brain Tumor Detection. Available at https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection. Accessed 19/06/2022 2022.
34. Shaik NS, Cherukuri TK: Multi-level attention network: application to brain tumor classification. Signal, Image and Video Processing 16:817-824, 2022
35. Raza A, et al.: A Hybrid Deep Learning-Based Approach for Brain Tumor Classification. Electronics 11:1146, 2022
36. Neelima G, Chigurukota DR, Maram B, Girirajan B: Optimal DeepMRSeg based tumor segmentation with GAN for brain tumor classification. Biomedical Signal Processing and Control 74:103537, 2022
37. Öksüz C, Urhan O, Güllü MK: Brain tumor classification using the fused features extracted from expanded tumor region. Biomedical Signal Processing and Control 72:103356, 2022
38. Ahuja S, Panigrahi BK, Gandhi TK: Enhanced performance of Dark-Nets for brain tumor classification and segmentation using colormap-based superpixel techniques. Machine Learning with Applications 7:100212, 2022
39. He K, Zhang X, Ren S, Sun J: Deep residual learning for image recognition: City, 2016 Year
40. Dosovitskiy A, et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:201011929, 2020
41. Tolstikhin IO, et al.: Mlp-mixer: An all-mlp architecture for vision. Advances in Neural Information Processing Systems 34, 2021
42. Trockman A, Kolter JZ: Patches are all you need? arXiv preprint arXiv:220109792, 2022
43. Allen-Zhu Z, Li Y: What can resnet learn efficiently, going beyond kernels? Advances in Neural Information Processing Systems 32, 2019
44. Koonce B, Koonce B: ResNet 50. Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization:63–72, 2021
45. Dogan A, et al.: PrimePatNet87: prime pattern and tunable q-factor wavelet transform techniques for automated accurate EEG emotion recognition. Computers in Biology and Medicine 138:104867, 2021
46. Goldberger J, Hinton GE, Roweis S, Salakhutdinov RR: Neighbourhood components analysis. Advances in neural information processing systems 17:513-520, 2004
47. Liu H, Setiono R: Chi2: Feature selection and discretization of numeric attributes. Proc. Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence: Herndon, VA, USA, 1995

48. Urbanowicz RJ, Meeker M, La Cava W, Olson RS, Moore JH: Relief-based feature selection: Introduction and review. Journal of biomedical informatics 85:189-203, 2018
49. Peterson LE: K-nearest neighbor. Scholarpedia 4:1883, 2009
50. Kang J, Gwak J: Deep Learning-Based Brain Tumor Classification in MRI images using Ensemble of Deep Features. Journal of the Korea Society of Computer and Information 26:37-44, 2021
51. Brain Tumor Classification (MRI). Available at https://www.kaggle.com/sartajbhuvaji/brain-tumor-classification-mri/discussion.
52. Tummala S, Kadry S, Bukhari SAC, Rauf HT: Classification of Brain Tumor from Magnetic Resonance Imaging Using Vision Transformers Ensembling. Current Oncology 29:7498-7511, 2022
53. Kaldera H, Gunasekara SR, Dissanayake MB: MRI based Glioma segmentation using Deep Learning algorithms: City, 2019 Year
54. Musallam AS, Sherif AS, Hussein MK: A New Convolutional Neural Network Architecture for Automatic Detection of Brain Tumors in Magnetic Resonance Imaging Images. IEEE Access 10:2775-2782, 2022
55. Rasool M, et al.: A Hybrid Deep Learning Model for Brain Tumour Classification. Entropy 24:799, 2022
56. Aurna NF, Yousuf MA, Taher KA, Azad AKM, Moni MA: A classification of MRI brain tumor based on two stage feature level ensemble of deep CNN models. Computers in Biology and Medicine 146:105539, 2022
57. Ullah N, et al.: An Effective Approach to Detect and Identify Brain Tumors Using Transfer Learning. Applied Sciences 12:5645, 2022
58. Kang J, Ullah Z, Gwak J: Mri-based brain tumor classification using ensemble of deep features and machine learning classifiers. Sensors 21:2222, 2021
59. Senan EM, Jadhav ME, Rassem TH, Aljaloud AS, Mohammed BA, Al-Mekhlafi ZG: Early Diagnosis of Brain Tumour MRI Images Using Hybrid Techniques between Deep and Machine Learning. Computational and Mathematical Methods in Medicine 2022, 2022
60. Gupta RK, Bharti S, Kunhare N, Sahu Y, Pathik N: Brain Tumor Detection and Classification Using Cycle Generative Adversarial Networks. Interdisciplinary Sciences: Computational Life Sciences:1–18, 2022
61. Alanazi MF, et al.: Brain tumor/mass classification framework using magnetic-resonance-imaging-based isolated and developed transfer deep-learning model. Sensors 22:372, 2022
62. Kibriya H, Masood M, Nawaz M, Nazir T: Multiclass classification of brain tumors using a novel CNN architecture. Multimedia Tools and Applications:1–17, 2022
63. Loh HW, Ooi CP, Seoni S, Barua PD, Molinari F, Acharya UR: Application of explainable artificial intelligence for healthcare: A systematic review of the last decade (2011–2022). Computer Methods and Programs in Biomedicine:107161, 2022

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Taha Muezzinoglu[1] · Nursena Baygin[2] · Ilknur Tuncer[3] · Prabal Datta Barua[4,5] · Mehmet Baygin[6] · Sengul Dogan[7] · Turker Tuncer[7] · Elizabeth Emma Palmer[8,9] · Kang Hao Cheong[10] · U. Rajendra Acharya[11,12,13]

Taha Muezzinoglu
tahamuezzinoglu@munzur.edu.tr

Nursena Baygin
nursena.baygin@erzurum.edu.tr

Ilknur Tuncer
ilknur.tuncer@icisleri.gov.tr

Prabal Datta Barua
Prabal.Barua@usq.edu.au

Mehmet Baygin
mehmetbaygin@ardahan.edu.tr

Turker Tuncer
turkertuncer@firat.edu.tr

Elizabeth Emma Palmer
elizabeth.palmer@unsw.edu.au

Kang Hao Cheong
kanghao_cheong@sutd.edu.sg

U. Rajendra Acharya
aru@np.edu.sg

[1] Department of Computer Engineering, Faculty of Engineering, Munzur University, Tunceli, Turkey

[2] Department of Computer Engineering, Faculty of Engineering, Erzurum Technical University, Erzurum, Turkey

[3] Interior Ministry, Elazig, Turkey

[4] School of Management & Enterprise, University of Southern Queensland, Toowoomba, Australia

[5] Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, Australia

[6] Department of Computer Engineering, Faculty of Engineering, Ardahan University, Ardahan, Turkey

[7] Department of Digital Forensics Engineering, College of Technology, Firat University, Elazig, Turkey

[8] Centre of Clinical Genetics, Sydney Children's Hospitals Network, Randwick 2031, Australia

[9] School of Women's and Children's Health, University of New South Wales, Randwick 2031, Australia

[10] Science, Mathematics and Technology Cluster, Singapore University of Technology and Design, Singapore S487372, Singapore

[11] Department of Electronics and Computer Engineering, Ngee Ann Polytechnic, Singapore 599489, Singapore

[12] Department of Biomedical Engineering, School of Science and Technology, SUSS University, Singapore, Singapore

[13] Department of Biomedical Informatics and Medical Engineering, Asia University, Taichung, Taiwan