International Information and
Engineering Technology Association

Advancing the World of Information and Engineering

Performance Evaluation of Feature Extraction and SVM for Brain Tumor Detection Using MRI Images

Check for updates

Zouhir Iourzikene*, Fawzi Gougam, Djamel Benazzouz

Laboratoire de Mécanique des Solides et Systèmes (LMSS), Faculté de technologie (FT), Université M'Hamed BOUGARA de Boumerdes (UMBB), Boumerdes 35000, Algeria

Corresponding Author Email: z.iourzikene@univ-boumerdes.dz

ABSTRACT

The aim of this study is to develop an automatic detection of brain tumors from magnetic resonance images based on artificial intelligence. The developed approach comprises three steps: pre-processing, feature extraction, and classification. The pre-processing consists of applying image processing techniques to improve contrast and reduce noise in magnetic resonance images. The feature extraction consists of transforming magnetic resonance images into numerical vectors that represent the discriminating attributes for tumor detection. Then the classification consists of using a machine-learning algorithm to separate magnetic resonance images into two classes: tumoral and non-tumoral. The performance evaluation of the proposed approach is tested under dataset of 3000 magnetic resonance images, where 1500 magnetic resonance images are with tumors and 1500 magnetic resonance images are without tumors. In the feature extraction step, two techniques have been used the bag of features and the ResNet50 convolution neural network then a comparison between them was performed. In the last step, the obtained images have been compared with the three different kernels function for the support vector machine classifier: Linear, Quadratic, and Cubic. The proposed magnetic resonance images classification approach was tested using confusion matrices and receiver operating characteristic curves, which revealed satisfactory performance in terms of Sensitivity, Precision, Specificity and Accuracy. The obtained results show that the BoF-SVMs combination achieves the best classification accuracy, with a recognition rate of 100%.

1. INTRODUCTION

Brain tumors are abnormal cell growths that form in the brain and can be benign or malignant, primary or secondary in origin. They can lead to serious complications for the health and well-being of patients [1]. It is therefore vital to detect them as early and accurately as possible, in order to make a correct diagnosis and apply the appropriate treatment. The magnetic resonance images (MRI) is a medical imaging technique that provides detailed images of the human brain [2]. These images can be used to diagnose brain pathologies, such as tumors that affect brain structures and functions. Once a tumor is identified, a biopsy can be performed to determine its nature, whether it is benign or malignant. The choice of treatment for brain tumors depends on various factors, such as the type and size of the tumor, as well as the overall health status of the patient. Therapeutic options include surgery, radiotherapy, and chemotherapy, among other possibilities depending on the specific situation of each patient [3]. It is therefore necessary to develop automated methods for classifying MRI images of the brain, enabling healthy tissue to be distinguished from tumor tissue.

The MRI brain analysis is essential for assessing, predicting, and monitoring the evolution of various neurological conditions, such as stroke, cancer, degenerative diseases, or congenital malformations. This analysis involves recognizing and marking areas of interest in MRI, according to clinical or biological criteria [4, 5]. The MRI brain analysis can be facilitated and optimized by the use of artificial intelligence (AI) and image processing. These tools can improve diagnostic accuracy by more effectively distinguishing neoplastic lesions from non-neoplastic lesions, thereby reducing diagnostic errors. Moreover, they can facilitate the tumor monitoring evolution over time by quantitatively analyzing radiographic images, which can be crucial for adjusting treatments and predicting treatment response. By identifying biomarkers associated with treatment resistance or sensitivity, AI can also help to personalize therapies, optimizing therapeutic interventions for each patient. Finally, AI contributes to better management of medical imaging data by facilitating storage, analysis, and interpretation of data, providing clinicians with valuable tools for informed decision-making in the fight against cancer [6]. These tools enable images to be manipulated, for example by filtering, segmenting, or applying mathematical operations, to extract useful information [7-9]. AI also makes it possible to learn from labeled or unlabeled data, to identify patterns, classify elements, or make predictions [10].

AI has several tools at its disposal, including machine-learning (ML) techniques, which are widely used to identify and classify brain MRI [11]. These techniques involve training a mathematical model from a set of examples so that it can then perform tasks on new data. Automatic segmentation techniques of MRI images are crucial in diagnosing brain conditions. MRI allows us to refine tumor tissues segmentation to be able to distinguish necrotic cores, active cells, and edema, compared to healthy brain tissues. This high-precision segmentation is made possible by the soft tissue contrasts provided by MRI, which is not possible in standard radiographic images such as conventional X-rays. These advanced technologies offer promising prospects for significantly improving for early detection, therapeutic planning, and monitoring of brain tumors, thus enabling more precise and individualized interventions for patients [12, 13]. For example, supervised learning uses data with labels to learn and associate an input with an output. Unsupervised learning uses unlabeled data to learn how to group or represent data [14]. Reinforcement learning uses a system of agents that interact with their environment and receive rewards or penalties to learn how to optimize their behavior [15, 16].

Support vector machine (SVMs) and convolution neural network (CNN) are two types of ML models frequently used to analyze and classify brain MRI [17]. SVMs are models that attempt to find a boundary between two categories of examples, by increasing the distance between them. SVMs can perform binary or multiclass classification, using kernel functions to transform data into a richer feature space [18]. In-depth analysis of the structural learning components of SVMs, hyper parameter optimization, and model parameter selection improve the performance of SVMs [19]. This holistic approach enhances understanding of SVMs and paves the way for significant improvements in their use for supervised classification [20, 21]. CNNs are models that use successive layers of artificial neurons that perform local operations on images, such as convolution, subsampling, or activation. CNNs can perform classification, segmentation, or object detection in images, learning to extract hierarchical and invariant features [22].

This article aims to present an innovative approach for automatically detecting brain tumors from MRI images. Early detection is critical for improving chances of recovery and reducing sequelae. However, manual classification of MRI images for tumor detection is complex, time-consuming, and prone to errors. The introduction of ML techniques opens new perspectives for improving the accuracy, speed, and reliability of this detection. Despite these advancements, challenges persist in making this detection more precise and effective, particularly due to the variability of tumor characteristics, the complexity of surrounding brain structures, and the need to develop models capable of generalizing to different tumor types and imaging conditions. Our research focuses on leveraging CNN for the automatic extraction of features from MRI images, as well as using SVMs for image classification. The proposed approach is based on three steps: pre-processing, feature extraction, and classification. The pre-processing aims to improve MRI quality and reduce noise. The feature extraction consists of representing MRI by numerical vectors that capture information relevant to tumor detection. The classification consists of using a ML algorithm to distinguish MRI images containing tumors from those that do not. The performance evaluation of the proposed approach is tested under dataset of 3000 MRI images, where half of MRI are with

tumors and the other half of MRI are without tumors. In the feature extraction step, two techniques have been used, the bag of features (BoF) and the ReseNet50 then a comparison between them has been performed. In the last step, the obtained images were compared with the three different kernels for the SVM classifier: Linear, Quadratic, and Cubic. The obtained results show that the BoF-SVMs combination achieves the best classification accuracy, with a recognition rate of 100%. The advances and outstanding performances of this study in these areas are detailed below.

-When features extracted by Resnet50 are classified by SVM with a quadratic kernel, their classification performance is improves significantly compared to softmax classifier. This method stands out to increase its ability to discriminate features and improve the overall accuracy of the brain tumor detection system.

-The features extracted by the BoF method outperform previous methods and ResNet50 in terms of classification. This method provides a robust representation of the characteristics of brain MRI images, resulting in improved discriminative ability and generalization of the SVM model.

-The hybrid system between the BoF automatic extractor and the SVM classifier demonstrates its ability to classify brain MRI images with high precision, without requiring prior segmentation or data augmentation. This approach significantly simplifies the brain tumor detection process while maintaining high performance.

-The features extracted by ResNet50 and BoF are well classified by SVM without requiring data augmentation. This approach results in simpler and faster models, which are crucial for effective clinical application in terms of processing time and required computational resources.

The rest of this paper is organized as follows: section II, presents the background theory of the BoF, SVM, k-folds, CNN and ResNet50 methods. In section III, we present the methodologies of Database, MRI pre-processing, Feature extraction from MRI and Classification of MRI. In section IV we present the obtained results, discussions and conclusion with perspectives.

2. BACK GROUND THEORIES OF THE USED APPROACHES

2.1 Bag of features

The feature bag produces the output object from the input image samples. Speeded-Up Robust Features (SURF), which are obtained from the image samples, form the default visual vocabulary given in Figure 1. The BoF technique is adapted to computer vision based on natural language processing. As the images do not contain distinct features, we first use the "extract features" function specific to each image category [23].

To extract features from images, simply call the "BoF" function, which:
-Extract SURF from images.
-Create feature vectors by reducing feature volume by quantifying feature space using K-means clustering.

The feature bag is created as follows:

Grid method: This is a procedure that selects the location of important objects in an image by cutting it into a grid of cells and samples points in each cell. This procedure reduces the number of features to be taken into account and distributes them more evenly [24]. To change the grid straight lines into

a grid curve we use Eq. (1) and Eq. (2).
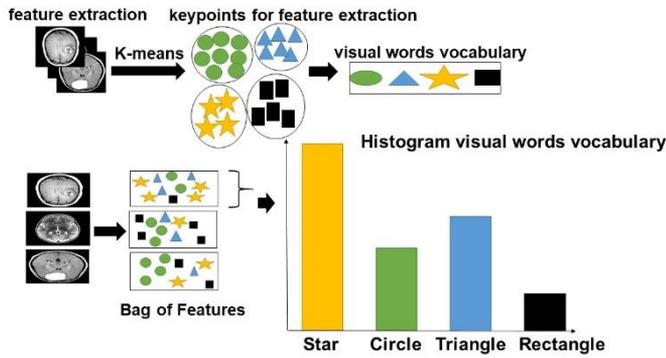
Figure 1. Illustration of feature extraction using the BoF technique

$$x = X(x', y') = \sum_{i=0}^{d} \sum_{j=0}^{d-i} u_{i,j} b_j(y') b_i(x') \qquad (1)$$

$$y = Y(x', y') = \sum_{i=0}^{d} \sum_{j=0}^{d-i} v_{i,j} b_j(y') b_i(x') \qquad (2)$$

where, x and y are the grid curve coordinates, x' and y' are the straight grid coordinates, $u_{i,j}$, $v_{i,j}$ are the control points of the Bézier surface, d is the Bézier surface degree and b_i are the Bernstein polynomials.

SURF: properties consisting of two parts: detector and decryptor. The detector finds the object positions to identify the salient features of the analyzed image, the decryptor uses the Hessian matrix. The decryptor then produces a vector describing the texture and orientation of pixels close to the points of interest [24, 25]. These vectors can be studied for comparison to determine similarities between different images, using Haar wavelet responses to describe the local neighborhood of each point (Eq. (3)).

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \qquad (3)$$

where, $L_{xx}(x, \sigma)$, $L_{xy}(x, \sigma)$, $L_{yy}(x, \sigma)$ are the second-order Gaussian partial derivatives at point x, and the scale σ.

The Haar wavelet response is acquired by Eq. (4):

$$\sum_{i=1}^{n} \sum_{i=1}^{n} w(i, j) I(x + i, y + j) \qquad (4)$$

where, $w(i, j)$ is the wavelet filter, and $I(x + i, y + j)$ is the image intensity at point $(x + i, y + j)$.

k-means: Group features into set of common words. Clustering is the process of dividing data set into homogeneous groups, called clusters. The k-means algorithm aims to minimize the sum of the squared distances between each data point and the center of the cluster to which it belongs [26]. The mathematical formula for this sum is given by Eq. (5).

$$\sum_{i=1}^{n} \min(\|x_i - \mu_j\|^2) \qquad (5)$$

where, n is the number of data items, x_i is the i-th data item and μ_j is the center of the j-th cluster.

Creating a histogram of visual words to describe the image as a vector is given by Eq. (6).

$$t_{id} = \frac{n_{id}}{n_d} log \frac{N}{n_i} \qquad (6)$$

where, t_{id} : histogram bin of word i for image d, n_{id} : occurances of word i in image d, n_d : number of word occurances in image d, n_i: number of images that contain word i, N: number of images.

2.2 Support vector machine

The SVM is an automatic algorithm used for classification, regression and data quality control. It does this by designing the best hyperplane that separates information into several classes or estimates the way they are output [27] (Figure 2). The mathematical formula for linear SVM is given by Eq. (7).

$$f(x) = wx + b \qquad (7)$$

For each training sample x_i, the function gives f(x_i) ≥ 0, for y_i = +1 and $f(x_i) \leq 0$ for $y_i = -1$. where w is the weight vector, b is the bias and x_i is the data.
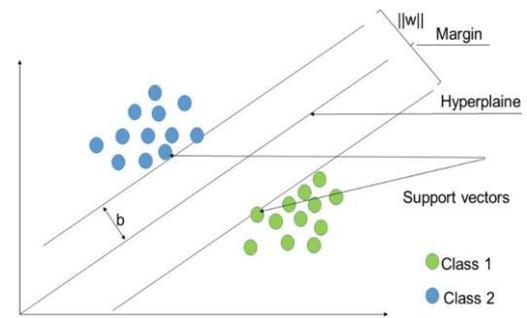
Figure 2. Data classification using SVM

In nonlinear SVM, the kernel function is used to transform the input data into a higher-dimensional. The kernel can capture complex relations between data features, enabling classes to be separated into higher-dimensional spaces [28].

2.3 K- folds cross validation

The k-fold method is a cross-validation procedure used to measure the performance of a ML model. It involves separating the data into k folds of equal size. The following operation is then performed k times: one fold is selected as the test set, and the other k folds are used as the training set (Figure 3). The model is adjusted on the learning set, and its error on the test set is evaluated. This gives us k errors, one per fold. The mean and standard deviation of the errors (Eq. (8) and Eq. (9)) can then be calculated to estimate the model's bias and variance. The k-fold method reduces the risk of over- or under-learning, by using all available data for training and testing, and avoiding dependence on a single random distribution of data. It also makes a possibility to compare different models or different parameters using the same validation protocol. The choice of k value depends on the amount of data available and the computation time required. In general, values between 5 and 10 are commonly used [29].
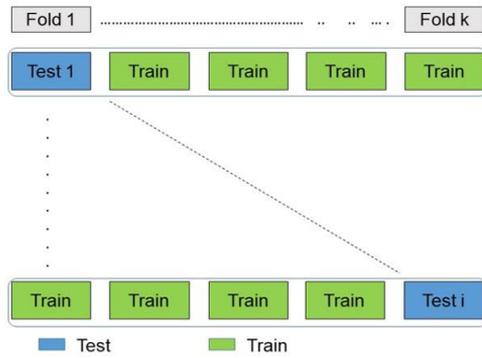
Figure 3. Data classification using SVM

N is the total number of data, k the number of folds, and n the size of each fold. The relation between these three quantities is N= kn.

The mathematical formula for calculating the average error is:

$$E = \frac{1}{k}\sum_{i=1}^{k} E_i \qquad (8)$$

The standard deviation errors are:

$$\sigma_E = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(E_i - E)^2} \qquad (9)$$

E is the average model error on k folds. k is the number of folds used for cross-validation. E_i is the model error on the i-th fold, calculated as the difference between the real value and the value predicted by the model. σ_E is the standard deviation of the model errors on the k folds.

2.4 Resnet50

The ResNet-50 is a CNN 50 layers deep with residual connections, making it easier to learn deep patterns. It can classify images into 1000 object categories (Figure 4).
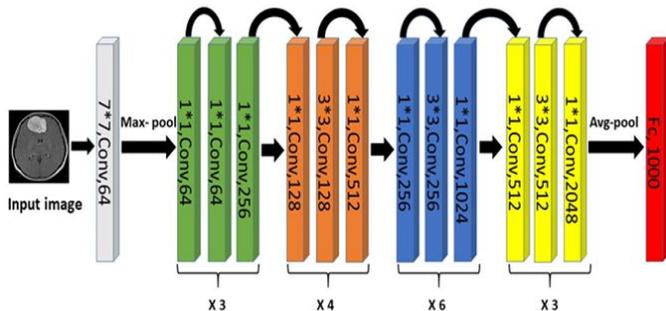
Figure 4. Architecture of ResNet50 model

ResNet-50 consists of five stages containing a variable number of residual blocks. The first stage has a convolution layer and a max-pooling layer. The second stage has three residual blocks with three convolution layers each. The third stage has four residual blocks with three convolution layers each. The fourth stage has six residual blocks with three convolution layers each [30]. The fifth stage has three residual blocks with three convolution layers each. The final layer is an

average-pooling layer followed by a fully-connected (fc) layer [31]. ResNet50 contains the following elements:

-Convolution with the Kernel size of 7*7 and 64 different Kernels, all of them are with step size 2, giving us 1 layer.

-Then we obtain the maximum pooling with stride size 2.

-In the next convolution, we get Kernel size of 1*1,64, then Kernel size of 3*3,64, and finally Kernel size of 1*1,256. These three layers are repeated 3 times totalizing 9 layers.

-Similarly, we do for Kernel size of 1*1,128, then for Kernel size of 3*3,128 and finally for Kernel size of 1*1,512. These three layers are repeated 4 times totalizing 12 layers.

-In the same manner we do for Kernel size of 1*1,256, then for Kernel size of 3*3,256 and finally for Kernel size of 1*1,1024. These three layers are 6 times totalizing 18 layers.

-We continue for Kernel size of 1*1,512, then for Kernel size of 3*3,512 and finally for Kernel size of 1*1,2048. These three layers are repeated 3 times totalizing 9 layers.

Finally, we estimate the average pool to end up with a fc layer containing 1000 nodes. Finally, the Softmax function provides us 1 layer [32, 33].

3. METHODOLOGY

Figure 5 shows the steps of the process to identify brain tumors from MRI of the brain. To achieve this, we considered a set of 3000 MRI images, half with tumors and half without. We first improved image quality by eliminating noise through a pre-processing step. Then, we extracted features from the images using two different techniques: BoF and ResNet50 neural network techniques. We evaluated the performance of both techniques using K-fold cross-validation and the SVM classifier with three different kernels: linear, cubic, and quadratic.
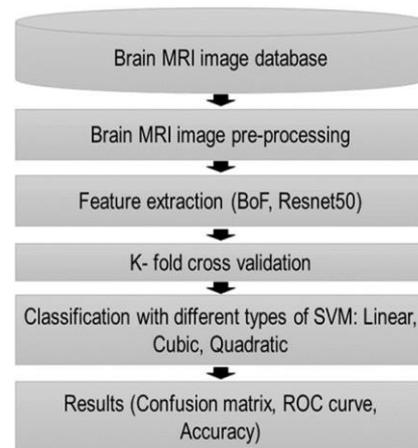
Figure 5. Automatic detection of brain tumors from MRI: Comparison between BoF and ResNet50 network techniques

3.1 Data base

In this article, an open-source dataset available on Kaggle is used for the classification of brain MRI images [34]. This dataset comprises three distinct folders: the first containing 1500 MRI images with tumor, the second containing 1500 MRI images without tumor, and the third containing 60 unlabeled MRI images (Figure 6). We chose to use only the first two folders, totaling 3000 images. The images were provided in sagittal, axial, and coronal planes and were in JPG format with RGB color and varying dimensions.
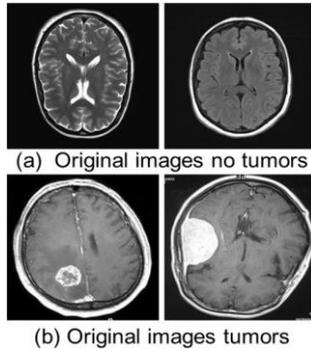
Figure 6. Examples of MRI: a) No tumor b) Tumor brain

3.2 MRI pre-processing

The aim of pre-processing MRI is to sharpen them and reduce the noise that can interfere with classification [35]. When preparing the brain MRI images for classification, we resized the images to a uniform size of 224×224 pixels. This resizing ensures consistency in image dimensions, which is essential for effective processing by the classification model. Additionally, the images were converted to grayscale to simplify processing and reduce data dimensionality. This conversion allows us to focus on pixel intensity variations, which are crucial for detecting structures and important features in MRI images.

3.3 Feature extraction from MRI

In this step, feature extraction from MRI images is used with two distinct methods, namely BoF and ResNet50.

BoF: used to build a dictionary of visual words from a set of images shown in Figure 7. A visual word is an image chunk that corresponds to an area of interest. This dictionary is created using a clustering algorithm, such as K-means, on the image chunks extracted from the images. Each image is then described by a frequency histogram of the visual words it contains. This histogram forms the image feature vector, which contains 500 numerical values [36, 37]. As we used a set of 3000 MRI images, we obtained a matrix of size [3000 500], where 3000 is the number of images and 500 is the number of features in each image. This matrix is then used as input for the SVM classifier.
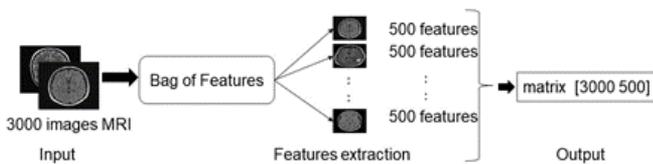
Figure 7. Used configuration of BoF for feature extraction from MRI

The SURF local feature extraction procedure is used to extract interest points and descriptors from an image. Here's how it works with the parameters we used:

Vocabulary size: We specified 'Vocabulary Size' = 500, which means that the clustering algorithm (k-means) used to group the extracted features will create a set of 500 visual words for each image.

Strongest Features = 0.8: The algorithm will select the top 80% of the strongest features from each image to form the visual vocabulary.

Grid: A grid is used to divide the image into regions before extracting features. This can help capture local information at different scales. With a GridStep of [8 8], the image is divided into regions of 8×8 pixels.

Block width parameter is used for feature description in the SURF method. By specifying a multi-scale approach of size [32, 64, 96, 128] to extract features at different resolutions:

-A block of size 32×32 pixels is used to extract features at a finer scale, capturing smaller details in the image.

-A block of size 64×64 pixels is used to extract features at a slightly larger scale, allowing to capture details slightly larger than the previous block.

-A block of size 96×96 pixels is used to extract features at an even larger scale, capturing larger details in the image.

-A block of size 128×128 pixels is used to extract features at the largest scale among those specified, capturing the largest details in the image.

ResNet50: we use a deep neural network that has been trained on a large collection of images, such as ImageNet. This network has 50 layers and can identify over 1000 object types. We use this network to extract features from the image. The output is the fc1000 layer, which is the last fc layer and contains 1000 numbers [38]. This vector corresponds to the high-level features of the image. Since we used a set of 3000 MRI images, we obtained a matrix of dimension [3000 1000], where 3000 is the number of images and 1000 is the number of features for each image shown in Figure 8. This matrix is then used as input for the SVM classifier.
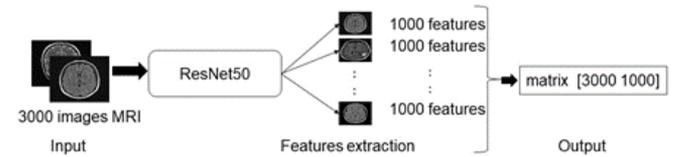
Figure 8. Used configuration of ResNet50 for feature extraction from MRI

The initial layer uses 7×7 filters with 64 filters to reduce the image size. Then, a max pooling operation with a 3×3 window and a stride of 2 is applied to further reduce dimensionality. The following convolutional layers use 1x×1, 3×3, and 1×1 filters with 64, 64, and 256 filters respectively for each kernel size. These layers are repeated several times (3 times for the first set, 4 times for the second set, 6 times for the third set and 3 times for the fourth set) to gradually increase the complexity of the extracted features. The final layer of the network is a fc layer with 1000 nodes. ReLU activation is used after each convolution operation, and batch normalization layers are included. The 'MiniBatchSize' parameter controls the number of images processed simultaneously during feature extraction. In our case, a 'MiniBatchSize' of 32 means that 32 images are processed together at each iteration. The 'OutputAs' parameter determines the output format of the extracted features. It is set to 'rows', which means that the extracted features are returned as rows in the matrix. Each row of this matrix represents the extracted features from a single image.

3.4 MRI classification

This step consists of classifying the MRI into two categories: healthy or tumoral. For this, we used the SVM classifier with three different kernels: linear, cubic, and quadratic [39, 40]. The K-fold cross-validation technique is

also used to evaluate model performance, as shown in Figure 9. We choose to divide the 3000 MRI images (1500 healthy and 1500 tumor) into 5 folds of 600 images (300 healthy and 300 tumor) each, to create the 600-image fold, the stratified k-fold cross-validation technique will randomly distribute the 300 healthy images and 300 tumor images. In this way, each fold will have a diversity of images representative of the data. Using the cross-validation technique with K = 5. Next, the SVM model with three different kernel types is applied to each group, based on features obtained from BoF and ResNet50.

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)
600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)
600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)
600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)
600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)	600 MRI (300 H+300 T)

■ Test　　■ Train　　H: Healthy　　T: Tumor

Figure 9. Diagram of the K-fold cross-validation technique (creates a random partition for stratified k-fold cross-validation. Each fold has the same number of MRI and contains the same type of class, health and tumor categories)

The function that defines the hyperplane that best separates the data into two classes can have different forms depending on the degree of the polynomial that represents it. We mean cubic, quadratic or linear function, depending on whether the degree is three, two or one [39].

Depending on the nature of the data to be classified, the function defining the hyperplane varies. A linear function is used if the data are linearly separable, that is a straight line can divide them without error. A non-linear function, such as a quadratic or cubic function, is used if the data are non-linearly separable, that is no straight line can divide them without error [40].

The following parameters are used to configure the SVM models:

Preset: Defines the type of SVM used, such as linear, cubic, or quadratic. This determines the shape of the decision function used to separate classes in the feature space.

Kernel function: Specifies the kernel function used to transform the feature space to make class separation easier in a higher-dimensional space.

Kernel scale: Controls the scaling of the kernel, which can affect the model's flexibility. Automatic scaling can be used to optimize model performance.

Box constraint level: Determines the model's regularization by specifying the constraint on the support vectors. A higher value indicates stronger regularization.

Multiclass method: Defines the method used to handle multi-class classification problems. The One-vs-One method trains a binary classifier for each pair of classes, while the One-vs-All method trains a classifier for each class against all other classes.

Standardize data: Indicates whether the data should be standardized before training the model.

The types of SVM models used share similarities in their parameters, except for the 'Preset' and 'Kernel function' parameters, which vary depending on the type of SVM.

For the linear SVM, the 'Preset' parameter is set to Linear SVM, and the kernel function is linear. The 'Box constraint level' parameter is set to 1, indicating moderate constraint on the support vectors. The model uses the One-vs-One method to handle multiple classes, which involves training a binary classifier for each pair of classes. Finally, the data were standardized before model training, with features scaled to have zero mean and unit variance.

For the cubic SVM, the 'Preset' parameter is set to cubic SVM, and the kernel function is cubic. The other parameters, such as Kernel scale, Box constraint level, Multiclass method, and Standardize data, are the same as for the linear SVM.

Finally, for the quadratic SVM, the 'Preset' parameter is set to quadratic SVM, and the kernel function is quadratic. The other parameters, such as Kernel scale, Box constraint level, Multiclass method, and Standardize data, are the same as for the linear SVM.

To assess the performance of each model, we calculated the CM and plotted the corresponding ROC curves. The obtained results are given in section V.

4. PERFORMANCE EVALUATION

4.1 Confusion matrix

The accuracy of a classification algorithm, such as SVMs, can be evaluated using the confusion matrix (CM). This tool compares the classes predicted by the algorithm with the real classes in the data. Figure 10 shows the CM for a two-class problem.

	Predicted class 1	Predicted class 2
True class 1	True positives (TP)	False positives (FP)
True class 2	False negatives (FN)	True negatives (TN)

Figure 10. CM for a two-class problem

The diagonal contains the elements that correspond to a correct prediction by the algorithm: true positives (TP) and true negatives (TN). On the other hand, Elements that are not on the main diagonal are erroneous predictions, that's to say elements that have been classified to a class that does not correspond to their real class. These are known as false positives (FP) and false negatives (FN).

The performance of a classification model can be evaluated using the CM, which can be used to calculate different indicators, such as:

Accuracy: Percentage of data that is correctly classified in relation to the total number of data. It indicates the success rate of classifications.

$$\text{Accuracy} = \frac{TP + TN}{TN + TP + FP + FN}$$

Sensitivity is the ratio of the number of correctly identified positives and the total number of positive cases. It indicates performance in recognizing all positives.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

Precision: The proportion between the number of correct results (TP) and the total number of affirmative results (the predicted positive). It represents the validity of positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Specificity: The model's ability to correctly identify non-positive examples is measured by specificity. It measures a model's ability to avoid false positives.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

4.2 Receiver operating characteristic curves

A graphical tool for measuring the performance of a binary classifier is an algorithm that assigns each piece of data positive or negative label according to a criterion. For example, in our case, the SVM classifier can separate MRI into two categories: normal or abnormal.

The receiver operating characteristic (ROC) graph shows the relationship between the true positive rate (TPR) and the false positive rate (FPR) for different classifier threshold levels. TPR is the fraction of positive data that are correctly identified and FPR is the fraction of negative data that are falsely attributed as positive. The classifier performs best when the TPR is close to 1 and the FPR is close to 0.

The following formulas are used to calculate TPR and FPR:

$$\text{TPR} = \frac{TP}{TP + FN} = \text{Sensitivity}$$

$$\text{FPR} = \frac{FP}{FP + TN} = 1 - \text{Specificity}$$

Classifier performance can be measured by the area under the curve (AUC), which indicates the probability that the classifier will give a higher score to positive data than to negative data, drawn at random. The classifier performs best when the AUC is close to 1.

5. RESULTS AND DISCUSSION

The measured performance of the six different models, using two feature extraction techniques (BoF and ResNet50) and three SVM kernels (linear, cubic, and quadratic) is studied. We employed K-fold cross-validation with K = 5 to estimate the accuracy of each model. The obtained results are given in the form of CM and ROC curves. Finally, a comparison study of the six developed models is presented.

The three SVM models using features extracted with the ResNet50:

ResNet50 features - Linear SVM: Figure 11 shows that the model's CM correctly classified 1444 healthy images and 1442 tumor images, but produced 56 FP errors (healthy images classified as tumors) and 58 FN errors (tumor images classified as healthy). Consequently, the accuracy of the model is (1441 + 1444) / 3000 = 0.962 = 96.2%.

Table 1 shows the performance of classification using ResNet50 features - Linear SVM.

Figure 12 shows the ROC curve evaluating the model's performance in classifying MRI into two categories. The orange point on the curve represents the model's current threshold, which has an FPR of 0.04 and a TPR of 0.96. This implies that the model correctly identifies 96% of images with a tumor, but also confuses 4% of tumor-free images with tumor images. The AUC is an indicator that summarizes the overall performance of the model, irrespective of the threshold chosen. The closer the AUC is to 1, the more efficient is the model. For the proposed approach, the obtained AUC is 0.99, which means that the model has a good ability to distinguish between the two classes.

ResNet50 features -Quadratic SVM: Figure 13 shows that the model's CM correctly classified 1480 healthy images and 1479 tumor images, but produced 20 FP errors (healthy images classified as tumors) and 21 FN errors (tumor images classified as healthy). The accuracy of the model is therefore (1480 + 1479) / 3000 = 0.986 = 98.6%.
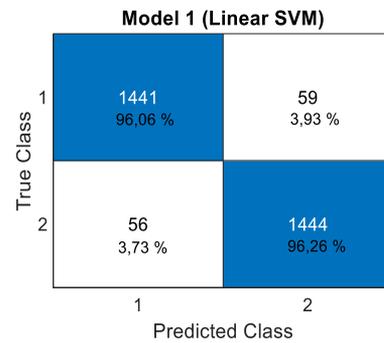
Figure 11. CM of ResNet50 features - Linear SVM for automatic brain tumor detection from MRI
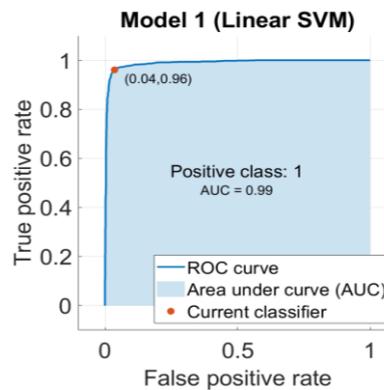
Figure 12. ROC curve for ResNet50 features - Linear SVM for automatic brain tumor detection from MRI

Table 1. Results of MRI classification using ResNet50 features - Linear SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1444	56	1442	58	96.26	96.06	96.26	96.2

Table 2. Results of MRI classification using ResNet50 Features-Quadratic SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1480	20	1479	21	98.6	98.66	98.66	98.6

Table 3. Results of MRI classification using ResNet50 features- Cubic SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1474	26	1479	21	98.6	98.46	98.27	98.5
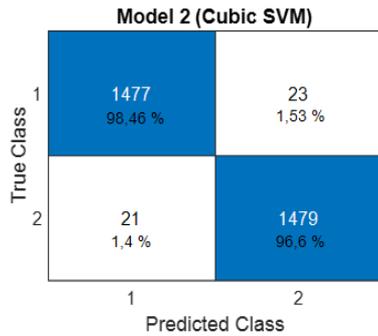
Figure 13. CM of ResNet50 features - Cubic SVM for automatic brain tumor detection from MRI
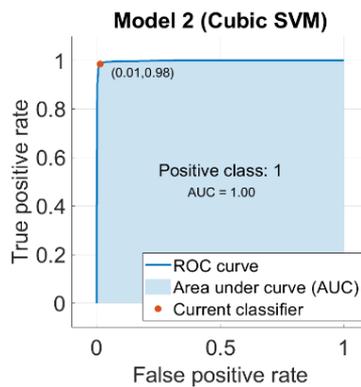
Figure 14. ROC curve for ResNet50 features - Cubic SVM for automatic brain tumor detection from MRI

Figure 14 illustrates the performance of the ResNet50 features-quadratic SVM model for binary classification of MRI. The red dot indicates the current threshold, which has an FPR of 0.01 and a TPR of 0.99. This implies that the model correctly identifies 99% of images with a tumor, but also makes an error on 1% of images without a tumor. The AUC is 1, which is the optimal value and indicates a good distinguishability between the two classes.

Table 2 resumes the performance of classification using ResNet50 Features-Quadratic SVM.

Table 3 resumes the performance of classification using ResNet50 features- Cubic SVM.

ResNet50 features- Cubic SVM: this model succeeded in identifying 1474 healthy images and 1479 tumoral images, as shown in Figure 15, which presents its CM. However, it made 26 FP errors (healthy images labeled as tumoral) and 21 FN errors (tumoral images labeled as healthy). This model therefore has an accuracy of (1477 + 1479) / 3000 = 0.985 = 98.5%. Its performance is comparable to that of the ResNet50 features - SVM quadratic model, which has an accuracy of 0.986. This indicates that both cubic and quadratic kernels are appropriate for the data, and that they allow us to define efficient separating hyperplanes.

Figure 16 illustrates the performance of the ResNet50 features - SVM model in the form of an ROC curve. The orange dot on the curve indicates the model's current threshold, which has an FPR of 0.01 and a TPR of 0.98. This means that the model correctly identifies 98% of images with tumors, but also confuses 1% of non-tumor images with tumor images. The AUC is 1, which is the optimal value and shows the ability of the model to distinguish between the two classes.
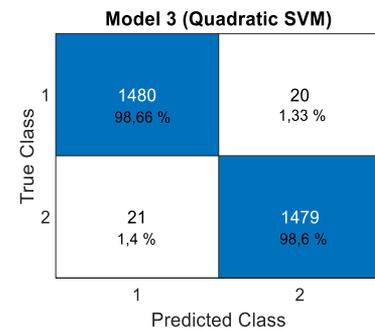
Figure 15. CM of ResNet50 features - Quadratic SVM for automatic brain tumor detection from MRI
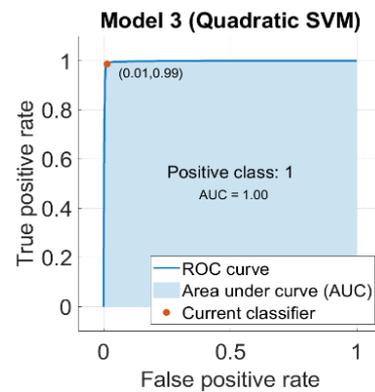
Figure 16. ROC curve for ResNet50 Features - Quadratic SVM for automatic brain tumor detection from MRI
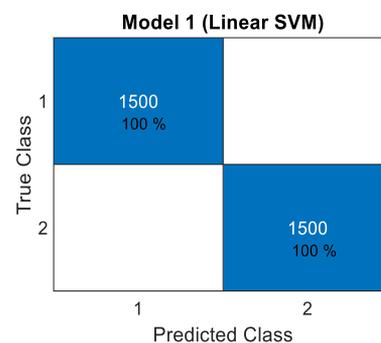
Figure 17. CM of BoF - Linear SVM for automatic brain tumor detection from MRI

Table 4. Results of MRI classification using BoF - Linear SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1500	0	1500	0	100	100	100	100

The three SVM models using features extracted BoF:

BoF - Linear SVM: this model succeeded in correctly classifying all the images in the dataset, without making any false predictions. Figure 17 shows the CM for this model, which displays 1500 healthy images and 1500 tumor images on the main diagonal. The accuracy of this model is therefore 100% (1500 + 1500) / 3000 = 1 = 100%), indicating that it performs optimally and that there is no overlap between the two classes.

Table 4 shows the performance of classification using BoF - Linear SVM.

The ROC curve in Figure 18 shows the model's performance. The model uses the current threshold indicated by the orange dot on the curve, which has an FPR of 0.00 and a TPR of 1.00. This means that the model correctly identifies all tumor images and that there is no FP among healthy images. The AUC is 1, which is the best possible score and indicates that the model has the ability to distinguish between the two classes perfectly.

BoF - Quadratic SVM and BoF - Cubic SVM: The BoF - Linear SVM model delivered optimal results, correctly classifying all images, whether healthy or tumoral. It achieved 100% accuracy. Neither of the other two models were able to distinguish it, and both achieved the same results. Figures 19 and 20 show the CM and ROC curves of BoF - Quadratic SVM technique respectively. Figures 21 and 22 show the CM and ROC curves, respectively, for the BoF - Cubic SVM technique.
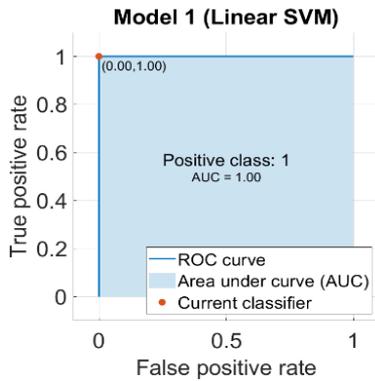
Figure 18. ROC curve for BoF - Linear SVM for automatic brain tumor detection from MRI
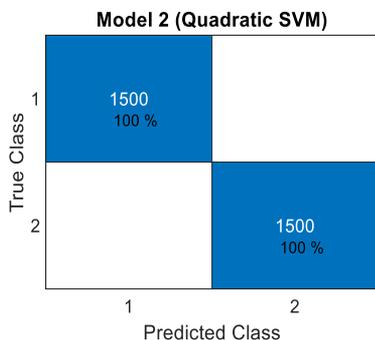
Figure 19. CM of BoF - Quadratic SVM for automatic brain tumor detection from MRI
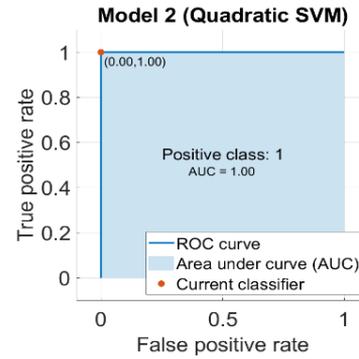
Figure 20. ROC curve for BoF - Quadratic SVM for automatic brain tumor detection from MRI
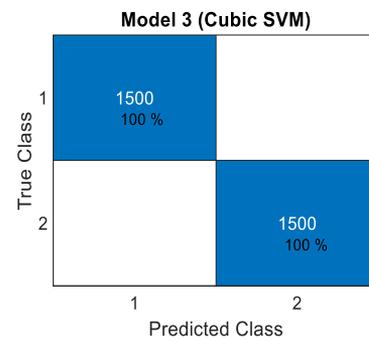
Figure 21. CM of BoF - Cubic SVM for automatic brain tumor detection from MRI
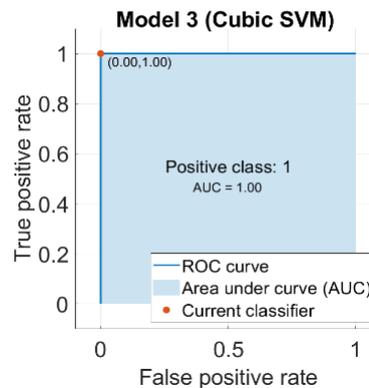
Figure 22. ROC curve for BoF - Cubic SVM for automatic brain tumor detection from MRI

Table 5 shows the performance of classification using BoF - Quadratic SVM. Table 6 shows the performance of classification using BoF - Cubic SVM.

Figure 23 presents the results of the comparative analysis of all models applied to the dataset. The performances of these six models for classifying brain MRI images are presented in Table 7. BoF models achieve perfect sensitivity, specificity, precision, and accuracy, indicating the absence of errors. ResNet50 models also performe well, with high values for all metrics. The high values of specificity and precision suggest that the models are reliable for correctly classifying positive cases.

Table 5. Results of MRI classification using BoF - Quadratic SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1500	0	1500	0	100	100	100	100

Table 6. Results of MRI classification using BoF - Cubic SVM

Database	TP	FP	TN	FN	Sensitivity%	Precision%	Specificity%	Accuracy%
3000 MRI	1500	0	1500	0	100	100	100	100

Table 7. Performance of the six models for the classification of brain MRI images

Model	Sensitiviy %	Specificiy %	Precision %	Accuracy %
ResNet50 - Linear SVM	96.26	96.26	96.06	96.2
ResNet50 - Quadratic SVM	98.6	98.66	98.66	98.6
ResNet50 - Cubic SVM	98.6	98.27	98.46	98.5
BoF - Linear SVM	100	100	100	100
BoF - Quadratic SVM	100	100	100	100
BoF - Cubic SVM	100	100	100	100
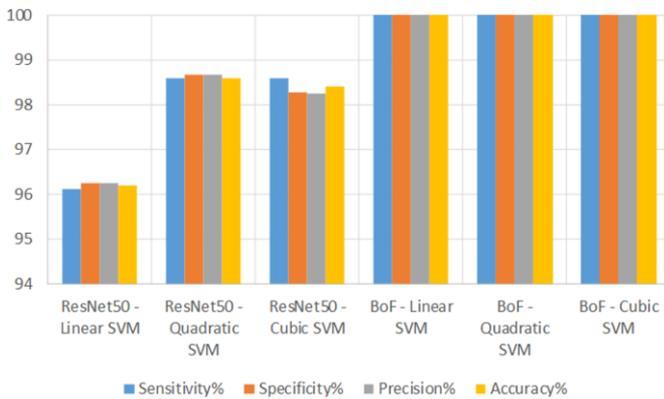
Figure 23. Comparative performance of the six models for brain MRI image classification

Table 8. Performance comparison of different techniques for automatic detection of brain tumors from MRI

Reference	Model	Accuracy (%)
Proposed	BoF - Linear SVM	100
	BoF - Quadratic SVM	100
	BoF - Cubic SVM	100
	ResNet50 - Linear SVM	96.2
	ResNet50 - Quadratic SVM	98.6
	ResNet50 - Cubic SVM	98.4
[41]	CNN Model	96
[42]	CNN	93
[43]	ResNet50_SVM	98.28

BoF models extract local features but ignore spatial relationships between features. They treat each feature independently, which may be crucial for accurate classification. Furthermore, deep learning models such as ResNet50 and BoF models are relatively simple, which can sometimes lead to better accuracy, especially when the dataset is not very large, as in our approach. Deep learning models often require larger datasets to learn complex representations.

Table 8 illustrates the comparison between the six methods used in this study and prior approaches.

From study [41], a CNN model is proposed for the segmentation of brain tumor MRI images into two classes: with tumors and without tumors. This architecture was trained and validated on a dataset of 3000 high-resolution MRI images. These medical images underwent preprocessing and

resizing before being processed by the CNN. The CNN architecture used in this study includes several layers, including convolutional layers for feature extraction, pooling layers for dimensionality reduction, fc layers for classification, and dropout layers to reduce overfitting. The overall accuracy of the model is 96%. Our hybrid method first extracts feature from the images using ResNet50, then uses these features for classification by quadratic SVM, which can capture more complex relationships between features and classes, leading to better class separation in feature space. This approach achieved an overall accuracy of 98.6%.

Another study [42] demonstrated that CNNs are effective in diagnosing brain tumors on MRI images, achieving an accuracy of 93%. Before training the CNN model, significant steps were taken to prepare the images. The first step was data augmentation, where each image with a tumor was transformed into 6 images, and each image without a tumor was transformed into 9 images, totaling 2065 images. Then, image preprocessing was performed to normalize the sizes and contrasts of the images, resize them to a standard size of (240, 240, 3), and normalize them to facilitate learning. The CNN model used in the study includes several layers, including convolutional, pooling, flattening, dropout, and dense layers, using convolutional layers with 3x3 kernels and 32 filters. After convolution and pooling operations, the results are passed to fc layers for the final classification of the images. Our method, combining a CNN model followed by quadratic SVM classification, achieved an overall accuracy of 98.6%, surpassing the results obtained by the data augmentation approach. This approach has the advantage of reducing computational complexity and training time while maintaining exceptional classification performance.

Kuraparthi et al. [43] proposed an effective transfer learning process applied to three pretrained models - AlexNet, ResNet-50, and VGG-16 - for classifying brain tumors. The performances of the three models are evaluated based on performance criteria. Simulation results show that ResNet50 outperforms the other two networks in classifying brain tumors. The proposed model, combining the Kaggle and BRATS datasets, achieved the best classification accuracy, reaching an accuracy of 98.28%, with reduced computation time after training the framework with data augmentation and the SVM classifier. In our study, the ResNet50 architecture used is the same as in the reference article, but three types of SVM models with different kernels are used. The quadratic

SVM model yielded better results, achieving an accuracy of 98.6%, surpassing the reference article.

6. CONCLUSION

In this study we presented two automatic brain tumor detection techniques from MRI, comparing the use of the BoF method with linear, cubic, and quadratic SVM kernels and the use of the ResNet50 network followed by linear SVM classification. The obtained results show that the BoF technique with linear, cubic, and quadratic SVM kernels achieved a classification accuracy of 100%, while the ResNet50-linear SVM approach showed slightly lower performance. The key contributions lie in demonstrating the effectiveness of the BoF technique in combination with SVMs for brain tumor classification. This approach proved to be simpler and faster than using the ResNet50 network, while still offering exceptional classification performance. The practical implications of our study lie in the clinical practice field, where a simpler and faster method of brain tumor detection can be extremely beneficial for patients. It can also have an impact on the industry by reducing the costs and resources required for MRI analysis.

However, our proposed approach has some limitations, especially regarding sample size and data diversity. These factors may have influenced our results and should be considered in interpreting the conclusions. Compared to other previous studies, the power of our approach is its accuracy classification (98%) surpassing the results obtained by other existing methods. In particular, our hybrid method combining feature extraction by ResNet50 followed by quadratic SVM classification outperformed the CNN models used in other studies, achieving an overall accuracy of 98.6%. These results suggest that our approach may be more effective for automatic brain tumor detection from MRI.

For future research, it would be interesting to extend this study for a larger number of clinical cases and compare more classification techniques.

REFERENCES

[1] Li, S., Wang, C., Chen, J., Lan, Y., Zhang, W., Kang, Z., Zheng, Y., Zhang, R., Yu, J., Li, W. (2023). Signaling pathways in brain tumors and therapeutic interventions. Signal Transduction and Targeted Therapy, 8(1): 8. https://doi.org/10.1038/s41392-022-01260-z

[2] Martucci, M., Russo, R., Schimperna, F., D'Apolito, G., Panfili, M., Grimaldi, A., Perna, A., Ferranti, A.M., Varcasia, G., Giordano, C., Gaudino, S. (2023). Magnetic resonance imaging of primary adult brain tumors: State of the art and future perspectives. Biomedicines, 11(2): 364. https://doi.org/10.3390/biomedicines11020364

[3] Nikam, R. M., Yue, X., Kaur, G., Kandula, V., Khair, A., Kecskemethy, H.H., Averill, L.W., Langhans, S.A. (2022). Advanced neuroimaging approaches to pediatric brain tumors. Cancers, 14(14): 3401. https://doi.org/10.3390/cancers14143401

[4] Mohan, G., Subashini, M.M. (2018). MRI based medical image analysis: Survey on brain tumor grade classification. Biomedical Signal Processing and Control, 39: 139-161.

https://doi.org/10.1016/j.bspc.2017.07.007

[5] Abd-Ellah, M.K., Awad, A. I., Khalaf, A.A.M., Hamed, H.F.A. (2019). A review on brain tumor diagnosis from MRI images: Practical implications, key achievements, and lessons learned. Magnetic Resonance Imaging, 61: 300-318. https://doi.org/10.1016/j.mri.2019.05.028

[6] Bi, W.L., Hosny, A., Schabath, M.B., et al. (2019). Artificial intelligence in cancer imaging: Clinical challenges and applications. CA: A Cancer Journal for Clinicians, 69(2): 127-157. https://doi.org/10.3322/caac.21552

[7] Suresha, D., Jagadisha, N., Shrisha, H.S., Kaushik, K.S. (2020). Detection of brain tumor using image processing. In 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, pp. 844-848. https://doi.org/10.1109/ICCMC48092.2020.ICCMC-000156

[8] Wadhwa, A., Bhardwaj, A., Singh Verma, V. (2019). A review on brain tumor segmentation of MRI images. Magnetic Resonance Imaging, 61: 247-259. https://doi.org/10.1016/j.mri.2019.05.043

[9] Devkota, B., Alsadoon, A., Prasad, P.W.C., Singh, A.K., Elchouemi, A. (2018). Image segmentation for early stage brain tumor detection using mathematical morphological reconstruction. Procedia Computer Science, 125: 115-123. https://doi.org/10.1016/j.procs.2017.12.017

[10] Adel, A., Hand, O., Fawzi, G., Walid, T., Chemseddine, R., Djamel, B. (2023). Gear fault detection, identification and classification using MLP Neural Network. In Recent Advances in Structural Health Monitoring and Engineering Structures, pp. 221-234. https://doi.org/10.1007/978-981-19-4835-0_18

[11] Gougam, F., Afia, A., Aitchikh, M.A., Touzout, W., Rahmoune, C., Benazzouz, D. (2024). Computer numerical control machine tool wear monitoring through a data-driven approach. Advances in Mechanical Engineering, 16(2): 16878132241229314. https://doi.org/10.1177/16878132241229314

[12] Amin, J., Sharif, M., Raza, M., Saba, T., Anjum, M.A. (2019). Brain tumor detection using statistical and machine learning method. Computer Methods and Programs in Biomedicine, 177: 69-79. https://doi.org/10.1016/j.cmpb.2019.05.015

[13] Gurusamy, R., Subramaniam, V. (2017). A machine learning approach for MRI brain tumor classification. Computers, Materials and Continua, 53(2): 91-109.

[14] Gougam, F., Chemseddine, R., Benazzouz, D., Benaggoune, K., Zerhouni, N. (2021). Fault prognostics of rolling element bearing based on feature extraction and supervised machine learning: Application to shaft wind turbine gearbox using vibration signal. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 235(20): 5186-5197. https://doi.org/10.1177/0954406220976154

[15] Gupta, N., Khanna, P. (2017). A non-invasive and adaptive CAD system to detect brain tumor from T2-weighted MRIs using customized Otsu's thresholding with prominent features and supervised learning. Signal Processing: Image Communication, 59: 18-26. https://doi.org/10.1016/j.image.2017.05.013

[16] Rundo, L., Militello, C., Tangherloni, A., Russo, G., Vitabile, S., Gilardi, M.C., Mauri, G. (2018). NeXt for

neuro-radiosurgery: A fully automatic approach for necrosis extraction in brain tumor MRI using an unsupervised machine learning technique. International Journal of Imaging Systems and Technology, 28(1): 21-37. https://doi.org/10.1002/ima.22253

[17] Agarap, A.F. (2017). An architecture combining convolutional neural network (CNN) and support vector machine (SVM) for image classification. arXiv preprint arXiv:1712.03541.
https://doi.org/10.48550/arXiv.1712.03541

[18] Afia, A., Gougam, F., Rahmoune, C., Touzout, W., Ouelmokhtar, H., Benazzouz, D. (2024). Intelligent fault classification of air compressors using Harris hawks optimization and machine learning algorithms. Transactions of the Institute of Measurement and Control, 46(2): 359-378. https://doi.org/10.1177/01423312231174939

[19] Afia, A., Gougam, F., Touzout, W., Rahmoune, C., Ouelmokhtar, H., Benazzouz, D. (2023). Spectral proper orthogonal decomposition and machine learning algorithms for bearing fault diagnosis. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 45(10): 550. https://doi.org/10.1007/s40430-023-04451-z

[20] Roman, I., Santana, R., Mendiburu, A., Lozano, J.A. (2021). In-depth analysis of SVM kernel learning and its components. Neural Computing and Applications, 33(12): 6575-6594. https://doi.org/10.1007/s00521-020-05419-z

[21] Gougam, F., Afia, A., Soualhi, A., Touzout, W., Rahmoune, C., Benazzouz, D. (2024). Bearing faults classification using a new approach of signal processing combined with machine learning algorithms. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 46(2): 65. https://doi.org/10.1007/s40430-023-04645-5

[22] Zhang, Y., Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv preprint arXiv:1510.03820.
https://doi.org/10.48550/arXiv.1510.03820

[23] O'Hara, S., Draper, B.A. (2011). Introduction to the bag of features paradigm for image classification and retrieval. arXiv preprint arXiv:1101.3354. https://doi.org/10.48550/arXiv.1101.3354

[24] Ashour, A.S., Eissa, M.M., Wahba, M.A., Elsawy, R.A., Elgnainy, H.F., Tolba, M.S., Mohamed, W.S. (2021). Ensemble-based bag of features for automated classification of normal and COVID-19 CXR images. Biomedical Signal Processing and Control, 68: 102656. https://doi.org/10.1016/j.bspc.2021.102656

[25] Surakarin, W., Chongstitvatana, P. (2015). Predicting types of clothing using SURF and LDP based on bag of features. In 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Hua Hin, Thailand, pp. 1-5. https://doi.org/10.1109/ECTICon.2015.7207101

[26] Cao, J., Wu, Z., Wu, J., Liu, W. (2013). Towards information-theoretic K-means clustering for image indexing. Signal Processing, 93(7): 2026-2037. https://doi.org/10.1016/j.sigpro.2012.07.030

[27] Iourzikene, Z., Benazzouz, D., Gougam, F. (2022). Edge detection of MRI brain images based on segmentation

and classification using support vector machines and neural networks pattern recognition. In International Conference on Artificial Intelligence in Renewable Energetic Systems, Tamanghasset, Algeria, pp. 99-105. https://doi.org/10.1007/978-3-031-21216-1_11

[28] Abd-Ellah, M.K., Awad, A.I., Khalaf, A.A., Hamed, H F. (2016). Classification of brain tumor MRIs using a kernel support vector machine. In Building Sustainable Health Ecosystems: 6th International Conference on Well-Being in the Information Society, WIS 2016, Tampere, Finland, pp. 151-160. https://doi.org/10.1007/978-3-319-44672-1_13

[29] Xiong, Z., Cui, Y., Liu, Z., Zhao, Y., Hu, M., Hu, J. (2020). Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation. Computational Materials Science, 171: 109203. https://doi.org/10.1016/j.commatsci.2019.109203

[30] Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A. (2019). CNN-generated images are surprisingly easy to spot... for now. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, pp. 8692-8701. https://doi.org/10.1109/CVPR42600.2020.00872

[31] Ayadi, W., Elhamzi, W., Charfi, I., Atri, M. (2021). Deep CNN for brain tumor classification. Neural Processing Letters, 53: 671-700. https://doi.org/10.1007/s11063-020-10398-2

[32] He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385.
https://doi.org/10.48550/arXiv.1512.03385

[33] Ikechukwu, A. V., Murali, S., Deepu, R., Shivamurthy, R.C. (2021). ResNet-50 vs VGG-19 vs training from scratch: A comparative analysis of the segmentation and classification of Pneumonia from chest X-ray images. Global Transitions Proceedings, 2(2): 375-381. https://doi.org/10.1016/j.gltp.2021.08.027

[34] Kaggle Database. https://www.kaggle.com/datasets/abhranta/brain-tumor-detection-mri.

[35] Borole, V.Y., Nimbhore, S.S., Kawthekar, D.S.S. (2015). Image processing techniques for brain tumor detection: A review. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 4(5): 28-32.

[36] Altaei, M.S.M., Kamil, S.Y. (2020). Brain tumor detection and classification using SIFT in MRI images. AIP Conference Proceedings, 2292(1): 030004. https://doi.org/10.1063/5.0031014

[37] Razzaq, S., Mubeen, N., Kiran, U., Asghar, M.A., Fawad, F. (2020). Brain tumor detection from MRI images using bag of features and deep neural network. In 2020 International Symposium on Recent Advances in Electrical Engineering & Computer Sciences (RAEE & CS), Islamabad, Pakistan, pp. 1-6. https://doi.org/10.1109/RAEECS50817.2020.9265768

[38] Bingol, H., Alatas, B. (2021). Classification of brain tumor images using deep learning methods. Turkish Journal of Science and Technology, 16(1): 137-143.

[39] Nandpuru, H.B., Salankar, S.S., Bora, V.R. (2014). MRI brain cancer classification using support vector machine. In 2014 IEEE Students' Conference on Electrical, Electronics and Computer Science, Bhopal, India, pp. 1-

6. https://doi.org/10.1109/SCEECS.2014.6804439

[40] Rajinikanth, V., Kadry, S., Nam, Y. (2021). Convolutional-neural-network assisted segmentation and SVM classification of brain tumor in clinical MRI slices. Information Technology and Control, 50(2): 342-356. https://doi.org/10.5755/j01.itc.50.2.28087

[41] Lamrani, D., Cherradi, B., El Gannour, O., Bouqentar, M.A., Bahatti, L. (2022). Brain tumor detection using MRI images and convolutional neural network. International Journal of Advanced Computer Science and Applications, 13(7): 452-460.

https://doi.org/10.14569/IJACSA.2022.0130755

[42] Febrianto, D.C., Soesanti, I., Nugroho, H.A. (2020). Convolutional neural network for brain tumor detection. IOP Conference Series: Materials Science and Engineering, 771(1): 012031. https://doi.org/10.1088/1757-899X/771/1/012031

[43] Kuraparthi, S., Reddy, M.K., Sujatha, C.N., Valiveti, H., Duggineni, C., Kollati, M., Kora, P., Sravan, V. (2021). Brain tumor classification of MRI images using deep convolutional neural network. Traitement Du Signal, 38(4): 1171-1179. https://doi.org/10.18280/ts.380428