International Conference on Control, Engineering & Information Technology (CEIT'14)
Proceedings - Copyright IPCO-2014, pp.185-196
ISSN 2356-5608

Acceleration of SVMs by reduction of samples using MSFLA for images recognition

Anis LADGHAM
Electrical Departement
CSR Research Unit of EμE Laboratory
National School of Engineering –
Monastir, Ibn Eljazzar 5019, Monastir,
Tunisia
ladghamanis02@gmail.com

Anis SAKLY
Electrical Department
National School of Engineering –
Monastir,
Ibn ElJazzar, 5019, Monastir,
Tunisie
sakly_anis@yahoo.fr

Abdellatif MTIBAA
Electrical Departement
CSR Research Unit of EμE Laboratory
National School of Engineering –
Monastir,
Av Ibn ElJazzar 5019, Monastir-Tunisia
abdellatif.mtibaa@enim.rnu.tn

Abstract*—Generally, the learning techniques such as the Support Vector Machines (SVM) require a large input features for their performances. A step of feature selection is very important to reduce the size of the dataset of these features without losing the quality of classification. Indeed it is a compromise between quality and time and resources consumed. This paper presents a hybrid approach for detection of brain tumor tissues in Magnetic Resonance Images (MRI). The main purpose of our novel method is the reduction of the training features of the SVM classifier. Two datasets are used. The first is the dataset of normal MRI and the second is the dataset of the infected MRI. Firstly, we use the Gabor filter for textural feature extraction from MRI of the two datasets. Then, the feature selection stage is performed using the Modified Shuffled Frog Leaping Algorithm (MSFLA). This step is performed to select the most informative input features. Finally, the optimal features are given as input to the SVM classifier to detect the brain tissues as normal and abnormal. The experimental results show that the proposed method is able to achieve a good recognition quality and an optimized execution time by comparing it with old methods.

***Keywords*—SVM; Gabor filter; MSFLA; feature selection; MRI**

I. INTRODUCTION

The medical images processing becomes a very fertile research subject because we need automated and efficient disease diagnosis in a short amount of time. Reducing the time of medical operations, such as the medical images reading and recognition, is very interesting for the safety of patients. Computer and information technologies are very used in this domain [1-3].

The Magnetic Resonance Imaging (MRI) [4, 5] is the most valuable tool in the clinical environment because of its high spatial resolution and its contrast. Usually, the MRI are examined by the radiologists to recognize normal and abnormal tissues. The large volume of MRI to be analyzed and the shortage of the radiologists make such manual recognition inaccurate and expensive. Hence, there is a huge need for optimization of automated techniques of recognition of such images [6-10].

The kernel methods (KMs) are the class of algorithms for pattern classification and recognition, whose most known member is the SVM [11- 15]. The general task of pattern classification is to study and to find the types of relations (for example rankings, clusters, principal components, correlations, classifications) in general types of data (such as text documents, vectors, videos, images, etc.).

The SVM address the problem by mapping the data into a feature space of largest dimension, wherein each coordinate notation constitutes a function of the data items by converting the data into a set of points in a Euclidean space [13-15]. In this area, a variety of processes can be used to find the relationships between data. Since the mapping may be quite general (not necessarily linear, for example), the relationships found in this manner are very general.

The SVM classifiers are able to operate in the feature area without ever computing the coordinates of the data in that area, but rather by simply computing the inner products between the images of all pairs of data in this feature space. This operation is often computationally cheaper than the explicit computation of the coordinates.

The SVM learns a hyperplane (i.e., boundary) separating different class data with maximizing the margin. It requires a large number of features. However, not all of these features are equally important for a particular mission. Some of them can be redundant or misplaced. Higher performance may be achieved by deleting some features. Under other circumstances, the dimensionality of the input space may be reduced to save the computational effort. Although, this can slightly affect the accuracy of classification. Therefore, the process of classification should be fast and precise using a smallest number of features. This objective can be achieved using the feature selection strategies which are often implicit to explore the effect of unnecessary attributes on the performance of the classifier systems [16-26].

Luiza A. et al. [27] and Wang et al [28] developed an algorithm for automated classification of Brain MRI using the SVM classifier. This technique is used to distinguish the normal from the abnormal slices with statistical features. Generally, the large volumes classifiers as SVMs consume large amounts of time and resources for the classification. These classifiers use large datasets to provide accurate results. Therefore, the latest classification researches have focused in employing approach based on optimizing the dataset of features before training the used classifier. For example, meta-heuristics serve for good feature selection in SVM in order to obtain better qualities of classification and reduced execution time [24, 25].

Many recent studies have reported that this optimization is able to deliver higher classification accuracy than the other existing data classification algorithms. Umamaheswari, J. et al [9] describes about the process of recognition and classification of brain images such as normal and abnormal based on Particle Swarm Optimization (PSO) and SVM. The collective approach by using PSO-SVM gives high approximation capability and much faster convergence. Lei et al [29] defined a method of segmentation based on the classification of pixels using the SVM and the Genetic Algorithms (GA). GA is used for the optimization of the parameters in SVM's kernel. Ladgham, A. et al [30] proposed an approach based on a Shuffled Frog Leaping Algorithm (SFLA) and SVM. In the latter method, The Gray-Level Co-occurrence Matrix (GLCM) is used to extract features. The SFLA is used for feature selection.

This study focuses on further increasing the recognition accuracy rate and on further decreasing the computational effect by employing an approach based on the Modified Shuffled Frog Leaping Algorithm (MSFLA) [31] for textural feature selection. In [30], MSFLA has proven to be faster and more efficient compared to several other metaheuristics as SFLA and GA. We used a combination of Gabor filters to extract the features from the brain MR images. The method employs the SVM classifier to sort the selected texture features into its classification. Our new method is termed MSFLA-SVM.

The report is organized as follows. In Section 2, we deeply presented the proposed method. Within, we explain all the steps of our method. The experimental results are given in Section 3. Some conclusions are made in Section 7.

II. The Proposed Method

In this paper, we describe an accelerated SVM paradigm for the recognition of MR brain tumors. The flowchart of the proposed tumor location scheme is given in Figure 1. It involves two stages which are: the Training stage and the Testing stage. In the training stage, firstly, the dataset of the training images is correlated by 40 Gabor filters (5x8 matrixes of Gabor masks). So a 27x18 image is transformed to 27x18x5x8. Indeed, an input image is transformed to 40 output feature images. Thus, the use of this large number of masks helps to provide accurate results. But, using all the 40 filters to extract features from MR brain images is much time consuming and we will have several repetition in the dataset.

For this, a step of reduction of number of Gabor masks is performed using MSFLA. In this step, we will optimize the number of masks in order to keep the smaller number that can give results comparable to those given using all the masks. Finally, the selected Gabor masks are used to extract textural features from the MR images and to train the SVM classifier. In the testing stage, we use the selected Gabor masks for textural feature extraction. Then, these features are applied to the SVM classifier. The training set is used to build the model and to determine its parameters. And the test set is used to measure its performance holding the parameters constant.

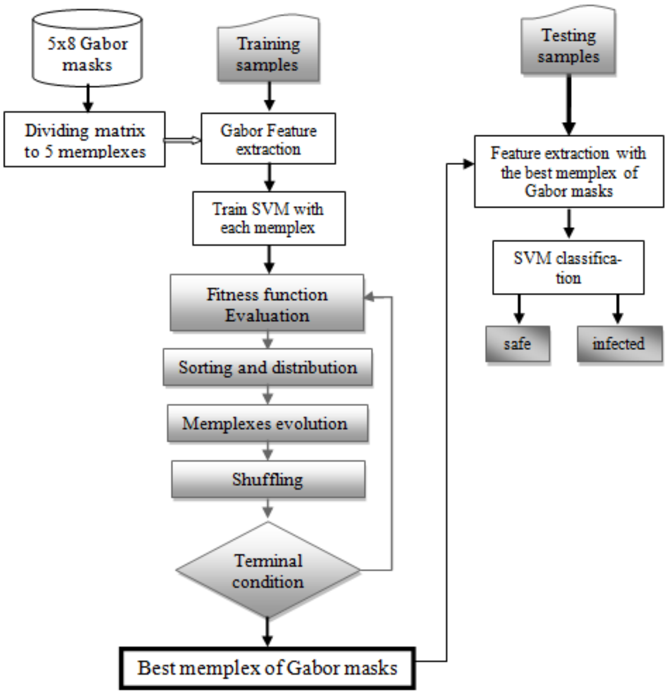


Fig. 1. Block diagram of the proposed method.

A. *Gabor feature extraction*

In this method, the textural features are extracted from the MR brain images using the 2D Gabor filtering. It is known among doctors that normal brain MR images are composed of three zones which are the White Matter (WM), the Gray Matter (GM) and the Cerebral Spinal Fluid (CSF). These three zones are similar for all the people. The tumors are abnormal tissues that come from uncontrolled multiplication of cells. The clinical signs accompanying a brain anomaly are numerous. The symptoms vary from a patient to another according to the location, the size and the shape of the tumor.

The Gabor wavelets were proposed by John Daugman [32]. They have the form of plane waves constrained by a Gaussian envelope function and defined by a sine wave. The Gabor filters are used to extract features from an image corresponding to a particular frequency band. The texture of an image can be regarded as a quasi-periodic signal. We can use Gabor filters with many frequencies and many directions for action on images. The two-dimensional Gabor function [32] can be expressed as using the Equation (1) bellow:

$$G(x;\ y)\ =\frac{1}{2\pi\sigma\beta}e^{-\pi\left[\frac{(x\text{-}x_0)_r{}^2}{\sigma^2}+\frac{(y\text{-}y_0)_r{}^2}{\beta^2}\right]}+e^{j(2\pi(u_0x+v_0y)+P)}\qquad(1)$$

Where P is the sinusoidal carrier phase. The parameters u_0 and v_0 define the spatial frequencies of the sinusoid in Cartesian coordinates. σ and β model the spatial deviations of the elliptical Gaussian throughout x and y respectively. The coefficient r stands for a rotation operation. The pair ($x_0;\ y_0$) denotes the of Gaussian envelope's peak location such that:

$$(x-x_0)_r=(x-x_0)\cos\theta+(y-y_0)\sin\theta\qquad(2)$$

$$(y-y_0)_r=-(x-x_0)\sin\theta+(y-y_0)\cos\theta\qquad(3)$$

The 2-D Fourier transform of this Gabor filter is given in the Equation (4):

$$\widehat{G}(u,v)=\frac{\sigma\beta}{2\pi}e^{(-\pi(\sigma^2(u-u_0)_r^2+\beta^2(v-v_0)_r^2))}+e^{(j(-2\pi(x_0(u-u_0)+y_0(v-v_0))+P))}\qquad(4)$$

or in polar coordinates,

$$Magnitude(\widehat{G}(u,v))=\frac{\sigma\beta}{2\pi}e^{(-\pi(\sigma^2(u-u_0)_r^2+\beta^2(v-v_0)_r^2))}\qquad(5)$$

$$Phase(\widehat{G}(u,v))=\text{-}2\pi(x_0(u\text{-}u_0)+y_0(v\text{-}v_0))+P\qquad(6)$$

A well designed set of Gabor filters may capture the relevant frequency spectrum in all directions. Phase may be taken as a feature as it contains information about the locations of the edge and other details about the image. Many meaningful features can be extracted using the Gabor filter banks [33].

The Equation (7) shows the response image of Gabor filter. It can be written as a correlation of the input image I with the Gabor kernel *G(r, o)* with the resolution *r* and the orientation *o*.

$$IG_{(r,\,o)}\ =\ I*G_{(r,\,o)}\qquad(7)$$

The center frequency of Gabor filter U_0 can be determined by the width of the strokes. In order to ensure that the sampling information cannot be lost, and the redundancy information can be reduced, the center frequency of Gabor filter was restricted as multiple relations. According to statistical experiences, as details of MRI are very fine and very close, we choose the ranges of stroke width in the interval of 1–10 pixels. In our work, we choose a combination of eight directions and the five smallest widths of Gabor filters (see the Figure 2). The Figure 3 shows the response of the set of Gabor filters when applied to a T2-weighted MR image. In the figure, we can see clearly that there are some filters that cannot extract the texture features from the MRI. These filters are not the same for each input image. For this, we use the MSFLA optimization to remove unnecessary Gabor masks.

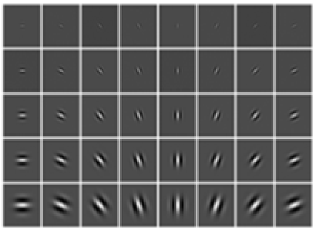


Fig. 2: The set of 40 Gabor filters used to extract the textural features: Each row represent a different scale (wavelengths top to down: 3, 6, 12, 24, 48); whereas each column stands for a different orientation.

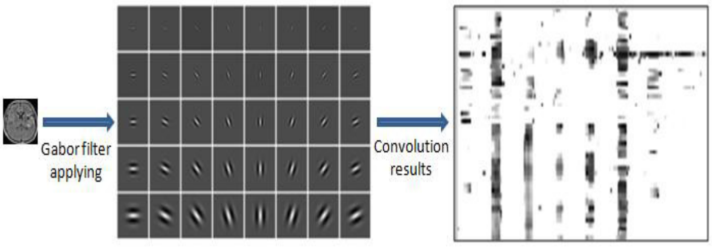


Fig. 3. The results of extraction of textural features from a brain MRI of the T2-weighted dataset using all the Gabor masks at five frequencies and eight orientations.

B. *Feature selection using MSFLA*

i. *The basic SFLA*

The SFLA is a newly developed memetic meta-heuristic [34]. It has been developed for solving the combinatorial optimization problems. The SFLA is a population of virtual frogs based cooperative search metaphor. It is inspired from natural memetics. The algorithm contains elements for local search and for global information exchange. The SFLA is composed of a set of interacting virtual population of frogs. This virtual population is partitioned into several memeplexes. The virtual frogs act as carriers of memes. A meme is a unit of cultural evolution. In the algorithm, an independent local search is performed simultaneously in each memeplex.

The SFLA combines the advantages of particle swarm optimization (PSO), introduced by [35, 36], which inspires its principle from the herding behavior of animals like fish floquant and from genetic algorithm (GA) which is a research technique developed by Holland and which models the principle of natural evolution [37]. The local search is achieved using a PSO-like method adapted for discrete problems but focusing on a local search. To ensure global exploration, the virtual frogs are periodically shuffled and reorganized into new memplexes in a technique similar to that used in the shuffled complex evolution algorithm. All virtual frogs can communicate and transmit information between them to improve their locations. Also, to give the opportunity for random generation of the improved solution, the random virtual frogs are generated and substituted in the population.

In SFLA, each virtual frog has a different solution from others according to its adaptability evaluated by its fitness function. The entire population of the virtual frogs is divided into a predefined number of subsets (memplexes). Frogs of each memplex have their own strategy to explore the environment in different directions. After a predefined number of memetic evolution, the exchange of information between memplexes starts. This procedure is called procedure of Shuffling. This procedure must guarantee that the evolution to a particular solution is free from all prejudices. The Memetic evolution and the Shuffling procedures are performed alternatively until reaching the convergence criteria.

The SFLA algorithm starts by creating arbitrarily a population X_i of F virtual frogs ($i= 1, 2... F$). The virtual frogs are equivalent to the GA chromosomes. All the virtual frogs are sorted in descending order and substituted to m memplexes, each memplex contains p virtual frogs, the first ranked virtual frog moves to the first memplex, the second one moves to the second memplex, the p ranked frog moves to the p^{th} memplex and the $(p+1)^{th}$ returns to the first memplex. In each memplex, the virtual particles having the best and the worst fitness are identified respectively by X_b and X_w. The virtual particle with the best fitness in the whole population is identified by the global best X_g. During the evolution of memplexes, the worst virtual particles jump to reach the best ones in the process of 'memeplex evolution' using the Equations (1) and (2):

➢ The initial population: To achieve an SFLA optimization, we start by generating an arbitrary population of *F* frogs *Xi (i=1, 2…F).*

➢ The step of Sorting and distribution: After their evaluation by the proposed fitness function, the virtual frogs are sorted in descending order and divided into *m* memplexes, each one contains *p* frogs, the first ranked frog is placed in the first memplex, the second in the second memplex, the p^{th} frog in the p^{th} memplex and the $(p+1)^{th}$ returns to the first memplex.

➢ The step of evolution of the virtual frogs in each memplex: The virtual frogs with the best and the worst fitness in each memplex are called respectively X_b and X_w. The best frog in the whole population is called the global best X_g. During the evolution of memplexes, worst frogs jump to reach the best ones in the memeplex evolution process using the Equation 8 and the Equation 9 bellow:

$$S = rand.(X_b - X_w) \tag{8}$$

$$IX_w = X_w + S \; ; \; S < Smax \tag{9}$$

where IX_w is the improvement of the worst solution, S indicates the jump step of the worst frog, S_{max} is the maximum jump distance of the frogs and rand is an arbitrary number in the range [0, 1]. If these equations do not improve the worst solution, X_b is replaced by X_g and the process of evolution is repeated using the Equation 10.

$$S = rand.(X_g - X_w) \tag{10}$$

If the worst solution is not improved, a new position is generated arbitrarily.

➢ The Shuffling step: After a predefined number of memeplex evolution steps, all the frogs of memeplexes are collected and sorted in descending order. Step 2 divides frogs into different memeplexes again, and then step 3 is done.

➢ The terminal condition: If a global solution or a fixed iteration number is reached, the algorithm stops.

ii. Features selection using the MSFLA

The MSFLA [31] is an enhanced version of the metaheuristic SFLA. MSFLA has demonstrated effectiveness in image segmentation and in image recognition [31, 32]. It gives accurate results in shorter time than SFLA.

In MSFLA, we evaluate the memplexes by the fitness function rather than the virtual frogs. Indeede, in the phase of sorting and distribution, the evaluation of the virtual frogs by the fitness function is replaced by the evaluation of the memplexes. And the step of evolution of the virtual frogs in each memplex is replaced by the step of evolution of memplexes. Then, poor memplex is concerned by the amelioration at each shuffling iteration.

Within population, memplexes are ranked in descending order according to their fitness values. Those with the best and the worst fitness are respectively named M_b and M_w. To improve the fitness of the worst solution, we change the virtual frogs of the corresponding memplex in the memplexes evolution stage using the Equation (11) and the Equation (12). The steps of MSFLA optimization are given bellow:

➢ The initial population: We start by generating an arbitrary population of *F* virtual frogs *Xi (i=1, 2…F).* In our work, the virtual frogs are the coordinates of Gabor masks. We use a 5x8 matrix of Gabor filters. Then, the number of virtual frogs of the initial population is 40. Our goal is to look for the smaller number of Gabor masks giving the best recognition result in less time.

- The step of sorting and distribution: In an arbitrary manner, the virtual frogs are divided into 5 memplexes, each one contains 8 Gabor masks. Then, we will complete the process MSFLA to choose the best combination of 8 masks. These eight masks will be used to extract features from MRI instead of the old 40. After achieving the memplexes evolution step and the shuffling step, if the desired performance does not reach, we increase a little the number of masks by memplex.

- The step of Memeplexes evolution: To improve the worst solution, an equation similar to the PSO is used. In our algorithm, we want to make this worst solution better than the best one. This amelioration takes place by swapping randomly the positions of the internal frogs in the worst memplexe, e.g. Eq. (11):

$$S_1 = rand\left(1, p\right).\left(M_b - M_w\right) \tag{11}$$

where rand(1,p) is a random vector which elements are between 0 and 1.

The new solution is given in the Equation (12). This new solution is also evaluated by the fitness function. If a better solution than the previous is produced, it will be memorized; else the Equation (11) is repeated for a predefined number of times.

$$IM_w = M_w + S_1 \tag{12}$$

If these equations produce a better solution, then it replaces the worst memplex. If they do not, then the factor M_b of the Equation (11) is changed by M_j *(1<j<m)* other than M_b and M_w and adapted to the Equation (13) and the Equation (14):

$$S_2 = rand\left(1, p\right).\left(Mj - Mw\right);\ 1 < j < m \tag{13}$$

$$IM_w = M_w + S_2 \tag{14}$$

If these equations produce a better solution, it replaces the worst memplex. If they do not improve it, then a new solution is randomly generated to replace the worst one.

- The Shuffling step: After improving the worst solution, it takes the rank of the last best solution. Memplexes are sorted in descending order again based on their fitness, and then the step of memplexes evolution is repeated. Indeed, we want to ameliorate the new worst solution by using the same strategy used with the former worst solution. The shuffling stage is repeated until a predefined terminal condition is reached.

- The terminal condition: If a predefined solution is reached, the algorithm stops. Increasing the value of the terminal condition is useless because our developed algorithm MSFLA shows that it stabilizes and gives a precise result rapidly from the first iterations.

iii. The proposed fitness function

A fitness function is a particular kind of the objective functions which is used to demonstrate, as a figure of merit, how close a given solution is to achieving the set objects.

Particularly, in the fields of metaheuristic programming like MSFLA, each design solution is given as a string of numbers (referred a chromosome). After each testing round, the idea is to delete the worst design solutions, and to reproduce new ones from the best design solutions. Therefore, each design solution needs to be assigned a figure of merit in order to show how close it came to meeting the overall specification, and this is generated by the application of the fitness function to the test results obtained from that solution.

The reason that metaheuristic algorithms cannot be regarded to be a lazy way of acheiving design work is because of the effort involved to design a realizable fitness function. If this function is designed badly, the algorithm will either converge to an inappropriate solution, or will have difficulty converging at all solutions.

Moreover, the fitness function must not only correlate to the designer's aim, it must also be calculated quickly. The execution speed is very important, as a typical metaheuristic algorithm must be iterated many times in order to produce an exploitable solution for a problem.

In our work, the proposed fitness function is an evaluation function used to rank each memplex according to its performance. For each memplex of Gabor masks, after the first SVM training, we calculate the execution time of the classification of the test samples *t(i)* and the accuracy of recognition *acc(i)* in order to calculate its fitness value which is given in the Equation (15) bellow:

$$Fitness(i) = \frac{1}{t(i)} + acc(i) \tag{15}$$

C. Tumor location using the SVM

i. Support vector machines

The Support vector machine (SVM) [39-42] presents a family of supervised learning methods. It may be used for the classification, the recognition tasks and many other tasks. The SVM is based on the principle of minimization of the structural risk of the statistical learning theory. The main idea of SVM is the projection of the input space to another space with higher dimension in which the samples are being linearly separable. This projection is implied as the learning and the decision processes involve only inner dot product in the feature space which may be computed using a kernel function. The extensive discussion of the SVM classifier can be referred in [43].

The classification by SVM is really an extension of the perceptron that tries to search a hyperplane that separates the input data. Indeed, the perceptron tries to locate a separating hyperplane without considering how it separates the data. However, it is preferable to find a hyperplane that is as far away as possible from all the classes of the dataset. Indeed, we expect this to generalize better to unseen data (an example is shown in the Figure 4). The technical measure of how a hyperplane separates data is its margin. The hyperplane is the distance of the hyperplane to the closest element in the dataset. Indeed, a large margin means that the hyperplane manifestly separates the data.

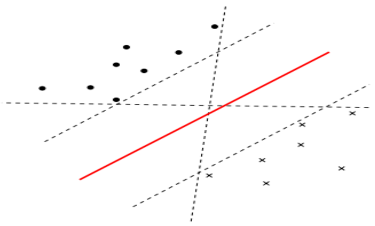


Fig. 4. The dashed lines may be the hyperplanes that separate the classes of a dataset, but the hyperplane with the largest margin (the red line) has the best separation.

Given some training data *D* which presents a set of *n* points having the form given in the Equation (16). Each x_i is a p-dimensional input vector and y_i is the number of classes of the input data x_i.

$$D = \{(x_i, y_i) / x_i \in \Re^p, y_i \in \{-1,1\}\}_{i=1}^{n} \tag{16}$$

For the linear separable case, we have a hyper-plane *wx+b=0* that separates all the training samples into two categories, which are the positive class data and the negative one. The separation is performed as follows:

$$y_i(w.x_i + b) \geq 1; \forall i \tag{17}$$

where *w* is the normal to the hyper-plane. The margin *M* of the last hyper-plane is the sum of the shortest distance from the hyper-plane to the closest positive data samples and also the closest negative ones. By using the geometry, the margin is found $\frac{2}{\|w\|}$. So, the maximum margin can be given by reducing $\|w\|^2$ subject to the constraints in the Equation (18).

$$M = (x^+ - x^-)N = (x^+ - x^-)\frac{w}{\|w\|} = \frac{2}{\|w\|} \tag{18}$$

where x^+ are the support vectors of the positive samples, x^- are the support vectors of the negative samples, *N* is the normal vector of the hyper-plane, $\|w\|$ is the Euclidean norm of *w*.

In order to handle non-linearly separable data, we expand the constraints slightly to allow for misclassified points. Vladimir N. Vapnik suggested a way for mapping the training samples $x_i.x_j$ by applying the Kernel functions (originally proposed by: Aizerman [44]) instead of dot product. These classifiers are used for maximum margin hyper-planes in a transformed feature space. The advantage is that it is unnecessary to know the mapping explicitly. In this context, slack variables ζ_i are added to the penalty errors. The learning task is to minimize the equation given bellow:

$$\frac{\|w\|^2}{2} + C\sum_{i=1}^{m} \zeta_i \tag{19}$$

$$\begin{aligned} & y_i(x_i w + b) \geq 1 - \zeta_i; \forall i \\ & \zeta_i \geq 0; \forall i \end{aligned} \tag{20}$$

where *C* is the penalty to errors and ζ_i are the positive slack variables that measure the amount of constraint violations.

The learning task equals to the maximization of the Lagrangian (see the equation 21):

$$\sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{21}$$

s. t.:

$$\alpha_i \geq 0$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

where it can be solved using the quadratic programming techniques. After we gain the optimal a_j, the classification of an unknown sample *z* will be decided based on the sign of the function given in the Equation (22):

$$G(z) = \sum_{j=1}^{n} \alpha_j y_j K(z, x_j) + b \tag{22}$$

ii. Training the SVM classifier by the extracted features

According to the World Health Organization, there are more than 120 kinds of brain and central nervous system tumors. The symptoms that accompany each kind of brain tumor vary greatly according to its location in the brain, its size and its shape. For this, we try to collect a varied dataset of MRI that contains a large number of types of tumors. Images used in the experiment are the T1-weighted MRI and the T2-weighted MRI. For the T2-weighted ones, we use a set of 82 MRI infected by a tumor as the positive class samples and 62 safe MRI as the negative class samples. Some samples of the positive class of the dataset are given in the Figure 5. Our algorithm was tested using two types of kernel which are the linear kernel and the quadratic one which are presented respectively by the equations (23) and (24).

$$K(x_i, x_j) = x_i . x_j \qquad (23)$$

$$K(x_i, x_j) = (1 + x_i . x_j)^2 \qquad (24)$$

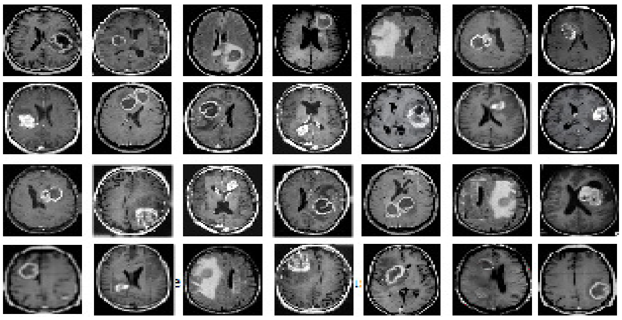


Fig. 5. Some samples from the T2-weighted MRI dataset.

The classification process using MSFLA-SVM classifier networks is performed using the following steps. For each image, eight images of Gabor filter output are fed into the SVM classifier networks. Each Gabor filter image is the output of each memplex of Gabor filters. *□(x)* are the feature extraction function of a sample image. It is composed of the mean μ, the variance σ^2 and the entropy *e*. The SVM classifier is used to classify the textural features of MRI, represented as *G(z)* in the Equation (22). For this, we use cross-validation to train the SVM classifier. The training samples are divided into two parts, one of which is used for initial training and the other is used for validation.

Each test result given by each memplex of Gabor filters is evaluated by the fitness function. Then, the evaluation results are sorted in descending order. Then, in the memplexes evolution stage, the Equations. (11) and (12) are applied to the worst memplex to improve its structure and then its fitness. After a predefined number of memeplex evolution stages, all the memeplexes are collected and sorted in descending order again. The training procedure will be stopped when the precision rate of the validation samples is more than a predefined stop criterion. The masks of the final best memplex are saved to achieve the recognition stage.

III. EXPERIMENTAL RESULTS

The platform adopted to develop the proposed approach is a Personnel Computer with the subsequent features: the Windows 7 Operating System, a CPU with 2.53 GHz and 4 Go RAM. The proposed algorithm is implemented on MATLAB environment. To evaluate the performance of MSFLA-SVM, we define two parameters which are the Recognition Rate and the Number of selected features. The Recognition rate is defined to measure the accuracy of the algorithm. It is the ratio of the number of recognized images and the total number of the input test images.

The number of selected features is the number of selected Gabor masks obtained after applying feature selection stage multiplied by the number of the training images of the dataset.

MSFLA-SVM is compared with the Gabor-SVM and SFLA-SVM algorithms using the two datasets of T1-weighted MRI and T2-weighted MRI and by using firstly the linear kernel for the SVM and secondly we use the quadratic kernel. Gabor-SVM denotes the method that uses the Gabor filters for textural feature extraction and the SVM classifier for the classification of MRI. In Gabor-SVM, we don't use a feature selection stage. SFLA-SVM denotes the method that uses the Gabor filters for textural feature extraction, the metaheuristic SFLA to achieve the feature selection step and the SVM classifier for the classification of MRI.

Several parameters must be initialized before the implementation of MSFLA. These parameters are the Number of virtual Frogs *F* which are 40 elements, the Number of memplexes *M* which are 5 memplexes, the Number of virtual Frogs in each memplex *P* which are 8 and the Number of iterations *itr* for the MSFLA algorithm which are 10 iterations. Even with the choice of reduced number of frogs, our algorithm gives excellent results. Table 1, Table 2, Table 3 and Table 4 give the results of the experimental comparisons achieved using the predefined datasets and kernels. We make ten attempts for the three algorithms and we choose the best one for each method in order to fill the table. Our method gives the best results compared to the others in terms of recognition time, accuracy, number of selected features and number of support vectors selected.

For our algorithm, we try that background pixels will not be learned to the SVM training process in order to optimize the computational time and resources. For this, we choose arbitrary a sparse set of pixels with different colors from the background color. The set of pixels is selected by the convolution of the test image with two MRI templates (see the Figure 6) using the Equation (25). Assume that the test image *A* has the dimensions (M_a, N_a) and the template *B* has the dimensions (M_b, N_b). After the convolution, we use the function that finds the regional maxima of the convoluted image. These regions are very fortunate not to be background regions. The selected pixels and their weights are learned jointly with the SVM classifier.

$$C(i, j) = \sum_{m=0}^{Ma-1} \sum_{n=0}^{Na-1} A(m,n) * B(i-m, j-n) \qquad (25)$$

$$0<i<Ma+Mb-1 \text{ and } 0<j<Na+Nb-1$$

Fig. 6. MRI templates used for convolution with test images.

The Figure 7 gives the results of recognition of a set of MRI using MSFLA-SVM. This figure could easily demonstrate that our method was easily recognized infected MRI among the set of MR brain images.

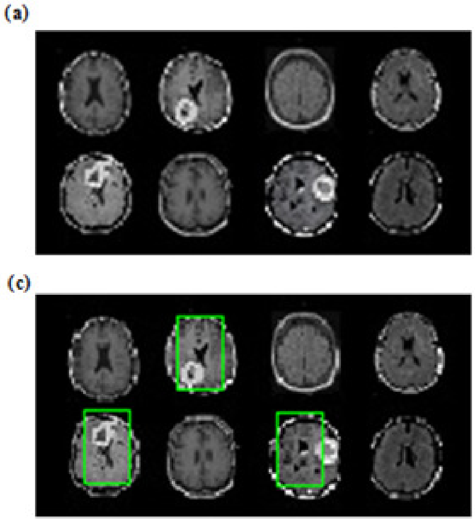


Fig. 7. Results of tumor recognition with MSFLA-SVM.

TABLE 1. PERFORMANCE COMPARISON OF OUR METHOD WITH GABOR-SVM AND SFLA-SVM USING THE DATABASE OF T1-WEIGHTED IMAGES USING THE LINEAR KERNEL

	Gabor-SVM	*SFLA-SVM*	*ASFLA-SVM*
Number of selected features	5480	3014	2466
Recognition Rate (%)	95.31	94.11	96.17
Recognition time (s)	9.11	5.45	2.70
Number of support vectors	204	198	237

TABLE 2. PERFORMANCE COMPARISON OF OUR METHOD WITH GABOR-SVM AND SFLA-SVM USING THE DATABASE OF T2-WEIGHTED IMAGES USING THE LINEAR KERNEL

	Gabor-SVM	*SFLA-SVM*	*ASFLA-SVM*
Number of selected features	5480	2740	2192
Recognition Rate (%)	94.93	94.20	95.67
Recognition time (s)	9.03	4.48	2.90
Number of support vectors	207	214	221

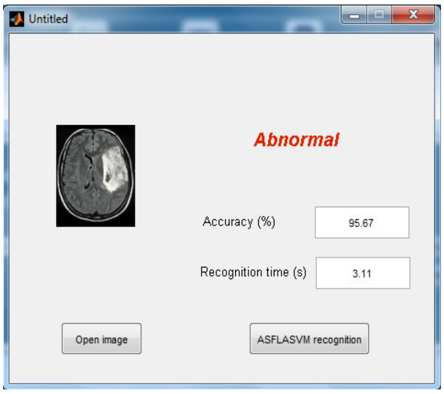
TABLE 3. PERFORMANCE COMPARISON OF OUR METHOD WITH GABOR-SVM AND SFLA-SVM USING THE DATABASE OF T1-WEIGHTED IMAGES USING THE QUADRATIC KERNEL

	Gabor-SVM	*SFLA-SVM*	*ASFLA-SVM*
Number of selected features	5480	3123	2612
Recognition Rate (%)	75.31	77.11	81.17
Recognition time (s)	13.11	5.45	5.76
Number of support vectors	176	201	199

TABLE 4. PERFORMANCE COMPARISON OF OUR METHOD WITH GABOR-SVM AND SFLA-SVM USING THE DATABASE OF T2-WEIGHTED IMAGES USING THE QUADRATIC KERNEL

	Gabor-SVM	*SFLA-SVM*	*ASFLA-SVM*
Number of selected features	5480	3040	2563
Recognition Rate (%)	74.03	78.20	79.8
Recognition time (s)	11.53	7.148	6.09
Number of support vectors	165	187	200

The interface of our application for MR brain tumor recognition using MSFLA-SVM is depicted in the Figure 8. After trainig the SVM classifier with the selected features, the following interface will appear. By clicking on the icon 'Open image', a test image to be classified is opened. And by clicking on the icon 'MSFLASVM recognition', the latter test image is classified by the traind SVM. The interface shows a MRI recognized by providing the elapsed time and the accuracy of the SVM classisier. This interface is prepared to facilitate the use of the algorithm by doctors and non professionals.



Abnormal

Accuracy (%) 95.67

Recognition time (s) 3.11

Fig. 8. The interface of the MSFLA-SVM.

The Figure 9 gives the curve of the shuffling stage iterations of our proposed metaheuristic MSFLA. From the very first iterations, MSFLA stabilizes and determines the best memplex of Gabor masks. This figure shows that MSFLA consumes only 2 or 3 iterations to determine the solution. And for this reason, our algorithm does not consume much time. This graph proves that MSFLA helps to choose the best set of Gabor masks for a very short time. For that, our algorithm gives the best results compared to those given by the basic algorithm Gabor-SVM and by SFLA-SVM.

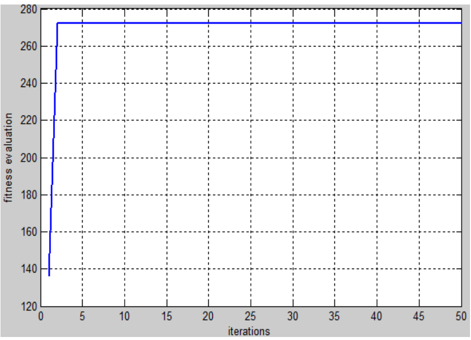


Fig. 9. The Shuffling trace of the proposed metaheuristic MSFLA.

IV. Conclusion

In this study, we propose a novel paradigm of automatic recognition of MRI slices. The main purpose of our method is the optimization of resources used in SVM classifier in order to increase its performances such as recognition accuracy and time of execution. The proposed method includes 3 steps. The steps are the textural features extraction step, the feature selection step and the classification step using the SVM classifier. In the textural feature extraction stage, a set of Gabor wavelets are used to extract the textures from the MRI. In the feature selection stage, we have reduced these features using MSFLA in order to keep the most significative ones. Finally, a SVM classifier is used to recognize normal and abnormal brain MRI. The experimental results indicate that MSFLA-SVM gives better performance than fundamental SVM and then the SFLA-SVM. These results indicate that our new method can be very helpful in computer aided intelligent health care systems.

References

[1] A. Ladgham, F. Hamdaoui, A.Sakly, and A.Mtibaa, "Real Time Implementation of Detection of Bacteria in Microscopic Images Using System Generator," Journal of Biosensors and bioelectronics, (ISSN: 2155-6210), 3 (5), October 2012.

[2] A. Ben Abdelali and A. Mtibaa, "Toward hardware implementation of the compact color descriptor for real time video indexing," Advances in Engineering Software Vol. 36, 2005, pp. 475–486

[3] F. Hamdaoui, A. Ladgham, A. Sakly, and A. Mtibaa, "Real Time Implementation of Medical Images Segmentation Using Xilinx System Generator," International REview on Computers and Software, IRECOS (ISSN:1828-6003); Vol 7 N 6, pp: 2861-2867, Novembre 2012.

[4] E. Haack, "Magnetic Resonance Imaging", Physical Principles and Sequence Design. Wieley-Liss, New York, 1999.

[5] L Kuncheva, J Rodriguez, C Plumpton, D Linden, and S Johnston, "Random subspace ensembles for FMRI classification," IEEE Trans. Med. Imaging, Vol. 29, No. 2, 2010, 531–542.

[6] J. L. Moyano-Cuevas, A. Plaza, I. Dopido, J. B. Pagador, J. A. Sanchez-Margallo, L. F. Sánchez, and F. M. Sánchez-Margallo, "3D Segmentation of MRI of the Liver Using Support Vector Machine," XIII Mediterranean Conference on Medical and Biological Engineering and Computing 2013, IFMBE Proceedings Vol. 41, 2014, pp 368 – 371

[7] H. Selvaraj, S. S. Thamarai, D. Selvathi, and L. Gewali, "Brain MRI Slices Classification Using Least Squares Support Vector Machine," Int. J. of Int. Comp. in Med. Sc. & Im. Pr. Vol. 1, No. 1, 2007, pp. 21 – 33.

[8] M. R. Mahmudur, C. D. Bipin, and B. Prabir, "Medical image retrieval with probabilistic multi-class support vector machine classifiers and adaptive similarity fusion," Computerized Medical Imaging and Graphics Vol. 32, 2008, pp. 95 – 108.

[9] J. Umamaheswari and G. Radhamani, "An Amalgam Approach for DICOM Image Classification and Recognition," International Journal of Computer and Communication Engineering Vol. 62, 2012, pp. 807 – 812.

[10] L. Shuo, F. Thomas, K. Adam, and L. Song, "Automatic clinical image segmentation using pathological modeling, PCA and SVM," Engineering Applications of Artificial Intelligence. Vol. 19, 2006, pp. 403–410.

[11] Z. Yungang, Z. Bailing, C. Frans, X. Jimin, and L. Wenjin, "One-class kernel subspace ensemble for medical image classification," EURASIP Journal on Advances in Signal Processing, Vol. 17, February 2014.

[12] UCI, Machine learning repository. http://archive.ics.uci.edu/ml/datasets/, Accessed 22 June 2013

[13] DM Tax and RP Duin, Support vector domain description. Pattern Recognit Lett. 20, 1191–1199 (1999)

[14] DM Tax and RP Duin, Support vector data description. Mach. Learn. 54, 45–66 (2004)

[15] R. Jayadeva and S. C. Khemchandani, "Twin support vector machines for pattern classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, No. 5, 2007, pp. 905–910.

[16] C. Fernandez-Lozano, C. Canto, M. Gestal, J. M. Andrade-Garda, J. R. Rabuñal, J. Dorado, and A. Pazos, "Hybrid Model Based on Genetic Algorithms and SVM Applied to Variable Selection within Fruit Juice Classification," The Scientific World Journal, Vol. 2013, 2013, Article ID 982438, 13 pages.

[17] M. U. Aneesh, A. K. M. Abhishek, and K. Manikantan, "Optimal Feature Selection based on Image Pre-processing using Accelerated Binary Particle Swarm Optimization for Enhanced Face Recognition," International Conference on Communication Technology and System Design, Procedia Engineering Vol. 30, 2011, pp. 750 – 758.

[18] I. Guyon, A. Elisseefi, and L. P. Kaelbling, "An introduction to variable and feature selection," Journal of Machine Learning Research, Vol. 3, 2003, pp. 1157–1182.

[19] P. Lewis, "The characteristic selection problem in recognition systems," IRE Transactions on Information Theory, Vol. 8, 1962, pp. 171–178.

[20] F. Alonso-Atienza, J. L. Rojo-Álvarez, A. Rosado-Muñoz, J. J. Vinagre, A. García-Alberola, and G. Camps-Valls, "Feature selection using support vector machines and bootstrap methods for ventricular fibrillation detection," Expert Systems with Applications, Vol. 39, No. 2, 2012, pp. 1956–1967, View at Publisher, View at Google Scholar, View at Scopus.

[21] C. Fernandez-Lozano, J. A. Seoane, P. Mesejo, Y. S. G. Nashed, S. Cagnoni, and J. Dorado, "2D-PAGE Texture classification using support vector machines and genetic algorithms," in Proceedings of the 4th International Conference on Bioinformatics Models, Methods and Algorithms, 2013, pp. 5–14, Scitepress.

[22] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "An MR brain images classifier system via particle swarm optimization and kernel support vector machine," The Scientific World Journal, Vol., 2013, Article ID 130134, 9 pages, View at Publisher, View at Google Scholar

[23] L. Shuaishi, T. Yantao, P. Cheng, and L. Jinsong, "Facial expression recognition approach based on least squares support vector machine with improved particle swarm optimization algorithm," IEEE International Conference on Robotics and Biomimetics (ROBIO), 2010, pp. 399 – 404.

[24] M. H. Nguyen, and F. De la Torrea, "Optimal Feature Selection for Support Vector Machines," 2009, http://www.contrib.andrew.cmu.edu/~minhhoan/papers/SVMFeatureWeight_PR.pdf.

[25] C. L. Huang and C. J. Wang, "A GA-based feature selection and parameters optimization for support vector machines," Expert Systems with Applications, Vol. 31, 2006, pp. 231–240.

[26] M. S. Chen, Ch. Hwang, and T. Y. Ho, "Terrain Image Classification with SVM Advances in Swarm Intelligence," Lecture Notes in Computer Science Vol. 7929, 2013, pp. 89 – 97.

[27] A. Luiza, Automated Segmentation and Classification of Brain Magnetic Resonance Imaging, http://www.cs.ualberta.ca/~luiza/c615/proj.pdf.

[28] X.Y Wang and Q.Y. Wang, "Color Image Segmentation Using Automatic Pixel Classification with Support Vector Machine," Neurocomputing, 2011, 3898–3911.

[29] L. Lei, D. Y. Shi, and X. Jun, "Color Image Segmentation Based-on SVM Using Mixed Features and Combined Kernel, Intelligence Science and Big Data Engineering," Lecture Notes in Computer Science Volume 8261, 2013, pp. 401-409.

[30] A. Ladgham, G. Torkhani, A. Sakly, and A. Mtibaa, "Modified Support Vector Machines for MR Brain Images Recognition," IN IEEEInternational Conference on Control, Decision and Information Technologies (CoDIT'13); May 6-8, 2013, Hammamet, Tunisia.

[31] A. Ladgham, F. Hamdaoui, A. Sakly, and A. Mtibaa, "Fast MR brain image segmentation based on modified Shuffled Frog Leaping Algorithm," Signal, Image and Video Processing, 2013, DOI: 10.1007/s11760-013-0546-y

[32] A. Bhaduri, "Color Image Segmentation using Clonal Selection-based Shuffled Frog Leaping Algorithm," International Conference on Advances in Recent Technologies in Communication and Computing, ARTCom, 2009, pp. 517–520.

[33] J. G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," Journal of the Optical Society of America A, Vol. 2, No. 7, 1985, pp. 1160–1169.

[34] Y. Kosaka and K. Kotani, "Facial Expression Analysis by Kernal Eigen Space Method based on Class Features (KEMC) Using Non-Linear Basis For Separation of Expression Classes," International Conference on Image Processing (ICIP), 2004.

[35] M. M. Eusuff and K. E. Lansey, "Optimization of water distribution network design using the shuffled frog leaping algorithm," J. Water Resour. Planning Manag, Vol. 129, No. 3, 2003, pp. 210–225.

[36] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," IEEE Int. Conf. Neural Netw, 1995.

[37] Hamdaoui, F., Ladgham, A., Sakly, and A., Mtibaa, A, "A new images segmentation method based on modified PSO algorithm," International Journal of Imaging Systems and Technology, 23 (3), pp. 265 - 271 (2013)

[38] D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley. Reading, MA, 1989.

[39] V. N. Vapnik, "The nature of statistical learning theory," New York: Springer, 1995.

[40] V. N. Vapnik and S. Kotz, "Estimation of Dependences Based on Empirical Data," Springer, ISBN 0-387-30865-2, 2006, 510 pages.

[41] AD Shieh and DF Kamm, "Ensembles of one class support vector machines," in Proceedings of the Multiple Classifier Systems(Springer, Berlin, 2009), pp. 181–190.

[42] C. Chang and C. Lin, LIBSVM: a Library for Support Vector Machines, 2011, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, last accessed: 24.05.2011.

[43] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," Data Mining and Knowledge Discovery Vol. 2, 1998, pp. 121–167.

[44] M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," Automation and Remote Control 25, 1964 pp. 821–837.