Detection of Alzheimer's disease and mild cognitive impairment based on structural volumetric MR images using 3D-DWT and WTA-KSVM trained by PSOTVAC

Yudong Zhang\abcs, Shuihua Wang\abcs, Preetha Phillips\d, Zhengchao Dong\b, Genlin Ji\ab, Jiquan Yang\c

\a School of Computer Science and Technology, Nanjing Normal University, Nanjing 210023, Jiangsu, China
\b Translational Imaging Division & MRI Unit, Columbia University and New York State Psychiatric Institute, New York, NY 10032, USA
\c Jiangsu Key Laboratory of 3D Printing Equipment and Manufacturing, Nanjing 210042, Jiangsu, China
\d School of Natural Sciences and Mathematics, Shepherd University, Shepherdstown, WV 25443, USA

\textbf{A R T I C L E  I N F O}

\begin{itemize}
  \item Article history:
  \item Received 16 December 2014
  \item Received in revised form 20 May 2015
  \item Accepted 21 May 2015
\end{itemize}

\textbf{Keywords:}
Magnetic resonance imaging
Multiclass SVM
Kernel SVM
Particle swarm optimization
Time-varying acceleration-coefficient

\textbf{A B S T R A C T}

\textbf{Background:} We proposed a novel classification system to distinguish among elderly subjects with Alzheimer’s disease (AD), mild cognitive impairment (MCI), and normal controls (NC), based on 3D magnetic resonance imaging (MRI) scanning.

\textbf{Methods:} The method employed 3D data of 178 subjects consisting of 97 NCs, 57 MCIs, and 24 ADs. First, all these 3D MR images were preprocessed with atlas-registered normalization to form an averaged volumetric image. Then, 3D discrete wavelet transform (3D-DWT) was used to extract wavelet coefficients of the volumetric image. The triplets (energy, variance, and Shannon entropy) of all subbands coefficients of 3-DWT were obtained as feature vector. Afterwards, principle component analysis (PCA) was applied for feature reduction. On the basic of the reduced features, we proposed nine classification methods: three individual classifiers as linear SVM, kernel SVM, and kernel SVM trained by PSO with time-varying acceleration-coefficient (PSOTVAC), with three multiclass methods as Winner-Takes-All (WTA), Max-Wins-Voting, and Directed Acyclic Graph.

\textbf{Results:} The 5-fold cross validation results showed that the “WTA-KSVM+PSOTVAC” performed best over the OASIS benchmark dataset, with overall accuracy of 81.5% among all proposed nine classifiers. Moreover, the method “WTA-KSVM+PSOTVAC” exceeded significantly existing state-of-the-art methods (accuracies of which were less than or equal to 74.0%).

\textbf{Conclusion:} We validate the effectiveness of 3D-DWT. The proposed approach has the potential to assist in early diagnosis of ADs and MCIs.

© 2015 Elsevier Ltd. All rights reserved.

\textbf{Abbreviations:} (D)(F)WT, (discrete) (fast) wavelet transform; (M)(K)SVM, (multiclass) (kernel) support vector machine; 1D/2D/3D-DWT, one/two/three dimensional DWT; AD, Alzheimer's disease; ALS, alternating least squares; ASF, atlas scaling factor; CA, classification accuracy; CDR, clinical Dementia rating; CM, confusion matrix; CV, cross validation; DAG, Directed Acyclic Graph; DS, down-sampling; eTIV, estimated total intracranial volume; GA, genetic algorithm; KNN, k-Nearest neighbor; MCI, mild cognitive impairment; MMSE, mini-mental state examination; MR(I), magnetic resonance (imaging); MSE, median square error; MWV, Max-Wins-Voting; NC, normal control; nWBV, normalized whole brain volume; OASIS, open access series of imaging studies; PC(A), principal component (analysis); PSO, particle swarm optimization; QP, quadratic programming; RBF, radial basis function; ROI, region of interest; RS, random search; SA, simulated annealing; SES, socio-economic status; SMO, sequential minimal optimization; SVMDT, SVM decision tree; TN/FN/(R), true/false negative (rate); TP/(P)/(R), true/false positive (rate); TVAC, time-varying acceleration coefficient; WTA, Winner-Takes-All.

\* Corresponding authors at: Nanjing Normal University, School of Computer Science and Technology, 1 Wenyuan, Nanjing 210023, China. Tel.: +86 15905183664.
E-mail addresses: zhangyudong@njnu.edu.cn (Y. Zhang), wangshuihua@njnu.edu.cn (S. Wang).

http://dx.doi.org/10.1016/j.bspc.2015.05.014
1746-8094/© 2015 Elsevier Ltd. All rights reserved.
1. Introduction

Alzheimer's disease (AD) is not a normal part of aging. It is a type of dementia that causes problems with memory, thinking, and behavior. Symptoms usually develop slowly and worsen over time, and may become severe enough to interfere with daily life, and lead to death [1]. There is no cure for this type of disease. In 2006, 26.6 million people suffered from this disease worldwide. AD is predicted to affect 1 in 85 people globally by 2050, and at least 43% of prevalent cases need high level of care [2]. As the world is evolving into an aging society, the burdens and impacts caused by AD on families and the society also increases significantly.

Early detection of AD is beneficial for the management of the disease. Mild cognitive impairment (MCI) is frequently seen as a prodromal stage of AD [3]. Hence, detecting AD and MCI accurately and effectively from normal healthy people is of great importance. At present, a multitude of neurologists and medical researchers have been deducing considerable time and energy toward this goal, and promising results have been continually springing up.

Magnetic resonance imaging (MRI) plays an important role in classifying AD subjects, MCI subjects, and normal controls (NC) [4–6]. In earlier times, much work was done to measure manually or semi-manually a priori region of interest (ROI) of magnetic resonance (MR) images, based on the fact that AD patients suffer more cerebral atrophy compared to NCs [7]. Most of these ROI-based analyses focused on the shrinkage of hippocampus & cortex, and enlarged ventricles [8].

However, the ROI-based methods suffer from some limitations. First, they focus on the ROIs pre-determined based on prior knowledge. Second, the accuracy of early detection depends heavily on the experiences of the examiners. Third, the mutual information among the voxels is difficult to operate [9]. Finally, there is no evidence that other regions (except hippocampus and entorhinal cortex) did not provide any information for AD and MCI. Besides, the auto-segmentation of ROI is not feasible in practice, and examiners tend to segment the brain manually. In contrast, we utilize machine-learning method in this study, in order to pick up and construct new features that can help guide the classification from the whole brain. Hence, no specific ROIs are needed.

ROI-based method is a kind of white-box, whose internal components are constructed by human experts and therefore can be viewed and understood by humans. The machine learning method belongs to the black-box model that is constructed by the data-driven learning algorithm. There is no intuitive knowledge of its internal working. In a word, the former is human-oriented, and the latter is machine-oriented. In the last decade, machine-learning methods are proven to achieve better performance than conventional human-oriented methods in various academic and industrial fields. Therefore, we choose the machine-learning model as the classifier in this study.

Revisiting the field of automated disease classification by machine learning, Chaplot, Patnaik and Jagannathan [10] used coefficients of discrete wavelet transform (DWT) as input to neural network self-organizing maps and support vector machine to classify MR images of the human brain. El-Dahshan, Hosny and Salem [11] suggested to use two-dimensional DWT (2D-DWT) with principal component analysis (PCA) in the first stage, and use forward back-propagation artificial neural network (FPANN) and k-nearest neighbors (KNN) in the second stage. Ramasamy and Anandhakumar [12] used fast-Fourier-transform based expectation-maximization Gaussian mixture model for brain tissue classification of MR images. Zhang and Wu [13] followed the method of El-Dahshan, Hosny and Salem [11], but replaced their classifiers with kernel support vector machine (K SVM). Three kernels (HPOL, IPOL, and GRB) were tested. Nathagopal and Rajamony [14] used wavelet texture parameters and neural network classifier to classify between benign and malignant brain tumors. Kalbkhani, Shayesteh and Zali-Varghahan [15] proposed a robust algorithm that classifies MR images into normal or one of the seven different diseases. Zhang, Wang, Ji and Dong [16] proposed a novel genetic pattern research algorithm to realize an MR brain image system. Saritha, Paul Joseph and Mathew [17] presented combined wavelet entropy based spider web plots and probabilistic neural network for classification of MR brain images. Zhang, Dong, Ji and Wang [18] suggested that removing spider-web-plot yielded the same classification performance. Das, Chowdhury and Kundu [19] proposed to use Ripple transform (RT) + PCA + least square SVM (LS-SVM), and the 5 × 5 CV shows high classification accuracies. En and Swee [20] used artificial neural network to develop a computer-aided knee Osteoarthritis Classification System. Vishnuvarthan and Pallikonda Rajasekar [21] proposed a segmentation method of MR brain images for tumor using fuzzy technique. Falahati, Westman and Simmons [22] reviewed recent studies that used machine learning and multivariate analysis in the field of AD research. This study included not only MRI but also positron emission tomography and cerebrospinal fluid biomarkers. Zhang, Wang and Dong [23] presented a classification method of AD based on structural MR images by Support Vector Machine Decision Tree (SVMDT). Padma and Sukanesh [24] used combined wavelet statistical texture features, to segment and classify AD benign and malignant tumor slices. El-Dahshan, Mohsen, Revett and Salem [25] used the feedback pulse-coupled neural network for image segmentation, the DWT for features extraction, the PCA for reducing the dimensionality of the wavelet coefficients, and the feed-forward back propagation neural network (FBPNN) to classify inputs into normal or abnormal. Zhou, Wang, Lei and Chen [26] suggested that 3D arterial spin labeling (ASL) detection combined with magnetic resonance spectroscopy (MRS) showed the regional hypo-perfusion with decrease of CBF and abnormal metabolic changes of the posterior cingulate cortex. Luo, Zhuang, Zhang, Wang, Yue and Huang [27] create 3D active appearance model (AAM) of hippocampus. Automatic recognition were carried out on both sides of the hippocampus in brain MR images of individuals with this model. Simoes, van Walsum and Slump [28] performed 3D texture analysis using local binary patterns computed at local image patches in the whole brain, combined in a classifier ensemble. Their method does not require either the segmentation of specific brain regions or the nonlinear alignment to a template. Zhou, Wang, Xu, Ji, Phillips, Sun and Zhang [29] used wavelet-entropy as the feature space, then they employed a Naive Bayes classifier (NBC) classification method. Their results over 64 images showed that the sensitivity of the classifier is 94.50%, the specificity 91.70%, the overall accuracy 92.60%. Zhang, Dong, Wang, Ji and Yang [30] used discrete wavelet packet transform (DWPT), and harnessed Tsallis entropy to obtain features from DWPT coefficients. Then, they used a generalized eigenvalue proximal SVM (GEP SVM) with RBF kernel. Yang, Zhang, Yang, Ji, Dong, Wang, Feng and Wang [31] selected wavelet-energy as the features, and introduced biogeography-based optimization (BBO) to train the SVM. Their method reached 97.78% accuracy on 90 T2-weighted MR brain images.

Those studies applied various machine leaning methods on various medical fields, and all achieved excellent results. Nevertheless, they treated 3D brain as multi-slice 2D images, and employed 2D image-processing techniques for each slice of images. For example, they used 2D-DWT to extract features from every slice of image. This did not take into account the compressibility of the slice-dimension of the 3D brain.

Therefore, we tried to employ the 3D (volumetric) image processing technique (i.e., 3D-DWT) to extract features directly from 3D MR brain images. The 3D-DWT expands the 2D-DWT to treat volumetric data. It was reported that 3D-DWT outperformed 2D-DWT in volumetric image processing [32]. The methodology of our
method was depicted briefly as follows: After 3D-DWT decomposition, we used three descriptors (energy, variance, and entropy) to extract information from each subband. Then, PCA was chosen since it reduces the dimensionality of the data and therefore reduces the computational cost of analyzing new data. Finally, we introduced nine classification methods based on the mix of three individual classifiers and three multiclass methods.

The structure of the rest is organized as follows: Next section shows the data procurement. Section 3 presents the feature extraction and reduction method, in which we concisely discuss the fundamental concept of 3D-DWT and PCA. Section 4 introduces in three individual classifiers: linear SVM, kernel SVM, and kernel SVM trained by PSOTVAC. Section 5 introduces three multiclass methods: Winner–Takes-All, Max-Wins-Voting, and Directed Acyclic Graph. Section 6 describes how to carry out a K-fold cross validation to obtain out-of-sample evaluation. Experiments in Section 7 give the results of the nine proposed classification methods, and compare the best of them with state-of-the-art methods. Section 8 summarizes our contributions, points out the reasons and limitations. Section 9 concludes the paper and plans future work. For the ease of reading, the acronyms appeared in this study are shown in the end of the paper.

2. Data procurement

We downloaded the public dataset from Open Access Series of Imaging Studies (OASIS, downloaded from http://oasis-brains.org) that contains cross-sectional MRI data, longitudinal MRI data, and additional data for each subject. We chose the cross-sectional dataset corresponding to the MRI scan of individuals at a single point in time. The OASIS dataset consists of 416 subjects aged 18 to 96. We excluded subjects under 60 years old, and then picked up 178 subjects (97 NCs, 57 MCIs, and 24 ADs) from the rest subjects. The attributes of the included subjects were briefly summarized in Table 1. For each subject, 3 or 4 individual T1-weighted 3D MR images obtained in a single scan session are included. The subjects are all right-handed, and include both men and women.

2.1. Volumetric data and its pre-processing

Fig. 1 showed the input and output data of the classifier. The input data contained the volumetric data and additional data. For each individual, all available 3 or 4 volumetric 3D MR brain images were motion-corrected, and coregistered to form an averaged 3D image. Then, those 3D images were spatially normalized to the Talairach coordinate space and brain-masked.

2.2. Additional data

Additional data involved three types of data: demographical feature, clinical examination, and derived anatomic volume, with the goal of increasing the performance of the classification. The demographical features contained gender (M/F), handedness, age, education, and socioeconomic status (SES). Education codes were listed in Table A1 in the Appendix. The handedness feature was not included since all subjects are right-handed.

The mini-mental state examination (MMSE) served as the clinical examination. It was a brief 30-point questionnaire test used to screen for cognitive impairment and dementia. The MMSE test included simple questions in a number of areas: the time and place, repeating lists of words, arithmetic, language use & comprehension, and basic motor skills. A typical MMSE test was shown in Table A2 in the Appendix.

Derived anatomic volumes contained the estimated total intracranial volume (eTIV) (mm³), atlas scaling factor (ASF), and normalized whole brain volume (nWBV). The eTIV was used as the correction for ‘intracranial volume (ICV)’ because certain structures scale with general head size. The ASF was defined as the volume scaling factor required matching each individual to the atlas target. The nWBV was acquired using the methods in literature [33].

In total, the additional data contains the following 8 features: gender, age, education, SES, MMSE, eTIV, ASF, and nWBV.

2.3. Target data

The clinical dementia rating (CDR) was interpreted as the target data. It is a numeric scale quantifying the severity of symptoms of dementia. The patient’s cognitive and functional performances were assessed in six areas: memory, orientation, judgment & problem solving, community affairs, home & hobbies, and personal care. In this study, we chose three types of CDR: (1) subjects with CDR of 0 were considered NC; (2) Subjects with CDR of 0.5 were considered MCI; (3) Subjects with CDR of 1 were considered AD.

3. Feature extraction and reduction

Table 2 showed related mathematical symbols used in this and following sections.

3.1. 3D wavelet transform

Traditional AD methods commonly used 2D-DWT, which did not take into consider the compressibility of the slice-dimension. Therefore, we tried to use 3D-DWT to extract features from a 3D brain.

The DWT is a powerful implementation of the wavelet transform (WT) using the dyadic scales and positions. Fig. 2(a) shows the procedure of a 3-level 1D-DWT, where the functions \( l(n) \) and \( h(n) \) denote the low-pass filter and high-pass filter, respectively.\( L \) and \( H \) denote the subbands of low-frequency and high-frequency, respectively. The downward arrow represents a DS operator used for downsampling [34].

In applying this technique to MR images, the 1D-DWT is applied separately to each dimension. As a result, there are 4 subbands (LL, LH, HH, HL) at each scale. The subband LL is used for the
next 2D-DWT (See Fig. 2b). The LL subband can be regarded as the approximation component of the image, while the LH, HL, and HH subbands can be regarded as the detailed components of the image. As the level of the decomposition increased, we obtained more compact yet coarser approximation components. Thus, wavelets provide a simple hierarchical framework for interpreting the image information.

The 3D-DWT expands the 2D-DWT to 3D for treating volume data. 2D-DWT decomposes data in row and column direction. Further, the 3D-DWT decomposes data in row, column, and slice direction. Similarly, the 8 subbands are: LLL, LLH, LHL, LHH, HLL, HLH, HHL, and HHH. Mathematically, 3D-DWT with separable wavelets can be regarded as the process of applying 1D-DWT on each vector along z-axis that share x-axis and y-axis coordinates after applying 2D-DWT for all comprising planes, pixels in which are located in the same z-axis positions [35]. In short, 3D-DWT can be implemented by 2D-DWT followed by 1D-DWT. The pseudo-codes for the 3D-DWT were illustrated in Table 3. Fig. 2(c) illustrated the schematic diagram of 2-level 3D-DWT. Fig. 2(d) showed the decomposition results of 1-level 3D-DWT of a simple cube model, by 3D rendering technique.

As the level of the decomposition increased, we obtained more compact yet coarser approximation components of the volumetric image, thus implementing the “fine-to-coarse” multi-resolution decomposition. For example, LHL3 is interpreted as the high-pass subband at multiresolution level 3 from directional filtering of low-pass along x-direction, high-pass along y-direction, and low-pass z-direction.

### 3.2. Volumetric feature extraction

Considering \( Y \) represents a 3D volumetric data, and \( Y(x, y, z) \) as a voxel located at the coordinate \((x, y, z)\). We utilize energy \( e \), variance

---

**Table 2**

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l, h )</td>
<td>Low-pass, high-pass filters</td>
</tr>
<tr>
<td>( x, y, z )</td>
<td>Spatial coordinates</td>
</tr>
<tr>
<td>( N_x, N_y, N_z )</td>
<td>Number of points along ( x, y, ) &amp; ( z )-axes</td>
</tr>
<tr>
<td>( L, H )</td>
<td>Low-pass, high-pass subbands</td>
</tr>
<tr>
<td>( \psi, \varphi )</td>
<td>Wavelet function and wavelet coefficients</td>
</tr>
<tr>
<td>( e, v, s )</td>
<td>Energy, variance, and Shannon entropy</td>
</tr>
<tr>
<td>( V, V' )</td>
<td>Volume data and its mean</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>Scaling factor in RBF kernel</td>
</tr>
<tr>
<td>( S )</td>
<td>Classification score</td>
</tr>
<tr>
<td>( A )</td>
<td>Number of attributes/features</td>
</tr>
<tr>
<td>( K )</td>
<td>Kernel function</td>
</tr>
<tr>
<td>( x_n, y_n, z_n )</td>
<td>The input data, target class, output (predicted) class of ( n )-th observation</td>
</tr>
<tr>
<td>( X, Y, O )</td>
<td>The set of input data, target class, output class of the whole observation</td>
</tr>
<tr>
<td>( \xi )</td>
<td>Slack variable measuring the distance between the margin and ( x_n )</td>
</tr>
<tr>
<td>( c, C )</td>
<td>Index and total number of classes</td>
</tr>
<tr>
<td>( n, N )</td>
<td>Index and total number of samples</td>
</tr>
<tr>
<td>( w, b )</td>
<td>Weights and biases of classifiers</td>
</tr>
<tr>
<td>( \epsilon )</td>
<td>Error penalty</td>
</tr>
<tr>
<td>( \text{MAX}, \text{Iter} )</td>
<td>Maximum iterations</td>
</tr>
<tr>
<td>( P, V, N_p )</td>
<td>Position, velocity, and total number of particles</td>
</tr>
<tr>
<td>( \omega )</td>
<td>Inertial weight</td>
</tr>
<tr>
<td>( r_1, r_2 )</td>
<td>Two uniformly distributed random variables</td>
</tr>
<tr>
<td>( c_x, c_y, c_z )</td>
<td>Two acceleration coefficients (initial value) (final value)</td>
</tr>
<tr>
<td>( i )</td>
<td>Current iteration</td>
</tr>
<tr>
<td>( k, K )</td>
<td>Index and total number of folds in CV</td>
</tr>
<tr>
<td>( F(k) )</td>
<td>Index of ( k )-th fold (( k )-th validation set)</td>
</tr>
<tr>
<td>( T(k) )</td>
<td>Index of ( k )-th training set</td>
</tr>
</tbody>
</table>

**Table 3**

<table>
<thead>
<tr>
<th>Algorithm 1: Pseudo-codes for 3D-DWT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load volumetric data ( Y );</td>
</tr>
<tr>
<td>([N_x, N_y, N_z] = \text{size}(Y);)</td>
</tr>
<tr>
<td>2D-DWT along ( x )-axis ( \forall x=1 \text{ to } N_x )</td>
</tr>
<tr>
<td>for ( y=1 \text{ to } N_y )</td>
</tr>
<tr>
<td>( Y(x, y, z) = \text{2D-DWT}(Y(x, y, z));)</td>
</tr>
<tr>
<td>( 1 )D-DWT across ( XY ) plane</td>
</tr>
<tr>
<td>for ( x=1 \text{ to } N_x )</td>
</tr>
<tr>
<td>for ( y=1 \text{ to } N_y )</td>
</tr>
<tr>
<td>( Y(x, y) = \text{1D-DWT}(Y(x, y, z));)</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>end</td>
</tr>
<tr>
<td>Output 3D-DWT coefficients ( Y ).</td>
</tr>
</tbody>
</table>
\( n \), and Shannon entropy \( s \) as the descriptors of the volume data \( V \). They are defined as

\[
e = \sum_{x} \sum_{y} \sum_{z} y^2(x, y, z)
\]

(1)

\[
v = \sum_{x} \sum_{y} \sum_{z} (y(x, y, z) - \bar{y})^2
\]

(2)

where \( \bar{y} \) represents the mean value of volume data \( V \).

\[
s = -\sum_{x} \sum_{y} \sum_{z} y^2(x, y, z) \log \left( 1 + y^2(x, y, z) \right)
\]

(3)

The triplets (energy, variance, and Shannon entropy) of all subbands of 178 brains were extracted, and then aligned into a row vector. Fig. 3 showed the flowchart of feature extraction. The volumetric images of each individual were of dimension 176×208×176 = 6,443,008. We performed 3-level 3D-DWT of “db2” wavelet on each 3D image, obtaining \( 8 \times 7 \times 7 = 22 \) subbands (8 subbands of level 3, 7 subbands of level 2, and 7 subbands of level 1). For each subband, the triplet \( (e, v, s) \) was extracted. Therefore, totally 66 volumetric features were obtained from the volume data for each subject.

3.3. Feature reduction

At this step, we had 66 volumetric features (Section 3.2) and 8 additional features (Section 2.2) for each subject. Although 66+8 = 74 features were not a burden for current computers, they may complicate the subsequent classifier and increase computational resources. PCA effectively reduces the dimensionality of the data and therefore reduces the computational cost of analyzing new data.

PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components (PC). Suppose the number of observations is \( n \), the number of variables is \( p \), and therefore, the size of input matrix \( X \) is \( n \times p \). There are three main methods to perform the PCA:

1. Singular value decomposition (SVD), which is the canonical method for PCA.
2. Eigenvalue decomposition (EIG) of the covariance matrix. The EIG algorithm is faster than SVD when the number of observations exceeds the number of variables, but it is less accurate because the condition number of the covariance is the square of the condition number of \( X \).
3. Alternating least squares (ALS) algorithm. It finds the best rank-\( k \) approximation by factoring input matrix into an \( n \times k \) left
For this study, the features of 178 subjects were arranged into two ‘input data matrix’. One was the volumetric feature matrix with size of 178 × 66, and the other was the additional feature matrix with size of 178 × 8. Each input data matrix was decomposed into the PC ‘score matrix’ and the ‘coefficients matrix’. Here, the rows and columns of ‘score matrix’ correspond to subjects and components, respectively. Each row of the ‘coefficients matrix’ contained the coefficients for one PC, and they were in the order of descending component variance.

\[
\min \frac{1}{2} \|w\|^2 + \epsilon \sum_{n=1}^{N} \xi_n \\
\text{s.t.} \quad y_n (w^T x_n - b) \geq 1 - \xi_n \\
\quad \xi_n \geq 0, \quad n = 1, \ldots, N
\]  

We used ALS-PCA to extract PCs from additional feature matrix since it had missing values, and SVD-PCA to extract PCs from the volumetric feature matrix. Fig. 4 showed the PCA implementation in this study. Here the number of PCs was selected as can explain at least 95% and 99.95% variances for volumetric feature and additional feature, respectively.

4. Individual classifier

For the classifier, we used SVM because it is the hottest classifier in the field of pattern recognition. Nearly half of open publications related to pattern recognition used the SVM. It is easy to implement and the results are usually excellent compared to other classifiers. In this study, we improved SVM by three aspects: (1) we included kernel technique with the aim of improving classification performance; (2) we used three multi-class techniques to deal with the 3-class classification problem, as SVM can only deal with binary classification problem; (3) we used PSO-TVAC method to optimize the kernel parameters.

4.1. Linear SVM

The introduction of Support Vector Machine (SVM) attracts scholars from various fields. The advantages of SVMs included high accuracy, elegant mathematical tractability, and direct geometric interpretation. Given a A-dimensional N-size training dataset as

\[
\{(x_n, y_n) | x_n \in \mathbb{R}^A, y_n \in \{-1, +1\} \}, \quad n = 1, \ldots, N
\]

where \(y_n\) is either -1 or 1 corresponds to the class 1 or 2, respectively. Each \(x_n\) is a A-dimensional vector. The maximum-margin hyperplane that divides class 1 from class 2 is the support vector machine we want. Considering that any hyperplane can be written in the form \(w^T x - b = 0\), where \(w\) denotes the normal vector to the hyperplane. We want to choose the \(w\) and \(b\) to maximize the margin between the two parallel hyperplanes as large as possible while still separating the data. Two parallel hyperplanes are defined as \(w^T x - b = \pm 1\). Therefore, the task is transformed to an optimization problem. That is, we want to maximize the distance between the two parallel hyperplanes, subject to prevent data falling into the margin. The problem is finalized as

\[
\min \frac{1}{2} \|w\|^2 \\
\text{s.t.} \quad y_n (w^T x_n - b) \geq 1, \quad n = 1, \ldots, N
\]  

However, in practical applications, there may exist no hyperplane that can split the samples perfectly. In such case, the “soft margin” method will choose a hyperplane that splits the given samples as cleanly as possible, while still maximizing the distance to the nearest cleanly split samples. Positive slack variables \(\xi_n\) are introduced to measure the misclassification degree of sample \(x_n\) (the distance between the margin and the vectors \(x_n\) that lying on the wrong side of the margin). Then, the optimal hyperplane separating the data can be obtained by the following optimization problem.

\[
\min \frac{1}{2} \|w\|^2 + \epsilon \sum_{n=1}^{N} \xi_n \\
\text{s.t.} \quad y_n (w^T x_n - b) \geq 1 - \xi_n \\
\quad \xi_n \geq 0, \quad n = 1, \ldots, N
\]

where \(\epsilon\) is the error penalty (i.e., box constraint in some other literatures). Therefore, the optimization becomes a trade-off between a large margin and a small error penalty. The constraint optimization problem can be solved using “Lagrange multiplier” as

\[
\min_{w \in \mathbb{R}^N, b \in \mathbb{R}} \frac{1}{2} \|w\|^2 + \epsilon \sum_{n=1}^{N} \xi_n - \sum_{n=1}^{N} \alpha_n \left[ y_n (w^T x_n - b) - 1 + \xi_n \right] - \sum_{n=1}^{N} \beta_n \xi_n
\]  

The min–max problem is not easy to solve, so Cortes and Vapnik further proposed a dual form technique to solve it.

The dual form of formula (6) can be designed as

\[
\max_{\alpha} \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m k(x_n, x_m) \\
\text{s.t.} \quad \sum_{n=1}^{N} \alpha_n y_n = 0, \quad n = 1, \ldots, N
\]

The key advantage is that the slack variables \(\xi_n\) vanish from the dual problem, with the constant \(\epsilon\) appearing only as an additional constraint on the Lagrange multipliers. Now, the optimization problem (8) becomes a quadratic programming (QP) problem, which is defined as the optimizing a quadratic function of several variables subject to linear constraints on these variables. Therefore, numerous methods can solve formula (6) within milliseconds, such as Sequential Minimal Optimization (SMO), least square, interior point method, augmented Lagrangian method, conjugate gradient method, simplex algorithm, etc. In this study, SMO is chosen due to its simplicity and rapidness.

4.2. Kernel SVM

Traditional linear SVMs cannot separate intrinsically distributed practical data. Recently, multiple improved SVMs have grown rapidly, among which the kernel SVMs (KSVMS) are the most popular and effective. KSVMS have the following advantages [37]: (1) work very well in practice and have been remarkably successful in such diverse fields as natural language categorization, bioinformatics and computer vision; (2) have few tunable parameters; and (3)
training often involves convex quadratic optimization. Hence, solutions are global and usually unique, thus avoiding the convergence to local minima exhibited by other statistical learning systems.

The radial basis function (RBF) kernel is one of the most widely used kernels with the form as

\[ K(x_m, x_n) = \exp\left(-\frac{\|x_m - x_n\|^2}{2\sigma^2}\right) \]  \hspace{1cm} (9)

where \( K \) is the kernel function, and \( \sigma \) the scaling factor in RBF kernel. We put formula (9) into (8), and got the final KSVM training function as

\[
\begin{aligned}
\max_{\alpha} & \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_m \alpha_n y_n y_m \exp\left(-\frac{\|x_m - x_n\|^2}{2\sigma^2}\right) \\
\text{s.t.} & \quad 0 \leq \alpha_n \leq C \\
& \quad \sum_{n=1}^{N} \alpha_n y_n = 0 \quad n = 1, \ldots, N
\end{aligned} \]  \hspace{1cm} (10)

It is still a quadratic programming problem, and we can use abovementioned methods in Section 4.1 to solve the problem. However, there is still an outstanding issue, i.e., the value of parameters \( \sigma \) in equation (10). We will discuss the problem in Section 4.3.

### 4.3. PSOTVAC

The weights \( w \) and biases \( b \) of KSVM were trained by SMO, Scholars pointed out the performance of the classifier was not sensitive to the value of error penalty \( \varepsilon \) [38], so we merely assign it the value of 100. However, the problem arose as how to determine the kernel parameter \( \sigma \).

KSVM-users were inclined to use trial-and-error or grid-searching method to find the optimal values of kernel parameter \( \sigma \), however, it will cause heavy computation burden [39]. The reason behind was that the number of kernel parameters was equal to the number of individual KSVMs, which increased either linearly with the class number \( C \) for WTA method, or increase quadratically with \( C \) for both MWV and DAG method. Fig. 5 illustrated their relationship.

Therefore, we used global algorithm to find the optimal scaling factor \( \sigma \) of kernels of each individual KSVM. Particle swarm optimization (PSO) was chosen since it had various successful applications in academic and industrial fields. Scholars have proven PSO had better performance than evolutionary computation [40], simulated annealing [41], cultural algorithm [42], etc. The input is the three scaling factors of RBF kernel (one for each individual SVM. See Fig. 5), the output is the classification accuracy on the whole dataset. The task is to find the optimal scaling factors that maximize the classification accuracy.

PSO performs searching via a swarm of particles that updates from iteration to iteration. To seek the optimal solution, each particle moves in the direction to its previously best (pbest) position and the global best (gbest) position in the swarm [43].

\[ \text{pbest}(i, t) = \arg \min_{k=1,\ldots,T} f(P_k(t)) \]  \hspace{1cm} (11)

\[ \text{pbest}(i, t) = \arg \min_{k=1,\ldots,T} f(P_k(t)), \quad i \in \{1, 2, \ldots, N_p\} \]

Fig. 5. The curve of the number of individual KSVM against the number of class.

Fig. 6. Illustration of different classifiers that predict a new instance as class 3: (a) WTA, (b) MWV, (c) DAG.
\[
gbest(t) = \arg\min_{i \in \{1, \ldots, N_p\}, k \in \{1, \ldots t\}} [f(P_i(k))] \quad (12)
\]

where \( i \) denotes the particle index, \( N_p \) the total number of particles, \( t \) the current iteration number, \( f \) the fitness function, and \( P \) the position. The velocity \( V \) and position \( P \) of particles are updated by the following equations.

\[
V_i(t + 1) = \omega V_i(t) + c_1 r_1 (\text{gbest}(i, t) - P_i(t)) \\
+ c_2 r_2 (\text{gbest}(t) - P_i(t)) \quad (13)
\]

\[
P_i(t + 1) = P_i(t) + V_i(t + 1) \quad (14)
\]

where \( V \) denotes the velocity. The inertia weight \( \omega \) is used to balance the global exploration and local exploitation. The \( r_1 \) and \( r_2 \) are uniformly distributed random variables within range \([0, 1]\). The \( c_1 \) and \( c_2 \) are positive constant parameters called “acceleration coefficients”.

In addition to standard PSO, a variety of improved versions as proposed. The Time Varying Acceleration Coefficients (TVAC) technique was introduced, and we termed the combined algorithm of both PSO and TVAC as PSOTVC. It can enhance the global search ability in the early stage, and encourage the local search ability of particles at the end of the search. In order to achieve this goal, PSOTVC gives more weight on cognitive component and less weight on social component at the former stage, and gives less weight on cognitive component and more weight on social component in the latter stage. Mathematically, PSOTVC tunes \( c_1 \) and \( c_2 \) as

\[
c_1 = (c_{1f} - c_{1i}) \cdot t/MAX_{iter} + c_{1i} \\
c_2 = (c_{2f} - c_{2i}) \cdot t/MAX_{iter} + c_{2i} \quad (15)
\]

where \( c_{1i} \) and \( c_{1f} \) represents the initial and final value of \( c_1 \), respectively, \( c_{2i} \) and \( c_{2f} \) represents the initial and final value of \( c_2 \), respectively.

5. Multiclass method

SVMs or K SVMs were originally designed for binary classification. Several methods were proposed for multi-class problems, and the dominant approach was to reduce the single multiclass problem into multiple binary classification problems. In this study, we introduced three types of Multiclass K SVM (MKSVM). They are Winner-Takes-All (WTA), Max-Wins-Voting (MWV), and Directed Acyclic Graph (DAG) methods. Combinations of multiclass methods with SVM are abbreviated to WTA-K SVM, MWV-K SVM, and DAG-K SVM, respectively.

5.1. Winner-Takes-All

Assume there are totally \( C \) (>2) classes. WTA strategy classifies new instances based on the idea of one-versus-all [44]. We first train \( C \) different binary K SVMs, each one trained to distinguish the data in a single class from the data of all the remaining classes. When applied to a new test data, all the \( C \) classifiers are run, and the classifier that outputs the largest value is chosen. If there are two identical output values, WTA selects the class with the smallest index.

The mathematical model is described as follows. Given a \( A \)-dimensional \( \mathbb{R}^A \)-size training dataset of the form

\[
\{(x_n, y_n) | x_n \in \mathbb{R}^A, y_n \in \{1, 2, \ldots, C\}\}, \quad n = 1, 2, \ldots, N \quad (16)
\]

where \( x_n \) is a \( A \)-dimensional vector, and \( y_n \) is the known class label of each \( x_n \). The classification score \( S \) for \( c \)th individual binary K SVM can be defined as

\[
S_c(x) = \sum_{n=1}^{N} y_n^c \alpha_n^c k(x_n, x) - b_c, \quad c = 1, 2, \ldots, C \quad (17)
\]

with the interpreted output as

\[
y_n^c = \begin{cases} 
+1 & \text{if } y_n \in \text{cth class} \\
-1 & \text{otherwise}
\end{cases} \quad (18)
\]

where \( c \) represents the class index, \( S \) the score, and \( N \) the number of training data; \( y_n^c \in \{+1, -1\} \) depends on the class label of \( x_n \), if \( y_n \) belongs to the \( c \)th class, \( y_n^c = +1 \), otherwise \( y_n^c = -1 \); \( k() \) is the predefined kernel function; \( \alpha_n^c \) is the Lagrange coefficient; and \( b_c \) is the bias term. \( \alpha_n^c \) and \( b_c \) are obtained by training the \( c \)th individual K SVM. The final output of the whole WTA-K SVM is

\[
o(x) = \arg\max_{c \in C} S_c(x) \quad (19)
\]

Here we use the “\( \arg\max \)” function instead of the “sign” function that is commonly used in binary classification. Fig. 6(a) gave an illustration. Suppose we had four different classes and therefore established four individual classifiers. The first classifier “1v meant all the samples in class 1 as positive and all samples out of class 1 as negative. The scores of the four classifiers were listed as \( S_1(x) = -0.9, S_2(x) = -0.7, S_3(x) = 0.5, S_4(x) = 0.3 \). We selected the third classifier because its score was the largest among all individual classifiers, and the given instance was recognized as class 3.

5.2. Max-Wins-Voting

MWV classifiers new instance based on the one-versus-one approach [44]. We construct a binary K SVM for each pair of classes, so in total we will get \( C(C - 1)/2 \) individual binary K SVMs. When applied to a new test data, each K SVM gives one vote to the winning class, and the test data is labeled with the class having most votes. If there are two identical votes, MWV selects the class with the smallest index.

The mathematical model is described as follow. The \( ij \)th \( (i = 1, 2, \ldots, C - 1, j = i + 1, \ldots, C) \)th individual binary K SVM is trained with all data in the \( i \)th class with \( +1 \) label and all data of the \( j \)th class with \( -1 \) label, so as to distinguish \( i \)th class from \( j \)th class. The classification score of \( ij \)th K SVM is

\[
S_{ij}(x) = \sum_{n=1}^{N_i} y_{ij}^n \alpha_{ij}^n k(x_n, x) - b_{ij} \quad (20)
\]

\[i = 1, 2, \ldots, C - 1, \quad j = i + 1, \quad i + 2, \ldots, C\]

The interpretation output result is

\[
y_{ij}^n = \begin{cases} 
+1 & x_{ij}^n \in \text{ith class} \\
-1 & x_{ij}^n \in \text{jth class}
\end{cases} \quad (21)
\]

where \( N_i \) and \( N_j \) denotes the total number of \( ij \)th class and \( j \)th class, respectively, \( y_{ij}^n \in \{+1, -1\} \) depends on the class label of \( x_{ij}^n \). If \( x_{ij}^n \) belongs to \( i \)th class, \( y_{ij}^n = +1 \); otherwise \( y_{ij}^n = -1 \). \( \alpha_{ij}^n \) is the Lagrange coefficient; and \( b_{ij} \) is the bias term. \( \alpha_{ij}^n \) and \( b_{ij} \) are obtained by training the \( ij \)th individual SVM. The output of \( ij \)th K SVM is the sign function of its score, namely, \( o_{ij}(x) = \text{sgn}(S_{ij}(x)) \). If \( S_{ij}(x) > 0 \), then the output \( o_{ij}(x) \) is +1, denoting \( x \) belongs to \( i \)th class;
otherwise output is \(-1\), denoting \(x\) belongs to \(j\)th class. The final output of the MWV is

\[
\sigma(x) = \arg\max_i \left( \sum_j a_{ij}(x) \right)
\]  

(22)

Fig. 6(b) showed an illustration of MWV. There were four classes in total, so six classifiers were established. For the new instance, the classifiers output 1, 3, 4, 3, 2, 3, respectively. We did a count of results that there were 3 classifiers that outputs 3, and only 1 classifier that outputs 1, 2, and 4. Therefore, class 3 was the final decision.

5.3. Directed Acyclic Graph

DAG is a graph whose edges have an orientation and no cycles. The DAG-KSVM constructs the same individual classifier set as the MWV using one-versus-one model, however, the output of each individual KSVM is explained differently. When \(a_{ij}(x)\) is \(+1\), it denotes that \(x\) does not belong to \(j\)th class; when \(a_{ij}(x)\) is \(-1\), it denotes that \(x\) does not belong to \(i\)th class. Therefore, the final decision cannot be reached until the leaf node is reached [45].

Fig. 6(c) showed the DAG for finding the optimal class out of 4 classes. Here, the root node and intermediate nodes represented the individual binary KSVM, whereas the leaf nodes represented the output label. Given a test sample \(x\) starting at the root node, the individual binary KSVMs were evaluated. The node was then exited via the evaluation result to either left edge or right edge. The next KSVM’s function was evaluated again until the leaf node was reached. Therefore, DAG-KSVM cost less computation time compared to MWV-KSVM. In this case, the MWV-KSVM needed to cover all nodes of 6 individual KSVMs, yet the DAG-KSVM only needed to evaluate only 3 individual KSVMs.

6. Implementation and evaluation

Based on above procedures (Sections 2–5), we can summarize the proposed method, designed the experiments, and gave the evaluation method.

6.1. Algorithm implementation

In all, the implementation of the proposed method was drawn in Fig. 7. The detailed pseudocodes were listed in Table 4.

6.2. K-fold CV

If the training set is used as validation set, then we will get an optimistically biased assessment. The estimate result is termed the “in-sample estimate”. Cross Validation (CV) is a model validation technique for assessing the real accuracy the classifier will achieve for an independent set. The CV estimate is termed “out-of-sample estimate”.

In this study, K-fold CV is employed, and \(K\) is assigned with a value of 5 considering the best compromise between computational cost and reliable estimates, i.e., the dataset is randomly divided into 5 mutually exclusively subsets of approximately equal size, in which 4 subsets are used as training set and the last subset
Table 4

Pseudocode of the proposed algorithm.

Algorithm 2—Pseudocode of the proposed algorithm

Step 1: Data import: Import the OASIS dataset, and select 178 objects (Table 1).
Step 2: Class setting: training data includes MRI data and additional data, target data is set according to CDR values (Section 2.3).
Step 3: Feature extraction and reduction

a. 3D-DWT: Perform 3-Level 3D-DWT on volumetric MR images, and get 22 subbands of wavelet coefficients (Section 3.1 & Table 3).

b. Volumetric feature: Extract energy, variation, and Shannon entropy from each subband, and align them to form a row vector (Fig. 3).

c. Feature reduction: Perform SVD–PCA on volumetric features of MRI data, and select PCs that explains 95% of total variances. Perform ALS–PCA on additional data. Select PCs that explains 99.95% of total variances (Fig. 4).

Step 4: Out-of-sample test

a. K-fold CV: Randomly divide the 178 samples to K folds (Fig. 8). Let k = 1 and begin the kth fold.

b. Select kth fold F(k) as the validation set, and the rest four folds as the training set T(k).

c. Training & Test (Fig. 9)

i. Classifier model: Submit the training set to the proposed three classifiers as WTA–K SVM (Fig. 6a), MWV–K SVM (Fig. 6b), and DAG–K SVM (Fig. 6c).

ii. Train: The parameters α of all individual K SVMs are trained by PSOTVAC (Section 4.3). The parameters w and b of all individual K SVMs are trained by SMO algorithm (Section 4.1).

iii. Validation: Submit the validation set to the trained classifier, and record the predicted class of ith fold.

a. If k = K, combine the predicted class of all folds, and jump to Step 5, otherwise return to Step k with k = k + 1.

Step 5: Evaluation

b. Generate the Confusion Matrix (CM) of the out-of-sample test.
Calculate the classification accuracy of in-class versus out-of-class.

is used as validation set. The abovementioned procedure repeated 5 times, so each subset is used once for validation.

Suppose we do a K-fold CV shown in Fig. 8, the index of the whole observation \{1, 2, ..., N\} is randomly divided into five folds \{F(1), F(2), ..., F(K)\}.

\[
\bigcup_{k=1}^{K} F(k) = \{1, 2, ..., N\} \tag{23}
\]

At kth run, the kth fold F(k) is designated as the index of validation set, and the rest folds are set as the index of training set T(k), i.e.,

\[
T(k) = \bigcup_{i \in K \setminus \{k\}} F(i) = \{1, 2, ..., N\} \setminus F(k) \tag{24}
\]

Here “\(\setminus\)" represents the set-subtraction symbol. It is self-evident that training set and validation set of each run are independent.

\[
T(k) \cap F(k) = \emptyset, \quad k = \{1, 2, ..., K\} \tag{25}
\]

Fig. 8 Illustration of 5-fold CV.

6.3. Out-of-sample evaluation

The out-of-sample test used out-of-sample observations, i.e., those not used in the training of the classifier. It is unbiased compared to in-sample test. In this section, we explained in depth how to perform an out-of-sample test based on K-fold CV.

Fig. 9 shows how an out-of-sample test is performed based on 5-fold CV. The MKSVM in this study will contain three K SVMs no matter what multiclass technique is selected. The scaling parameters \(\sigma\) of RBF kernel of every individual K SVM are set as the predictor variables, and the median square error (MSE) between the target classes and output classes of training set \(X(T(k))\) are set as the fitness function as

\[
\min_{\sigma} \|Y(T(k)) – O(T(k)|\sigma\|^2, \quad k \in \{1, 2, ..., K\} \tag{26}
\]

Here \(Y\) and \(O\) denote the target class and output class of the whole set, respectively. PSOTVAC runs iteratively until the MSE reaches minimal. Afterwards, the validation set \(X(F(k))\) is submitted to the trained classifier so as to obtain the predicted output \(O(F(k))\) on kth validation set \(F(k)\).

Recall that \(T(k)\) and \(F(k)\) are independent in formula (25) and \(O(F(k))\) is the output class of kth validation set \(F(k)\) from a trained classifier with training data as \(T(k)\), we can infer that \(O(F(k))\) is an out-of-sample estimate. The above procedure repeats for each fold, and then we perform a union operation on all \(O(F(k))\) of each fold as

\[
\bigcup_{k=1}^{K} O(F(k)) = O \left( \bigcup_{k=1}^{K} F(k) \right) = O(X) = O \tag{27}
\]

The result \(O\) is the output class of the whole set under the out-of-sample test condition. Therefore, the confusion matrix is unbiased. It element at \(i\)th row and \(j\)th column is counted as the number of observations of class \(i\) predicted to class \(j\), with the definition of \(CM(i,j) = \sum O(Y = i) \rightarrow j\).

7. Experiments, results, and analysis

The programs were in-house developed using Matlab 2014a, and run on IBM laptop with 3 GHz Intel i3 dual-processor and 8GB RAM. The readers can repeat our results on any machine where Matlab is available.
7.1. Dataset preprocessing

Fig. 10 shows the snapshots of a prescribed subject by our preprocessing procedure. For each individual, all available 3 or 4 volumetric MR brain images were motion-corrected, and coregistered to form an averaged image as shown in Fig. 10a. Then, the averaged image was spatially normalized to the Talairach coordinate space (Fig. 10b) and brain-masked (Fig. 10c).

7.2. 3D-DWT results

Table 5 gives the results of the 3-level 3D-DWT of a prescribed subject. The results are shown from not only 3D rendering, but also the axial, sagittal, and coronal planes. The ISO-value of 3D rendering was assigned with a value of 0.01.

The 3D textures in Table 5 are clearly visible in Level 1, and they turn coarser as the spatial resolution decreases, attributed to the fact that a low-pass subband dyadically cut-off into eight higher-level subbands. We set the level of decomposition to 3, since too low resolution contained little information. The 3D wavelet decomposition can be implemented by 2D-DWT (x-y plane) followed by 1D-DWT (z-axis). In addition, “2D-DWT (x–z plane) followed by 1D-DWT (y-axis)” and “2D-DWT (y–z plane) followed by 1D-DWT (x-axis)” make no significant difference.

7.3. PCA results

We extracted the energy, variance, and Shannon entropy features from the 22 subbands of 3D-DWT. For each subject, there were 66 volumetric features and 8 additional features in total. The data were centered to zero-mean. SVD and ALS algorithms were used to the volumetric features and additional features, respectively.

The PCA results are shown in Fig. 11. The x-axis denotes the number of PCs, and the y-axis the percentage of the total variance explained by each principal component. The threshold lines were assigned with a value of 95% and 99.95%, respectively. Fig. 11(a) shows the results of 66 volumetric features. The 9 PCs accumulate to more than 95% of the total variances. Fig. 11(b) shows the results of additional data, of which 3 PCs accumulate to more than 99.95% of the total variances.

7.4. K-fold CV

We used 5-fold CV to divide the whole dataset randomly. The detailed randomly division was shown in Fig. 12, with different colors representing different folds. The number of subjects of the three classes in each fold was shown in Table 6.

The index of CV in Fig. 12 shows that those folds were complementary, independent, and uniformly distributed in the whole range [0, 178]. The advantages of K-fold CV over repeated random
Table 5 3-level 3D-DWT of a prescribed subject.

<table>
<thead>
<tr>
<th>3D</th>
<th>Level</th>
<th>LLL</th>
<th>LLH</th>
<th>LHL</th>
<th>LHH</th>
<th>HLL</th>
<th>HLH</th>
<th>HHL</th>
<th>HHH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ISO-Value = 0.01

<table>
<thead>
<tr>
<th>Axial</th>
<th>Level</th>
<th>LLL</th>
<th>LLH</th>
<th>LHL</th>
<th>LHH</th>
<th>HLL</th>
<th>HLH</th>
<th>HHL</th>
<th>HHH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sagittal</th>
<th>Level</th>
<th>LLL</th>
<th>LLH</th>
<th>LHL</th>
<th>LHH</th>
<th>HLL</th>
<th>HLH</th>
<th>HHL</th>
<th>HHH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Coronal</th>
<th>Level</th>
<th>LLL</th>
<th>LLH</th>
<th>LHL</th>
<th>LHH</th>
<th>HLL</th>
<th>HLH</th>
<th>HHL</th>
<th>HHH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 6
Statistics of number of subjects in all five folds.

<table>
<thead>
<tr>
<th>Class</th>
<th>Fold-1</th>
<th>Fold-2</th>
<th>Fold-3</th>
<th>Fold-4</th>
<th>Fold-5</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC</td>
<td>21</td>
<td>17</td>
<td>19</td>
<td>19</td>
<td>21</td>
<td>97</td>
</tr>
<tr>
<td>MCI</td>
<td>9</td>
<td>15</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>57</td>
</tr>
<tr>
<td>AD</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>24</td>
</tr>
<tr>
<td>Total</td>
<td>35</td>
<td>36</td>
<td>36</td>
<td>35</td>
<td>36</td>
<td>178</td>
</tr>
</tbody>
</table>

Table 7
Overall classification accuracy comparison of the proposed nine classifiers.

<table>
<thead>
<tr>
<th>Linear SVM (%)</th>
<th>KSVM (%)</th>
<th>KSVM + PSOTVAC (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WTA</td>
<td>75.3</td>
<td>79.8</td>
</tr>
<tr>
<td>MWV</td>
<td>68.5</td>
<td>76.4</td>
</tr>
<tr>
<td>DAG</td>
<td>68.5</td>
<td>75.3</td>
</tr>
</tbody>
</table>

Table 8
Comparison with existing methods on OASIS dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D-DWT + WTA-SVM + LIN [10]</td>
<td>67.3</td>
</tr>
<tr>
<td>2D-DWT + WTA-SVM + POLY [10]</td>
<td>67.7</td>
</tr>
<tr>
<td>2D-DWT + WTA-SVM + RBF [10]</td>
<td>68.2</td>
</tr>
<tr>
<td>2D-DWT + PCA + KNN [11]</td>
<td>73.1</td>
</tr>
<tr>
<td>2D-DWT + PCA + WTA-KSVM + HPOLY [13]</td>
<td>73.8</td>
</tr>
<tr>
<td>2D-DWT + PCA + WTA-KSVM + IPOLY [13]</td>
<td>73.9</td>
</tr>
<tr>
<td>2D-DWT + PCA + WTA-KSVM + GRB [13]</td>
<td>73.9</td>
</tr>
<tr>
<td>2D-DWT + Entropy + WebPlot + PNN [17]</td>
<td>73.5</td>
</tr>
<tr>
<td>PCA + SVM [23]</td>
<td>74.0</td>
</tr>
<tr>
<td>3D-DWT + PCA + WTA-KSVM + PSOTVAC (proposed)</td>
<td><strong>81.5</strong></td>
</tr>
</tbody>
</table>

sub-sampling (splits the whole dataset into training and validation set randomly) is that all observations are used for both training and validation, and each observation is used for validation exactly once. Therefore, the CM by K-fold CV is an out-of-sample estimate.

7.5. Classification comparison

For kth run, we took kth fold as the test set, and other four folds as the training set. The CM of each run was recorded. The ith row and jth column in the confusion matrix denoted the count of observations known in class i and predicted to class j. The three classes in order were NC, MCI, and AD.

We designed three classifiers: First was linear SVM, i.e., linear kernel; the second as KSVM, in which σ of RBF kernel was assigned with a fixed value of 1 for all 3 individual KSVMs; the third was WTA-KSVM in which σ was optimized by the PSOTVAC method. Meanwhile, three different subclassifiers (WTA, MWV, and DAG) were used for each classifier. Hence, we tested nine classifiers in total.

The results are shown in Table 7. For linear SVM, the classification accuracy (CA) of WTA is 75.3%, higher than MWV of 68.5% and DAG of 68.5%. For KSVM, WTA method obtains CA of 79.8%, with MWV of 76.4% and DAG of 75.3%. For the KSVMS + PSOTVAC, WTA method obtains the highest CA of 81.5%, compared to MWV of 80.9% and DAG of 80.3%.

Table 9
K-fold result of WTA-KSVM + PSOTVAC.

<table>
<thead>
<tr>
<th>Fold 1</th>
<th>Fold 2</th>
<th>Fold 3</th>
<th>Fold 4</th>
<th>Fold 5</th>
<th>Whole Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1</td>
<td>0</td>
<td>18</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 10
Evaluation of class-i versus out-of-class-i based on WTA-KSVM + PSOTVAC.

<table>
<thead>
<tr>
<th>Class</th>
<th>TP</th>
<th>FP</th>
<th>TN</th>
<th>FN</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC</td>
<td>91</td>
<td>13</td>
<td>68</td>
<td>6</td>
<td>0.9381</td>
<td>0.8395</td>
<td>0.8933</td>
</tr>
<tr>
<td>MCI</td>
<td>37</td>
<td>8</td>
<td>113</td>
<td>20</td>
<td>0.6491</td>
<td>0.9339</td>
<td>0.8427</td>
</tr>
<tr>
<td>AD</td>
<td>17</td>
<td>12</td>
<td>142</td>
<td>7</td>
<td>0.7083</td>
<td>0.9221</td>
<td>0.8933</td>
</tr>
</tbody>
</table>

Table 7 indicates that the WTA method is in general superior to MWV and DAG methods in all three classifiers in this study. The reason may fall within the fact that individual classifiers of WTA record the scores, and the whole multiclass classifier selects the highest score from C individual classifier, and outputs its index as the predicted class, as formulated in (18). Meanwhile, the individual classifiers of other two multiclass methods only record the binary output other than the score. Therefore, individual classifiers of WTA acquire more precise results than the MWV and DAG, which may lead to the higher CA of WTA. Table 7 suggests that WTA-KSVM + PSOTVAC obtains the highest overall CA of 81.5% among all nine classification-methods.

7.6. Comparison to existing methods

We compared the proposed method “WTA-KSVM + PSOTVAC” with existing state-of-the-art methods. Some methods were proposed for binary classification (AD and NC), hence, we used their method and redo the experiments on the same dataset, by either applying WTA technique to SVM or increasing output neuron number to 3 for ANN, with the aim of fair comparison. Some methods were proposed for 2D image, so we extracted key-slices from volumetric image, and used their method to the 2D key-slices. We tested methods over OASIS dataset for fair comparison. The comparison results are shown in Table 8.

The results in Table 8 show that the proposed “3D-DWT + PCA + WTA-KSVM + PSOTVAC” performs better than other state-of-the-art methods in classification accuracy. The overall CA of other existing methods are all less than or equal to 74.0%, while our method achieves 81.5%. Hence, the improvement of our method is significant due to the power of 3D-DWT.

7.7. In-depth analysis of WTA-KSVM + PSOTVAC

In past sections, we proved the “WTA-KSVM + PSOTVAC” performed best among nine proposed classifiers and better than existing methods. In what follows, we continued investigating this method. The 5-fold CV classification result is shown in Table 9. The evaluations of classification performances for the individual class (NC, MCI, and AD) are listed in Table 10, here TP denotes true positive, FP false positive, TN true negative, FN false negative.

Results in Table 10 give the classification evaluation of individual class, viz., class-i versus out-of-class-i. The sensitivities of NC, MCI, and AD are 93.81%, 64.91%, and 70.83%, respectively. The specificities of NC, MCI, and AD are 83.95%, 93.39% and 92.21%, respectively. The accuracies of NC, MCI, and AD are 89.33%, 84.27%, and 89.33%, respectively. Hence, the proposed method achieves an
Table 11  
Comparison of global optimization algorithms for finding optimal $\sigma$ of each individual classifier.

<table>
<thead>
<tr>
<th>WTA-KSVM</th>
<th>NC-v-all</th>
<th>MCI-v-all</th>
<th>AD-v-all</th>
<th>Success cases</th>
<th>CA (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS</td>
<td>2.65</td>
<td>2.95</td>
<td>2.83</td>
<td>115</td>
<td>64.6%</td>
</tr>
<tr>
<td>GA</td>
<td>2.70</td>
<td>2.76</td>
<td>2.0</td>
<td>139</td>
<td>78.1%</td>
</tr>
<tr>
<td>SA</td>
<td>2.64</td>
<td>2.69</td>
<td>3.12</td>
<td>136</td>
<td>76.4%</td>
</tr>
<tr>
<td>PSO</td>
<td>2.66</td>
<td>2.71</td>
<td>2.98</td>
<td>140</td>
<td>78.7%</td>
</tr>
<tr>
<td>PSOTVAC</td>
<td>2.68</td>
<td>2.65</td>
<td>2.91</td>
<td>145</td>
<td>81.5%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MWV-KSVM</th>
<th>NC-v-MCI</th>
<th>NC-v-AD</th>
<th>MCI-v-AD</th>
<th>Success cases</th>
<th>CA (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS</td>
<td>2.09</td>
<td>2.56</td>
<td>2.14</td>
<td>112</td>
<td>62.9%</td>
</tr>
<tr>
<td>GA</td>
<td>2.48</td>
<td>2.63</td>
<td>2.80</td>
<td>141</td>
<td>79.2%</td>
</tr>
<tr>
<td>SA</td>
<td>2.34</td>
<td>2.52</td>
<td>2.68</td>
<td>138</td>
<td>77.5%</td>
</tr>
<tr>
<td>PSO</td>
<td>2.51</td>
<td>2.39</td>
<td>2.69</td>
<td>142</td>
<td>79.8%</td>
</tr>
<tr>
<td>PSOTVAC</td>
<td>2.55</td>
<td>2.58</td>
<td>2.71</td>
<td>144</td>
<td>80.9%</td>
</tr>
</tbody>
</table>

excellent classification performance, but it is still too early to apply to practical use.

A crucial fact that appeared is the difficulty to detect MCI accurately. For instance, the sensitivity of MCI is merely 0.6491, less than those of NC and AD. Low sensitivity of MCI may be explained in the way that MCI can be regarded as a transitional stage between NC and AD, so it is easy to be misclassified to any of the other two classes.

7.8. Performance of PSOTVAC

The parameters of PSOTVAC were set as $c_1=2.5$, $c_2=0.5$, $c_3=0.5$, $c_4=2.5$. We analyzed two instances of WTA-KSVM and MWV-KSVM. DAG was not included since it used the same one-versus-one model as MWV. We compared PSOTVAC method with Random Search (RS), Genetic Algorithm (GA), Simulated Annealing (SA), and PSO method. The searching space of $\sigma$ was restricted within the range of [0.5, 5]. The comparison of best results is listed in Table 11.

The algorithm comparison in Table 11 indicates that the parameter $\sigma$ of individual K SVM of one-versus-all model falls mostly within the range [2.6, 3], and those of one-versus-one model falls chiefly within the range [2.3, 2.7]. It may give us a hint that the one-versus-all model yields a higher parameter than the one-versus-one model. From the last column of the table, we can see that the PSOTVAC performs the best among all algorithms, the next is PSO followed by GA, and the worst two algorithms are SA and RS. The reason may lie in that the annealing schedule of SA is difficult to set, to balance the exploration and exploitation. RS was proved to inevitably yield a point arbitrarily close to global optimal within a potential infinite steps, but it is not feasible in practice since a finite number of iterations can only be executed. GA and PSO are mature global optimizations that also obtained good results in this study. The PSOTVAC improves PSO by using TVAC, which yields the highest CA.

The shortcomings of PSOTVAC due to the random initialization cause the results are not always globally optimal. Parameter setting is another factor that limits the performance of PSOTVAC. The setting ($c_1=2.5$, $c_2=0.5$, $c_3=0.5$, $c_4=2.5$) was obtained by experience. In the future, we will analyze their effect to the classification performance.

8. Discussions

The scientific and technical contributions of the paper include the following points: (1) We took the whole brain (not the ROIs) as the research objective, so there was no need for brain segmentation and interpretation. (2) We used 3D-DWT to capture the 3D texture feature of brain, and proved it was effective. (3) We used ALS-PCA for feature reduction of dataset containing missing attributes. (4) We used PSOTVAC to get the optimal kernel parameter of each individual classifier (SVM/K SVM). (5) We used $K$-fold CV to obtain out-of-sample error estimate. (6) We found that "WTA-KSVM + PSOTVAC" performs best among nine proposed classification methods, and it even performed better than state-of-the-art methods.

For the weight/biases training of K SVM, the SMO runs much faster than QP method. QP tends to create a large matrix to find the separating planes; meanwhile, SMO can control the memory consumption by storing only a sub-matrix of the kernel matrix. However, SMO moves slowly if the number of samples is sizeable, since it has to re-calculate the kernel matrix elements. In this study, the number of the sample is suitable, which is the reason we chose SMO as the training algorithm.

Cost matrix is an efficient way to adjust distribution of errors. In this study, it is implied that we use the default cost matrix (i.e., the cost is zero if we successfully predict the sample with true class of $i$ to class $i$, and the costs are all equal to 1 if we fail). In some conditions, it is expected that predicting MCI/AD to NC is extremely deleterious which will delay the deserved treatment, and predicting NC to MCI/AD is not so harmful, since it can be corrected and rechecked by other measures. Therefore, we will try to tune the cost matrix in the future.

Reconsidering the whole classifier, the first limitation is that it establishes machine-oriented rules not human-oriented rules. Although machine-oriented rules yield higher classification performance than human-oriented rules, technicians cannot understand or interpret what the weights/biases of the classifier. Another limitation is how to increase the classification accuracy. Current studies indicate AD is associated with metabolites in the gray matter; however, what we used in this research is structural MRI, which does not cover the sufficient information containing the cause of AD. Adding other imaging techniques, such as magnetic resonance spectroscopy imaging (MRSI) to measure the metabolites in the brain, may increase the classification accuracy of the classifier.

9. Conclusions and future research

In this paper, we proposed three individual classifiers and three different multiclass methods, with the aim of distinguishing NC, MCI, and AD, based on additional data and 3D-DWT of structural volumetric MR images. We found the best classifier was the "WTA-KSVM + PSOTVAC". The result was relatively outstanding considering it was a three-classification problem not binary classification.

The future tentative work will focus on the following aspects: (1) Include other imaging techniques, such as MRSI. (2) To extract more efficient 3D features from volumetric data. (3) Try other advanced classifiers such as ANN [46], Bayesian classifier, hidden Markov models, and deep learning. (4) Check the effect produced by other wavelet family on classification performance. (5) Include the phase image of imaging data. (6) Tune the classifier structure so their weights/biases will have physical meanings that are easier for technicians to understand. (7) Adjust cost matrix to avoid predicting MCI/AD to NC.

Acknowledgement

This paper was supported by NSFC (61001102, 61273243, 51407095), Program of Natural Science Research of Jiangsu Higher Education Institutions (13KJB460011, 14KJB520021), Jiangsu Key Laboratory of 3D Printing Equipment and Manufacturing (BM2013006), Key Supporting Science and Technology Program (Industry) of Jiangsu Province (BE2012201, BE2013012-2,
Appendix A.

Table A1

<table>
<thead>
<tr>
<th>Code</th>
<th>Education codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Less than high school graduate</td>
</tr>
<tr>
<td>2</td>
<td>High school graduate</td>
</tr>
<tr>
<td>3</td>
<td>Some college</td>
</tr>
<tr>
<td>4</td>
<td>College graduate</td>
</tr>
<tr>
<td>5</td>
<td>Beyond college</td>
</tr>
</tbody>
</table>

Table A2

A typical MMSE test.

<table>
<thead>
<tr>
<th>Category</th>
<th>Points</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation to time</td>
<td>5</td>
<td>From broadest to most narrow</td>
</tr>
<tr>
<td>Orientation to place</td>
<td>5</td>
<td>From broadest to most narrow</td>
</tr>
<tr>
<td>Registration</td>
<td>3</td>
<td>Repeating named prompts</td>
</tr>
<tr>
<td>Attention and calculation</td>
<td>5</td>
<td>Serial Sevens, or spelling “world” backwords</td>
</tr>
<tr>
<td>Recall</td>
<td>3</td>
<td>Registration Recall</td>
</tr>
<tr>
<td>Language</td>
<td>2</td>
<td>Naming a pencil and a watch</td>
</tr>
<tr>
<td>Repetition</td>
<td>1</td>
<td>Speaking back a phrase</td>
</tr>
<tr>
<td>Complex commands</td>
<td>6</td>
<td>Drawing figures shown, etc.</td>
</tr>
</tbody>
</table>

References


