MJaya-ELM: A Jaya Algorithm with Mutation and Extreme Learning Machine based Approach for Sensorineural Hearing Loss Detection

Deepak Ranjan Nayak[a,*], Yudong Zhang[b,*], Dibya Sundar Das[a], Subinita Panda[a]

[a]*Pattern Recognition Lab, Department of Computer Science and Engineering, National Institute of Technology, Rourkela, India 769 008*

[b]*Department of Informatics, University of Leicester, Leicester, UK LE1 7RH*

Abstract

Sensorineural hearing loss (SNHL) is a common hearing disorder or deafness which accounts for about 90% of the reported hearing loss. Magnetic resonance imaging (MRI) has been found to be an effective neuroimaging technique for detecting SNHL. However, manual detection methods, mainly based on the visual inspection of MRI, are cumbersome, time-consuming and need skilled supervision. Hence, there is a great need for the development of a computer-aided detection system for fast, accurate and automated detection of SNHL. This paper presents a new method for automated diagnosis of SNHL through brain MR images. Fast discrete curvelet transform is employed for image decomposition. The features are extracted from various decomposed subbands at different scales and orientations. A set of discriminant features is then derived using PCA+LDA algorithm. A hybrid classifier is suggested using extreme learning machine and Jaya optimization with mutation (MJaya-ELM) to distinguish hearing loss images from healthy MR images which overcomes the drawbacks of traditional ELM and other learning algorithms for single layer feedforward neural network. The concept of mutation has been introduced to conventional Jaya optimization (MJaya) for improving the global search ability of the solutions by providing additional diversity. The proposed system has been evaluated on a well-studied database. The comparison results demonstrate that the proposed scheme outperforms the existing schemes in terms of overall accuracy and sensitivity over different classes. The effectiveness of the proposed MJaya-ELM algorithm is also compared with its counterparts such as PSO-ELM, DE-ELM, and Jaya-ELM, and the results indicate the superiority of MJaya-ELM.

Keywords: Sensorineural hearing loss (SNHL), Fast discrete curvelet transform (FDCT),

*Corresponding author

Email addresses: depakranjannayak@gmail.com (Deepak Ranjan Nayak), yudongzhang@ieee.org (Yudong Zhang), dibyasundarit@gmail.com (Dibya Sundar Das), subinita.panda@gmail.com (Subinita Panda)

1. Introduction

Sensorineural hearing loss (SNHL) is a common hearing disorder, the root of which lies in vestibulocochlear nerve, inner ear, and central auditory processing system [1]. The hearing disorder can be either due to heredity or acquired and is caused because of several external factors like trauma, noise, infection, etc. Generally, SNHL can be of three types such as neural hearing loss, sensory hearing loss and both [2, 3]. The diagnosis methods for SNHL mostly involve conventional diagnostic techniques such as Weber test and Rinne test [4]. Research on SNHL [5, 6] show that it degenerates the brain tissues and this atrophy can easily be detected from the brain magnetic resonance (MR) images. However, the manual detection of this disease with the assistance of MRI scanning is unpredictable, cumbersome and time-consuming.

In past few years, significant efforts have been made to detect SNHL using computer vision and AI techniques. For instance, Ikawa [7] has used Auditory brainstem response (ABR) indexing for the diagnosis of hearing problems and brain functioning. He has employed 1D discrete wavelet multiresolution analysis to reconstruct a waveform from a distorted waveform with low PSNR value. Monzack *et al.* [8] have utilized the live confocal imaging techniques to trace the relationship between hair cell phagocytosis and sensorineural hearing loss. Nayak *et al.* [9] have proposed a new method which combines 2D discrete wavelet transform (DWT), probabilistic principal component analysis (PCA) and AdaBoost with random forests for pathological brain detection. In [10], the authors have suggested an approach for pathological brain detection which employs DWT, Kernel PCA and least squared support vector machine (LS-SVM) for feature extraction, feature reduction, and classification respectively. Wright *et al.* [11] have suggested an imaging technique on mouse cochlea which utilizes the confocal imaging methods followed by 3D analysis using the Imaris software. Wang *et al.* [12] have proposed an approach for SNHL detection which exploits fractional Fourier transform (FRFT) for extraction of salient features. The Levenberg-Marquardt (LM) algorithm is applied for training the single hidden layer feedforward network (SLFN) classifier. Vasta *et al.* [13] have employed Markov random fields (MRF) and Naïve Bayesian Classifier (NBC). Later, Zhang *et al.* [14] have used stationary wavelet entropy (SWE) and SLFN for unilateral hearing loss detection. Chen *et al.* [15] have recently developed an approach utilizing DWT features and generalized eigenvalue proximal SVM (GEPSVM) classifier. In another contribution, Wang *et al.* [16] have combined wavelet entropy (WE) and directed acyclic graph based SVM (DAG-SVM)

for identification of unilateral hearing loss. Liu *et al.* [17] have introduced an automated model based on wavelet packet decomposition (WPD) and LS-SVM for detection of left and right sensorineural hearing loss.

The key observations from the aforementioned literature are summarized as follows. DWT has become a significant choice for feature extraction in most of the earlier approaches although it fails to capture directional features. DWT can capture features in limited directions which might not sufficient to extract accurate edge information from MR images where edges are often curved. Further, earlier studies [12, 14, 16, 17] mostly include feedforward neural network (FNN) and SVM as classifiers. FNN, in general, uses gradient-based approaches such as backpropagation and LM for training the parameters which pose several critical issues like slower convergence, getting trapped at local minima, and overfitting. SVM overcomes these problems of gradient descent learning, however, suffers from high computational complexity.

In this paper, we have made an effort to develop an approach which aims at mitigating the above issues. The main contributions of our work are outlined as follows.

(a) We exploit unequally spaced fast Fourier transform based fast discrete curvelet transform (FDCT-USFFT) to extract multi-directional features from the MR images.

(b) We employ PCA+LDA method to derive a reduced and more discriminant feature set.

(c) We introduce the concept of mutation to traditional Jaya optimization (MJaya) to alleviate a large number of generations and search delay as well as improve the global search capability of the solutions.

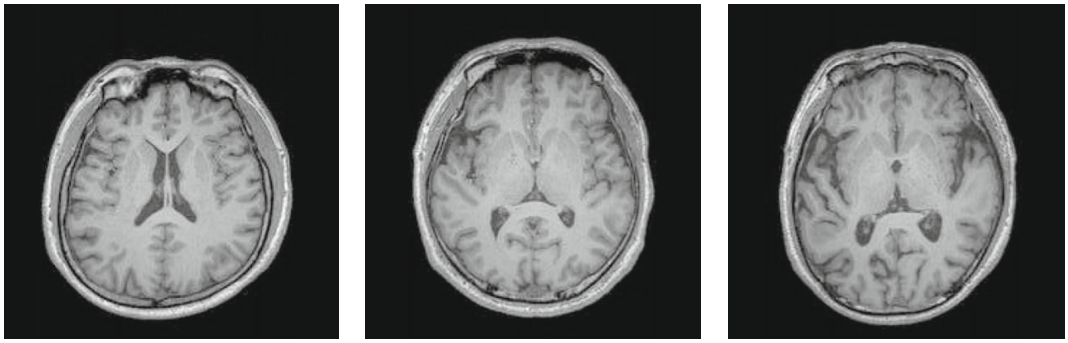
(d) To overcome the issues caused by conventional classifiers, we use extreme learning machine (ELM) in the current study which provides good generalization performance at faster learning speed. Besides, we develop a novel learning algorithm for SLFN by hybridizing ELM with MJaya (MJaya-ELM) which primarily focuses on improving the generalization performance of standard ELM.

The remainder of this paper is structured as follows. We provide the descriptions of the materials used for simulation in Section 2. The proposed methodology is detailed in Section 3. The experimental results and comparisons are presented in Section 4. Eventually, we conclude this paper in Section 5. For the ease of reading, the acronyms used in this paper are defined in Table A.1 (please refer to the appendix).

2. Materials

The study includes 49 subjects among which 14 cases belong to right-sided hearing loss (RHL), 20 cases belong to healthy controls (HC), and 15 cases belong to left-sided hearing loss (LHL). The sample MRI scans from three different categories of hearing loss are shown in Figure 1. The considered dataset is moderate sized and the results obtaining from it is convincing. This study was approved by the Ethics Committee of Southeast University, and a signed informed consent form was obtained from every subject prior to entering this study.

We utilized a clinical audiometer to perform pure tone audiometry with six different octave frequencies (0.25, 0.5, 1, 2, 4 and 8 kHz) in order to measure the pure tone average (PTA) and reflect hearing performance. All patients were diagnosed with unilateral hearing loss (UHL) with hearing deficit in either unilateral ear (PTA $\geq$ 40 dB) and normal hearing in both ears (PTA $\leq$ 25 dB). The patients included were all right-handed and their ages are between 41 and 60. For each patient, the hearing loss was sudden and persistent. None of them used a hearing aid on the impaired ear. Table 1 shows the parametric details of the dataset.



(a) Healthy controls (b) Left-sided hearing loss (c) Right-sided hearing loss

Figure 1: Different categories of hearing loss MR images

Table 1: Subject characteristics

Parameter	HC	RHL	LHL
Age (year)	53.6 ± 5.4	53.9 ± 7.6	51.7 ± 9.6
Duration of disease	-	14.2 ± 14.9	17.6 ±17.3
Gender (f\m)	12 \ 8	8 \ 6	7\ 8
Level of education	11.5±3.2	12.1±2.4	12.5±1.7
PTA of right ear (dB)	21.3±2.2	80.9±17.4	20.4±4.2
PTA of left ear (dB)	22.2± 2.1	21.8±3.2	78.1±17.9

3. Proposed methodology

In this section, we discuss the proposed methodology for automated detection of SNHL. As shown in Figure 2, there are four major steps involved in detection process: (i) preprocess the MR hearing loss images based on brain extraction tool (BET) (ii) extract the salient features based on FDCT-USFFT, (iii) derive the reduced and significant features based on PCA+LDA method, and (iv) classify the images based on a novel MJaya-ELM algorithm. Each step is described in detail in the following.

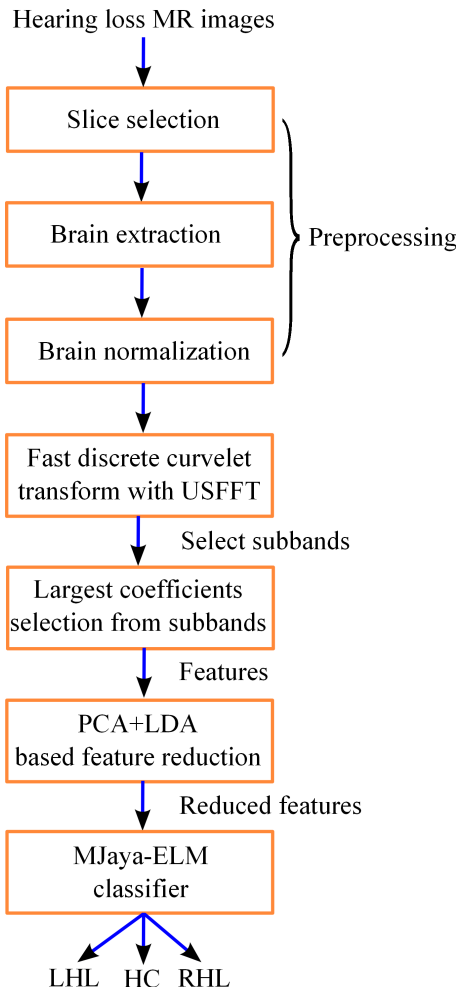


Figure 2: Detailed block diagram of the proposed approach

3.1. Preprocessing

For preprocessing, we have used the FMRIBS Software Library (FSL) v5.0. The skull is separated from the brain tissues using the brain extraction tool (BET). Further, with the help of FLIRT

and FNIRT tools, the preserved brains were normalized to a standard Montreal Neurological Institute (MNI) template. Gaussian kernel is then used to perform smoothing of the normalized brain images. The most distinctive slice (around 40^{th} slice) has been utilized to conduct the experiments as per the instruction of experienced neurologists. The results obtained from BET in three different planes are illustrated in Figure 3. It is worth mentioning here that the axial MR images (shown in middle of the figure) are taken into consideration for feature extraction.

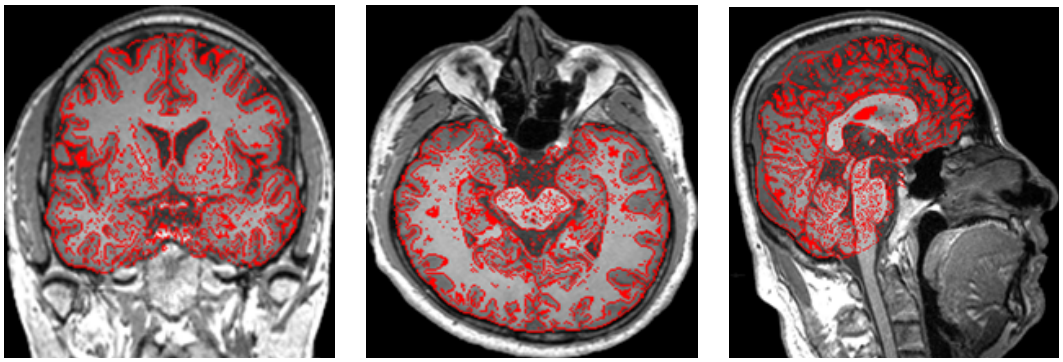


Figure 3: BET results

3.2. Feature extraction

Due to the notable characteristics like time-frequency localization and multiresolution, the wavelet transform has been extensively applied for feature extraction. It shows better performance in capturing 1D singularities; however, it fails to capture 2D singularities (line, curves, etc.) from the images. The line singularity issue is resolved by ridgelet transform (RT), but it could not effectively deal with the curve singularities. In contrary, the first-generation curvelet transform efficiently handles 2D singularities. The salient characteristics of curvelet transform as a feature extraction tool are– multiresolution, high directional selectivity, anisotropy and localization, and therefore it has gained considerable attention from researchers in recent years [18, 19]. More recently, second-generation curvelet transform is introduced to overcome the problems faced by the standard curvelet transform such as the unclear geometry of ridgelets and more time-consuming [20, 21].

Let g be a 2D signal, then the curvelet transform can be stated via inner product as

$$\mathbb{C}(\alpha, \beta, \gamma) = \langle g, \phi_{\alpha,\beta,\gamma} \rangle \tag{1}$$

Here, $\phi_{\alpha,\beta,\gamma}$ indicates the curvelet basis function, α, γ, and β denote the scale, position, and direction (orientation) parameter respectively. The curvelet transform decomposes the image into

a number of windows at various scales and orientations. The discrete form of curvelet transform for an Cartesian array $g[x_1, y_1]$ with $0 \leq x_1, y_1 < n$ is defined as

$$\mathbb{C}^D(\alpha, \beta, \gamma) = \sum_{0 \leq x_1, y_1 < n} g[x_1, y_1] \overline{\phi^D_{\alpha,\beta,\gamma}[x_1, y_1]} \tag{2}$$

where, $\phi^D_{\alpha,\beta,\gamma}$ indicates a digital curvelet waveform. The proposed approach makes use of second-generation curvelet transform, also called as fast discrete curvelet transform (FDCT), for deriving features from MR images. There exist two procedures to implement FDCT such as wrapping based FDCT (FDCT-WR) and unequally spaced fast Fourier transform based FDCT (FDCT-USFFT). In contrast to first generation curvelets, these two procedures are fast, simple, and less redundant. In this study, we have considered FDCT-USFFT as the feature extractor since it provides proper discretization of the continuous definition. In addition, compared to other relevant image transforms, FDCT-USFFT has the capability to capture multi-directional features which are essential for MR image classification.

Feature vector generation. To generate the feature vector, at first we have decomposed the MR images using FDCT-USFFT. Then, the coefficients of different subbands at each scale α and orientation β are collected. The number of scales ($\mathbf{s}$) for an image with size $n_r \times n_c$ is decided as [20]

$$\mathbf{s} = \lceil \log_2(min(n_r, n_c)) - 3 \rceil. \tag{3}$$

For example, $\mathbf{s}$ is computed as 5 for an MR image with resolution 256×256. Another parameter, the number of orientations (or angles) ($\mathbf{L}$) at coarsest level is set as 16. With these settings, FDCT-USFFT generates 32, 32 and 64 angles at scale 2, 3 and 4 respectively. It is well-known that curvelet at angle β and $\beta + \pi$ generates similar information. Therefore, we have discarded the coefficients of the symmetric subbands at scale 2, 3 and 4 which on the other hand helps in removing the redundant features from the original feature set. The final feature vector is then obtained by collecting a set of largest coefficients from the chosen subbands. The steps involved in the feature extraction method are described in Algorithm 1.

3.3. Feature reduction

It has been observed that the dimension of the resultant feature vector is relatively large than the number of samples that we consider in this study. Feature reduction methods help in reducing computational burden, visualizing data and improving the classification performance. In the past

Algorithm 1 Feature extraction using FDCT-USFFT method

Input: Input image: $g[x_1, y_1];\ 0 \leq x_1, y_1 < n$, $\mathbf{L}$, $\mathbf{s}$

Output: Feature vector F_v

1: Apply 2D FFT on input $g[x_1, y_1]$ and generate its Fourier coefficients $\hat{g}[n_1, n_2]$ as

$$\hat{g}[n_1, n_2] = \sum_{x_1, y_1 = 0}^{n-1} g[x_1, y_1] e^{-i2\pi(n_1 x_1 + n_2 y_1)/n};\ -n/2 \leq n_1, n_2 < n/2$$

2: For each scale and angle pair (α, β), interpolate $\hat{g}[n_1, n_2]$ to generate sample values $\hat{g}[n_1, n_2 - n_1 \tan\theta_\beta]$

3: Perform multiplication of the interpolated object $\hat{g}$ with the parabolic window $\tilde{U}_\alpha$

$$\tilde{g}_{\alpha,\beta}[n_1, n_2] = \hat{g}[n_1, n_2 - n_1 \tan\theta_\beta]\tilde{U}_\alpha, \hat{g}[n_1, n_2]$$

4: Obtain the discrete curvelet coefficients $\mathbb{C}^D(\alpha, \beta, \gamma)$ by applying inverse 2D FFT to each $\tilde{g}_{\alpha,\beta}$

5: Discard the symmetric subbands from each scale except first and last scale

6: Collect v largest coefficients from each of the chosen subband and construct a feature vector F_v

decades, PCA and linear discriminant analysis (LDA) have attracted tremendous attention from researchers for feature dimensionality reduction. While applying LDA on a high dimensional and small sample size problem, the within-scatter matrix (S_w) becomes always singular [22, 23] and in this case, at least $D + C$ (where D indicates the dimension of the feature vector and C indicates the number of classes) number of samples is required to confirm that the S_w is not singular which is not achievable practically in general. To address this issue, we have employed an approach called PCA+LDA [23], where we at first employ PCA to reduce a D-dimensional data to an M-

Algorithm 2 Feature reduction using PCA+LDA

Require: Input feature matrix: $\mathbf{F}$ of size $N \times D$ (N: number of training samples)

Ensure: Reduced feature matrix: $\hat{\mathbf{F}}$ of size $N \times l$

▷ Functions $pca(.)$ and $lda(.)$ denote PCA and LDA methods respectively

1: Choose a dimension M ▷ PCA method

2: $\mathbf{F}'(N \times M) \leftarrow pca(\mathbf{F}, M)$

3: Select a dimension l ▷ LDA method

4: $\hat{\mathbf{F}}(N \times l) \leftarrow lda(\mathbf{F}', l)$

5: Output the reduced matrix $\hat{\mathbf{F}}$

dimensional data and then LDA is applied to reduce to a l-dimensional data, $l << M < D$. The overall steps of PCA+LDA method are listed in Algorithm 2.

3.4. Classification using MJaya-ELM

To overcome the limitations of standard gradient-based learning schemes, Huang et al. [24] have proposed a simple non-iterative learning paradigm (called as ELM) for single-hidden layer feedforward neural networks (SLFNs). Because of its faster learning ability, ELM has been applied in a wide range of applications [25]. ELM tends to reach the smallest training error as well as the smallest norm of the output weights. It is well-known from Bartlett's theory [26] that for feedforward neural networks arriving at smaller training error, the smeller the norm of the output weight is, the better generalization performance of the networks tend to acquire. ELM needs only one hyperparameter (i.e., output weights) to determine mathematically as the hidden biases and input weights are generated randomly. It is noted here that the Moore-Penrose (MP) generalized inverse operation is usually used to determine the output weights.

Let N be the number of training samples (x_i, y_i), where $x_i = [x_{i1}, x_{i2}, \ldots, x_{il}]^T \in R^l$ and $y_j = [y_{i1}, y_{i2}, \ldots, y_{iC}]^T \in R^C$, h_n be the number of hidden neurons and $\phi(.)$ be a limiting function., then the basic ELM algorithm is enumerated in three-steps as follows.

1. Randomly initialize the input weights and biases (h_z^w,b_z), $z = 1, 2, \ldots, h_n$
2. Calculate the hidden layer output matrix $\mathbf{H}$
3. Calculate the output weight matrix $o^w = \mathbf{H}^\dagger \mathbf{Y}$

where, $\mathbf{H}^\dagger$ denotes the MP generalized inverse of matrix $\mathbf{H}$. The size of $\mathbf{H}$, o^w and $\mathbf{Y}$ are $N \times h_n$, $h_n \times C$ and $N \times C$ respectively.

ELM poses two major issues because of the random generation of hidden biases and input weights [27]: (i) it needs more hidden neurons and (ii) it causes an ill-conditioned hidden layer output matrix $\mathbf{H}$. These issues cause ELM to respond slowly on unknown testing data and induce poor generalization performance. The conditioning of a matrix can be measured by a well-known metric called 2-norm condition number and is defined in Section 4. A few pieces of works have been proposed in the literature to address the issues of standard ELM [27, 28, 29, 30, 31, 32]. In these works, several meta-heuristic techniques such as genetic algorithms (GA), differential evolution (DE) and PSO, are hybridized with ELM to optimize its hidden node parameters, while the MP generalized inverse is employed to find the output weights. These meta-heuristic approaches involve hyperparameter tuning as their performances are strongly influenced by the hyperparameters. In

this paper, we have hybridized ELM with a recent meta-heuristic technique called Jaya which in general does not require any hyperparameter tuning in the optimization process.

Jaya is a simple optimization scheme proposed by Rao [33]. "Jaya" is a Sanskrit word which means "victory". Unlike other optimization schemes, Jaya is not dependent on any algorithm-specific parameters. The principle behind Jaya is that it always pulls the solution towards the best solution by avoiding the worst solution. Hence, the solution update equation requires both the best and worst solution and is defined as

$$P'_{j,d}(k) = P_{j,d}(k) + r_{1,d}(k)(P_{best,d}(k) - |P_{j,d}(k)|) - r_{2,d}(k)(P_{worst,d}(k) - |P_{j,d}(k)|) \tag{4}$$

where, $P_{j,d}(k)$ indicates the value of the d^{th} variable for the j^{th} solution during k^{th} iteration, $P_{best,d}(k)$ and $P_{worst,d}(k)$ denote the value of d^{th} variable for the *best* and *worst* candidate solution during k^{th} iteration, n indicates the number of candidate solutions (i.e., $j = 1, 2, \ldots, n$), m indicates the number of variables (i.e., $d = 1, 2, \ldots, m$), k indicates the iteration number (i.e., $k = 1, 2, \ldots, MAX_{itr}$), MAX_{itr} is the maximum number of iterations, and $r_{1,d}(k)$ and $r_{2,d}(k)$ are the two random numbers in the range $[0, 1]$ for d^{th} variable at iteration k. The second term in Eq. 4 pulls the solution towards the best solution and the third term pushes the solution away from the worst solution. For a minimization problem, the final decision is made using the following condition [34].

$$P_{j,d}(k+1) = \begin{cases} P'_{j,d}(k) & \text{if } f(P'_{j,d}(k)) < f(P_{j,d}(k)) \\ P_{j,d}(k) & \text{otherwise} \end{cases} \tag{5}$$

The prime objective of all candidate solutions of Jaya algorithm in each iteration is only to find better solution compared to the previous worst value and this behavior delineates that all candidates of Jaya algorithm are suffering from lack of strong determination and dedication to quickly track the global best solution [34]. In order to control this phenomenon, we have introduced a polynomial mutation operator to Jaya algorithm (MJaya). The mutation operator provides additional diversity and thereby, improving the search toward the global best solution. Let P be a solution and the mutated solution P' is given by

$$P' = \begin{cases} P + [2 * rand(.)]^{\frac{1}{\mu+1}} - 1 & \text{if } r \leq 0.5 \\ P + 1 - [2 * (1 - rand(.))]^{\frac{1}{\mu+1}} & \text{if } r > 0.5 \end{cases} \tag{6}$$

where, r is a random number in the range [0,1] and μ is the user-defined mutation variable. In MJaya, we select a random candidate solution and add some random perturbation to it by a

mutation probability. The overall steps involved in the proposed MJaya algorithm are outlined in Algorithm 3.

Algorithm 3 Proposed MJaya algorithm

Initialize a set of random candidate solutions (P_j)
Calculate the fitness of each solution
Find the best candidate solution (P_{best}) and worst candidate solution (P_{worst})
while ($t < MAX_{itr}$) **do**
    for each candidate solution **do**
        Modify the solution using Eq. (4)
        Update the solution if the modified solution is better than previous solution using Eq. (5)
    end for
    Compute the fitness of each updated solution
    Update P_{best} and P_{worst}
    Initialize r_{11}, r_{12}, m_p, and μ ▷ Mutation
    Select a random candidate solution (P_{rnd})
    if ($r_{11} < m_p$) **then**
        if ($r_{12} \leq 0.5$) **then**
            $P'_{rnd} = P_{rnd} + [2 * rand1(.)]^{\frac{1}{\mu+1}} - 1$
        else
            $P'_{rnd} = P_{rnd} + 1 - [2 * (1 - rand2(.))]^{\frac{1}{\mu+1}}$
        end if
        if ($f(P'_{rnd})$ is better than $f(P_{rnd})$) **then**
            $P_{rnd} = P'_{rnd}$
            if ($f(P'_{rnd})$ is better than $f(P_{best})$) **then**
                $P_{best} = P'_{rnd}$
            end if
        end if
    end if
end while
Return P_{best}

In the above algorithm, the parameters r_{11}, r_{12}, $rand1(.)$, $rand2(.)$ represent four different

random numbers in the range [0,1], rnd denotes a random number between 1 to n (where n is the number of candidate solutions), m_p indicates the mutation probability, and μ dictates the mutation variable.

MJaya-ELM is a hybridization of ELM and MJaya optimization in which MJaya is employed to determine the optimal values of the input weights and hidden biases. In this study, we have also verified the effectiveness of Jaya-ELM algorithm which is a hybridization of ELM and standard Jaya algorithm. MJaya searches the global optima with the support of both the norm of the output weights and the root-mean-squared error (RMSE) to improve the generalization performance. The main motive is to ameliorate the convergence performance by minimizing the norm of the output weights and adjusting the solutions in a specified range. Besides, MJaya algorithm attempt to establish a well-conditioned and compact SLFN. The proposed MJaya-ELM algorithm is detailed as follows.

(a) Randomly initialize candidate solutions in the population, each solution consists of a set of input weights and hidden biases in the range [-1,1] as

$$P_j = \left[h_{11}^w, h_{12}^w, \ldots, h_{1l}^w, h_{21}^w, h_{22}^w, \ldots, h_{2l}^w, h_{h_n1}^w, h_{h_n2}^w, \ldots, h_{h_nl}^w, b_1, b_2, \ldots, b_{h_n}\right] \tag{7}$$

(b) For each solution, compute o^w ($o^w = \mathbf{H}^\dagger\mathbf{Y}$) and fitness on the validation set as

$$f() = \sqrt{\frac{\sum_{i=1}^{N_v} || \sum_{z=1}^{h_n} o_z^w \phi(h_z^w \cdot x_i + b_z) - y_i||_2^2}{N_v}} \tag{8}$$

where, N_v indicates the number of validation samples.

(c) Find the best candidate solution P_{best} and worst candidate solution P_{worst}.

(d) Modify each solution using Eq.(4).

(e) Check whether the solutions go beyond the search space i.e., [-1, 1]

$$P'_{j,d}(k) = \begin{cases} -1 & \text{if } P'_{j,d}(k) < -1 \\ +1 & \text{if } P'_{j,d}(k) > +1 \end{cases} \tag{9}$$

(f) Calculate the fitness of modified solutions and check the following condition to find the solution for next generation

$$P_j(k+1) = \begin{cases} P'_j(k) & \left(|f(P_j(k)) - f(P'_j(k))| < \epsilon f(P_j(k)) \text{ and } ||o_{P'_j}^w|| < ||o_{P_j}^w||\right) \\ P_j(k) & \text{otherwise} \end{cases} \tag{10}$$

where, $f(P_j(k))$ and $f(P'_j(k))$ are the fitness of candidate solution j and its corresponding modified solution respectively, $||o^w_{P_j}||$ and $||o^w_{P'_j}||$ represent the norm of the output weights of solution j and its modified solution respectively, and $\epsilon > 0$ denotes a user-defined tolerance rate.

(g) Update P_{best} and P_{worst}.

(h) Initialize two random numbers r_{11} and r_{12}, mutation probability m_p and a user-defined parameter μ.

(i) Select a random solution from the updated candidate solutions and apply mutation to it (using steps 13 to 25 in Algorithm 2), and update the solution if there is better solution.

(j) Repeat (d)-(i) until the maximum number of iterations reached. Eventually, obtain the optimal input weights and hidden biases.

The proposed method tends to achieve a smaller norm of the output weight as we decide the optimal solution using Eq. (10). The lower the norm value of output weights is, the lower is its corresponding condition value of $\mathbf{H}$. Summarizing, the benefits of the proposed MJaya-ELM are: (i) it generates a well-condition matrix, (ii) it provides a lower norm value, and (iii) it offers greater generalization performance with a compact SLFN architecture.

4. Results and discussion

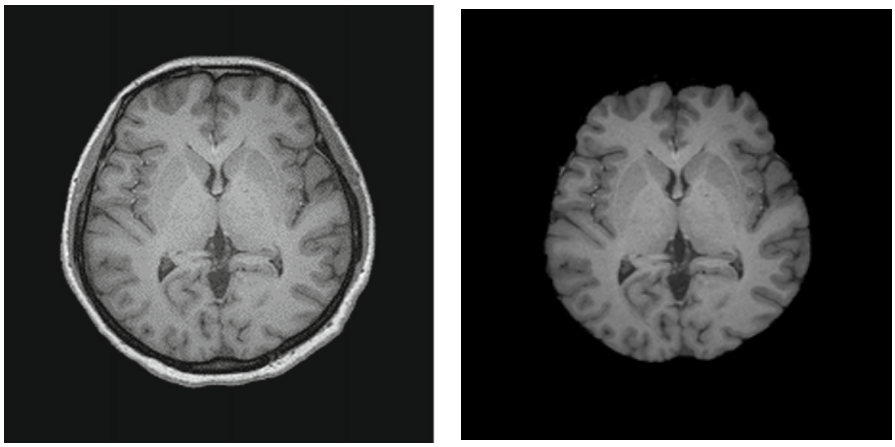
All the programs were developed using MATLAB 2017a and were run on a PC with 16 GB RAM, 3.5 GHz processor, and Windows 10 OS. In order to derive fair comparisons, we kept the statistical set up similar to the competent schemes. That is, we used 5-fold stratified cross-validation strategy in this study and repeated the procedure for 10 times [14, 15]. In each trial, out of 5 folds, three folds are used for training, one fold each for validation and testing.

To facilitate the performance comparison between our proposed approach and other competent approaches, several evaluation metrics such as sensitivity (S_e), specificity (S_p), precision (P_r) and overall accuracy (A_o), are taken into consideration. Moreover, we used two additional measures such as norm and 2-norm condition number to evaluate the proposed classifier and its counterparts. The 2-norm condition number of the matrix $\mathbf{H}$ is calculated as

$$\mathcal{K}_2(\mathbf{H}) = \sqrt{\frac{\lambda_{max}(\mathbf{H}^T\mathbf{H})}{\lambda_{min}(\mathbf{H}^T\mathbf{H})}} \tag{11}$$

where, $\lambda_{min}(\mathbf{H}^T\mathbf{H})$ and $\lambda_{max}(\mathbf{H}^T\mathbf{H})$ are the smallest and largest eigenvalues of matrix $\mathbf{H}^T\mathbf{H}$.

The result of preprocessing is illustrated in Figure 4. Figure 4(a) shows a sample LHL MR image and Figure 4(b) depicts its corresponding brain extracted image. It can be observed that the BET tool efficiently removes the skull and other unwanted areas from the MR image.



(a) Original scanning image (b) Brain extracted image

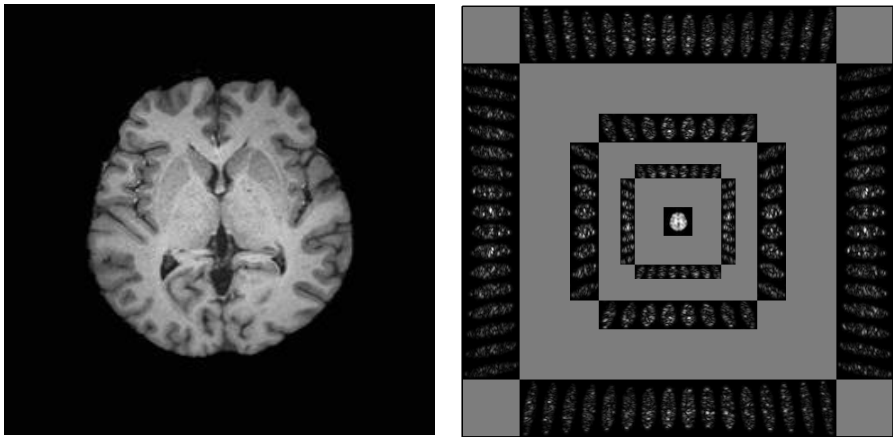
Figure 4: Sample LHL MR image and its preprocessing result obtained from BET

Next, we derived the features from the preprocessed images using FDCT-USFFT decomposition. We set the parameter **s** to 5 based on Eq. (3) and **L** to 16 since the results were found ideal with these values. Figure 5 depicts the FDCT-USFFT decomposed subbands at different scales and angles for an LHL MR image with **s** and **L** values 5 and 16 respectively. It is observed that a total of 130 subbands are obtained using FDCT-USFFT decomposition, out of which 66 subbands (excluding 64 symmetric subbands) are taken into account for feature extraction. To construct the feature vector, we collected v largest coefficients from each selected subband rather than all coefficients. The v value was determined experimentally as 20 and therefore, the dimension of the feature vector is 1320 which is relatively larger than the number of samples.

Thereafter, we employed PCA+LDA algorithm to reduce the dimensionality of the feature vector and obtain a more discriminative feature set. The value of M is set to $N-1$, where N is the number of training samples. The value of l is computed using a well-known measure called normalized cumulative sum variance (NCSV) [21] which is defined as

$$NCSV(a) = \frac{\sum_{j=1}^{a} \lambda(j)}{\sum_{j=1}^{M} \lambda(j)} \quad , \ 1 \leq a \leq M \tag{12}$$

where, $\lambda(j)$ represents the eigenvalue of the j^{th} feature and M denotes the total number of the



(a) Sample MR image (b) FDCT decomposed subbands

Figure 5: Illustration of a sample hearing loss image and its FDCT coefficients at different scales and orientations

eigenvectors (features) sorted in descending order. A threshold value was manually set and the number of features (for instance l) for which the NCSV value surpasses the threshold were selected. In this study, the threshold value was empirically set as 0.95 and the reduced number of features (l) was obtained as 2.

Eventually, the proposed MJaya-ELM algorithm is employed to perform classification. The performance of MJaya-ELM is compared with its counterparts such as PSO-ELM, DE-ELM, Jaya-ELM, and standard ELM. For fair comparison purpose, we used the same number of iterations (i.e., 30) and population size (i.e., 20) in all these methods. The inputs to all the methods were normalized into the range [-1,1] and the sigmoid function was used as the activation function. The value of ϵ in Eq. (10) was set to 0.02. The parameters m_p and μ were set to 0.5 and 20 respectively. The parameters of DE and PSO were set as follows. In PSO-ELM, c_1 and c_2 were set to 2, while in DE-ELM, the scaling factor (F) and crossover rate (C_r) were set as 0.8 and 0.7 respectively.

Table 2 shows the performance comparisons among all the classifiers in terms of classification accuracy, number of hidden neurons, norm of output weights, and condition number. It is observed that the proposed MJaya-ELM classifier achieves greater classification accuracy than its counterparts with only two neurons. The proposed method also obtains the smallest norm and condition number compared to others which ensures better generalization performance. It can also be noticed that the standard ELM yields the smallest accuracy and requires more hidden neurons in comparison to others. A competitive performance is noticed among PSO-ELM, DE-ELM, and Jaya-ELM methods, however, an improved performance is observed with MJaya-ELM when compared with

Table 2: Performance comparisons of the proposed classifier with its competent methods

Classifiers	A_o (%)	h_n	$\|\|o^w\|\|$	Condition number ($\mathcal{K}_2$)
ELM	96.94	4	56.0127	3.4217e+03
PSO-ELM	97.96	2	13.6560	14.9558
DE-ELM	98.78	2	9.1835	11.3077
Jaya-ELM	98.57	2	9.4641	11.9539
MJaya-ELM (Proposed)	**99.18**	**2**	**6.3225**	**7.7380**

Table 3: Fold-wise statistical results for MJaya-ELM method

Run	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5	Total	A_o(%)
R-1	3/4/3	3/4/2	3/4/3	3/4/3	3/4/3	15/20/14	100.00
	3/4/3	**3/4/2**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	10 (10)	9 (9)	10 (10)	10 (10)	10 (10)	49 (49)	
R-2	3/4/3	2/4/3	3/4/2	3/4/3	3/4/3	14/20/14	97.96
	3/4/3	**3/4/3**	**3/4/2**	**3/4/3**	**3/4/3**	**15/20/14**	
	10 (10)	9 (10)	9 (9)	10 (10)	10 (10)	48 (49)	
R-3	3/4/2	3/4/3	3/4/3	3/4/3	3/4/3	15/20/14	100.00
	3/4/2	**3/4/3**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	9 (9)	10 (10)	10 (10)	10 (10)	10 (10)	49 (49)	
R-4	3/4/3	3/4/3	3/4/1	3/4/3	3/4/3	15/20/13	97.96
	3/4/3	**3/4/3**	**3/4/2**	**3/4/3**	**3/4/3**	**15/20/14**	
	10 (10)	10 (10)	8 (9)	10 (10)	10 (10)	48 (49)	
R-5	3/4/2	3/4/3	3/4/3	3/4/3	3/4/3	15/20/14	100.00
	3/4/2	**3/4/3**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	9 (9)	10 (10)	10 (10)	10 (10)	10 (10)	49 (49)	
R-6	3/4/2	3/4/3	3/4/3	3/4/3	3/4/2	15/20/13	97.96
	3/4/2	**3/4/3**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	9 (9)	10 (10)	10 (10)	10 (10)	9 (10)	48 (49)	
R-7	3/4/2	3/4/3	3/4/3	3/4/3	3/4/3	15/20/14	100.00
	3/4/2	**3/4/3**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	9 (9)	10 (10)	10 (10)	10 (10)	10 (10)	49 (49)	
R-8	3/4/3	3/4/2	3/4/3	3/4/3	3/4/3	15/20/14	100.00
	3/4/3	**3/4/2**	**3/4/3**	**3/4/3**	**3/4/3**	**15/20/14**	
	10 (10)	9 (9)	10 (10)	10 (10)	10 (10)	49 (49)	
R-9	3/4/2	3/4/3	3/4/2	3/4/3	3/4/3	15/20/13	97.96
	3/4/3	**3/4/3**	**3/4/2**	**3/4/3**	**3/4/3**	**15/20/14**	
	9 (10)	10 (10)	9 (9)	10 (10)	10 (10)	48 (49)	
R-10	3/4/3	3/4/3	3/4/3	3/4/3	3/4/2	15/20/14	100.00
	3/4/3	**3/4/3**	**3/4/3**	**3/4/3**	**3/4/2**	**15/20/14**	
	10 (10)	10 (10)	10 (10)	10 (10)	9 (9)	49 (49)	
Total						149/200/137	99.18
						150/200/140	
						486 (490)	

Jaya-ELM. To draw fair comparisons, the number of hidden neurons were kept similar for PSO-ELM, DE-ELM and Jaya-ELM classifiers. The overall accuracy of the standard ELM with only two neurons was also evaluated and found significantly smaller as compared to others, i.e., 80.00%. It is evident from the results that the concept of mutation to conventional Jaya optimization ensures better performance.

The results of correctly predicted samples in each fold obtained by the proposed MJaya-ELM method are tabulated in Table 3. In the table, $a/b/c$ represents the correctly classified samples for class LHL, HC and RHL respectively, whereas **a/b/c** (in bold font) indicates the total number of samples in each fold. It is worth mentioning here that a total of 490 samples (150 LHL, 200 HC, and 140 RHL) are possible with 10 repetitions of 5-fold cross-validation strategy. The proposed system is able to correctly classify 486 samples (149 LHL, 200 HC, and 137 RHL) which accounts an overall accuracy of 99.18%.

The confusion matrices of the proposed MJaya-ELM, Jaya-ELM, DE-ELM, PSO-ELM and standard ELM methods are illustrated in Table 4, 6, 8, 10, and 12 respectively. The TP (true positive), TN (true negative), FP (false positive), FN (false negative) values, and the performance metrics computed using these four parameters for all the classifiers are listed in Table 5, 7, 9, 11, and 13. It is evident from the tables that the proposed MJaya-ELM obtains superior performance than its counterparts such as Jaya-ELM, DE-ELM, PSO-ELM and standard ELM. However, a competitive performance is observed with DE-ELM and Jaya-ELM.

Table 4: Confusion matrix of proposed MJaya-ELM method

		Predicted class		
		LHL	HC	RHL
Target class	LHL	149	1	0
	HC	0	200	0
	RHL	2	1	137

Table 5: Results of classification for MJaya-ELM method

Class	TP	TN	FP	FN	S_e (%)	S_p (%)	P_r (%)	A_{cc} (%)
LHL	149	338	2	1	99.33	99.41	98.68	99.39
HC	200	288	2	0	100.00	99.31	99.01	99.59
RHL	137	350	0	3	97.86	100.00	100.00	99.39

Table 6: Confusion matrix of Jaya-ELM method

		Predicted class		
		LHL	HC	RHL
Target class	LHL	150	0	0
	HC	3	196	1
	RHL	3	0	137

Table 7: Results of classification for Jaya-ELM method

Class	TP	TN	FP	FN	S_e (%)	S_p (%)	P_r (%)	A_{cc} (%)
LHL	150	334	6	0	100.00	98.24	96.15	98.78
HC	196	290	0	4	98.00	100.00	100.00	99.18
RHL	137	349	1	3	97.86	99.71	99.28	99.18

Table 8: Confusion matrix of DE-ELM method

		Predicted class		
		LHL	HC	RHL
Target class	LHL	149	0	1
	HC	0	200	0
	RHL	3	2	135

Table 9: Results of classification for DE-ELM method

Class	TP	TN	FP	FN	S_e (%)	S_p (%)	P_r (%)	A_{cc} (%)
LHL	149	337	3	1	99.33	99.11	98.03	99.18
HC	200	288	2	0	100.00	99.31	99.01	99.59
RHL	135	349	1	5	96.43	99.71	99.26	98.78

Table 10: Confusion matrix of PSO-ELM method

		Predicted class		
		LHL	HC	RHL
Target class	LHL	143	4	3
	HC	0	198	2
	RHL	0	1	139

Table 11: Results of classification for PSO-ELM method

Class	TP	TN	FP	FN	S_e (%)	S_p (%)	P_r (%)	A_{cc} (%)
LHL	143	340	0	7	95.33	100.00	100.00	98.57
HC	198	285	5	2	99.00	98.28	97.54	98.57
RHL	139	345	5	1	99.29	98.57	96.53	98.78

Table 12: Confusion matrix of ELM method

		Predicted class		
		LHL	HC	RHL
Target class	LHL	147	1	2
	HC	0	196	4
	RHL	7	1	132

Table 13: Results of classification for ELM method

Class	TP	TN	FP	FN	S_e (%)	S_p (%)	P_r (%)	A_{cc} (%)
LHL	147	333	7	3	98.00	97.94	95.45	97.96
HC	196	288	2	4	98.00	99.31	98.99	98.78
RHL	132	344	6	8	94.29	98.29	95.65	97.14

4.1. Comparison with state-of-the-art schemes

An overall comparative analysis has been made with the state-of-the-art schemes and the results are tabulated in Table 14. The parameter S_e^c in the table represents the sensitivity over the class c. It is observed that the proposed scheme achieves higher sensitivity and overall accuracy than the existing schemes. The proposed system yields a sensitivity value of 99.33%, 100.00%, and 97.86% over class LHL, HC, and RHL respectively. From these results, it can be deduced that the proposed scheme is promising for detection of SNHL.

Table 14: Performance comparison with state-of-the-art approaches

Reference	Scheme	S_e^1 (%)	S_e^2 (%)	S_e^3 (%)	A_o (%)
Yang *et al.* [16]	WE + DAG-SVM	94.00	97.00	93.57	95.10
Wang *et al.* [12]	FRFT + PCA + SLFN	94.00	96.54	94.29	95.10
Chen *et al.* [15]	DWT + PCA + GEPSVM	95.33	96.00	95.71	95.71
Liu *et al.* [17]	WPD + LS-SVM	95.33	95.00	96.43	95.51
Vasta *et al.* [13]	WE + MRF + NBC	-	-	-	91.02
Zhang *et al.* [14]	SWE + SLFN	95.33	97.00	94.29	95.71
Proposed scheme	**FDCT + PCA+LDA + MJaya-ELM**	**99.33**	**100.00**	**97.86**	**99.18**

5. Conclusions and future research directions

In this paper, we have developed an automated system for classification of hearing loss MR images. The fast discrete curvelet transform has been used for image decomposition. The feature vectors are then constructed by collecting the biggest coefficients from the selected subbands. A

mutation operator is introduced into traditional Jaya optimization (MJaya) to enhance the global search ability of the candidate solutions. A hybrid MJaya and ELM method (MJaya-ELM) is proposed to perform classification which ensures better generalization performance than its counterparts. The proposed scheme has been validated on a well-studied hearing loss dataset and its performance is compared with the state-of-the-art approaches. The results demonstrate the superiority of the proposed scheme in terms of overall accuracy and other performance measures. In addition, it is noticed that the introduction of mutation concept in MJaya-ELM ameliorates the classification performance significantly.

The limitation of the proposed work is that the system has been validated on a small dataset and its performance may deteriorate with a larger and diverse dataset. Hence, the efficacy of the system needs to be verified over a larger dataset. In future, we intend to investigate the applicability of the proposed methodology for the classification of other medical images. Further, the effectiveness of the proposed MJaya-ELM method could be evaluated on real-world binary/multiclass classification and regression problems. Contemporary meta-heuristic algorithms could also be hybridized with ELM to optimize its hidden node parameters.

Acknowledgments

This work is supported by Natural Science Foundation of China (61602250) and Key Laboratory of Measurement and Control of Complex Systems of Engineering, Southeast University, Ministry of Education (MCCSE2017A02).

Appendix A.

References

[1] T. Nakagawa, M. Yamamoto, K. Kumakawa, S.-i. Usami, N. Hato, K. Tabuchi, M. Takahashi, K. Fujiwara, A. Sasaki, S. Komune, et al., Prognostic impact of salvage treatment on hearing recovery in patients with sudden sensorineural hearing loss refractory to systemic corticosteroids: A retrospective observational study, Auris Nasus Larynx 43 (5) (2016) 489–494.

[2] A. Rathinavelu, H. Thiagarajan, A. Rajkumar, Three dimensional articulator model for speech acquisition by children with hearing loss, Universal Acess in Human Computer Interaction. Coping with Diversity (2007) 786–794.

Table A.1: Abbreviation list

Abbreviation	Definition
ABR	Auditory Brainstem Response
DE	Differential Evolution
DWT	Discrete Wavelet Transform
ELM	Extreme Learning Machine
FDCT	Fast Discrete Curvelet Transform
FNN	Feedforward Neural Network
FRFT	Fractional Fourier Transform
GEPSVM	Generalized Eigenvalue Proximal Support Vector Machine
LDA	Linear Discriminant Analysis
LS-SVM	Least squares Support Vector Machine
MJaya	Jaya with Mutation
MRF	Markov Random Field
MRI	Magnetic Resonance Imaging
NBC	Naive Bayes Classifier
PCA	Principal Component Analysis
PSO	Particle Swarm Optimization
SLFN	Single Layer Feedforward Network
SNHL	Sensorineural Hearing Loss
SVM	Support Vector Machine
SWE	Stationary Wavelet Entropy
UHL	Unilateral Hearing Loss
USFFT	Unequally Spaced Fast Fourier Transform
WPD	Wavelet Packet Decomposition

[3] I. Saliba, K. Sidani, Prognostic indicators for sensorineural hearing loss in temporal bone histiocytosis, International Journal of Pediatric Otorhinolaryngology 73 (12) (2009) 1616–1620.

[4] R. Vaswani, L. Parikh, N. Udochi, S. K. Vaswani, Rinne test modified to quantify hearing., Southern Medical Journal 101 (1) (2008) 107–108.

[5] S. Karelle, L. Demanez, P. Zangerle, P. Blaise, G. Moonen, A.-L. Poirrier, Sudden sensorineural hearing loss: when ophthalmology meets otolaryngology, B-ENT 8 (2) (2012) 135–139.

[6] H. Xiong, H. Chu, X. Zhou, X. Huang, Y. Cui, L. Zhou, J. Chen, J. Li, Y. Wang, Q. Chen, et al., Simultaneously reduced NKCC1 and na, K-ATPase expression in murine cochlear lateral wall contribute to conservation of endocochlear potential following a sensorineural hearing loss, Neuroscience Letters 488 (2) (2011) 204–209.

[7] N. Ikawa, Automated averaging of auditory evoked response waveforms using wavelet analysis,

International Journal of Wavelets, Multiresolution and Information Processing 11 (04) (2013) 1–21.

[8] E. Monzack, L. May, S. Roy, J. Gale, L. Cunningham, Live imaging the phagocytic activity of inner ear supporting cells in response to hair cell death, Cell Death & Differentiation 22 (12) (2015) 1995–2005.

[9] D. R. Nayak, R. Dash, B. Majhi, Brain MR image classification using two-dimensional discrete wavelet transform and adaboost with random forests, Neurocomputing 177 (2016) 188–197.

[10] R. Dash, B. Majhi, Least squares SVM approach for abnormal brain detection in mri using multiresolution analysis, in: International Conference on Computing, Communication and Security (ICCCS), IEEE, 2015, pp. 1–6.

[11] G. D. Wright, H. F. Horn, Three-dimensional image analysis of the mouse cochlea, Differentiation 91 (4) (2016) 104–108.

[12] S. Wang, M. Yang, Y. Zhang, J. Li, L. Zou, S. Lu, B. Liu, J. Yang, Y. Zhang, Detection of left-sided and right-sided hearing loss via fractional fourier transform, Entropy 18 (5) (2016) 194.

[13] R. Vasta, A. Augimeri, A. Cerasa, S. Nigro, V. Gramigna, M. Nonnis, F. Rocca, G. Zito, A. Quattrone, ADNI, Hippocampal subfield atrophies in converted and not-converted mild cognitive impairments patients by a Markov random fields algorithm, Current Alzheimer Research 13 (5) (2016) 566–574.

[14] Y. Zhang, D. R. Nayak, M. Yang, T.-F. Yuan, B. Liu, H. Lu, S. Wang, Detection of unilateral hearing loss by stationary wavelet entropy, CNS & Neurological Disorders-Drug Targets 16 (2) (2017) 122–128.

[15] Y. Chen, M. Yang, X. Chen, B. Liu, H. Wang, S. Wang, Sensorineural hearing loss detection via discrete wavelet transform and principal component analysis combined with generalized eigenvalue proximal support vector machine and Tikhonov regularization, Multimedia Tools and Applications 77 (3) (2018) 3775–3793.

[16] M. Yang, S. Du, J. Yang, B. Liu, J. M. Gorriz, J. Ramírez, T.-F. Yuan, Y. Zhang, Wavelet entropy and directed acyclic graph support vector machine for detection of patients with

unilateral hearing loss in MRI scanning, Frontiers in Computational Neuroscience 10 (2016) 1–11.

[17] B. Liu, J. Yang, P. Chen, X. Zhou, X. Wu, Y. Zhang, Computer-aided detection of left and right sensorineural hearing loss by wavelet packet decomposition and least-square support vector machine, Journal of The American Geriatrics Society 64 (2016) S350.

[18] J.-L. Starck, E. J. Candès, D. L. Donoho, The curvelet transform for image denoising, IEEE Transactions on Image Processing 11 (6) (2002) 670–684.

[19] D. R. Nayak, R. Dash, B. Majhi, Pathological brain detection using curvelet features and least squares SVM, Multimedia Tools and Applications 77 (3) (2018) 3833–3856.

[20] E. Candes, L. Demanet, D. Donoho, L. Ying, Fast discrete curvelet transforms, Multiscale Modeling & Simulation 5 (3) (2006) 861–899.

[21] D. R. Nayak, R. Dash, B. Majhi, V. Prasad, Automated pathological brain detection system: A fast discrete curvelet transform and probabilistic neural network based approach, Expert Systems with Applications 88 (2017) 152–164.

[22] J. Yang, J.-y. Yang, Why can LDA be performed in PCA transformed space?, Pattern Recognition 36 (2) (2003) 563–566.

[23] A. M. Martínez, A. C. Kak, PCA versus LDA, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2) (2001) 228–233.

[24] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.

[25] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 42 (2) (2012) 513–529.

[26] P. L. Bartlett, The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network, IEEE Transactions on Information Theory 44 (2) (1998) 525–536.

[27] Q.-Y. Zhu, A. K. Qin, P. N. Suganthan, G.-B. Huang, Evolutionary extreme learning machine, Pattern Recognition 38 (10) (2005) 1759–1763.

[28] Y. Xu, Y. Shu, Evolutionary extreme learning machine based on particle swarm optimization, in: International Symposium on Neural Networks, Springer, 2006, pp. 644–652.

[29] D. R. Nayak, R. Dash, B. Majhi, An improved pathological brain detection system based on two-dimensional PCA and evolutionary extreme learning machine, Journal of medical systems 42 (1) (2018) 1–15.

[30] F. Han, H.-F. Yao, Q.-H. Ling, An improved evolutionary extreme learning machine based on particle swarm optimization, Neurocomputing 116 (2013) 87–93.

[31] S. Suresh, R. V. Babu, H. Kim, No-reference image quality assessment using modified extreme learning machine classifier, Applied Soft Computing 9 (2) (2009) 541–552.

[32] A. S. Alencar, A. R. R. Neto, J. P. P. Gomes, A new pruning method for extreme learning machines via genetic algorithms, Applied Soft Computing 44 (2016) 101–107.

[33] R. Rao, Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems, International Journal of Industrial Engineering Computations 7 (1) (2016) 19–34.

[34] N. Kumar, I. Hussain, B. Singh, B. K. Panigrahi, Rapid MPPT for uniformly and partial shaded PV system by using JayaDE algorithm in highly fluctuating atmospheric conditions, IEEE Transactions on Industrial Informatics 13 (5) (2017) 2406–2416.