Indonesian Journal of Electrical Engineering and Computer Science
Vol. 40, No. 2, November 2025, pp. 654~666
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v40.i2.pp654-666

Classification of voice pathologies using one dimensional feature vector and two dimensional scalogram

Ranita Khumukcham[1], Sharmila Meinam[2], Kishorjit Nongmeikapam[3]
[1]Department of Electronics and Communication Engineering, Indian Institute of Information Technology Manipur, Imphal, India
[2]Department of Electronics and Communication Engineering, MIT, Manipur University, Imphal, India
[3]Department of Computer Science and Engineering, Indian Institute of Information Technology Manipur, Imphal, India

Article Info

Article history:

Received Jan 23, 2025
Revised Jul 30, 2025
Accepted Oct 14, 2025

Keywords:

Analytical Morlet (amor)
Bump
Discriminant analysis
K-nearest neighbor
Morse
Naïve Bayes

ABSTRACT

Most research work focus only on binary classification of voice pathologies such as normal and pathological classification. However, the current work gives importance to multiclass classification too. The paper compares one-dimensional (1D) feature vectors based machine learning (ML) techniques and two-dimensional (2D) scalogram image based deep learning (DL) model for binary and multiclass classification of voice pathology. The multiclass classification classifies the voice signal into four categories which are healthy, hyperkinetic dysphonia, hypokinetic dysphonia, and reflux laryngitis. The current work demonstrates the evaluation of 1D feature vectors extracted from speech signal such as MFCC (mel-frequency cepstral coefficient) and pitch with various ML techniques like K-nearest neighbor (KNN), Naïve Bayes, and discriminant analysis (DA). Another technique that uses time-frequency scalograms derived using three different wavelets, i.e., analytical Morlet (amor), Bump, and Morse, are used for training a pre-trained GoogleNet architecture, which is a very popular DL model. Experimental results show that 2D scalogram image based DL model for binary (96.05%) and multiclass (89.8%) classification of voice pathology gives better performance while comparing with 1D feature vectors based ML techniques.

Corresponding Author:

Ranita Khumukcham
Department of Electronics and Communication Engineering
Indian Institute of Information Technology Manipur
Imphal 795002, India
Email: ranitakh89@gmail.com

1. INTRODUCTION

Speech is the most basic form of expression, and any change to the vocal cord interrupts its seamless flow. Vocal fatigue, pressure, dysphonia, roughness, glottal assault, sore throat, and other symptoms are exacerbated by speech problems. Long-term vocal cord abuse can result in diseases such as laryngeal malignancy, folding, polyp, and nodule. The hoarseness of one's voice might define these conditions. Aside from self-abuse, a sedentary lifestyle may lead to an increase in voice problems [1], [2]. Deep learning (DL) has surpassed traditional classifiers such as Naïve Bayes, decision trees, K-nearest neighbor (KNN), and support vector machine (SVM). Since the last several decades, handcrafted speech or acoustic characteristics have been critical for detecting voice disorder and this cannot be overlooked [3]-[7]. For diagnosing voice pathology, a wide range of long and short feature descriptors have been employed. Long-term characteristics have been employed in certain significant research studies [8]-[11]. Wahed [12] suggested a study to develop a detector for vocal larynx abnormalities by extracting a mixture of various feature descriptors from a

diseased voice sample. Orozco-Arroyave *et al.* [13] presented a method for diagnosing Parkinson's illness, palate dysfunction, and vocal fold abnormalities, hypernasal lip. Another seminal study that used entropy to distinguish between healthy and diseased voices was suggested in [14]. Uloza *et al.* [15] describes a multi-class voice pathology classifier that employs a rich feature vector generated from varied and common speech characteristics. Reynolds and Rose [16] removed characteristics from short regular utterances using a mel-frequency filter bank. Pravena *et al.* [17] used the Gaussian mixture model (GMM) model to train 11 distinct mel-frequency cepstral coefficient (MFCC) characteristics to distinguish a normal voice from a disordered one. The machine learning techniques is applied in most of the medical application [18].

2. METHOD

For studying the implementation results of one-dimensional (1D) and 2D based machine learning (ML) and DL systems, two separate workflows are proposed in the current methodology, as discussed in the following two subsections. The summarized architectures of Figures 1 and 2 are similar, except, the former deals with a binary class prediction and the latter is a multiclass predictor. In case of 1D binary classification, the first step is to collect data which is input speech, the second step go for feature extraction, the third step is for ML which is training and testing the sample provided and the last step undergoes classification of healthy and pathological. The MFCC and pitch characteristics are extracted from the input signal as a feature extraction. KNN, Naive Bayes, and discriminant analysis (DA) are used for training and testing the samples. In 2D binary classification, the input speech is converted into time-frequency scalogram and goes for DL using GoogleNet and lastly classification. The time frequency sclogram and DL method is explained in the later section. In case of 1D multiclass classification, the first step is to collect data which is input speech, the second step go for feature extraction, the third step is for ML which is training and testing the sample provided and the last step undergoes classification of healthy, hyperkinetic dysphonia, hypokinetic dsyphonia and laryngitis. As feature extraction, MFCC and pitch characteristics are extracted from the input signal. KNN, Naive Bayes, and DA are used for ML which explained in later section. In 2D mutliclass classification, the input speech is converted into time-frequency scalogram and goes for DL and lastly classification which is explained in later section.

2.1. Dataset

Cesari *et al.* [19] suggested a vocal pathology dataset, which will be used in this study. The collection contains 151 diseased and 55 healthy speech samples, respectively. There are three types of abnormal voices: hypokinetic dysphonia, hyperkinetic dysphonia, and reflux laryngitis. All recordings feature a 4.76 second sustained ‘a' vowel sound at an 8 kHz sampling rate. To avoid overfitting, each speech sample is split into 10 equal length segments of 0.476 second duration, 3,808 sampling points, and an 8 kHz sampling frequency. Overfitting or excessive variance might lead to misleading positive outcomes. As indicated in Table 1, this arrangement yielded 1,510 and 550 diseased and healthy speech samples, respectively. To prevent the issue of class imbalance, the total number of samples that will be trained and tested is 550 for each class. The number of segmented samples for the healthy class, 550, is used as the upper limit in this case. This balanced no. will subsequently take part in training and testing.

There are 41, 72, and 38 samples from the hypokinetic dysphonia, hyperkinetic dysphonia and reflux laryngitis categories, respectively, among the 151 un-segmented voice samples. It's also worth noting that they're all divided into ten equal-length speech samples. Table 2 shows that there are now 720, 410, and 380 samples available for each of the three classes. To prevent the issue of class imbalance, the number of samples for all four classes is kept at 380, with reflux laryngitis having the fewest. This balanced no. will take part in future training and testing.

Table 1. Dataset distribution for binary prediction

Class	Original no.	Segmented no.	Balanced no.
Healthy	55	550	550
Pathological	151	1510	550

Table 2. Dataset distribution for multiclass prediction

Class	Original no.	Segmented no.	Balanced no.
Healthy	55	550	380
Hypokinetic dysphonia	41	410	380
Hyperkinetic dysphonia	72	720	380
Reflux laryngitis	38	380	380

2.2. Framework for classification using 1D features and machine learning

The workflow of the proposed 1D features-based ML architecture is shown in Figures 1(a) and 2(a). It will consist of three stages as explained in subsection.

Figure 1. The proposed architectures for (a) 1D and (b) 2D learning models for binary prediction

2.2.1. Speech input

Speech samples from either Table 1 or Table 2 will be used depending on the type of prediction model needed, i.e., binary or multiclass. Regardless of prediction model, all samples have an 8 kHz sampling frequency and 3,808 sampling points.

2.2.2. Feature descriptors

The MFCC and pitch characteristics are extracted from the input signal. These two characteristics are retrieved from a single input voice sample and concatenated into a single vector. Concatenated vectors of this kind are created for all training samples. They will participate in training.

MFCC is an acoustic signal description predicated on the linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency [20]. The MFCC features are the coefficients that make up the mel-frequency cepstrum. This frequency warping improves the representation of sound and speech data. The window length is set at 3% of the sampling rate, which is 240. And the overlap length is fixed at 2.5% of the sampling rate, which is 200. The original sampling rate, i.e., 8 kHz is utilized.

Pitch. The fundamental frequency or pitch of a voice relates to the number of times the vocal folds come together during phonation per second. The auto-correlation function is used in time-domain pitch period estimate methods (ACF). The main principle behind correlation-based pitch tracking is that the correlation signal will have a significant magnitude peak during the pitch period's lag. The autocorrelation computation is performed directly on the waveform and is a simple calculation [21]. Salhi *et al.* [21] computes the autocorrelation function for a signal x(n).

$$\varphi x(m) = \lim_{n\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x(n)x(n+m) \quad (1)$$

The autocorrelation function of a signal is basically a transformation of the signal which is useful for displaying structure in the waveform. Thus, for pitch detection, if we assume x(n) is exactly periodic with period P, i.e. x(n)=x(n+P) for all n, then the autocorrelation function of (1) is also periodic with the same period.

$$\varphi x(m) = \varphi x(m+P) \quad (2)$$

2.2.3. Machine learning classifiers

There are numerous classification algorithms available today, but none of them outperform the others in every case [22]. We chose three classifiers for the current work study: KNN, Naive Bayes, and DA. These classifiers are trained individually using the concatenated feature vectors obtained from the training samples.

Akbulut *et al.* [23] states, the KNN technique is among the earliest and easiest kinds of nonparametric classifier. The drawback is that when a low k value is used, the separation border becomes excessively adapted to the training data, resulting in over-training. At higher k values, the border tends to be smoother, resulting in improved prediction results for fresh samples. The best value of k must be found

empirically. To identify the optimal value of k, we empirically evaluated different values of k using the Euclidean distance metric. More specifically, we tested k=1,2,3,4,5,6,7. It was discovered that a value of k=5 produces the greatest results.

Naive Bayes: the fundamental feature of Naive Bayes is a strong naive assumption of independence from each condition or occurrence. It is a straightforward model that may be used to huge datasets. The basis of the Naive Bayes theorem is the Bayes formula, which is given by

$$P(C|X) = \frac{P(C)\,P(X|C)}{P(X)} \quad (3)$$

where, X= (x_1, x_2, x_3, …, x_n) is the attribute, C is the class, P(C|X): probability of event C given X has occurred, P(X|C): probability of event X given C has occurred, P(C): probability of event C, P(X): probability of event X.

We must maximise the probability value of each class in the Nave Bayes classifier, which is represented as the hypothesis maximum a posteriori (HMAP).

$$H_{MAP} = \arg\max P(C|X_1, X_2, \ldots\ldots\ldots, X_n) = \arg\max P(C) \prod_{i=1}^{n} P(X_i\,|C) \quad (4)$$

Where, P represents opportunity, x_i is the i_{th} attribute value, C is class.

Linear discriminant analysis (LDA): it can be used for classification as well as dimensionality reduction. This classifier evaluates a projection hyperplane that accomplishes two goals: 1) interclass variance should be reduced, and 2) projected means of classes should be as close to each other as possible [4]. Consider the following example in which a class is to be predicted. Let X represent the predictor variables. Suppose X is the single predictor variable, i.e. X=x. Let $f_k(x)$ be the estimated discriminator score that the observation belongs to the C_k class. Then, $f_k(x)$ can be evaluated by the formula:

$$f_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\prod_k) \quad (5)$$

where, $\prod_k$ is the prior probability that the class of observation is C_k. $\mu_k$is the average of training observations belonging to class C_k. For each of the K classes the weighted average of sample variances is represented by σ^2. The LDA classifier will predict that class k for the given observation whose discriminant score is largest.

2.3. Framework for classification using 2D scalograms and deep learning

The current subsection will discuss the effects of using an image-based analysis for performing both binary and multiclass predictions. The workflow is highlighted in Figures 1(b) and 2(b). The first step is to generate scalogram images from all samples of every class.

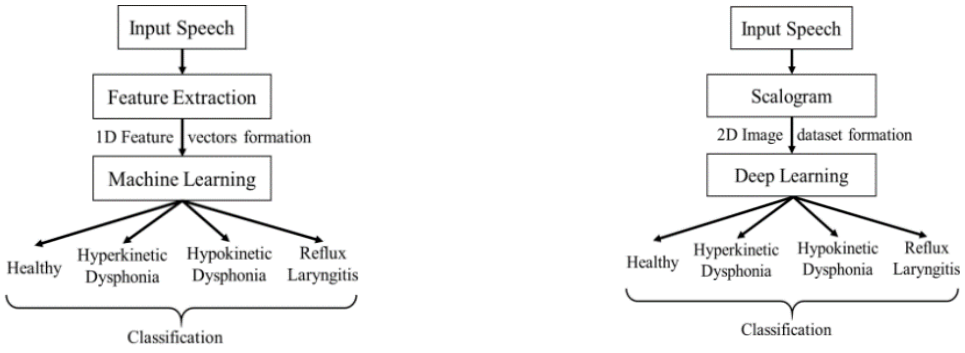

Figure 2. The proposed architectures for (a) 1D and (b) 2D learning models for multiclass prediction

Time-frequency scalograms: the next step is to convert the segmented speech samples from each of these three classes into Morse scalogram (M.S) 2D images. The continuous wavelet transform (CWT) of a given signal having function f(t) is evaluated by using the mother wavelet through the expression:

$$CWT(x,y) = \frac{1}{\sqrt{x}}\int_{-\alpha}^{+\alpha} f(t) * m\left(\frac{t-y}{x}\right) dt \quad (6)$$

where, x and y are the scaling and shifting factor for the mother wavelet and * signifies convolutions operation. The above expression can be translated as an integration of summation of the input audio sample multiplied by the time scaled and shifted forms of the mother wavelet (m).

The Morse wavelet is being chosen for the current work because it displays strong localization in both the frequency and temporal domains, making it ideal for studying localized discontinuities. The fourier transform of a Morse wavelet is expressed as:

$$m_{d,n}(\omega) = \varepsilon(\omega)\alpha_{d,n}\omega_n^{d^2}e^{-\omega^n} \tag{7}$$

where, $\xi(\omega)$ is a unit step function, d^2 is the time-bandwidth product, $\alpha_{d,\eta}$ signifies normalization constant and η is the symmetry parameter. Different combination of d^2 and η can produce diverse Morse wavelets.

Similarly, the coefficients of (6) can be implemented with bump wavelet transformation [24] to derive the glottal derivative Bump scalogram. The fourier transform of a bump wavelet is:

$$\psi(s\omega) = e^{\left(1-\frac{1}{1-\frac{(s\omega-\mu)^2}{\sigma^2}}\right)}1_{\left[\frac{\mu-\sigma}{s},\frac{\mu+\sigma}{s}\right]} \tag{8}$$

where, σ and μ are parameters that controls the transformed signal's frequency and time localization.

Applying time-domain to frequency-domain transformation using wavelet, the 1-D input signal is transformed into a 2D signal. And an analytical morlet (amor) wavelet based time- frequency version of the input audio is:

$$\omega = e^{2i\pi ft}e^{\frac{-4\ln(2)t^2}{h^2}} \tag{9}$$

where h is full-width at half-maximum (FWHM) which is the distance in time between 50% gain before the peak to 50% gain after the peak [23].

2.3.1. GoogLeNet

It is a cutting-edge convolutional neural network (CNN) suggested by Google. It had a top-five mistake rate of 6.67 % [25]. The GoogleNet employs nine (9) 1D-inception modules, each of which employs three distinct convolutional kernels, namely 1x1, 3x3, and 5x5. This network has a total of 142 layers. The input layer is a 2D image input layer with 224x224x3 dimensions. It is linked to a convolutional layer with a kernel size of 7x7, stride of 2, and 512 filters. This layer will collect features from the preceding layer (the input layer) and store them as activation maps with 512 depths (equal to the number of filters). It is linked to a max-pooling layer with kernel size or filter size 3x3 and stride equal to 2 through a rectified linear unit (ReLU) layer. The max-pooling layer's goal is to downsample (or minimise) the size of the activation maps created by the previous layer. To minimise overfitting, this new activation map is now put into a normalising layer. Overfitting is a phenomenon that reduces DL network accuracy by supplying features in a non-uniform manner. Overfitting is minimised by utilising either a dropout layer or a normalising layer; currently, dropout is seldom employed, and batch normalisation or cross channel normalisation has largely replaced it. The normalised layer is linked to two further convolutional layers with kernel sizes of 3x3, stride 2 through a ReLU layer. With this second convolutional layer, a cross channel normalisation layer is employed, followed by a max-pooling layer. This max-pooling layer's activation maps are linked to an inception module. Each inception module includes 13 layers, 6 of which are convolutional layers and the rest are a mix of ReLU and max-pooling layers. A depth concatenation module is utilised at the conclusion of each inception module to merge the activation maps from the inception module's four columns. The GoogLeNet's final layers include dropout, fully connected, softmax, and a classification output layer. The dropout layer employs a dropout probability of 70%. The dimension of the completely linked layer is 2,048. The related probabilities will be computed using the softmax layer. The last layer is a classification output layer, which will be programmed to identify the number of classes requested.

3. RESULTS AND DISCUSSIONS

After carefully implementing the precodure in the model, the experimental results are evaluated as follows:

3.1. Evaluation metrics

The current work will be evaluated using nine (9) metrics, which are – sensitivity (Sen.), accuracy (Acc.), Cohen's kappa index error (Err.), precision (Pre.), specificity (Spe.), Matthews correlation coefficient (MCC), false positive rate (FPR), and F1 score. Here, TP, TN, FP, FN stands for true positive, true negative, false positive, and false negative respectively.

- Sensitivity: it identifies the actual number of positive samples of all the positives samples. It is also called as true positive rate (TPR) and is given by:

$$Sensitivity = \frac{TP}{TP+FN} \tag{10}$$

- Accuracy: it is the simplest and most common metric for model evaluation. It is the ratio of the correct prediction which is the sum of TP and TN to the total number of predictions of the given dataset or samples which is given by:

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \tag{11}$$

- Cohen's Kappa index: it is used to measure the fedility of two raters. If the value is less, then zero than there is no agreement and if it is in between 0.81 to 1 than ther is perfect agreement.

$$CKI = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e} \tag{12}$$

- Error: it determines the wrong classification which is given by (13).

$$Err = 100 - Acc \tag{13}$$

- Precision: it is the ratio of the true positives to all the positives of the samples.

$$Precision = \frac{TP}{TP+FP} \tag{14}$$

- Specificity: it identifies the actual number of negative samples of all the negative samples. Here, it is more important to classify the negative then to classify the positive. So, it is also called TNR.

$$Specificity = \frac{TN}{TN+FP} \tag{15}$$

- MCC: it is a measure for binary classification's quality. It gives best result for an unbalanced class while taken into consideration TPs, TNs, FPs, FNs.

$$MCC = \frac{TP\ X\ TN - FP\ X\ FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{16}$$

- False positive rate: it is the probability that positive result is predicted when the true value is negative which a false prediction.

$$FPR = \frac{FP}{FP+TN} \tag{17}$$

- F1 score: it combines the precision and recall of the samples which is given by the harmonic mean of precision and recall and is known as dice similarity coefficient (DSC). It gives better performance for unbalanced dataset.

$$F_1 = \frac{2TP}{2TP+FP+FN} \tag{18}$$

3.2. Implementation results of the 1D feature-based machine learning approaches

The current section shows the implementation of a 1D image-based ML approach for performing binary and multiclass prediction. There are two subsections – binary prediction and multiclass prediction. For the binary prediction, there are 550 samples in each category as shown in Table 3 and mean classification

score is shown in Table 4. Whereas, the multiclass prediction utilizes four categories which have 550 samples in each category in Table 5.

3.2.1. Binary prediction

For each of the two classes, the training and testing samples are split in the ratio of 80:20%. It results in the formation of 440 training and 110 test samples respectively. The training samples are made to undergo training with three ML techniques individually by using the parameters mentioned in section 2.2.3. It may be kindly noted that the training samples are initially converted to MFCC and pitch feature vectors and then fed to the ML algorithms. Due to the use of window for feature extraction as mentioned in section 2.2, a total of 8,941 speech frames are generated from the 110 healthy test samples. Similarly, a total of 7,637 frames are generated from 110 pathological test samples. It is these resulting test frames that will undergo exhaustive testing. The test frames mentioned above are tested against the KNN, Naïve Bayes, and LDA classifiers. The per-class performance is highlighted in Figures 3 to 5. It is observed that all the three classifiers provide significantly low per-class performance, with 62.4% as shown in Figure 3 being the highest for the healthy class and 50.6% shown in Figure 4 for the pathological class. The mean scores derived from these per-class scores are also highlighted in Table 4. It is observed that KNN provides the highest accuracy 57.89% in comparison to the other two classifiers.

Table 3. Number of training and test samples for binary prediction

Class	Training	Testing	Total	Test frames
Healthy	440	110	550	8,941
Pathological	440	110	550	7,637

Table 4. Mean classification score of 1D feature-based binary prediction

Algorithm	Sen.	Acc.	Kappa	Err	Pre.	Spe.	MCC	FPR	F1
KNN	67.89	57.89	15	42.11	58.27	47	15.23	53	62.71
Naïve Bayes	62.39	55.02	9.43	44.98	56.2	47	9.5	53	59.13
LDA	59.63	54.55	8.66	45.45	56.03	49	8.68	51	57.78

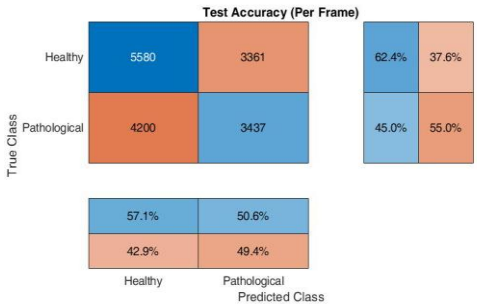

Figure 3. Per-class and per-frame classification

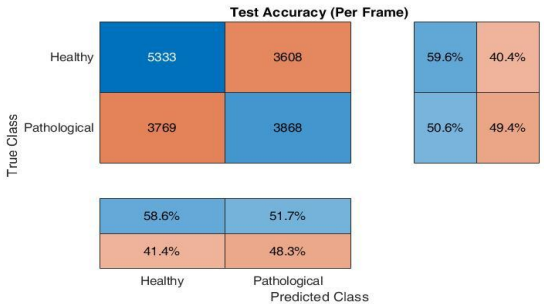

Figure 4. Per-class and per-frame classification result for the Naïve Bayes method result for the KNN method

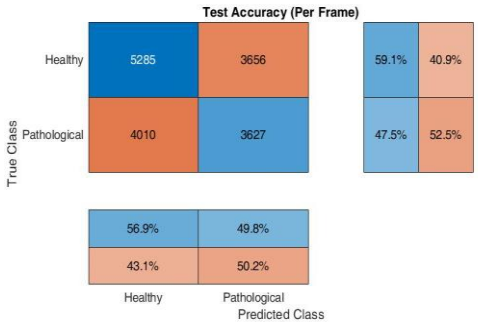

Figure 5. Per-class and per-frame classification result for the LDA method

Therefore, it is noted that use of 1D feature with ML classifiers cannot give good results in cases of low number of training samples. It is also desired to study the performance of 2D image-based datasets with DL algorithms. They are performed in section 3.3.

3.2.2. Multiclass prediction

For the current multiclass prediction also, the training and test samples are again split in the ratio 80:20%. It has also been discussed in Table 2 that there are 380 samples in each of the four classes. By applying the above splitting ratio, the number of training and test samples in each class are 304 and 76 respectively. From all the training samples, feature vectors which is a combination of MFCC and pitch feature vectors are extracted. These feature vectors are utilized in training three ML classifiers individually by using the parameters mentioned in section 2.2.3. Also, due to the use of window for feature extraction as mentioned in section 2.2, the number of test frames for each of the four classes are 5,198; 5,514; 3,393; and 5,099 (see Table 5). These test frames will undergo exhaustive testing.

The test frames mentioned in Table 5 are tested against the KNN, Naïve Bayes, and LDA classifiers. The per-class performance is highlighted in Figures 6 to 8. It is observed that the KNN provided the best per-class accuracies for healthy (i.e., 40.2%), hyperkinetic dysphonia (i.e., 32.5%), hypokinetic dysphonia (i.e., 56.5%). The Naïve Bayes classifier provided the best performance for the reflux laryngitis category by demonstrating an accuracy of 42.9%.

Table 5. Number of training and test samples for multiclass prediction

Class	Training	Testing	Total	Test frames
Healthy	304	76	380	5,198
Hyperkinetic dysphonia	304	76	380	5,514
Hypokinetic dysphonia	304	76	380	3,393
Reflux Laryngitis	304	76	380	5,099

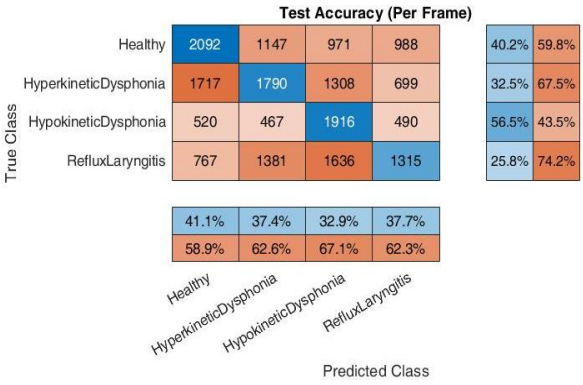

Figure 6. Per-class and per-frame classification result for the KNN method

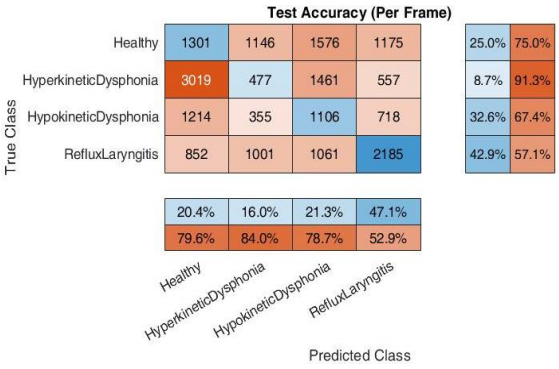

Figure 7. Per-class and per-frame classification result for the Naïve Bayes method

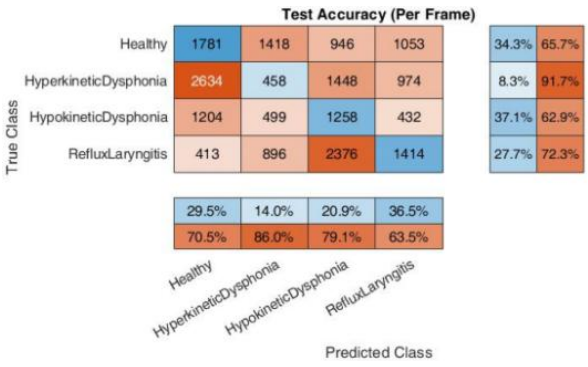

Figure 8. Per-class and per-frame classification result for the LDA method

The mean scores derived from these per-class scores are also highlighted in Table 6. It is observed that similar to the binary prediction approach discussed above, the KNN classifier demonstrated the best performance for the multiclass prediction also. It is also desired to study the performance of 2D image-based datasets with DL algorithms. They are performed in section 3.3.

Table 6. Mean classification score of 1D feature-based binary prediction

Algorithm	Sen.	Acc.	Kappa	Error	Pre.	Spe.	MCC	FPR	F1
kNN	37.82	37.02	40.45	62.98	37	79.09	16.47	20.91	36.18
Naïve Bayes	28.14	27.68	44.71	72.32	25.02	75.97	14.03	24.03	26.16
LDA	32.78	32.18	44.71	67.82	30.23	77.51	19.95	22.49	30.74

3.3. Implementation results of the 2D image-based deep learning approach

The current section shows the implementation of a 2D image-based DL approach for performing binary and multiclass prediction. There are two subsections –binary prediction an multiclass prediction. For the binary prediction, there are 380 training samples in each category as Table 7. Whereas, the multiclass prediction utilizes four categories which have 380 samples in each category in Table 8.

Table 7. Number of training and test samples for binary prediction

Class	Training	Testing
Healthy	304	76
Pathological	304	76

Table 8. Mean classification score of 2D feature-based binary prediction

Algorithm	Dataset	Sen.	Acc.	Kappa	Error	Pre.	Spe.	MCC	FPR	F1
GoogLeNet	M.S.	96.05	96.05	92.11	3.95	96.05	96.05	92.11	3.95	96.05
GoogLeNet	B.S.	96.05	96.05	92.11	3.95	96.05	96.05	92.11	3.95	96.05
GoogLeNet	A.S.	96.05	94.74	89.47	5.26	6.58	93.42	89.5	94.81	94.81

3.3.1. Binary prediction

It can be seen in Table 7 that there are two categories in which there are 380 samples each. The training and testing ratio were divided in the ratio of 80:20% respectively. It translates to around 304 training and 76 test samples respectively. The training and test samples were kept in different folders so that none of the test samples were used (or seen) during the training process. Furthermore, the training samples were further divided into training and validation samples in the ratio 80:20% respectively. This means that out of 304 training samples there are 243 actual training and 61 validation samples respectively.

Therefore, it can be summarized that there are 243 training, 61 validation and 76 test samples respectively for each class. The parameters mentioned in section 3.3 (above) is used for developing the M.S database. The training and validation samples are made to undergo training by setting the following parameters: minibatch size as 16, validation frequency 30 and flat learning rate of 0.0001. The number of epochs is set as 15, however the training process is terminated when the validation accuracy and loss curves become flat. Figure 9 shows the training progression for the GoogLeNet model with the M.S dataset for binary class prediction. Figure 10 gives its confusion matrix. For an extensive evaluation, the Amor scalogram (A.S) and Bump scalogram (B.S) datasets are also developed as shown in Figures 11 to 14. Another two GoogLeNet models are also training with these datasets by using the same set of DL training parameters. The per class performance of the GoogleNet with the M.S dataset is shown by the confusion matrix of Figure 10. It is observed that 73 out of 76 healthy test samples are correctly predicted, thereby giving a per-class accuracy of 96.1%. Similarly, the pathological test samples are also classified with a per-class accuracy of 96.1%.

The mean classification scores are also recorded in Table 8 for a comparison with other scalograms such as the B.S and A.S dataset. The separate evaluation of the GoogLeNet with the M.S. and B.S datasets shows a similar performance, i.e., 96.05% each. The value of MCC and Kappa are slightly low (i.e., 92.11% each). The A.S. dataset with the GoogLeNet provides the lowest mean accuracy, which is 94.74%.

3.3.2. Multiclass prediction

There are four classes in this type of prediction. They are – (i) healthy, (i) hyperkinetic dysphonia, (iii) hypokinetic dysphonia, and (iv) reflux laryngitis. The same training, validation and test samples splitting

pattern mentioned in section 3.3.1 is also adopted here. There are 243 training, 61 validation and 76 test samples respectively for each class. By applying the same set of parameters mentioned in section 3.3, the M.S, B.S, and A.S datasets are generated by using all the samples mentioned in Table 9. The same set of DL training parameters mentioned in section 3.3.1 is used here.

Figure 15 shows the training progress of the GoogleNet with the M.S dataset for multiclass class prediction. The per class performance of the GoogleNet with the M.S dataset is shown by the confusion matrix of Figure 16. It is observed that since the number of classes has increased in comparison to the binary prediction, all the four classes demonstrate around 90% per-class accuracy. For instance, the healthy test samples are classified with an accuracy of 88.2%, the hyperkinetic dysphonia and reflux laryngitis are classified with 90.8% accuracy each. Finally, the hypokinetic dysphonia records a per-class accuracy of 89.5%. The mean classification scores are also recorded in Table 10 for a comparison with other scalograms such as the B.S and A.S dataset. The separate evaluation of the GoogLeNet with the M.S., B.S. and A.S. datasets show that use of M.S. with GoogLeNet provides the best performance over 9 metrics.

Figure 17 shows the training progress of the GoogleNet with the A.S dataset for multiclass class prediction. The per class performance of the GoogleNet with the A.S dataset is shown by the confusion matrix of Figure 18. It is observed that the healthy test samples are classified with an accuracy of 94.7%, the hyperkinetic dysphonia of 78.9%, the hypokinetic dysphonia of 77.6% and reflux laryngitis are of 94.7%. Meanwhile, the training progress of the GoogleNet with the B.S dataset for for multiclass class prediction is shown in Figure 19. And its confusion matrix is shown in Figure 20. It is observed that the healthy test samples are classified with an accuracy of 84.2%, the hyperkinetic dysphonia of 84.2%, the hypokinetic dysphonia of 77.6% and reflux laryngitis are of 89.5%.

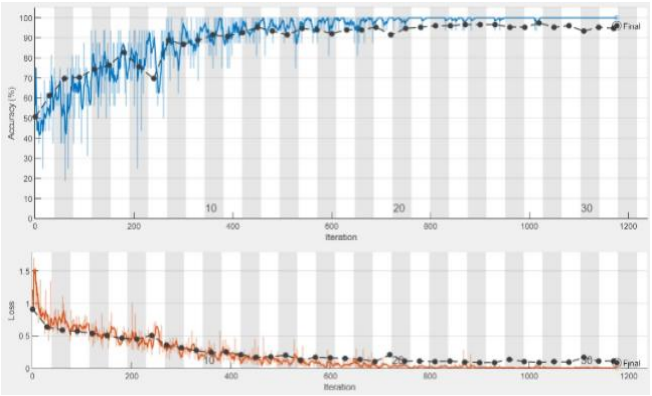

Figure 9. Training progress of the GoogLeNet with the per-class M.S dataset for binary class prediction

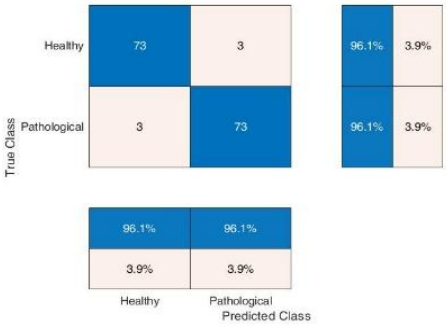

Figure 10. Confusion matrix showing the result GoogLeNet with the M.S

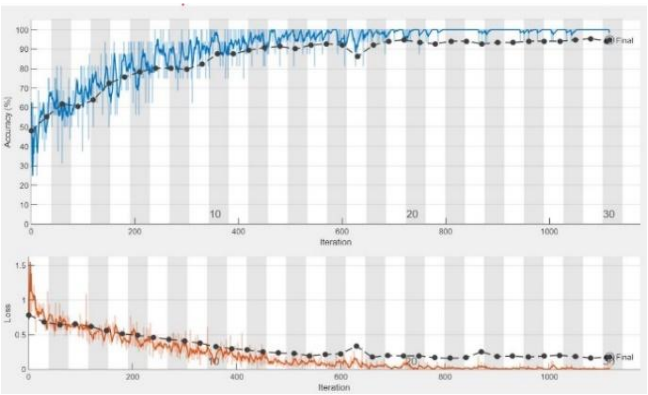

Figure 11. Training progress of the GoogLeNet with the A.S dataset for binary classprediction

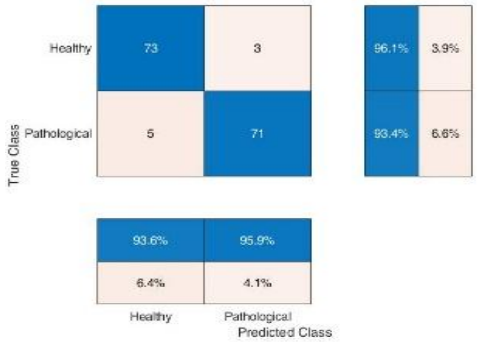

Figure 12. Confusion matrix showing the per-class result GoogLeNet with the A.S

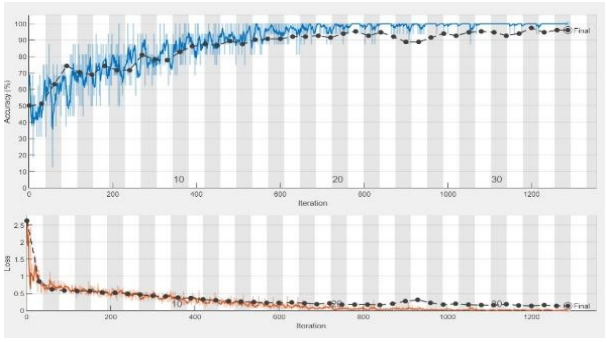

Figure 13. Training progress of the GoogLeNet with the B.S dataset for binary class prediction

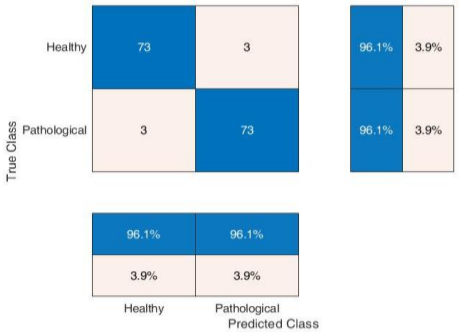

Figure 14. Confusion matrix showing the per class result GoogLeNet with the B.S

Table 9. Number of training and test samples for multiclass prediction

Class	Training	Testing
Healthy	304	76
Hyperkinetic dysphonia	304	76
Hypokinetic dysphonia	304	76
Reflux Laryngitis	304	76

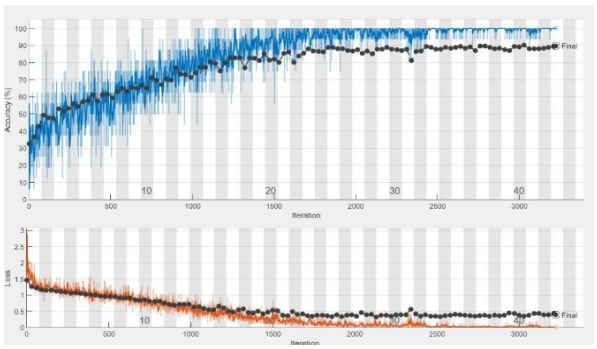

Figure 15. Training progress of the GoogLeNet with the M.S dataset for multiclass class prediction

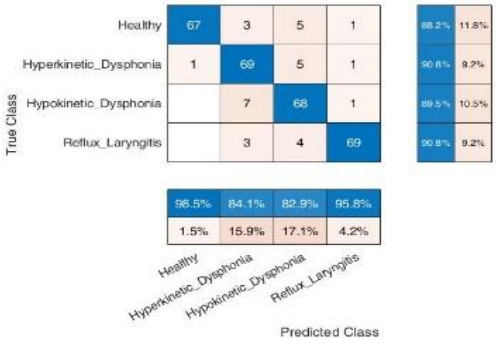

Figure 16. Confusion matrix showing the per-class result GoogLeNet with the M.S

Table 10. Mean classification score of 2D feature-based multiclass prediction

Algorithm	Dataset	Sen.	Acc.	Kappa	Error	Pre.	Spe.	MCC	FPR	F1
GoogLeNet	M.S.	89.8	89.8	72.81	10.2	90.36	96.6	86.65	3.4	89.93
GoogLeNet	B.S.	83.88	83.88	57.02	16.12	84.48	94.63	78.77	5.37	83.92
GoogLeNet	A.S.	86.51	86.51	64.04	13.49	87.02	95.5	82.24	4.5	86.36

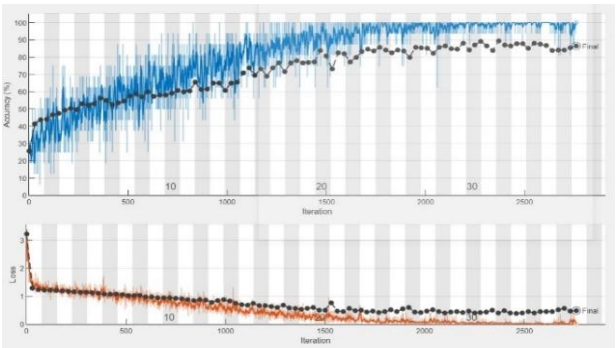

Figure 17. Training progress of the GoogLeNet with the A.S dataset for multiclass class prediction

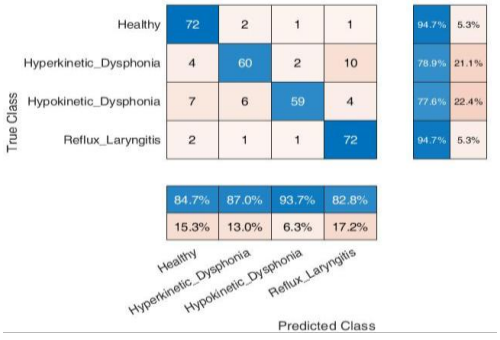

Figure 18. Confusion matrix showingthe per-class result GoogLeNet with the A.S

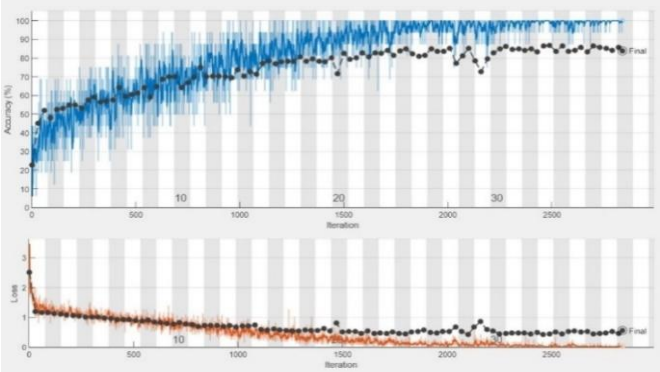

Figure 19. Training progress of the GoogLeNet with the B.S dataset for multiclass class prediction

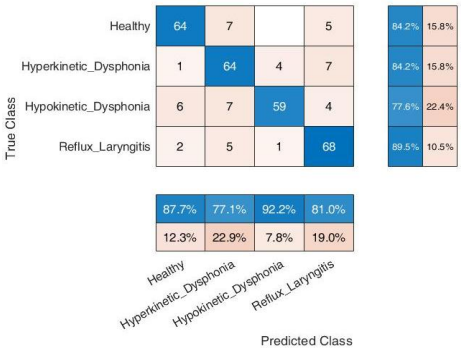

Figure 20. Confusion matrix showing the per- class result GoogLeNet with the B.S

4. CONCLUSION

The current work has performed an exhaustive evaluation of the performance of 1D-based ML and 2D-based DL binary and multiclass predictions. It is observed that even though same number of training and test samples are used for 1D and 2D methods, the 1D based method demonstrates significantly poor performance than 2D or image-based prediction. For instance, the highest mean accuracy obtained from the 1D based classifier are 57.89% and 37.02% for binary and multiclass prediction. Whereas, the highest mean accuracy obtained from the 2D based classifier are 96.05% and 89.8% for binary and multiclass classifier respectively. Therefore, it is understood that voice pathology classification can be successfully performed with image-based DL techniques.

FUNDING INFORMATION

Authors state no funding involved.

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability does not apply to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] I. Hammami, L. Salhi, and S. Labidi, "Voice pathologies classification and detection using EMD-DWT analysis based on higher order statistic features," *Irbm*, vol. 41, no. 3, pp. 161–171, Jun. 2020, doi: 10.1016/j.irbm.2019.11.004.

[2] A. Al-nasheri, G. Muhammad, M. Alsulaiman, and Z. Ali, "Investigation of voice pathology detection and classification on different frequency regions using correlation functions," *Journal of Voice*, vol. 31, no. 1, pp. 3–15, Jan. 2017, doi: 10.1016/j.jvoice.2016.01.014.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, no. 2, pp. 1097–1105, May 2012.

[4] K. Nongmeikapam, K. Wahengbam, O. N. Meetei, and T. Tuithung, "Handwritten Manipuri Meetei-Mayek classification using convolutional neural network," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, no. 4, pp. 1–23, May 2019, doi: 10.1145/3309497.

[5] K. Nongmeikapam, W. K. Kumar, O. N. Meetei, and T. Tuithung, "Increasing the effectiveness of handwritten Manipuri Meetei-Mayek character recognition using multiple-HOG-feature descriptors," *Sadhana - Academy Proceedings in Engineering Sciences*, vol. 44, no. 5, p. 104, Apr. 2019, doi: 10.1007/s12046-019-1086-0.

[6] W. K. Kumar, K. Nongmeikapam, and A. D. Singh, "An urban parametric scene parsing technique through an improved multispectral image fusion," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3516699.

[7] L. M. Devi, K. Wahengbam, and A. D. Singh, "Dehazing buried tissues in retinal fundus images using a multiple radiance pre-processing with deep learning based multiple feature-fusion," *Optics and Laser Technology*, vol. 138, p. 106908, Jun. 2021, doi: 10.1016/j.optlastec.2020.106908.

[8] H. Kasuya, S. Ogawa, K. Mashima, and S. Ebihara, "Normalized noise energy as an acoustic measure to evaluate pathologic voice," *The Journal of the Acoustical Society of America*, vol. 80, no. 5, pp. 1329–1334, Nov. 1986, doi: 10.1121/1.394384.

[9] B. Boyanov, T. Ivanov, S. Hadjitodorov, and G. Chollet, "Robust hybrid pitch detector," *Electronics Letters*, vol. 29, no. 22, pp. 1924–1926, Oct. 1993, doi: 10.1049/el:19931281.

[10] L. Gavidia-Ceballos and J. H. L. Hansen, "Direct speech feature estimation using an iterative EM algorithm for vocal fold pathology detection," *IEEE Transactions on Biomedical Engineering*, vol. 43, no. 4, pp. 373–383, Apr. 1996, doi: 10.1109/10.486257.

[11] A. Al-nasheri *et al.*, "An investigation of multidimensional voice program parameters in three different databases for voice pathology detection and classification," *Journal of Voice*, vol. 31, no. 1, pp. 113.e9-113.e18, Jan. 2017, doi: 10.1016/j.jvoice.2016.03.019.
[12] M. A. Wahed, "Computer aided recognition of pathological voice," in *2014 31st National Radio Science Conference (NRSC)*, Apr. 2014, pp. 349–354, doi: 10.1109/NRSC.2014.6835096.
[13] J. R. Orozco-Arroyave *et al.*, "Characterization methods for the detection of multiple voice disorders: neurological, functional, and laryngeal diseases," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 6, pp. 1820–1828, Nov. 2015, doi: 10.1109/JBHI.2015.2467375.
[14] M. K. Arjmandi and M. Pooyan, "An optimum algorithm in pathological voice quality assessment using wavelet-packet-based features, linear discriminant analysis and support vector machine," *Biomedical Signal Processing and Control*, vol. 7, no. 1, pp. 3–19, Jan. 2012, doi: 10.1016/j.bspc.2011.03.010.
[15] V. Uloza *et al.*, "Categorizing normal and pathological voices: automated and perceptual categorization," *Journal of Voice*, vol. 25, no. 6, pp. 700–708, Nov. 2011, doi: 10.1016/j.jvoice.2010.04.009.
[16] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995, doi: 10.1109/89.365379.
[17] D. Pravena, S. Dhivya, and D. Devi A, "Pathological voice recognition for vocal fold disease," *International Journal of Computer Applications*, vol. 47, no. 13, pp. 31–37, Jun. 2012, doi: 10.5120/7250-0314.
[18] I. Obaid, M. A. Mohammed, M. K. A. Ghani, S. A. Mostafa, F. T.AL-Dhief, "Evaluating the performance of machine learning techniques in the classification of wisconsin breast cancer." *International Journal of Engineering and Technology*, vol 7, 2018, pp. 160-166.
[19] U. Cesari, G. De Pietro, E. Marciano, C. Niri, G. Sannino, and L. Verde, "A new database of healthy and pathological voices," *Computers and Electrical Engineering*, vol. 68, pp. 310–321, May 2018, doi: 10.1016/j.compeleceng.2018.04.008.
[20] H. Beigi, "Speaker recognition," in *Fundamentals of Speaker Recognition*, Springer US, 2011, pp. 543–559.
[21] L. Salhi, T. Mourad, and A. Cherif, "Voice disorders classification using multilayer neural network," in *2008 2nd International Conference on Signals, Circuits and Systems*, Nov. 2008, pp. 1–6, doi: 10.1109/ICSCS.2008.4746953.
[22] Y. Akbulut, A. Sengur, Y. Guo, and F. Smarandache, "NS-k-NN: neutrosophic set-based k-nearest neighbors classifier," *Symmetry*, vol. 9, no. 9, p. 179, Sep. 2017, doi: 10.3390/sym9090179.
[23] D. H. Griffel, *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics (SIAM), 1992.
[24] M. X. Cohen, "A better way to define and describe Morlet wavelets for time-frequency analysis," *NeuroImage*, vol. 199, pp. 81–86, Oct. 2019, doi: 10.1016/j.neuroimage.2019.05.048.
[25] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

BIOGRAPHIES OF AUTHORS

Ranita Khumukcham received the B.E. degree in electronics and communication engineering from Manipur University, Manipur and the M. tech degree in mobile communication and computing from the National Institute of Technology, Arunachal Pardesh. She is currently pursuing Ph.D. at Indian Institute of Information Technology, Manipur. She can be contacted at email: ranitakh89@sgmail.com.

Sharmila Meinam received her B.E. degree in electronics and telecommunication engineering from Assam Engineering College, Assam, India and M.E degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India. She is currently working as an Assistant Professor in Electronics and Communication Engineering Department at Manipur Institute of Technology, Manipur University and pursuing her Ph.D. from Manipur Institute of Technology, Manipur University. She has 23 years of teaching experience and publications in international journals and conferences. Her areas of interest are communications and image processing. She can be contacted at email: sharmila_meinam@yahoo.com.

Kishorjit Nongmeikapam is working as Head of Department (HoD) in the Department of Computer Science and Engineering, Indian Institute of Information Technology (IIIT), Manipur. Has the teaching experience of 21 years. Has obtained the Bachelor degree in computer science and engineering from PSG College of Technology, Coimbatore, India with Master degree and Ph.D. degree from Jadavpur University, Kolkata, India. Has published more than 75 (seventy-five) peer reviewed papers and also authored the book, "See the C programming". Completed multiple R&D projects on driverless vehicle and NLP. Has delivered more than 50 invited talks. He can be contacted at email: kishorjit@iiitmanipur.ac.in.