On the Comparison of Line Spectral Frequencies and Mel-Frequency Cepstral Coefficients Using Feedforward Neural Network for Language Identification

Teddy Surya Gunawan[1], Mira Kartiwi[2]
[1] Department of Electrical and Computer Engineering, Kulliyyah of Engineering
[2] Department of Information Systems, Kulliyyah of ICT
International Islamic University Malaysia

Article Info

Article history:

Received Jan 3, 2018
Revised Mar 5, 2018
Accepted Mar 23, 2018

Keywords:

Language Identification
LSF
MFCC
Feedforward Neural Network
Classifier
Recognition Rate

ABSTRACT

Of the many audio features available, this paper focuses on the comparison of two most popular features, i.e. line spectral frequencies (LSF) and Mel-Frequency Cepstral Coefficients. We trained a feedforward neural network with various hidden layers and number of hidden nodes to identify five different languages, i.e. Arabic, Chinese, English, Korean, and Malay. LSF, MFCC, and combination of both features were extracted as the feature vectors. Systematic experiments have been conducted to find the optimum parameters, i.e. sampling frequency, frame size, model order, and structure of neural network. The recognition rate per frame was converted to recognition rate per audio file using majority voting. On average, the recognition rate for LSF, MFCC, and combination of both features are 96%, 92%, and 96%, respectively. Therefore, LSF is the most suitable features to be utilized for language identification using feedforward neural network classifier.

Corresponding Author:

Teddy Surya Gunawan,
Department of Electrical and Computer Engineering, Kulliyyah of Engineering,
International Islamic University Malaysia,
Jalan Gombak, 53100 Kuala Lumpur, (+603) 6196 4521.
Email: tsgunawan@iium.edu.my, tsgunawan@gmail.com

1. INTRODUCTION

There are about 7105 living languages owned by 6.7 billion populations in this world [1] and these languages definitely differ from each other. Many researches have been conducted in the area of language identification system (LID). A tutorial on LID has been presented in [2] in which syntactic, morphological, and acoustic, phonetic, phonotactic, and prosodic level information have been discussed in details. Around 87 prosodic features has been used for LID system in [3] which provides better recognition performance, while [4] utilizes visual features with error rate less than 10%. In [5], a highly accurate and computationally efficient framework of i-vector presentation is proposed for rapid language identification. A hierarchical LID framework is proposed in [6], in which a series of classification decisions is performed at multiple levels with individual languages identified only at the final level.

Although many researches have been conducted on LID, but most of the researchers are only identifying around two to three languages. Therefore, in this paper, five languages including Arabic, Chinese, English, Korean, and Malay, spoken by both males and females will be analyzed. For LID system, the most used features is Mel-Frequency Cepstral Coefficients (MFCC) and Line Spectral Frequencies (LSF) [7]-[9]. Systematic experiments will be conducted to find the optimum parameters. The combination of both LSF and MFCC features along with various structures of feedforward neural networks will be evaluated. The performance criteria used is mainly the recognition rate, as well as the neural network training time.

2. LANGUAGE IDENTIFICATION SYSTEM

Language identification system contains at least three basic blocks, including preprocessing, feature extraction, and classifier. Preprocessing is a process of speech signal refinement. The raw speech signal that we obtained is not proper to use directly as input. The weak signal that we obtained has to be amplified, removed the longer silence, and also extracted the background noise or music for further processing. There are many feature extractions that can be used for LID system, in order to extract the speech signal from each different speaker of different language, for example Line Spectral Frequencies (LSF), Mel-Frequency Cepstral Coefficients (MFCC), Shifted Delta Cepstra (SDC), Perceptual Linear Prediction (PLP), Dynamic Time Warping (DTW), and Bark Frequency Cepstral Coefficients (BFCC). There are a few classifiers that can used, including Vector Quantization (VQ), Gaussian Mixture Model (GMM), Support Vector Machine (SVM), Ergodic Hidden Markov Model (HMM), K-Means Clustering Algorithm and Artificial Neural Network (ANN) [2]. In this paper, two most popular audio features will be evaluated, including LSF and MFCC, and feedforward neural network will be used as the classifier as shown in Figure 1.

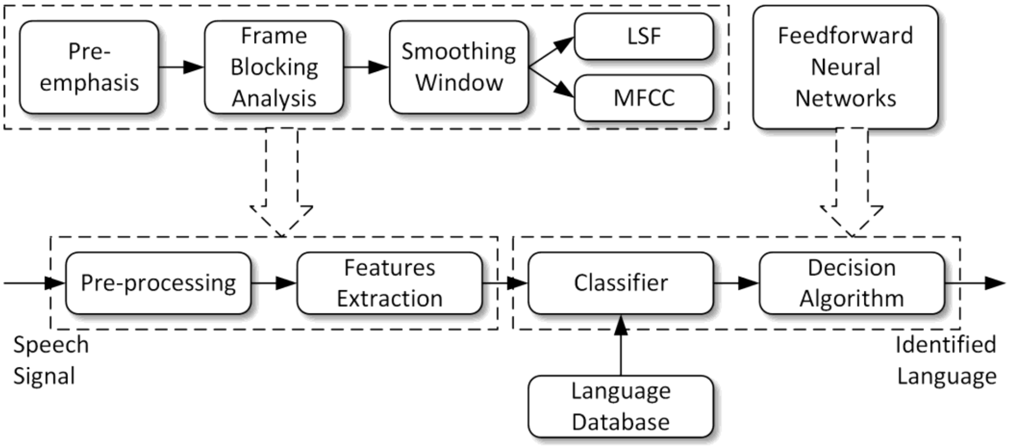


Figure 1. Proposed Language Identification System

2.1. Line Spectral Frequencies

A widely used source-filter model of speech is the linear prediction coefficient (LPC) model. LPC models are used for speech coding, recognition and enhancement. A LPC model with order p can be expressed as shown in Eq. (1).

$$x(n) = \sum_{k=1}^{p} a_k x(n-k) + e(n) \tag{1}$$

where $x(n)$ is speech signal, a_k is the LP parameters and $e(n)$ is speech excitation. Note that, the coefficients a_k model the correlation of each sample with the previous p samples whereas $e(n)$ models the part of speech that cannot be predicted from the past p samples.

The line spectral frequencies (LSF) is an alternative representation of linear prediction parameters. LSFs are used in speech coding, and in the interpolation and extrapolations of LP model parameters, for their good interpolation and quantization properties. LSFs are derived as the roots of the following two polynomials as shown in Eq. (2) and (3).

$$P(z) = A(z) + z^{-(P+1)}A(z^{-1}) \tag{2}$$

$$Q(z) = A(z) + z^{-(P+1)}A(z^{-1}) \tag{3}$$

where $A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k} = 1 - a_1 z^{-1} - \cdots - a_p z^{-p}$ is the inverse liner predictor filter and $A(z) = \frac{1}{2}[P(z) + Q(z)]$. The polynomial equations (Eq. (2) and (3)) can be rewritten in the factorized form as shown in Eq. (4) and (5).

$$P(z) = \prod_{i=1,3,5,\cdots}(1 - 2cos\omega_i z^{-1} + z^{-2}) \tag{4}$$
$$Q(z) = \prod_{i=2,4,6,\cdots}(1 - 2cos\omega_i z^{-1} + z^{-2}) \tag{5}$$

where ω_i are the LSF parameters. It can be shown that all the roots of the two polynomials have a magnitude of one and they are located on the unit circle and alternate each other. Hence, in LSF representation, the liner predictor coefficients $(a_1, a_2, \dots, a_p)$ is converted to LSF vector $(\omega_1, \omega_2, \dots, \omega_p)$. Matlab implementation function lpc() and poly2lsf() were used for this purpose.

2.2. Mel-Frequency Cepstral Coefficients

Mel-Frequency Cepstral Coefficients (MFCC) are computed using a filter bank of M filters ($m = 0,1,\dots,M-1$), each one has a triangular shape and is spaced uniformly on the mel scale using Eq. (6). Each filter is defined as in Eq. (7).

$$f_{mel} = 1127ln\left(1 + \frac{f_{Hz}}{700}\right) \quad (6)$$

$$H_m[k] = \begin{cases} 0 & k < f[m-1] \\ \frac{k-f[m-1]}{f[m]-f[m-1]} & f[m-1] < k \leq f[m] \\ \frac{f[m+1]-k}{f[m+1]-f[m]} & f[m] < k \leq f[m+1] \\ 0 & f \geq f[m+1] \end{cases} \quad (7)$$

The log-energy of mel spectrum is calculated as:

$$S[m] = ln[\sum_{k=0}^{N-1} |X[k]|^2 H_m[k]], \quad m = 0,1,\dots,M-1 \quad (8)$$

where $X[k]$ is the output of discrete Fourier Transform (DFT) of the input signal. Although traditional cepstrum uses inverse discrete Fourier transform (IDFT), MFCC is normally implemented using discrete cosine transform as follows:

$$\hat{x}[n] = \sum_{m=0}^{M-1} S[m] cos\left[\left(m + \frac{1}{2}\right)\frac{\pi n}{M}\right], \quad n = 0,1,\dots,M-1 \quad (9)$$

Typically, the number of filters M ranges from 20 to 40, and the number of coefficients is 13.

2.3. Feed Forward Neural Network Classifier

In artificial neural network, the basic processing unit is a perceptron. A feedforward neural network organizes perceptrons into a layer, cascade these layers into a network, and the connections between layers follow only one direction. The layer that receives connections from the input feature vectors is the input layer, the outermost layer is the output layer which is the classifier output, and the rest of the layers between the input and output layers are called hidden layers. The computation of a feedforward neural network or multilayer perceptron can be described as follows

$$h^{(l)} = \Psi^{(l)}(W^{(l)}.h^{(l-1)} + b^{(l)}) \quad (10)$$

where $h^{(l)}$ is the output vector of layer $l, l = 1,\dots,L$ wherevbnm, L is the number of layers in the neural network. $h^{(0)}$ is the input, while $W^{(l)}$, $b\text{^}(l)$, and $\Psi^{(l)}$ are the weight matrix, the bias vector, and the activation function of layer l. In classification of K classes, the activation function is normally a sigmoid for $K = 2$ or softmax function for $K > 2$.

Given a set of samples $\{(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})\}$ and a feedforward neural network with initial parameters ψ (characterized by weight matrices and bias vectors), we would like to train the neural network so that it can learn the mapping. If we see the whole network as the following function

$$\hat{y} = F(x;\psi) \quad (11)$$

and define some loss function $E(x,y,\psi)$, then the goal of training our network becomes minimizing $E(x,y,\psi)$. The gradient of E indicates the direction to increase E as follows

$$\nabla E(\psi) = \left[\frac{\partial E}{\partial \psi_1}, \dots, \frac{\partial E}{\partial \psi_n}\right] \quad (12)$$

Since the gradient E specifies the direction to increase E, at each step parameters will be updated proportionally to the negative of the gradient

$$\psi_i \leftarrow \psi_i + \Delta\psi_i \tag{13}$$

where $\Delta\psi_i = -\epsilon \frac{\partial E}{\partial \psi_i}$. The training procedure is called gradient descent, and ϵ is a small positive training parameter called learning rate. Cross entropy error is normally used as a loss function

$$E = -\frac{1}{K}\sum_{k=1}^{K} y_k^{(m)} \log \hat{y}_k^{(m)} \tag{14}$$

where m is the index of an arbitrary sample, K is the number of classes, $y_k^{(m)}$ is the k-th column corresponding to the probability of class k of vector $y^{(m)}$. The gradient components of the output layer can be computed directly, while they are harder to compute in lower layers. Normally, the current gradient is calculated using the error of the previous step. Since errors are calculated in the reverse direction, this algorithm is known as backpropagation.

3. RESULTS AND DISCUSSION

This section will discuss the language database preparation, experimental setup, various experiments to find optimum parameters, and the performance evaluation of the proposed LID system.

3.1. Experimental Setup and Language Database

A high performance system was used for processing, i.e. a multicore system with Intel Core i7 6700 K 4.00 GHz (4 cores with 8 threads), 32 GBytes RAM, 256 GBytes SSD and 2 TBytes hard disk, installed with Windows 10 operating system and Matlab 2017b with Signal Processing and Neural Network Toolboxes. During simulation, other running applications were minimized as much as possible.

For the language database preparation, audio file of ten speakers with different language were taken from online language database. There were six males and four females of speakers that will be used as subject for this project. All the speakers were divide into two group for training (four males and one female) and testing (two males and three females) respectively. Besides, each of the speaker spoke different languages and sentences such as Arabic, Chinese (specifically Mandarin), English, Korean and Malay. The database presented in [10] was used with some rearrangement, in which 15 files were used for training and 5 files were used for testing.

3.2. Experiments on Sampling Frequencies, Frame Sizes, Model Orders, and Feedforward Neural Network Structures

There are many parameters which could be optimized to achieve the highest performance, i.e. in terms of language recognition rate. In this paper, several important parameters will be analysed, including F_s (sampling frequency), N_F (frame size), p (model order), and the structure of feedforward neural networks. The structure of feedforward neural networks could be varied in terms of number of hidden layers and number of nodes in each hidden layer. Note that, a 50% overlapping windows was used for both LSF and MFCC feature extraction so that both will have the same number of frames for each audio file. In [8], we used non overlapping window for LSF feature extraction.

Our previous researches have reported that sampling frequency has an effect on the recognition rate [10], while it has negligible effect on the other [8]. Therefore, the first experiment will vary the sampling frequency, i.e. 8000 Hz and 16000 Hz. For this experiment, the other two parameters were fixed as follows, $p = 12$, $N_F = 20$ ms. While the structure of the feedforward neural network was fixed to have one hidden layer with 20 nodes. Table 1 shows the recognition rate versus training time for two sampling frequencies, i.e. 8000 and 16000 Hz. Based on Table 1, the recognition rate for 16 kHz sampling frequency is higher than 8 kHz sampling frequency, especially for LSF features. Therefore, the 16 kHz sampling frequency will be selected as one of the optimum parameter.

Table 1. Experimental Results on Varying Sampling Frequencies

F_s	Training time (s)		Recognition Rate (%)	
	LSF	MFCC	LSF	MFCC
8000	2.76	8.28	63.02	73.86
16000	10.37	4.05	67.78	73.01

The next experiment will evaluate the effect of varying window size (frame size) to the recognition rate. For this experiment, the other two parameters were fixed as follows, $Fs = 16000$, $p = 12$. In addition, the structure of the feedforward neural network was fixed to have one hidden layer with 20 nodes.

Figure 2 shows the results of recognition rate and training time for various frame size from 10 to 100 ms. The red line represents LSF, while the blue line represents MFCC. The square marker represents the recognition rate (see the left axis), while the triangle marker represents the training time (see the right axis). Based on Figure 2, the frame size of 30 ms was selected due to it provides reasonable training time and recognition rate. The frame size of 50 ms was another good candidate, however, larger windows size tends not to capture enough the dynamic of speech signals. One can argue that the neural network training plays more significant role to the recognition rate.

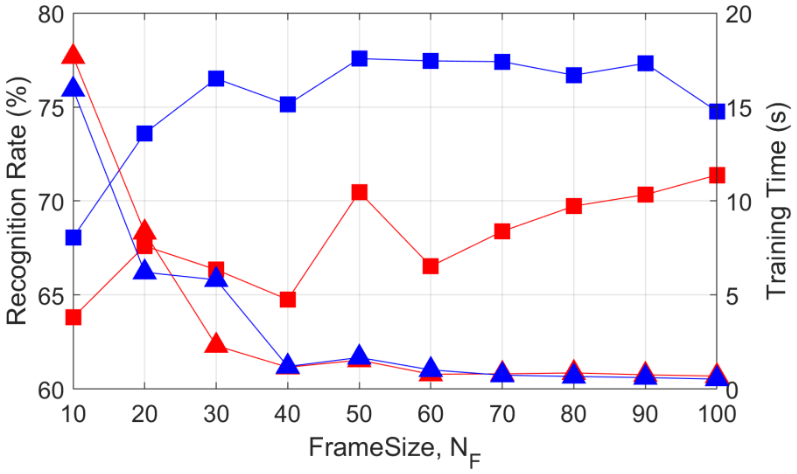


Figure 2. Recognition Rate for Various Frame Sizes N_F

The subsequent experiment will evaluate the effect of varying model order of LPC and MFCC to the recognition rate. For this experiment, the other two parameters were fixed as follows, $Fs = 16000$, $N_F = 20$. In addition, the structure of the feedforward neural network was fixed to have one hidden layer with 20 nodes. Figure 3 shows the results of recognition rate and training time for various model order of LPC and MFCC from 6 to 48 with interval of 2. Based on Figure 3, the model order of 42 was selected as one of the optimum parameter as it provides high recognition rate for both LSF and MFCC. Furthermore, the neural network training time is not that affected by the increment of model order.

The last experiment is regarding the neural network structure configuration. The feedforward neural network with various structure of hidden layer(s) was used. Number of epoch was set to 1000, number of maximum validation fail was set to 100, and the scaled conjugate gradient was used as the training algorithm. Table 2 shows the recognition rate and training time for various structure of neural network, i.e. one hidden layer with a number of nodes $[N_1]$, two hidden layers $[N_1\ N_2]$, and three hidden layers $[N_1\ N_2\ N_3]$. The Matlab patternnet () function was used with hidden layer(s) variation. Note that, our preliminary results using learning vector quantization (LVQ) neural network as in [8] is not as promising as simple feedforward neural network with various hidden layer configuration. Moreover, LVQ requires longer training time as well compared to the simple feedforward neural networks.

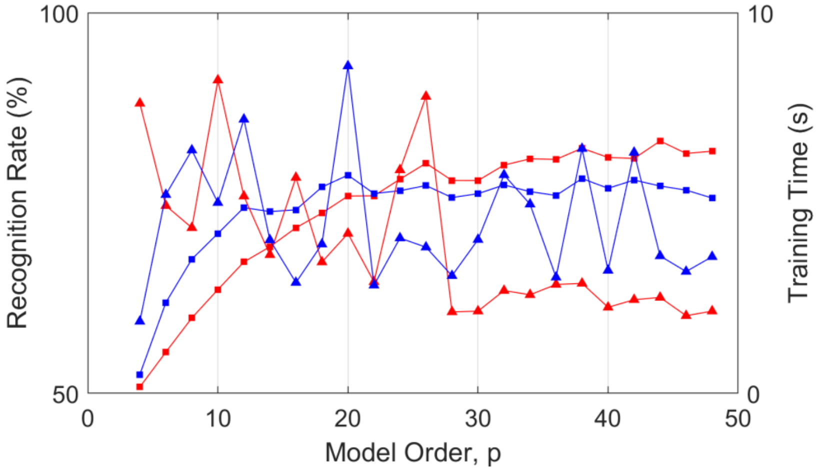


Figure 3. Recognition Rate for Various Model Orders p

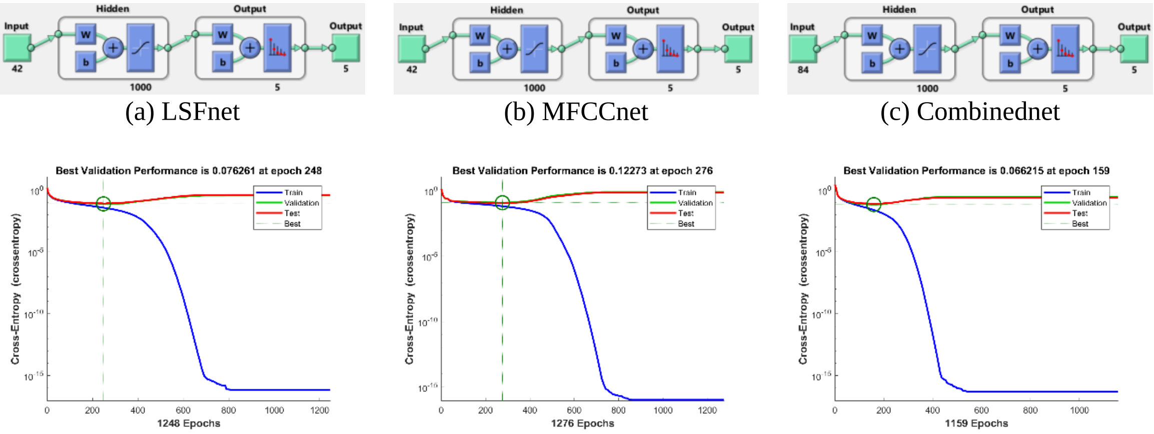
Table 2. Experiments of Feedforward Neural Network Structures

Hidden Layer(s)	Training Time (s)		Recognition Rate (%)	
	LSF	MFCC	LSF	MFCC
[10]	6.45	4.62	76.62	68.62
[20]	2.23	2.36	81.09	73.78
[30]	3.14	3.56	85.1	80.28
[40]	3.16	3.68	84.23	80.93
[50]	3.81	3.66	84.62	79.02
[10 5]	2.93	5.03	76.09	70.64
[20 10]	3	4.29	82.67	79.18
[40 20]	5.65	5.31	89.19	79.76
[40 20 10]	7.64	7.21	86.73	81.87
[100]	6.23	6.48	87.94	80.87
[200]	12.11	10.77	87.77	81.32
[300]	17.47	15.65	89.91	80.48
[400]	25.07	20.32	91.51	82.43
[500]	25.68	28.33	89.71	84.65
[1000]	60.2	55.28	91.13	86.04
[40 40]	6.99	6.26	88.63	82.73
[40 40 20]	8.65	10.7	87.66	86.36
[40 40 40]	10.56	9.58	87.78	84.03
[2000]	120.2	142.46	90.72	89.01
[3000]	201.21	127.03	93.55	80.36
[4000]	421.92	371.81	93.75	89.25
[5000]	356.5	243.19	92.01	69.91
[10000]	872.93	1533.78	88.88	90.33
[15000]	1592.14	1324.68	92.87	84.74
[20000]	2255.15	2724.14	93.41	88.63
[30000]	3234.22	4016.23	90.39	74.67
[40000]	5379.07	3468.15	94.19	71.81
[50000]	4076.94	6283.14	71.43	71.03

From Table 2, it is found that the optimum number of hidden layer is one hidden layer, while the number of nodes is 1000 as highlighted in bold. The neural network structure [1000] provides a high recognition rate with acceptable training time. The other structure, i.e.[3000], is one of the good candidate as well, but the training time is more than three times longer compared to [1000].

3.3. Experiments on the Optimum Parameters on the Training Data

From the previous experiments, the optimum parameters and neural network configuration are $Fs = 16000$ Hz, $N_F = 20$ ms, $p = 42$, and feedforward neural network with [1000] structure of hidden layer. The neural network will be trained using 15 files for each of the 5 languages. Moreover, as the number of frames for LSF and MFCC is now the same as both using 50% overlapping window, we combined both features to evaluate whether the recognition rate is higher or not. In this experiment, to allow longer training time we further change the number of epochs to 1000, maximum validation fail to 1000, and minimum gradient to 10^{-30}. Figure 4 and Table 3 shows the training performance for LSF, MFCC, and combined features. Note that, input layer of Combinednet is the addition of input layer LSF and MFCC, i.e. 84. It can concluded that the combined features will contribute to the recognition rate, while LSF is the dominant feature.



(a) LSFnet (b) MFCCnet (c) Combinednet

(d) LSF Performance (e) MFCC Performance (f) Combined Performance

Figure 4. Neural Network Structure and its Performance for LSF, MFCC, and Combination of LSF and MFCC

Table 3. Performance for LSF, MFCC, and Combined Features

Feature	Training Time (s)	NEpochs	MSE	Recognition Rate (%)
LSF	239	1248	7.04e-17	91.58
MFCC	242	1277	1.06e-16	85.53
Combined LSF and MFCC	259	1160	4.89e-17	93.77

3.4. Experiments on the Testing Data

The last experiment would be to evaluate the trained neural network on the unknown or testing data, i.e. 5 files for each languages. We have trained the feedforward neural network to classify the current frame into 5 trained languages. At the end, we need to decide the identified language for the whole file and not the current frame. For this purpose, we utilized the majority voting rule as explained in [11], in which the identified language is the majority voting in that particular file. Table 4 shows the recognition rate for each language per frame and per file after majority voting. Note that, although it has lower recognition rate per frame but sometime it as 100% recognition rate when it is calculated per file using majority voting, vice versa.

The detailed analysis revealed that for English language, it has been wrongly classified as Malay language for 1 file and 2 file using LSF and MFCC, respectively. For Malay language, it has been wrongly classified as Malay language for 1 file using combined LSF and MFCC. Interestingly, the combined features mostly improved the recognition rate except for the Malay language. Further experiment is required with additional database, especially for English and Malay language to validate the obtained results. From the average of recognition rate, it has been found that using LSF features alone is sufficient for language identification.

Table 4. Recognition Rate for Each Language on the Unknown/Testing Data

Language	Recognition Rate (%) Per Frame			Recognition Rate (%) Per File		
	LSF	MFCC	Combined	LSF	MFCC	Combined
Arabic	58.16	55.21	60.49	100	100	100
Chinese	79.65	43.28	73.81	100	100	100
English	65.39	46.08	65.58	80	60	100
Korean	68.34	67.05	64.47	100	100	100
Malay	65.68	56.79	69.11	100	100	80
Average	*67.44*	*53.68*	*66.69*	*96*	*92*	*96*

4. CONCLUSION AND FUTURE WORKS

In this paper, two popular features for language identification, i.e. LSF and MFCC, have been compared and evaluated. Language identification system using feedforward neural network has been trained on five different languages, i.e. Arabic, Chinese, English, Korean, and Malay. Systematic experiments have been conducted to obtain the optimum parameters, i.e. sampling frequency, frame size, model order, and structure of neural network. The optimum parameter obtained were $Fs = 16000$ Hz, $N_F = 20$ ms, $p = 42$, and feedforward neural network with one hidden layer and 1000 hidden nodes. On average, the recognition rate for LSF, MFCC, and combination of both features are 96%, 92%, and 96%, respectively. Results showed that LSF alone is the most suitable feature for language identification using feedforward neural network classifier. Further research includes using extensive database, deep neural network for feature extraction and classifier, or use different audio features and different classifiers.

ACKNOWLEDGEMENT

The authors would like to express their gratitude to the Malaysian Ministry of Higher Education (MOHE), which has provided funding for the research through the Fundamental Research Grant Scheme, FRGS15-194-0435.

REFERENCES

[1] M. P. Lewis, *et al.*, “Ethnologue: Languages of the world,” SIL international Dallas, TX, vol. 16, 2009.

[2] E. Ambikairajah, *et al.*, “Language identification: A tutorial,” *IEEE Circuits and Systems Magazine*, vol. 11, pp. 82-108, 2011.

[3] R. W. Ng, *et al.*, “Spoken Language Recognition With Prosodic Features,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 1841-1853, 2013.

[4] J. L. Newman and S. J. Cox, “Language identification using visual features,” *IEEE Transactions on audio, speech, and language processing*, vol. 20, pp. 1936-1947, 2012.

[5] M. V. Segbroeck, *et al.*, “Rapid language identification,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 23, pp. 1118-1129, 2015.

[6] S. Irtza, *et al.*, “A hierarchical framework for language identification,” in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 5820-5824, 2016.

[7] K. Simonchik, *et al.*, “Comparative Analysis of Classifiers for Automatic Language Recognition in Spontaneous Speech,” in *International Conference on Speech and Computer*, pp. 174-181, 2016.

[8] T. S. Gunawan, *et al.*, “Development of Language Identification System using Line Spectral Frequencies and Learning Vector Quantization Networks,” *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 9, pp. 21-27, 2017.

[9] T. M. H. Asda, *et al.*, “Development of Quran Reciter Identification System Using MFCC and Neural Network,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 1, pp. 168-175, 2016.

[10] T. S. Gunawan, *et al.*, “Development of Language Identification System using MFCC and Vector Quantization,” in *Proceeding of 4th IEEE International Conference on Smart Instrumentations, Measurement, and Aplications (ICSIMA) 2017, Putrajaya*, pp. 1-4, 2017.

[11] T. S. Gunawan, *et al.*, “Higher-Order Statistics and Neural Network Based Multi-Classifier System for Gene Identification,” in *Proceedings of International Conference on Signal Processing and Communication Systems (ICSPCS 2007), Australia*, pp. 1-7, 2007.