A classification of marked hijaiyah letters' pronunciation using hidden Markov model

Cite as: AIP Conference Proceedings **1867**, 020036 (2017); https://doi.org/10.1063/1.4994439
Published Online: 01 August 2017

Untari N. Wisesty, M. Syahrul Mubarok, and Adiwijaya

A Classification of Marked Hijaiyah Letters' Pronunciation Using Hidden Markov Model

Untari N. Wisesty[a)], M. Syahrul Mubarok[b)], Adiwijaya[c)]

School of Computing, Telkom University
Bandung 40257, Indonesia

[b)]Corresponding author: msyahrulmubarok@telkomuniversity.ac.id
[a]untarinw@telkomuniversity.ac.id, [c]adiwijaya@telkomuniversity.ac.id

Abstract. Hijaiyah letters are the letters that arrange the words in Al Qur'an consisting of 28 letters. They symbolize the consonant sounds. On the other hand, the vowel sounds are symbolized by harokat/marks. Speech recognition system is a system used to process the sound signal to be data so that it can be recognized by computer. To build the system, some stages are needed i.e characteristics/feature extraction and classification. In this research, LPC and MFCC extraction method, K-Means Quantization vector and Hidden Markov Model classification are used. The data used are the 28 letters and 6 harakat with the total class of 168. After several are testing done, it can be concluded that the system can recognize the pronunciation pattern of marked hijaiyah letter very well in the training data with its highest accuracy of 96.1% using the feature of LPC extraction and 94% using the MFCC. Meanwhile, when testing system is used, the accuracy decreases up to 41%.

Keywords: Linear Predictive Coding, Mel Frequency Cepstral Coefficients, Hidden Markov Model, Hijaiyah Letters' Pronunciation.

INTRODUCTION

Hijaiyah letters are the letters that arrange the words in Al Qur'an. Hijaiyah letters consist of 28 letters symbolizing consonant sounds. Meanwhile, the vocal sounds are symbolized by harokat/marks. Harokat or marks are used to clarify the pronunciation of the letters. The first step to learn Al Quran is by reading and understanding each hijaiyah letter. After understanding the pronunciation of the letters, the next step of learning Al Quran in well and appropriate ways can be realized according to the rules. Speech recognition is a system used to process sound signal into data so that it can be recognized by computer [1]. The extracted sound signal can then result in analyzable information for every sound signal variation. From each character of the phoneme, the recognition is tried and it is changed into text [2]. To build the speech recognition system, some methods are needed in order to gain maximum result. The basic method used is feature extraction. Feature extraction is a feature taking from a form whose gained value will be analyzed for the further process. There are some frequently used feature extraction methods i.e Linear Predictive Coding (LPC), Mel Frequency Cepstral Coefficient (MFCC), and Human Factor Cepstral Coefficient (HFCC). LPC is the strongest sound analysis technique and one of the most useful methods to codify the sounds with good quality in low bit rate. LPC is a model made based on human sound production. This system uses the conventional filter model in which glottal, sound channel and mouth radiation removal function are integrated to a filter stimulating the sound channel with sound science. MFCC is a method used to do the feature extraction. This method adopts the mechanism of human hearing organs so that it is able to recognize the most important sound characteristics [3].

In Untari's research [4] a testing of LPC, MFCC and Probabilistic Neural Network with Indonesian city names pronunciation has been conducted. PNN (Probabilistic Neural Network) is a deterministic method used as classifier of the system so that the feature extraction can gain maximum result. Based on the result, 100% accuracy is gained

International Conference on Mathematics: Pure, Applied and Computation
AIP Conf. Proc. 1867, 020036-1–020036-11; doi: 10.1063/1.4994439
Published by AIP Publishing. 978-0-7354-1547-8/$30.00

with number of classes of 2,3 meanwhile the lowest accuracy of 73% is gained with 10 classes. After feature extraction is done, classifier method is needed for training and recognition in order to result in maximum output in the form of expected text by using Hidden Markov Model (HMM). Hidden Markov Model is an easy-to-apply method and has training algorithm to predict the parameter of one sound data set [5]. This method is a quite famous method used in speech processing problem. The basic problem of HMM is evaluation, decoding and training [1, 6]. In this paper, an analysis on feature extraction method will be conducted using LPC, MFCC and HMM to classify hijaiyah sound data into 168 classes.

LITERATURE REVIEW

Marked Hijaiyah Letter Pronunciation

Hijaiyah letters or Arabic alphabets are the letters that arrange words in Al Quran. Hijaiyah letters consist of 28 letters symbolizing consonant sound while the vocal sounds are symbolized by harokat/marks. Harokat or marks are used to clarify the letter pronunciation. There are 8 types of harokat i.e fathah, kasroh, dhomah, fathahtain, kasrohtain, dhomahtain, sukun, and tasydid. In this paper, a system will be made. The system will recognize 28 letters and 6 harokat i.e fathah, kasroh, dhomah, fathahtain, kasrahtain, and dhomahtain. The gained data are resulted from sound recording conducted by 6 people i.e 3 males and 3 females. The recording is conducted in a closed and soundproof room. Each letter is pronounced by everyone four times in .wav format of 16 bit channel and sample rate of 441000 Hz as well as Mono channel. Next, sound trimming is done suitable with the needed database. So, one file with .wav extension will only consist of one letter and harokat pronunciation. This stage is called sound trimming and is done using Audacity.

Speech Recognition System

Speech recognition system is a technology used to recognize and understand the words pronounced and then change them into data representation recognized by computer using digitalization ways and match the digital signal with certain pattern. The pronounced words' form are changed into digital signal by changing sound wave into collection of numbers which are suited with certain codes to identify the words. The result of pronounced words identification can be presented in the form of writing [5]. The information in the actual sound signal is actually represented by short term amplitude spectrum in the form of sound wave. This enables to extract characteristics based on short term amplitude from the speaker's spectrum (phoneme). In making a sound recognition system, a method is needed to conduct characteristics extraction (feature extraction) and a method for classification (classifier). The purpose of speech recognition system is to convert sound signal into text form suitable with the pronounced words, not depending on the used device to record [7]. The first step done is normalization.

Normalization

Normalization is a process that must be done. It is to equalize all the sound signals into system feedback in order to be compatible with the system specification. In this paper, normalization is only conducted in the stage of preprocessing. The purpose of this process is to equalize maximum amplitude and sound signal sample rate so that there will not be any change in amplitude and sample rate in the next processing. The normalization process is begun by inputting the input sound. Next, resampling is done to be 16kHz and continued with centering process which is aimed to move the discrete amplitude distribution location on the axis of y=0 that also enables sound signal amplitude to have the mean =0. After that, dividing amplitude value with maximum amplitude value is conducted. Next, the normalized sound is gained. It will be used in the next process.

Feature Extraction

The characteristics extraction or feature extraction is a way of important feature taking in a signal whose gained value will be analyzed for the next process. In this paper, the feature extraction method used is Linear Predictive Coding (LPC) and Mel Frequency Cepstral Coefficient (MFCC).

Linear Predictive Coding (LPC)

LPC is the strongest sound analysis technique and one of the most useful methods to codify sound with good quality in low bit rate. The following is the LPC process diagram.

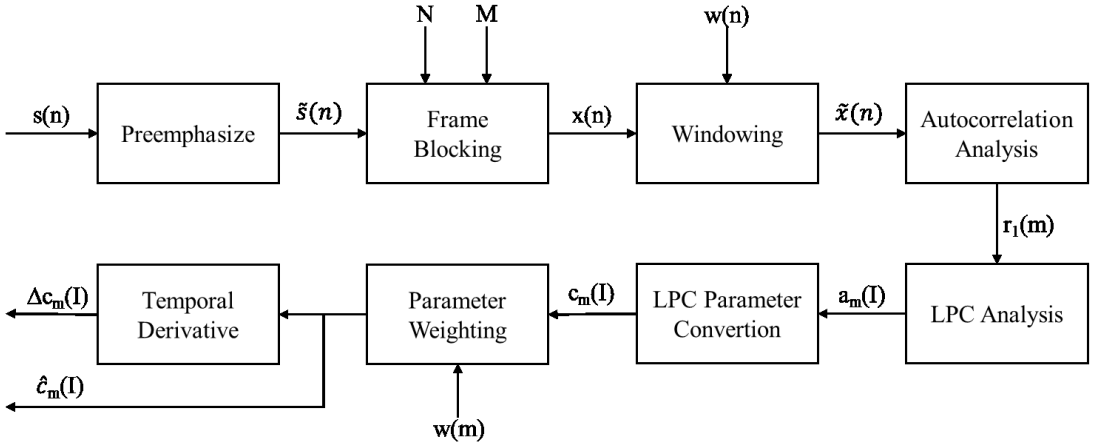

FIGURE 1. LPC Diagram Block [6].

Mel Frequency Cepstral Coefficient (MFCC)

Mel-Frequency Cepstral Coefficient (MFCC) is the most frequently used method for feature extraction in speech recognition. MFCC will convert sound signal into vector. MFCC results in big data so it needs long time. The stages in MFCC are Pre-Emphasis, Frame Blocking, Windowing, Fast Fourier Transform, Mel Frequency Warping, Discrete Cosine, Transform, and Cepstral Liftering [8].

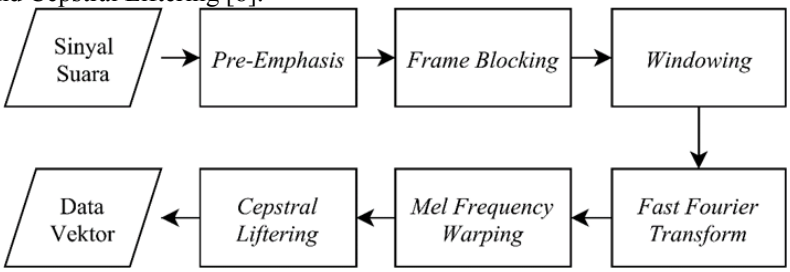

FIGURE 2. MFCC Diagram Block.

Vector Quantization

Vector Quantization is needed to create observation row (codebook index) which will then be processed using HMM (Hidden Markov Model) for training process [2]. Codebook is a group of dots (vectors) representing sound signal representation in digital form in sound room [1]. In the case of sound recognition, the codebook should be made for the sound to be recognized using the clustering method on the sound feature. The clustering will produce the sound having the similarity to be located in the same group. There are various types of clustering. In this paper, clustering algorithm used is K-means clustering. Meanwhile, the codebook index is gained during the training and testing by using the vector characteristics to be codebook index with the smallest Euclidean distance [1].

Hidden Markov Model

Codebook Index Row resulted in vector quantization process is used as input for HMM training process. This index can be called as HMM observation symbol. In this training process, it will be used to model several letters to be used later in testing process. Training is the most difficult HMM problem [1,6,9]. In this training process Baumwelch algorithm is used. HMM model resulted in this training process is $\lambda = (A, B, \pi)$, the notes from the result is A as inter-

state transition opportunity matrix, B as observation symbol opportunity matrix, and π initial state opportunity. The type of HMM used in this paper is ergodic discrete.

The HMM parameter like A, B and π is raised randomly with normalized value into one [1, 6, 9]. Next, the parameters are repredicted until optimum value is gained in the training process. For other HMM parameters, they are taken based on the number of states (N) and the number of each state observation symbols (M). In HMM, the hidden feature is the state where the state is the sound type itself. Meanwhile, the part to be observed is the feature of sound signal [6]. HMM parameter that is re estimated is the result of implementation of Baumwelch algorithm or Expectation Maximum (EM). The result of HMM parameter re estimation is the new value of A, B and π matrix elements. Iteration on the HMM parameter re estimation stops if the iteration is already maximum, or if the new model does not give meaningful value correction. Before solving the re estimation problem, variable forward (α) and backward $(\beta)$are calculated.

After HMM initialization parameter model of $\lambda = (A, B, \pi)$is done, raised randomly with the normalized value into one, then the calculation of $\alpha_t(i)$ and $\beta_t(i)$ with forward and backward algorithm. The calculation of $a_t(i)$ can be done inductively with three stages using forward algorithm that is [1,6,9,11]:

1. Initialization

$$\alpha_1 = \pi_i b_i(O_1), \qquad 1 \le i \le N \qquad \text{i)}$$

2. Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{n} \alpha_t(i)\alpha_{ij}\right] b_j(O_{t+1}) \qquad (2)$$

Where: $1 \le t \le T-1$
$1 \le j \le N$

3. Termination

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \qquad (3)$$

Then, $\beta_t(i)$ can be calculated using backward algorithm as follows:

1. Initialization

$$\beta_T(i) = 1, \qquad 1 \le i \le N \qquad (4)$$

2. Induction

$$\beta_t(i) = \sum_{j=1}^{N} \alpha_{ij}\, b_j(O_{t+1})\beta_{t+1}(j) \qquad (5)$$

Where: $t = T-1, T-2, \dots, 1, 1 \le i \le N$

The thing that should be taken into consideration is the calculation using forward and backward algorithm can result in very small value, for example for $\alpha_t(i)$ tends to be 0 exponential when T increases [6,9]. This must be considered because the very small parameter value can cause the value considered 0 and if it is used as dividing factors can result in very big value. [1]. Therefore, scaling is used on the parameters calculation in HMM model so that new algorithm is gained to calculate $\alpha_t(i)$ and $\beta_t(i)$, i.e. by adding scale factors. Scale factor (C_t) used is [1, 6, 9, 11]:

$$C_t = \frac{1}{\sum_{i=1}^{N} \hat{\alpha}_t(i)} \qquad (6)$$

After that, HMM parameter re estimation is done to gain a new model to replace the previous model. The value of A, B and π can be estimated using this formula:

$$\pi_i = \frac{\hat{\alpha}_1(i)\hat{\beta}_1(i)}{\sum_{j=1}^{N} \hat{\alpha}_t(j)} \qquad (7)$$

$$\alpha_{ij} = \frac{\sum_{t=1}^{T-1} \hat{\alpha}_t(i)a_{ij}b_j(O_{t+1})\hat{\beta}_{t+1}(j)}{\sum_{t=1}^{T-1}\sum_{j=1}^{T} \hat{\alpha}_t(i)a_{ij}b_j(O_{t+1})\hat{\beta}_{t+1}(j)} \qquad (8)$$

$$b_j(k) = \frac{\sum_{t=1}^{T-1} \hat{\alpha}_t(i)\hat{\beta}_t(i)\delta(O_t, V_k)}{\sum_{t=1}^{T-1} \hat{\alpha}_t(i)\hat{\beta}_t(i)} \qquad (9)$$

Notes:

N : State numbers

M: The number of observation symbol of each state

a : Interstate transition opportunity distribution

b : Observation symbol opportunity distribution
π: First state distribution
O: Observation
t: The number of orders

The result of HMM parameter reestimation will stop if the iteration is already maximum or can use threshold of minimum improvements (the new model cannot make a change in the previous model).

After reestimation process is finished, the system will save $\lambda = (A, B, \pi)$ model. The gained model will be used in testing. The number of models saved later will be the same as the recognized letter numbers.

RESEARCH METHOD

In this paper a system will be designed to recognize hijaiyah letter consisting of 28 letters and 6 harokat based on human sound signal input. Generally, there are two processes done by the system i.e. training and testing. In training process, a training to be conducted is the one with sound sample inputted so that sound model and used in testing process. In testing process where the system can recognize one of several sound inputs, the training and testing process are generally the same. The difference is the input and output of the two processes. The training process requires several sound sample inputs in which training will be conducted in the system based on the sound input to create a model. This gained model will be used in sound recognition in testing process. Meanwhile in system testing process only one or several sound inputs are required. The sound input will then be recognized by the system using maximum likelihood value search in the model created in training process and output from the testing process is a letter that is the most suitable with the input. The following is a diagram block of the developed system.

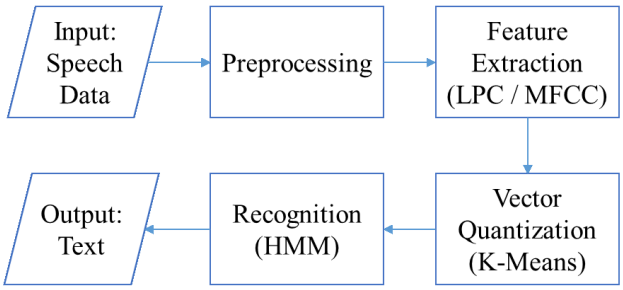

FIGURE 3. Diagram Block of Hijaiyah Letter Pronunciation Recognition System

Based on diagram block in Figure 3, pseudo code of each recognition system process that is developed are as follows:

1. The data used as input are the 28 hijaiyah letter and 6 harokat pronunciation and then division of data conducted of about 70% for training data and 30% for testing data.

Preprocessing:

2. Resampling of sound signal suitable with the needs.
3. Do the centering process, aimed to move the location of discrete amplitude distribution on y axis =0 with the following equation.

$$x(n) = x(n) - \bar{x}(n) \tag{10}$$

4. Do the sound signal normalization to equalize range value of all data with the following equation.

$$x(n) = \frac{x(n)}{\max(|x(n)|)} \tag{11}$$

Feature Extraction, feature extraction method used is LPC and MFCC and then will be compared to the performance after the classification process.

5. Calculate the filter value from pre-emphasize.

$$h(z) = 1 - a\,z^{-1} \qquad 0 \le a \le 1 \tag{12}$$

Where:

a = pre-emphasize filter Constant

$$\hat{x}(n) = x(n) - a\,x(n-1) \tag{13}$$

Where:

$\hat{x}$ (n) = output signal
x(n) = input signal

6. Trimming the sound into frames with short duration (frame blocking).

$$n = \frac{l-k}{s} \tag{14}$$

where:

n: The number of frames
l: The length of sound
k: The overlap length
s: Inter-frame distance

7. Calculate the windowing function with hamming window:

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1}) \quad 0 \leq n \leq N-1 \tag{15}$$

where:

N = The number of samples in each frame
s(n) = Output signal
x(n) = Input signal
w(n) = hamming windows

8. Calculate the signal windowing value based on function result of hamming window:

$$S(n) = x(n) \times w(n) \quad 0 \leq n \leq N-1 \tag{16}$$

Feature extraction if the LPC is used (steps 9-14):

9. Calculate the autocorrelation coefficient value of the signal resulted from windowing:

$$r_l(m) = \sum_{n=0}^{N-1-m} \hat{x}_1(n)\hat{x}_1(n+m) \quad m = 0,1,2,\dots,p \tag{17}$$

Where:

$r_l(m)$ = autocorrelation coefficient to-m on frame to-t
p = LPC analysis order

10. Calculate LPC analysis coefficient.

$$a_m = a_m^{(p)} \quad ; 1 \leq m \leq p \tag{18}$$

11. Convert the LPC analysis result to be LPC coefficient set:

$$c_o = l \text{ x } n \text{ x } \sigma^2 \tag{19}$$

$$c_m = a_m + \sum_{k=1}^{m-1}(\frac{k}{m})\, c_k a_{m-k} \quad 1 \leq m \leq p \tag{20}$$

$$c_m = \sum_{k=1}^{m-1}(\frac{k}{m})\, c_k a_{m-k} \quad m > p \tag{21}$$

Where:

c_m = Cepstral Coefficient
σ^2 = LPC model gain

12. Calculate the window value especially for parameter weighting.

$$w_m = 1 + \frac{Q}{2}\sin(\frac{\pi m}{Q}) \quad 1 \leq m \leq Q \tag{22}$$

Do the parameter weighting in the cepstral coefficient set.

$$\hat{c}_m = w_m c_m \quad 1 \leq m \leq Q \tag{23}$$

Where:

Q = cepstral coefficient set order
$\hat{c}_m$ = cepstral coefficient of weighting result
w_m = windowing value

13. Differentiate the cepstral coefficient set on time.

$$\Delta\hat{c}_m(t) = \frac{\partial \hat{c}_m(t)}{\partial t} \tag{24}$$

14. Feature of characteristics extraction result with LPC is gained.

Feature extraction if using MFCC (steps 15-18):

15. Change signal data from time domain to frequency domain using Fast Fourier Transform Method.

$$f(n) = \sum_{K=0}^{N-1} Yke^{-2\pi jkn/N} \tag{25}$$

where:

N = Sample number
n = Sample index
Y = Signal on frequency domain

16. Do the Mel Frequency Warping using the following equation.

$$\text{Mel(f)} = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (26)$$

where:

f = linear frequency (Hz)
Mel(f) = Mel-frequency scale

17. Do the cepstral liftring using Discrete Cosine Transform to gain signal cepstrum of Mel-Frequency Cepstral Coefficient (MFCC).

$$\text{Cep}_s\,(n; m) = \sum_{j=1}^{N} \log(\text{fmel}_j) \cos\left(\frac{k(j-1)\pi}{2N}\right) \quad (27)$$

Where:

N = sample numbers
k = cepstral coefficients
fmel = mel frequency

18. Feature characteristics extraction with MFCC is gained.

Vector Quantization, using K-means:

19. Decide the dot/vector as many as c cluster, where each dot/vector representing a centroid.
20. Input each object (record data) into cluster, choose the cluster having the closest distance with the centroid. The calculation of distance using Eucledian distance function:

$$J = \sum_{j=1}^{K} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \quad (28)$$

Where:

J = objective function
K = cluster numbers
n = data numbers
$x_i^{(j)}$ = data to-i
c_j = centroid from cluster to-j

21. After all objects enter the group, calculate the new centroid dot/vector by calculating average of value of group members.
22. Repeat steps 20 and 21 until the lowest centroid dot/vector does not change.
23. The cluster numbers (observation symbol) formed will be the input in HMM process.

Classification, using Hidden Markov Model:

24. Initialization of HMM $\lambda = (A, B, \pi)$ that is raised randomly.
25. Calculate the value of $a_t(i)$ using forward algorithm:

$$\alpha_1 = \pi_i b_i(O_1), \qquad 1 \le i \le N \quad (29)$$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(O_{t+1}) \quad (30)$$

26. Calculate the value of $\beta_t(i)$using backward algorithm:

$$\beta_T(i) = 1, \qquad 1 \le i \le N \quad (31)$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \quad (32)$$

27. The value of A, B and π can be estimated using this formula:

$$\pi_i = \frac{\hat{a}_1(i)\hat{\beta}_1(i)}{\sum_{j=1}^{N} \hat{a}_t(j)} \quad (33)$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \hat{a}_t(i) a_{ij} b_j(O_{t+1}) \hat{\beta}_{t+1}(j)}{\sum_{t=1}^{T-1} \sum_{j=1}^{T} \hat{a}_t(i) a_{ij} b_j(O_{t+1}) \hat{\beta}_{t+1}(j)} \quad (34)$$

$$b_j(k) = \frac{\sum_{t=1}^{T-1} \hat{a}_t(i) \hat{\beta}_t(i) \delta(O_t, V_k)}{\sum_{t=1}^{T-1} \hat{a}_t(i) \hat{\beta}_t(i)} \quad (35)$$

28. Reestimate the value of A, B and π until convergent.
29. Save the process result model of Hidden Markov Model.

For the testing process, there will be process 2 until 23 and further:

30. Matching the vector with saved model by calculating likelihood log for every model.
31. System output in the form of hijaiyah letter pronunciation introduction.

RESULTS AND DISCUSSION

Conducted experiment is aimed to analyze performance of the feature extraction and classification method used in building the system of hijaiyah letter pronunciation introduction. The variables to be tested include the division of data class, sample rate, codebook and state. In each of the testing two methods of feature extraction i.e LPC and MFCC as well as Hidden Markov Model classification method. After that, the performance of the testing is measured using accuracy and running time.

Data Class Division Experiment

In the first analysis, several scenarios are done to process the data that have big class i.e 168 classes. In scenario 1 all sounds are put into one group of 168 classes. The data is then divided into training data and testing data. For the scenario 2 the data are divided into two groups i.e. the first group consists of pronunciation with fathah, kasroh and dhomah marks. Meanwhile, for the second group, it consists of the pronunciation of fathatain, kasrohtain and dhomahtain marks. In the scenario two, every group consists of 84 classes. For the scenario 3, the data is divided into 6 groups based on the marks and each group consists of 28 classes. The following is the observation result of data division on system accuracy:

TABLE 1. Testing Result of Data Class Division on System Accuracy

Number of Classes	LPC		MFCC	
	Training	Testing	Training	Testing
168	96.10%	40.18%	94.00%	41.00%
84	97.45%	44.94%	95.70%	42.50%
28	99%	51%	98.37%	47.52%

From the above three scenarios, it can be concluded that the data processing and the decision of the number of sound data has influence on the accuracy gained by the system. When the data are divided into several groups, the accuracy gained will be higher. Meanwhile, if the data are directly classified into 168 classes then the testing accuracy can decrease. This is possible because there is a model having similarity in cepstral way so that the system is wrong in doing the classification. The solution to gain good accuracy with big data is by adding audio file in each model during the training. This is because the more audio file data means the more observation row during the training. As a result, the re estimation process of HMM parameter will be better and create an optimum model.

Sample Rate Testing

Sample rate shows the vast signal values taken from one second range when the sampling process during the sound recording is done. The higher value of sample rate, the more detail the sound is represented which means the better sound quality. In this testing, there will be several different sample rates to see the effect of the sample rate to the performance of system on accuracy value. The following is the system measurement using several different sample rates.

From figure 4, it can be observed that the testing using 16000 Hz sample rate gains the highest accuracy value compared to the other sample rate using either LPC or MFCC feature extraction. This is because 8000 Hz sample rate has not been able to represent sound characteristics. In addition, 44100 Hz sample rate that theoretically creates better sound during the testing does not get higher accuracy compared to the testing using 16000 Hz sample rate. That occurs because the audio file with high sample rate will have bigger data size and the saved bit value is greater so that there are several redundant data sizes. Therefore, the high sample rate usage makes the computer calculate more bit value which makes it difficult to model one sound.

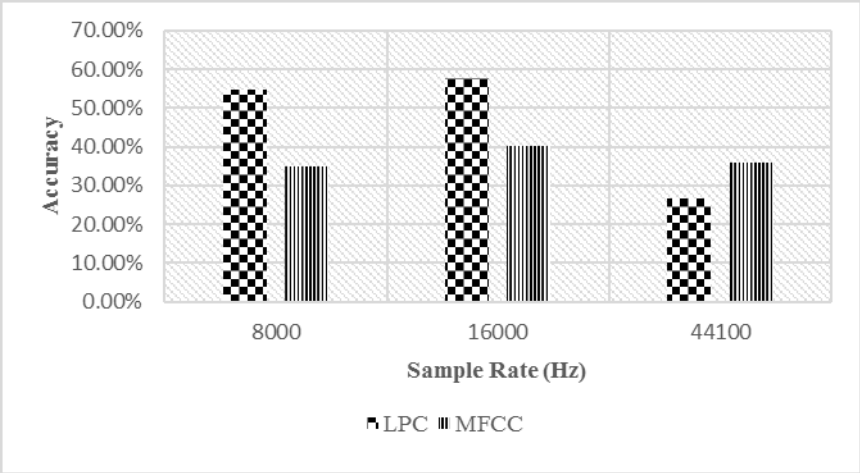

FIGURE 4. Testing Result of Sample Rate on System Accuracy.

Codebook Size and Number of State Experiment

In the system design made, K-means clustering and Hidden Markov Model classification methods are used. Therefore, the size of the codebook and the number of states becomes the parameter really influencing the system performance. The used parameter during the clustering will influence the number of resulted codebooks. Additionally, the number of states used in Hidden Markov Model has a role during the system testing. In the current testing the used codebooks are 16, 32, 64 128, 256 and the number of states are 3, 4, 5, 6, 7 as well as the number of classes is 168. The following is the testing result of influence of codebook size on system accuracy:

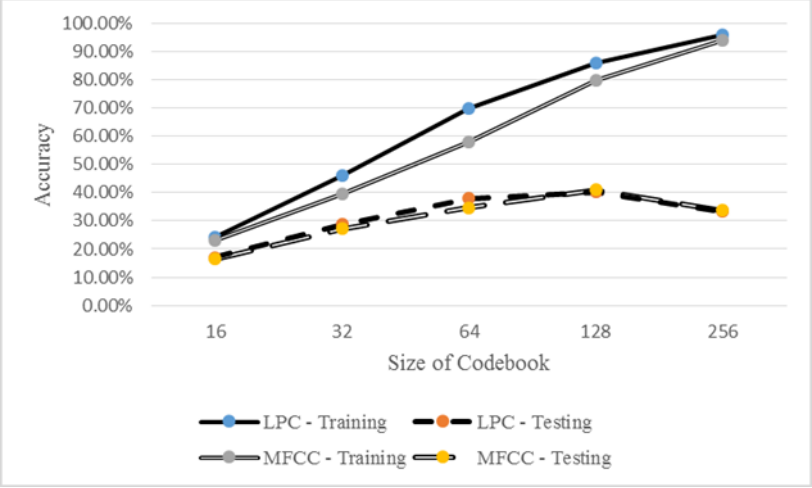

FIGURE 5. Testing Result of Codebook Size on System Accuracy

Based on figure 5, it can be seen that when the size of codebooks is higher, then the accuracy becomes higher too, either using LPC or MFCC feature extraction. However it is also important to note that if the size of codebooks is too big, then the codebooks become bad. This is because there is a possibility that several dot/vectors, supposed to be in one cluster, are separated due to many limitations [1].

It can be seen in above figure that 256 codebooks tend to create decreased accuracy. Meanwhile, the accuracy of training, the value resulted always increases. This occurs because the system can recognize the letter very well. Besides the codebook, the number of states also influences the accuracy level as presented in the following figure:

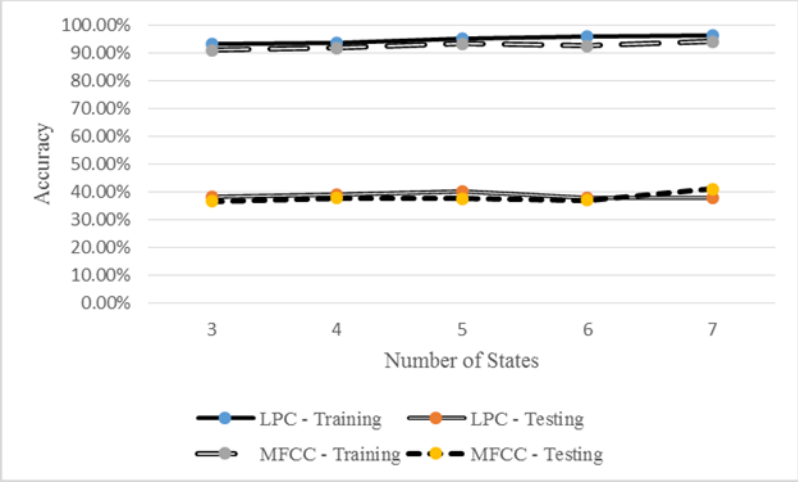

FIGURE 6. Testing Result of State Numbers on System Accuracy.

Based on figure 6, it can be seen that the number of states does not really influence the accuracy resulted by the system both in testing accuracy and training accuracy. The increase on resulted accuracy value is not really big. This shows that the number of states does not really influence the system accuracy. Based on several testing, it can be concluded that if the codebook size is higher, then the testing accuracy tends to increase. However, if the codebook size is too large, then the accuracy tends to decrease. This is because there is a possibility that several dot/vectors, supposed to be in one cluster, are separated due too many limitations [1]. However, the state increase does not really influence the accuracy resulted by the system. It is different from the training accuracy. Training accuracy always increases on several testing above. This shows that the system made can recognize training data very well.

CONCLUSION

Based on several conducted testings, it can be concluded that the system can recognize the pattern of punctuated hijaiyah letter pronunciation very well in data training with the highest accuracy of 96.1% using LPC feature extraction and 94% using MFCC. However, when testing system is used the accuracy decreases into 41%. This is because of the number of classes in punctuated hijaiyah letter is quite large i.e 168 classes yet the provided data are still few. This is possible since there is a model having the similarity in cepstral way so that the system was wrong in doing the classification. In the sample rate testing conducted in sound signal resampling, it can be concluded that the best sample rate is 16 KHz. If the sample rate is too small, then the model cannot represent sound signal characteristics, while if the sample rate is too big makes the sound representation with redundant values. The other influencing parameter is the size of codebook in the time of vector quantization and HMM state number. The biggest the size of the codebook the highest accuracy gained. However, if the codebook size is too big, then the accuracy of the system is decreased.

This occurs because there is a possibility that some dot/vectors, that are to be in one cluster, are separated due to many limitations. Meanwhile, for the state improvement it does not really influence the resulted accuracy by system. In the testing the best codebook size is gained i.e 128 and state number of 5. The method of LPC feature extraction and MFCC averagely results in the same accuracy pattern in every testing. However, LPC has higher accuracy average and more time efficient process.

ACKNOWLEDGMENTS

The authors would like to thank Telkom University for financially supporting in this research.

REFERENCES

1. Fauzi, R. M., Adiwijaya, Maharani, W. 2016. The Recognition of Hijaiyah Letter Pronunciation Using Mel Frequency Cepstral Coefficients and Hidden Markov Model. Advanced Science Letters, 22(8), 2043-2046.

2. H. Fandy. H., Suyanto, T. Iwan.I., 2011, Pengenalan Sinyal Suara Pada Speech To-Text Menggunakan Linear Predictive Coding (LPC) dan Hidden Markov Model (HMM). Institut Telnologi Telkom.
3. Anu L.B, Suresh D, Kubakaddi Sanjeev. (June, 2015). Person Identification using MFCC and Vector Quantization. IPASJ International Journal of Electronics & Communication (IIJEC). ISSN: 2321-5984. Volume 3. Issue 6.
4. Wisesty. U.N.; Adiwijaya.; Astuti W., 2015, Feature Extraction Analysis on Indonesian Speech Recognition System, Indonesia, ICoICT. DOI:10.1109/ICoICT.2015.7231396.
5. Wisesty. U.N.; Adiwijaya. Thee. H.L., 2012, Indonesian Speech Recognition System Using Discriminant Feature Extraction – Neural Predicitive Coding(DFE-NPC) and Probabilistic Neural Network, Indonesia, Cyberneticscom. DOI:10.1109/CyberneticsCom.2012.6381638.
6. Yulita, I. N., Houw Liong The, Adiwijaya. 2012. Fuzzy Hidden Markov Models for Indonesian Speech Classification. JACIII, 16(3), 381-387.
7. Rabiner. L.R., Juang. B.H., 2006, Speech Recognition: Statistical Methods,Amerika, Elsevier Ltd.
8. Gupta Shikha, Jaafar Jafreezal, Ahmad Wan Fatimah A, Bansal Arpit, (August, 2013), Feature Extraction Using MFCC. Signal & Image Processing: An International Journal (SIPIJ). Vol. 4.
9. Rabiner.L.R., Juang.B.H., 1993, Fundamental Of Speech Recognition, Amerika, Prentice-Hall International, Inc.
10. Juang. B.H., Rabiner. L.R., 1991, Hidden Markov Models for Speech Recognition, Amerika, Technometrics.
11. Rabiner. L.R., 1989, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Amerika, IEEE.
12. Adiwijaya, Wisesty, U. N., Wirayuda, T. A. B., Baizal, Z. K. A., & Haryoko, U. 2013. An Improvement of Backpropagation Performance by Uisng Conjugate Gradient on Forecasting of Air Temperatur and Humidity in Indonesia. Far East Journal of Mathematical Sciences (FJMS), (Part I), 57-67.
13. Adiwijaya, U. N., & Nhita, F. (2014). Study of Line Search Techniques on the Modified Backpropagation for Forecasting of Weather Data in Indonesia. Far East Journal of Mathematical Sciences (FJMS) **86:2**, pp. 139 - 148.