11

Mel-Frequency Cepstrum Coefficients as Higher Order Statistics Representation to Characterize Speech Signal for Speaker Identification System in Noisy Environment using Hidden Markov Model

Agus Buono[1], Wisnu Jatmiko[2] and Benyamin Kusumoputro[3]
[1]Computer Science Department, Bogor Agriculture University,
[2]Faculty of Computer Science, University of Indonesia,
[3]Faculty of Engineering, University of Indonesia
Indonesia

1. Introduction

Sound is an effective and efficient magnitude for biometric characterization. However, the sound is a phenomenon that is a fusion of multidimensional and influenced many aspects, such as speaker characteristics (articulator configuration, emotions, health, age, sex, dialect), languages, and the environment (background and transmission media), so that the system has been developed until now has not been able to work well in real situations. This is the background of this research.

In this research, we investigate higher order statistics (HOS) and Mel-Frequency Cepstrum Coefficients (MFCC) as a feature extraction, and integrated with a Hidden Markov Model (HMM) as a classifier to get a more robust speaker identification system, especially for Gaussian Noise. Research carried out more focused on feature extraction part of the speaker identification system. In classifier process stage, we use the HMM. This is a technique that has been widely used in voice processing provides good results. At the beginning, we empirically showed the failure of conventional MFCC using power spectrum in noisy environment. Then proceed with reviewing the matter, and proposed HOS-based extraction techniques to overcome these problems. Next is an experiments to demonstrate the effectiveness of the proposed method. Data used in this study came from 10 people who say the phrase 'PUDHESA' as much as 80 times with different ways of utterance. In this research, we use signals that are spoken with different variations of pressure, duration, emotional, loud and weak. Figure 1 presents the forms of signals for different utterances of a speaker. In accordance with the focus of this research is to build models that are more robust to noise, then we add a Gaussian noise signal to each original signal with a signal-to-noise ratio (SNR) of 20 dB, 10 dB, 5 dB and 0 dB.

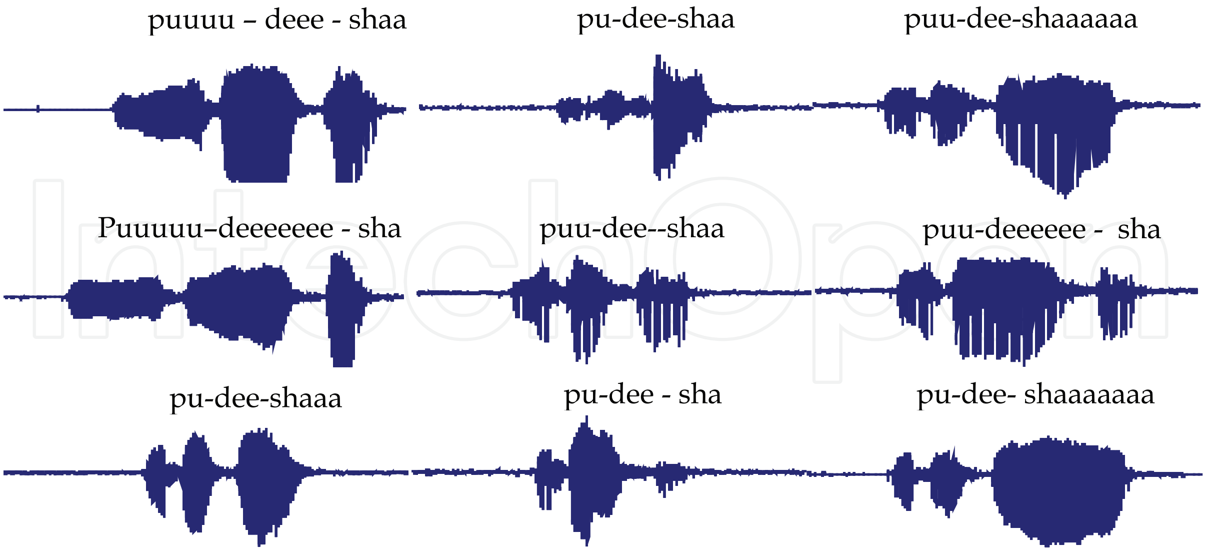


Fig. 1. Various forms of signals according to a speaker utterance mode

Figure 2 presents a comparison between the original speech signal with the original signal that has been contaminated by gaussian noise signal with a level of 20 dB, 10 dB, 5 dB and 0 dB. From the pictures it can be seen that the more severe the noise is given, then the more the signal is distorted from its original form.

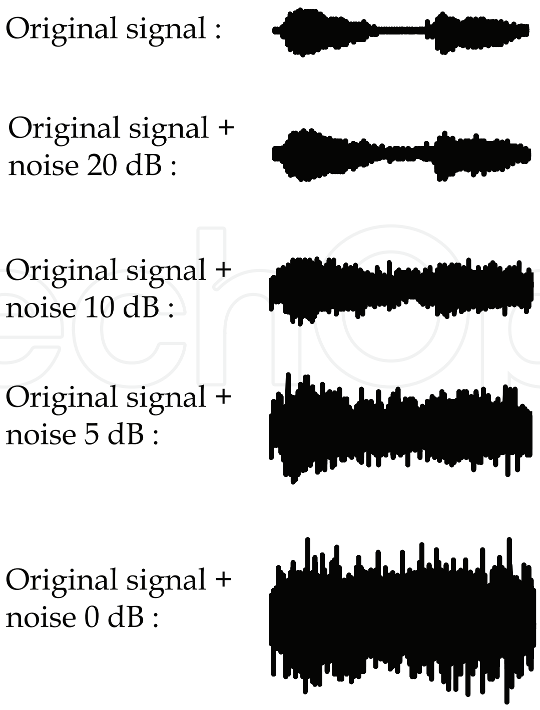


Fig. 2. Comparison of the original signal with the signal that is contaminated by noise

3. Speaker identification system

3.1 Overview

Speaker identification is an automatic process to determine who the owner of the voice given to the system. Block diagram of speaker identification system are shown in Figure 3. Someone who will be identified says a certain word or phrase as input to the system. Next, feature extraction module calculates features from the input voice signal. These features are processed by the classifier module to be given a score to each class in the system. The system will provide the class label of the input sound signal according to the highest score.

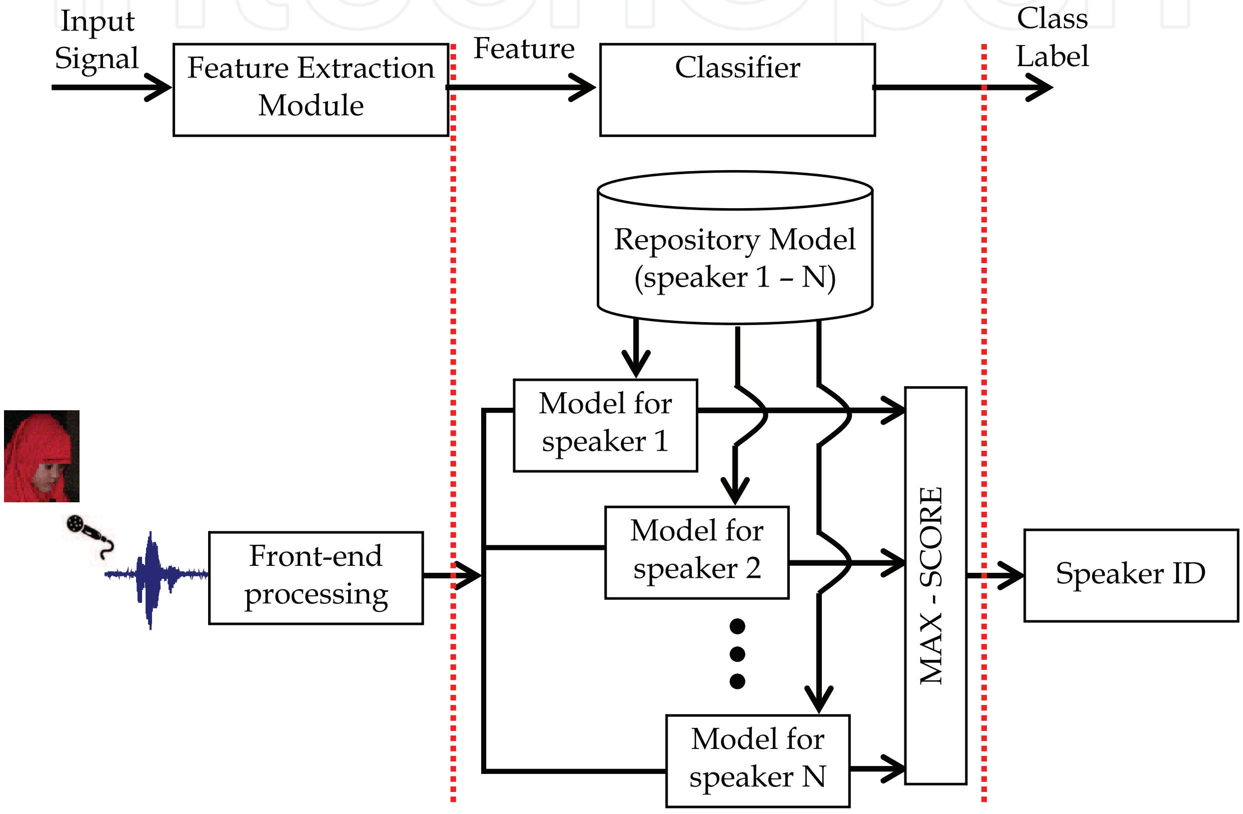


Fig. 3. Block diagram of speaker identification system

Input to the speaker identification system is a sound wave signal. The initial phase is to conduct sampling to obtain digital signals from analogue voice signal. Next perform quantization and coding. After the abolition of the silence, these digital signals are then entered to the feature extraction module. Voice signals are read from frame to frame (part of signal with certain time duration, usually 5 ms up to 100 ms) with a certain length and overlapped for each two adjacent frames. In each frame windowing process is carried out with the specified window function, and continued with the process of feature extraction. This feature extraction module output will go to the classifier module to do the recognition process. In general there are four methods of classifier (Reynold, 2002), namely: template matching, nearest neighbour, neural network and hidden Markov model (HMM). With the template matching method, the system has a template for each word/speaker. In the nearest neighbour, the system must have a huge memory to store the training data. While the neural network model is less able to represent how the sound signal is produced naturally. In the Hidden Markov Model, speech signal is statistically modelled, so that it can represent how the sound is produced naturally. Therefore, this model was first used in modern speaker recognition system. In this research we use the HMM as a classifier, so the features of each frame will be processed sequentially.

3.2 MFCC as feature extraction

Feature extraction is the process for determining a value or a vector that can characterize an object or individual. In the voice processing, a commonly used feature is the cepstral coefficients of a frame. Mel-Frequency Cepstrum Coefficients (MFCC) is a classical feature extraction and speech parameterization technique that widely used in the area of speech processing, especially in speaker recognition system.

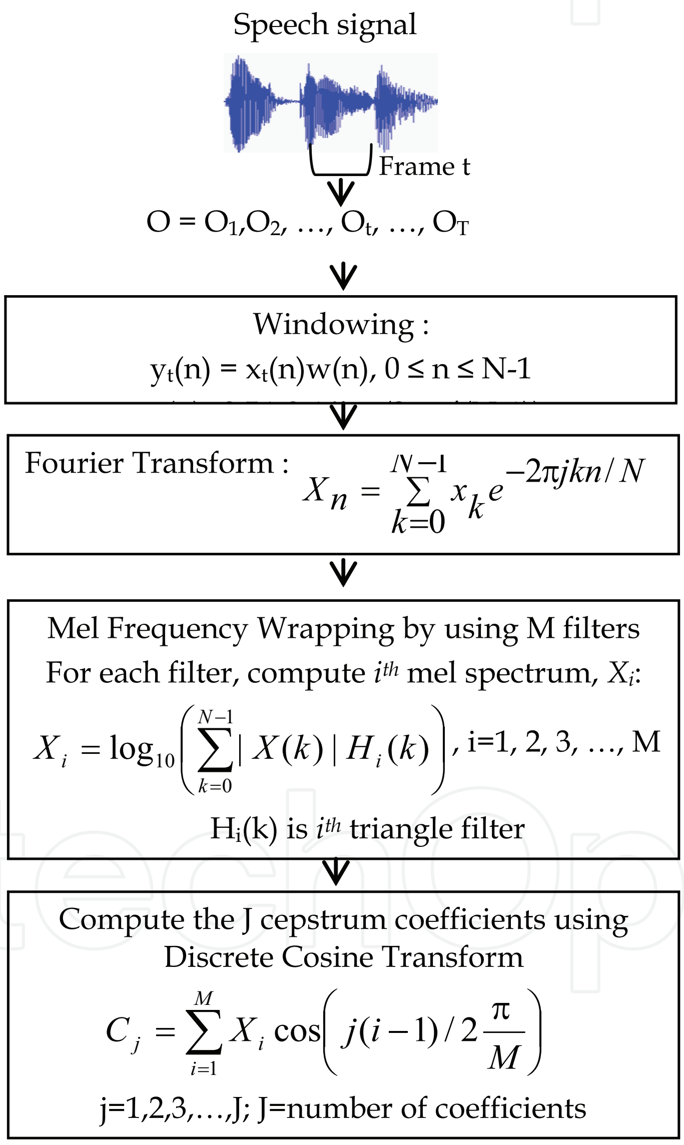

Speech signal
Frame t
$O = O_1, O_2, \ldots, O_t, \ldots, O_T$
Windowing :
$y_t(n) = x_t(n)w(n),\ 0 \le n \le N-1$
Fourier Transform : $X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N}$
Mel Frequency Wrapping by using M filters
For each filter, compute i^{th} mel spectrum, X_i:
$X_i = \log_{10}\left(\sum_{k=0}^{N-1} |X(k)|\, H_i(k)\right)$, i=1, 2, 3, …, M
$H_i(k)$ is i^{th} triangle filter
Compute the J cepstrum coefficients using Discrete Cosine Transform
$C_j = \sum_{i=1}^{M} X_i \cos\left(j(i-1)/2\frac{\pi}{M}\right)$
j=1,2,3,…,J; J=number of coefficients

Fig. 4. MFCC process flowchart

Compare to other feature extraction methods, Davis and Mermelstein have shown that MFCC as a feature extraction technique gave the highest recognition rate (Ganchev, 2005). After its introduction, numerous variations and improvements of the original idea are developed; mainly in the filter characteristics, i.e, its numbers, shape and bandwidth of filters and the way the filters are spaced (Ganchev, 2005). This method calculates the cepstral coefficients of a speech signal by considering the perception of the human auditory system to sound frequency. Block diagram of the method is depicted in Figure 4. For more detailed explanation can be read in (Ganchev, 2005) and (Nilsson, M & Ejnarsson, 2002).

After a process of windowing and Fourier transformation, performed wrapping of signals in the frequency domain using a number of filters. In this step, the spectrum of each frame is wrapping using M triangular filter with an equally highest position as 1. This filter is developed based on the behavior of human ear's perception, in which a series of psychological studies have shown that human perception of the frequency contents of sounds for speech signal does not follow a linear scale. Thus for each tone of a voice signal with an actual frequency f, measured in Hz, it can also be determined as a subjective pitch in another frequency scale, called the 'mel' (from Melody) scale, (Nilsson, M & Ejnarsson, 2002). The mel-frequency scale is determined to have a linear frequency relationship for f below 1000 Hz and a logarithmic relationship for f higher than 1000Hz. One most popular formula for frequency higher than 1000 Hz is, (Nilsson, M & Ejnarsson, 2002):

$$\hat{f}_{mel} = 2595 * \log_{10}\left(1 + \frac{f}{700}\right) \tag{1}$$

as illustrated by Figure 5 below:

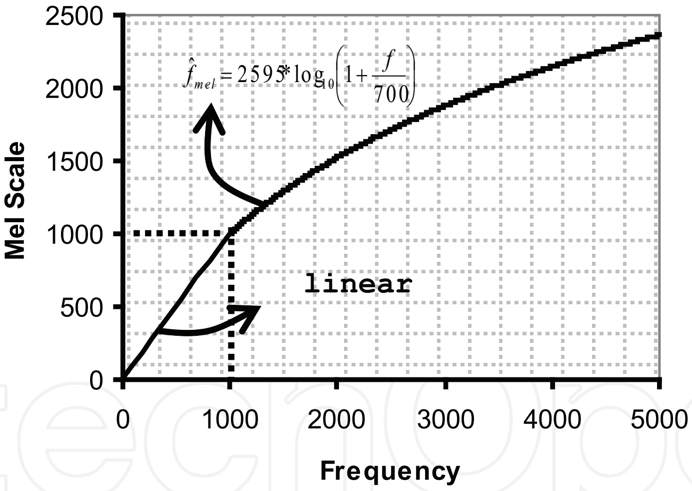


Fig. 5. Curve relationship between frequency signal with its mel frequency scale

Algorithm 1 depicted the process for develop those M filters, (Buono et al., 2008).

Algorithm 1: Construct 1D filter

a. Select the number of filter (M)

b. Select the highest frequency signal (f_{high}).

c. Compute the highest value of $\hat{f}_{mel}$:

$$\hat{f}_{mel}^{high} = 2595 * \log_{10}\left(1 + \frac{f_{high}}{700}\right)$$

d. Compute the center of the i^th filter (f_i), i.e.:

d.1. $f_i = \frac{1000}{0.5 * M} * i$ for i=1, 2, 3, …, M/2

d.2. for i=M/2, M/2+1, …, M, the f_i formulated as follow :

1. Spaced uniformly the mel scale axis with interval width Δ , where:

   $\Delta = \frac{\hat{f}_{mel}^{high} - 1000}{0.5 * M}$

   According to the equation (1), the interval width Δ can be expressed as:

   $\Delta = \frac{5190}{M} \log\left(\frac{700 + f_{high}}{1700}\right)$

2. The mel-frequency value for the center of ith filter is:

   $a = 1000 + (i - 0.5 * M) * \Delta$

3. So, the center of ith filter in frequency axes is:

   $f_i = 700 * \left(10^{a/2595} - 1\right)$

Figure 6 gives an example of the triangular i^th filter:

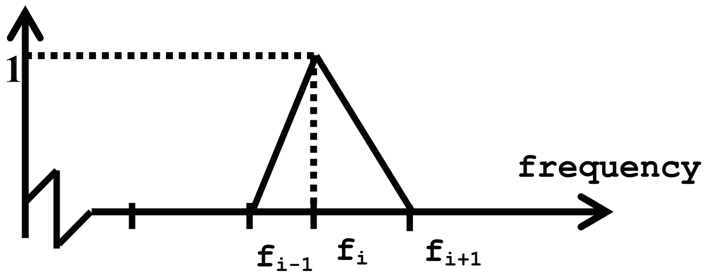


Fig. 6. A triangular filter with height 1

The mel frequency spectrum coefficients are calculated as the sum of the filtered result, and described by:

$$X_i = \log\left(\sum_{f=0}^{N-1} abs(X(j)) * H_i(f)\right) \tag{2}$$

where i=1,2,3,…,M, with *M* the number of filter; *N* the number of FFT coefficients; *abs(X(j))* is the magnitude of j^th coefficients of periodogram yielded by Fourier transform; and $H_i(f)$ is the i^th triangular at point *f*.

The next step is cosine transform. In this step we convert the mel-frequency spectrum coefficients back into its time domain using discrete cosine transform:

$$C_j = \sum_{i=1}^{M} X_i * \cos\left(\frac{j * (i - 0.5) * \pi}{20}\right) \tag{3}$$

where j=1,2,3,…,K, with *K* the number of coefficients; *M* the number of triangular filter; X_i is the mel-spectrum coefficients, as in (2). The result is called mel frequency cepstrum coefficients. Therefore the input data that is extracted is a dimensionless Fourier coefficients, so that for this technique we refer to as 1D-MFCC.

3.3 Hidden Markov model as classifier

HMM is a Markov chain, where its hidden state can yield an observable state. A HMM is specified completely by three components, i.e. initial state distribution, Л, transition probability matrix, A, and observation probability matrix, B. Hence, it is notated by λ = (A, B, Л), where, (Rabiner, 1989) and (Dugad & Desai, 1996):

A: NxN transition matrix with entries $a_{ij}=P(X_{t+1}=j\,|\,X_t=i)$, N is the number of possible hidden states

B: NxM observation matrix with entries $b_{jk}=P(O_{t+1}=v_k\,|\,X_t=j)$, k=1, 2, 3, …, M; M is the number of possible observable states

Л: Nx1 initial state vector with entries $п_i=P(X_1=i)$

For HMM's Gaussian, B consists of a mean vector and a covariance matrix for each hidden state, μ_i and Σ_i, respectively, i=1, 2, 3, …, N. The value of $b_j(O_{t+1})$ is $N(O_{t+1},\mu_j,\Sigma_j)$, where :

$$N(\mu_j,\Sigma_j)=\frac{1}{(2\pi)^{d/2}\,|\,\Sigma_j\,|^{1/2}}\exp\left[-\frac{1}{2}(O_{t+1}-\mu_j)\Sigma_j^{-1}(O_{t+1}-\mu_j)'\right] \qquad (4)$$

There are three problems with HMM, (Rabiner, 1989), i.e. evaluation problem, P(O|λ); decoding problem, P(Q|O, λ); and training problem, i.e. adjusting the model parameters A, B, and Л. Detailed explanation of the algorithms of these three problems can be found in (Rabiner, 1989) and (Dugad & Desai, 1996).

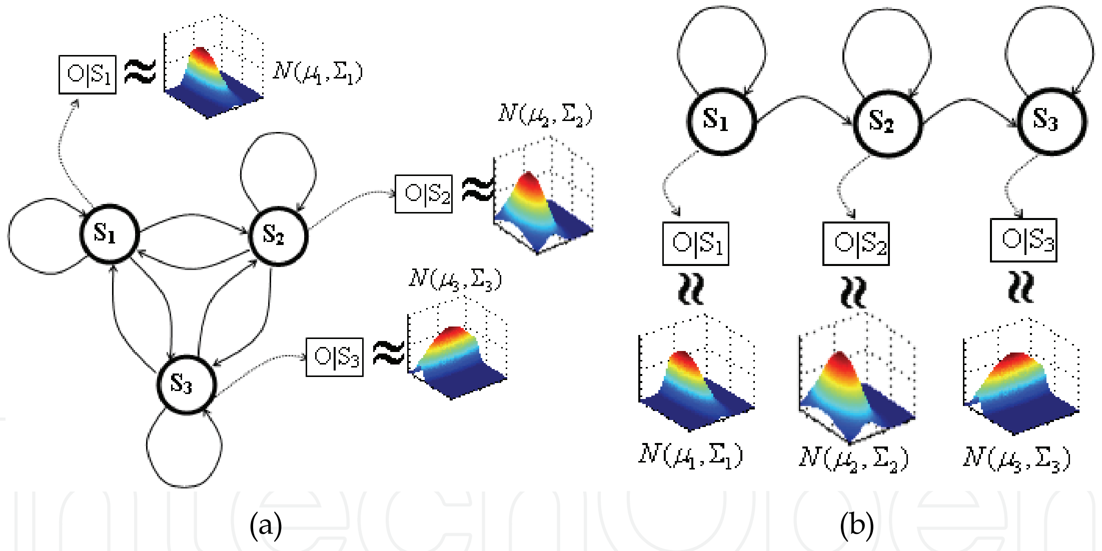
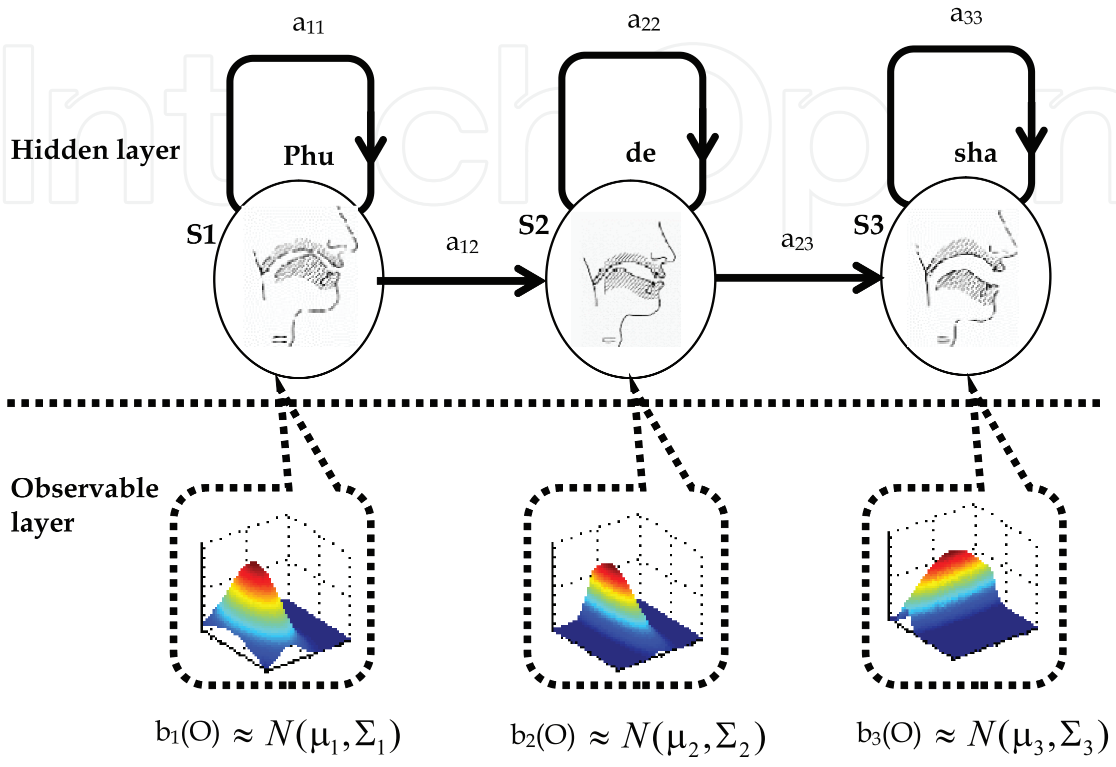


Fig. 7. Example HMM with Three Hidden State and distribtion of the evidence variable is Gaussian, (a) Ergodic, (b) Left-Right HMM

In the context of HMM, an utterance is modeled by a directed graph where a node/state represents one articulator configuration that we could not observe directly (hidden state). A graph edge represents transition from one configuration to the successive configuration in the utterance. We model this transition by a matrix, A. In reality, we only know a speech signal produced by each configuration, which we call observation state or observable state. In HMM's Gaussian, observable state is a random variable and assumed has Normal or Gaussian distribution with mean vector μ_i and covariance matrix Σ_i (i=1, 2, 3, …, N; N is number of hidden states). Based on inter-state relations, there are two types of HMM, which is ergodic and left-right HMM. On Ergodic HMM, between two states there is always a link, thus also called fully connected HMM. While the left-right HMM, the state can be arranged from left to right according to the link. In this research we use the left-right HMM as depicted by Figure 8.


a_{11}
a_{22}
a_{33}
Hidden layer
Phu
de
sha
S1
S2
S3
a_{12}
a_{23}
Observable layer
$b_1(O) \approx N(\mu_1,\Sigma_1)$
$b_2(O) \approx N(\mu_2,\Sigma_2)$
$b_3(O) \approx N(\mu_3,\Sigma_3)$

Note : a_{ij} is the transition probability from state i into state j
$b_i(O)$ is the distribution of observable O given hidden state Si

Fig. 8. Left-Right HMM model with Three State to Be Used in this Research

3.4 Higher order statistics

If {x (t)}, t = 0, ± 1, ± 2, ± 3, ... is a stationary random process then the higher order statistics of order n (often referred as higher order spectrum of order n) of the process is the Fourier transform of $\{c_n^x\}$. In this case $\{c_n^x\}$ is a sequence of n order cumulant of the {x (t)} process. Detailed formulation can be read at (Nikeas & Petropulu, 1993). If n=3, the spectrum is known as bispectrum. In this research we use bispectrum for characterize the speech signal. The bispectrum, $C_3^x(\omega_1,\omega_2)$, of a stationary random process, {x (t)}, is formulated as:

$$C_3^x(\omega_1,\omega_2)=\sum_{\tau_1=-\infty}^{+\infty}\sum_{\tau_2=-\infty}^{+\infty}c_3^x(\tau_1,\tau_2)\exp\left\{-j\left(\omega_1\tau_1,\omega_2\tau_2\right)\right\} \tag{5}$$

where $c_3^x(\tau_1,\tau_2)$ is the cumulant of order 3 of the stationary random process, {x (t)}. If n=2, it is usually called as power spectrum. In 1D-MFCC, we use power spectrum to characterize the speech signal. In theory the bispectrum is more robust to gaussian noise than the power spectrum, as shown in Figure 9. Therefore in this research we will conduct a development of MFCC technique for two-dimensional input data, and then we refer to as 2D-MFCC.

Basically, there are two approaches to predict the bispectrum, i.e. parametric approach and conventional approach. The conventional approaches may be classified into the following three classes, i.e. indirect technique, direct technique and complex demodulates method. Because of the simplicity, in this research we the conventional indirect method to predict the bispectrum values. Detail algorithm of the method is presented in (Nikeas & Petropulu, 1993).

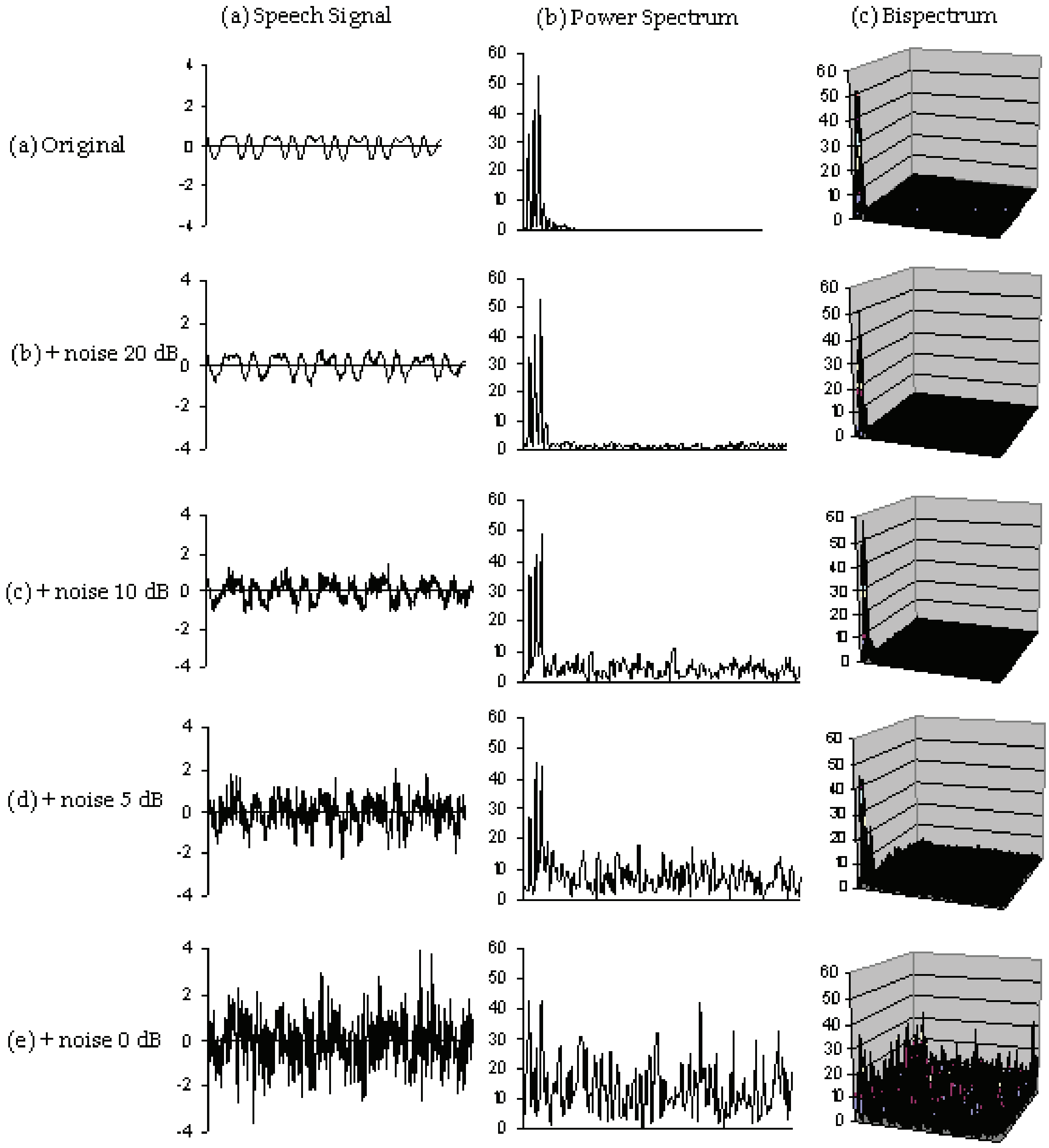


Fig. 9. Comparison between the power spectrum with the bispectrum for different noise

4. Experimental setup

First we show the weakness of 1D-MFCC based on power spectrum in capturing the signal features that has been contaminated by gaussian noise. Then we proceed by conducting two experiments with similar classier, but in feature extraction step, we use 2D-MFCC based on the bispectrum data.

4.1 1D-MFCC + HMM

Speaker identification experiments are performed to follow the steps as shown in Figure 10.

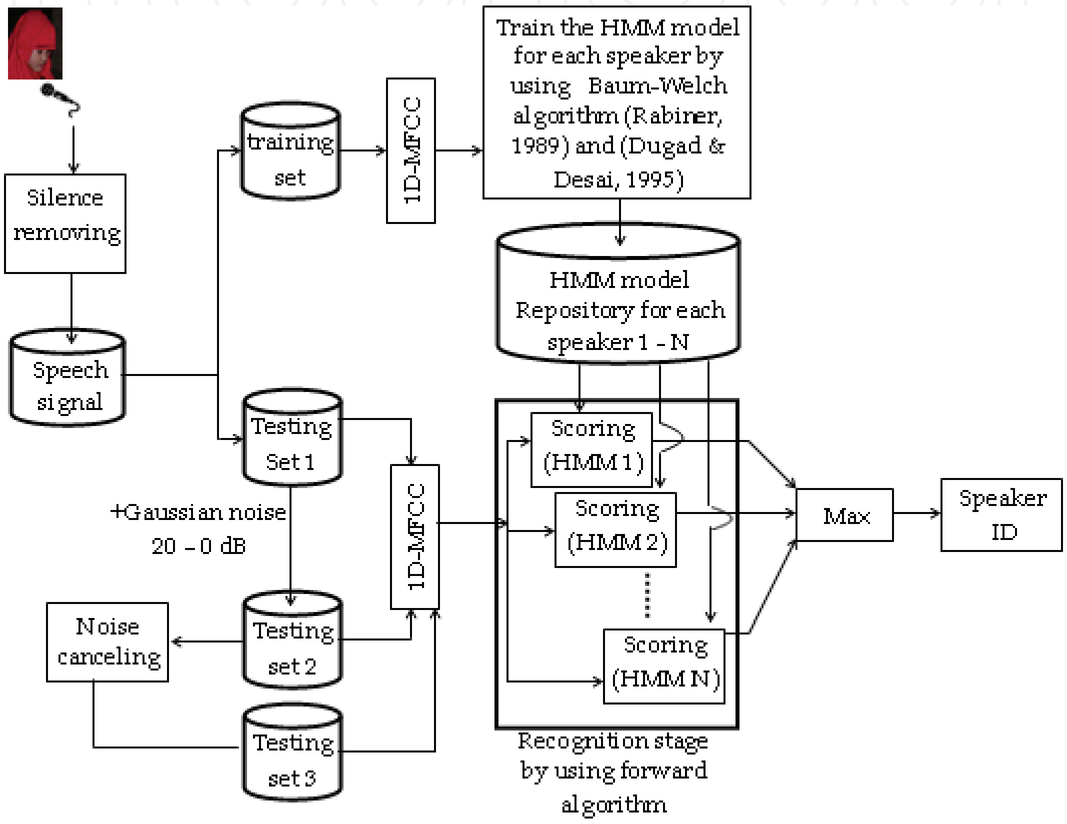

Train the HMM model for each speaker by using Baum-Welch algorithm (Rabiner, 1989) and (Dugad & Desai, 1995)
training set
1D-MFCC
Silence removing
Speech signal
HMM model Repository for each speaker 1 - N
Testing Set 1
Scoring (HMM 1)
+Gaussian noise 20 – 0 dB
1D-MFCC
Scoring (HMM 2)
Max
Speaker ID
Noise canceling
Testing set 2
Scoring (HMM N)
Testing set 3
Recognition stage by using forward algorithm

Fig. 10. Block diagram of experimental 1D-MFCC + HMM

The data used comes from 10 speakers each of 80 times of utterance. Before entering the next stage, the silence of the signal has been eliminated. Then, we divide the data into two sets, namely training data set and testing data set. There are three proportion values between training data and the testing data, ie 20:60, 40:40 and 60:20. Furthermore, we established three sets of test data, ie data sets 1, 2 and 3. Data set 1 is the original signal without adding noise. Data set 2 is the original signal by adding gaussian noise (20 dB, 10 dB, 5 dB and 0 dB), without the noise removal process. Data set 3 is the original signal by adding gaussian noise and noise removal process has been carried out with noise canceling algorithm, (Widrow et al., 1975) and (Boll, 1979). Next, the signal on each set (there are four sets, namely training data, testing data 1, testing data 2, and testing data 3) go into the feature extraction stage. In this case all the speech signals from each speaker is calculated its characteristic that is read frame by frame with a length 256 and the overlap between adjacent frames is 156, and forwarded to the appropriate stage of 1D-MFCC technique as has been described previously. The next stage is to conduct the experiment according to the specified proportion, so that there are three experiments. In each experiment, in general there are two main stages, namely training stage and the recognition stage. In the training phase, we use the Baum-Welch algorithm to estimate the parameters of HMM, (Rabiner, 1989) and (Dugad & Desai, 1995). Data used in this training phase is the signal in training data that has been through the process of feature extraction. Our resulting HMM parameters stored in the repository, which would then be used for the recognition process. After the model is obtained, followed by speaker identification stage. In this case each signal on the test data (one test data, test data second and third test data) that has been through the process of feature extraction will be given a score for each speaker model. For a signal to be identified, compute the score for model 1 to model the N (N is the number of models in the repository). Score for model i, Si, is calculated by running the forward algorithm with the HMM model i. Further to these test signals will be labeled J, if $S_j > S_i$, for i=1,2,3, ...,j-1, j+1, ..., N.

Experimental result

Table 1 presents the accuracy of the system for various noise and various proportions of training data and test data.

Tipe of test data set	Training:test		
	20:60	40:40	60:20
Original signal	85.5	93.8	99.0
+noise 20 dB	37.0	41.1	52.8
+noise 10 dB	14.4	15.4	22.5
+noise 5 dB	12.7	13.8	17.3
+noise 0 dB	10.4	10.0	11.3

Table 1. The accuracy of the system at various proportions of training data and test data

From the table it can be said that for the original signal, the system with feature extraction using 1D-MFCC and HMM as a classifier able to recognize very well, which is around 99% for the original data on the proportion of 75% training data. The table also shows that with increasing noise, the accuracy drops drastically, which is to become 52% to 20 dB noise, and for higher noise, the accuracy below 50%. It is visually apparent as shown in Figure 11. The failure of this system is caused by the power spectrum is sensitive to noise, as shown in Figure 9 above.

To see the effect of number of hidden states to the degree of accuracy, in this experiment, the number of hidden state in HMM model varies from 3 to 7. Based on the results, seen that level of accuracy for the original signal is ranged from 99% to 100%. This indicates that the selection of number of hidden state in HMM does not provide significant effect on the results of system accuracy.

Table 1 also indicates that the amount of training data will affect the HMM parameters that ultimately affect the accuracy of the system. In this research, a signal consisting of about 50 frames. Therefore, to estimate HMM parameters that have a state of 3 to 7 is required sequence consisting of 3000 (50x60) samples.

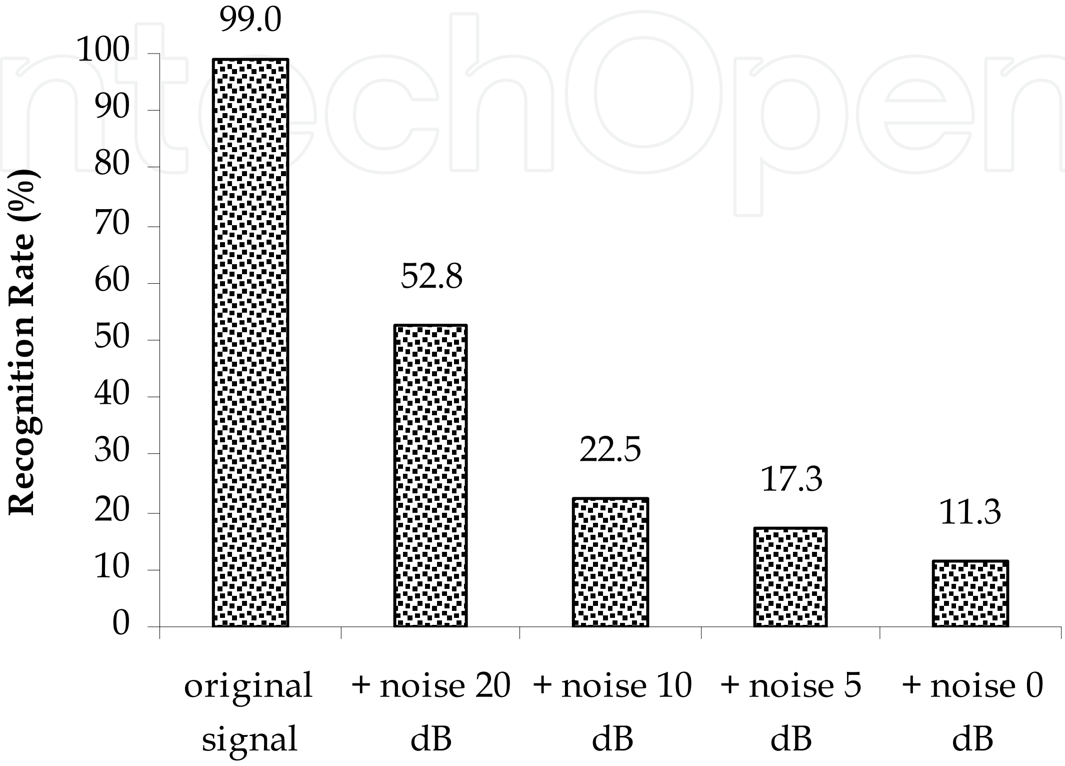


Fig. 11. The accuracy of the system for a variety of noise on the proportion of training data and test data 60:20

To improve the accuracy of the system, then we continue the experiment with Data set 3 (with noise removal or cancellation process, NC) and the results are presented in Figure 12. After going through the process of adaptive noise canceling, system performance increases, especially for signals with the addition of noise, as shown in the figure. For the original signal without adding noise, the NC system provides 96.6% accuracy, about 3% below the system without going through the NC. While for signals with the addition of noise, adaptive noise canceling improve system robustness against noise up to the level of 20 dB with an accuracy of 77.1%. For larger noise, the system failed to work properly.

Based on the above findings, we conducted further experiments using the bispectrum as input for the feature extraction stage. By using this bispectrum, it is expected effect of noise can be suppressed. Bispectrum for a given frame is a matrix with dimensions NxN, where N is the sampling frequency. In this research, we chose N=128, so that for one frame (40 ms) will be converted into a matrix of dimension 128x128. Therefore we perform dimension reduction using quantization techniques. This quantization results next through the process of wrapping and cosine transformation as done in the 1D-MFCC. To abbreviate, then we call this technique as 2D-MFCC.

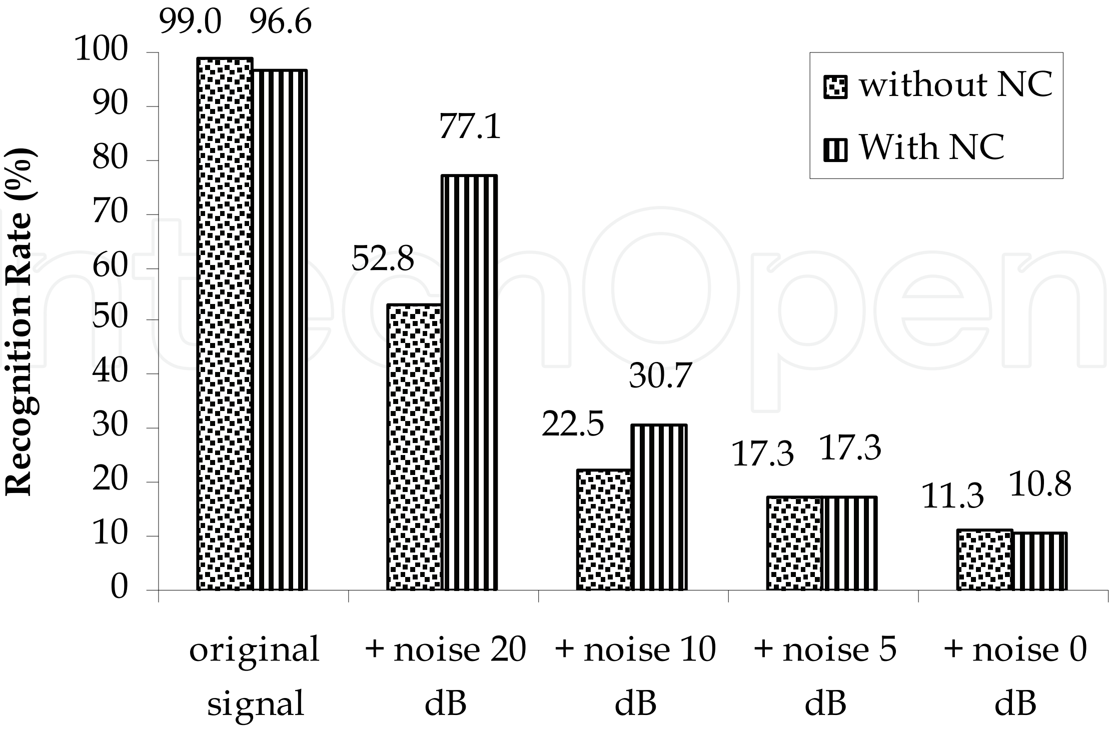


Fig. 12. Accuracy of the system with and without noise cancellation (NC)

4.3 2D-MFCC + HMM

Flow diagram of the experiments conducted in this section are presented in Figure 13. In general there are three parts of the picture, namely the establishment of the channel center (which would be required for quantitation of the bispectrum), the training of HMM models, and the testing model. The process of determining the center of the channel that carried out the research followed the procedure as described in (Fanany & Kusumoputro, 1998). In the training stage of HMM models, each voice signal in the training set is read frame by frame, is calculated its bispectrum values, quantized, and the process of wrapping and cosine transform, so that the feature is obtained. After the feature is obtained, then forwarded to the stage of parameter estimation of HMM with Baum-Welch algorithm. This is done for each speaker, thus obtained 10 HMM models. In testing or recognition phase, a voice signal is read frame by frame, then for each frame is calculated its bispectrum, quantized, followed by wrapping and cosine transform. After that, followed by the recognition process using a forward algorithm for each HMM model (which resulted in the training phase).

Channel center reconstruction

Due to the bispectrum is simetric, then we simply read it in the triangle area of the domain space bispectrum (two-dimensional space, F1xF2). Center channel is determined such that the point (f1, f2) with high bispektrum will likely selected as determination of the channel center. Therefore, the center will gather at the regional channels (f1, f2) with large bispectrum values and for regions with small bispectrum value will have less of channel center. With these ideas, then the center channel is determined by the sampling of points on F1xF2 domain. Sampling is done by taking an arbitrary point on the domain, then at that point generated the random number r€[0,1]. If this random number is smaller than the ratio of the bispectrum at these points with the maximum of the bispectrum, then the point will be selected as the determination point. For another thing, then the point is ignored. Having obtained a number of determination points, followed by clustering of these points to obtain the K cluster centers. Then, the cluster center as the channel center on the bispectrum quantization process. From the above explanation, there are three phases to form a center channel, namely the establishment of a joint bispectrum, bispectrum domain sampling and determination of the channel center.

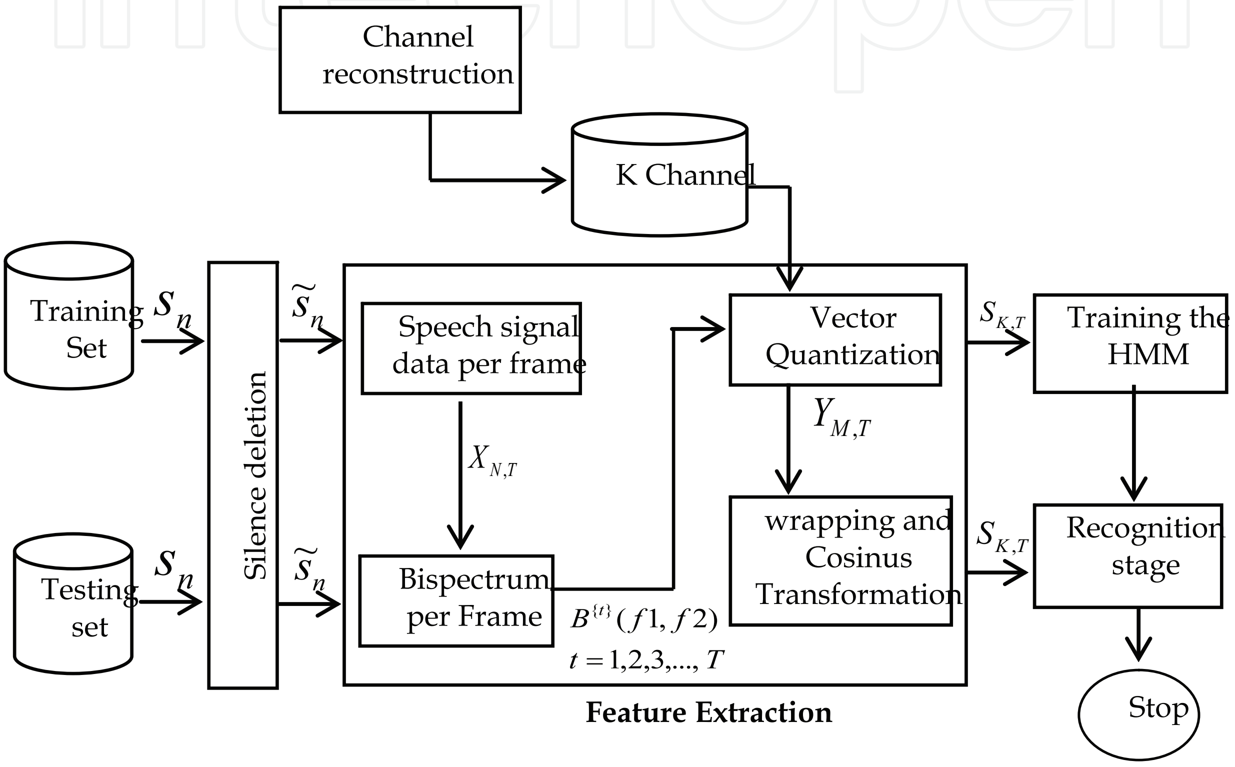


Fig. 13. Flow diagram of the experiments

Figure 14 presents the process of determining the combined bispactrum. a voice signal for each speaker is calculated its bispectrum frame by frame, and then averaged. After this process is done for all speakers, then the combined bispectrum is the sum of the average bispectrum of each speaker divided by the number of speakers (in this case 10).

After obtaining the combined bispectrum, the next is to conduct sampling of the points on the bispectrum domain. Figure 15 presents the sampling process in detail. The first time raised a point A (r1, r2) in the bispectrum domain and determined the point B (f1, f2) which is closest to A. Then calculated the ratio (r) between the combined bispectrum value at point B with the largest combined bispectrum value. After it was raised again a number r3. If r3<r, then inserted the point A into the set of point determination, G. If the number of points on the G already enough, followed by classifying the points on G into P clusters. Cluster centers are formed as the channel center. Next, the P channel's centers is stored for use in a quantization process of the bispectrum (in this research, the P value is 250, 400 and 600).

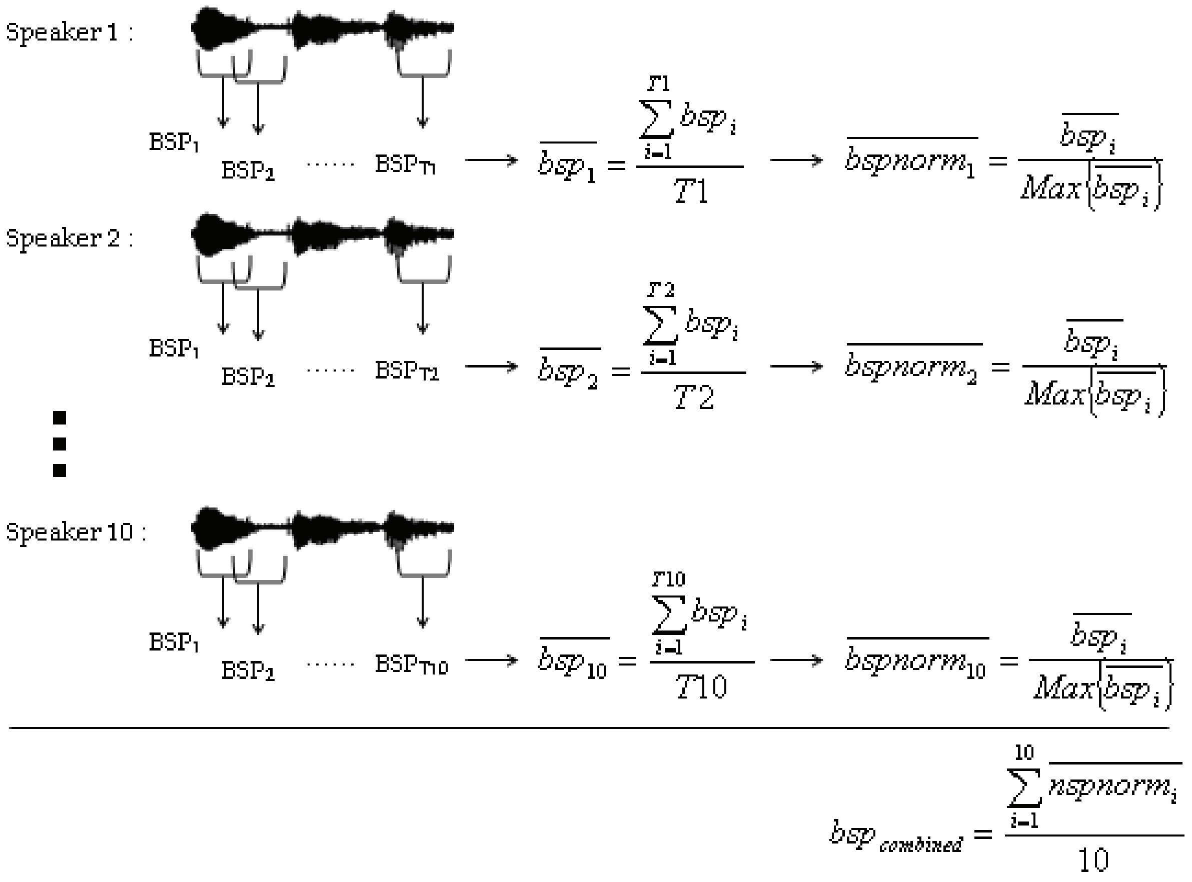

Speaker 1 :
BSP1
BSP2
BSPT1
$\overline{bsp_1} = \frac{\sum_{i=1}^{T1} bsp_i}{T1}$
$\overline{bspnorm_1} = \frac{\overline{bsp_i}}{Max\{\overline{bsp_i}\}}$
Speaker 2 :
BSP1
BSP2
BSPT2
$\overline{bsp_2} = \frac{\sum_{i=1}^{T2} bsp_i}{T2}$
$\overline{bspnorm_2} = \frac{\overline{bsp_i}}{Max\{\overline{bsp_i}\}}$
Speaker 10 :
BSP1
BSP2
BSPT10
$\overline{bsp_{10}} = \frac{\sum_{i=1}^{T10} bsp_i}{T10}$
$\overline{bspnorm_{10}} = \frac{\overline{bsp_i}}{Max\{\overline{bsp_i}\}}$
$bsp_{combined} = \frac{\sum_{i=1}^{10} \overline{nspnorm_i}}{10}$

Fig. 14. The process of determining the combined bispectrum

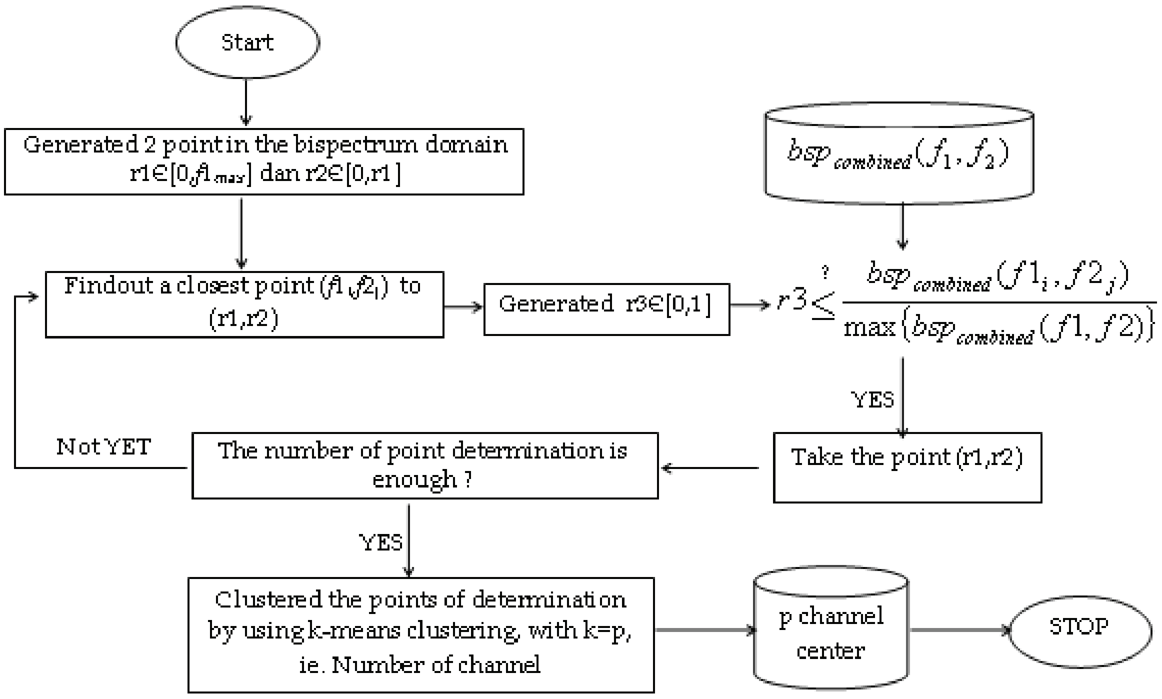

Start
Generated 2 point in the bispectrum domain r1∈[0,f1max] dan r2∈[0,r1]
$bsp_{combined}(f_1, f_2)$
Findout a closest point (f1i,f2j) to (r1,r2)
Generated r3∈[0,1]
$r3 \overset{?}{\leq} \frac{bsp_{combined}(f1_i, f2_j)}{\max\{bsp_{combined}(f1, f2)\}}$
YES
Not YET
The number of point determination is enough ?
Take the point (r1,r2)
YES
Clustered the points of determination by using k-means clustering, with k=p, ie. Number of channel
p channel center
STOP

Fig. 15. Bispectrum domain sampling process

Having obtained the P channel's centers, next will be described the process of quantization the bisepctrum of a frame. Bispectrum is read only performed on half of the domain. Each point in the first half of this domain is labeled in accordance with the nearest channel center. Bispectrum values for each channel is obtained by calculating the bispectrum statistic.
The next stage of feature extraction is the process of wrapping. For this, the P channel are sorted based on the distance to the central axis. Wrapping process using a filter like that used in 1D-MFCC. Having obtained the coefficient for each filter, followed by a cosine transform. Output of the feature extraction process is then entered to the recognition stage.

Result and discussion

Figure 16. presents a comparison of the accuracy of the system using the number of channels 250, 400 and 600, followed by wrapping and cosine transform for the reduction of channel dimensions. From the figure, it seen that the 2D-MFCC as feature extraction system provides the average accuracy of 90%, 89%, 75% for the original signal, the original signal plus noise 20 dB and the original signal plus noise 10 dB. With level of noise 5 dB and 0 dB, the system has failed to recognize properly. From these images can also be seen that the number of channels did not provide significant differences effect.

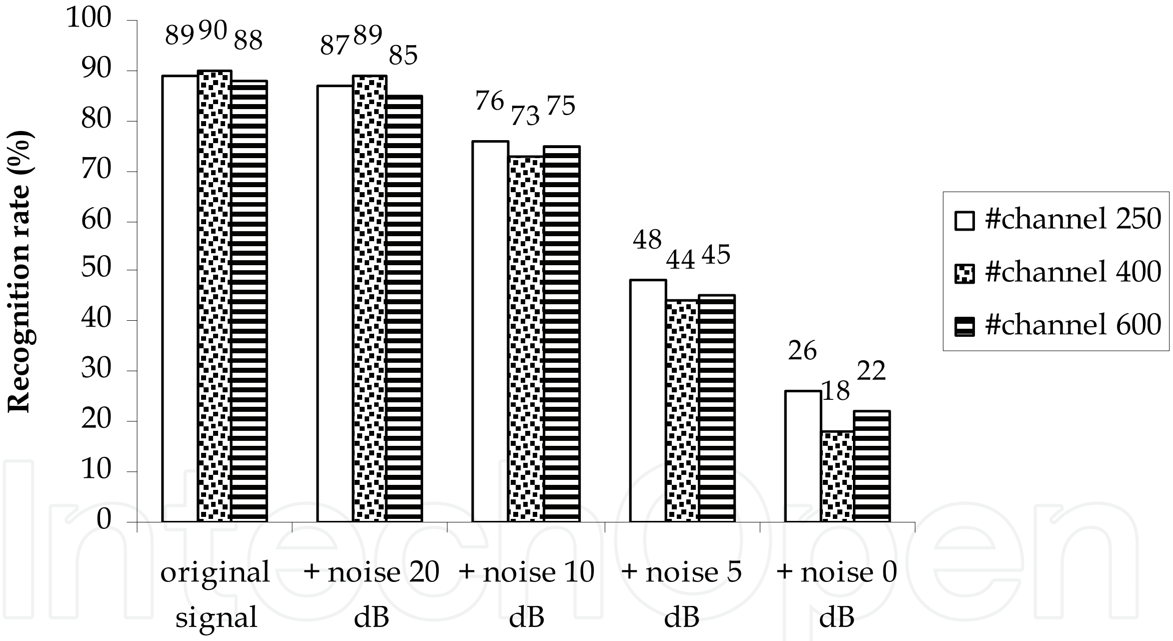


Fig. 16. Comparison of accuracy with different number of channels

When compared with previous techniques based on power spectrum (1D-MFCC) shows that the bispectrum-based technique is more robust to noise. This is as shown in Figure 17. Even if compared with the 1D-MFCC with the elimination of any noise, the 2D-MFCC technique still gives much better results. However, for the original signal, seen this technique still needs improvement. Some parts that can be developed is in the process of wrapping of the bispectrum which is quantized. In this case, there are several options, including whether to continue using the one-dimensional filter (as in the 1D-MFCC) with modifications on the shape and width filter. Or, by developing two-dimensional filter.

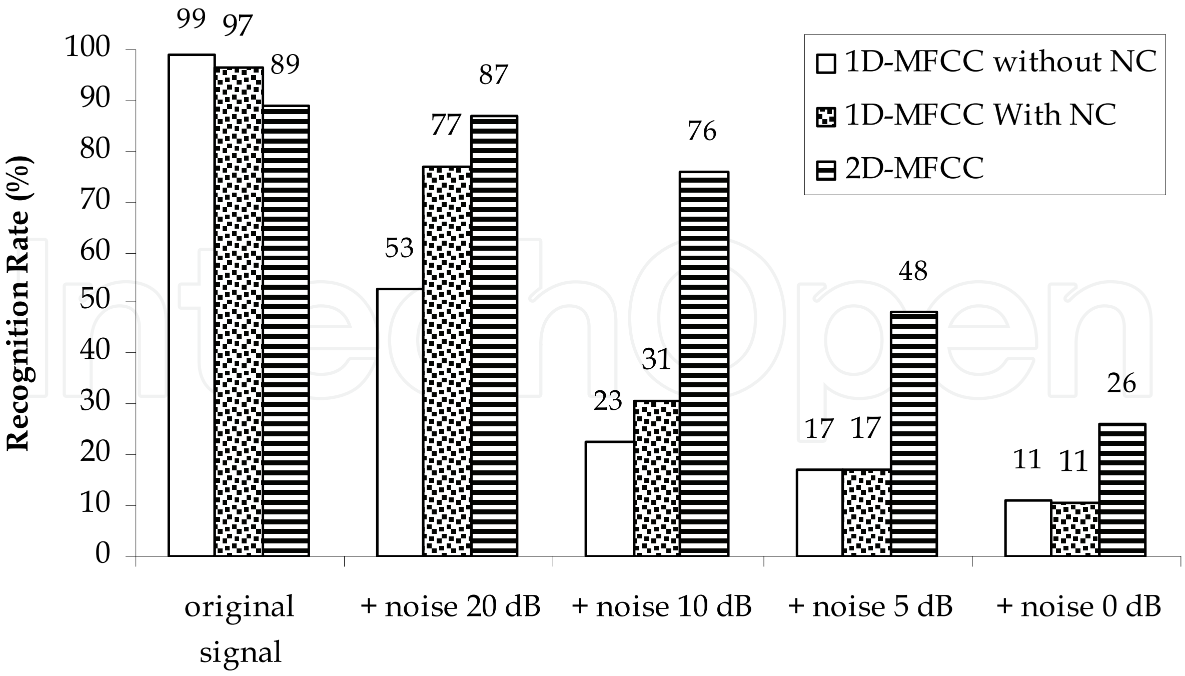


Fig. 17. Comparison of recognition rate between the 1D-MFCC with 2D-MFCC

5. Conclusion and future work

1. Conventional speaker identification system based on power spectrum can give results with an average accuracy of 99% for the original signal without adding noise, but failed to signal with the addition of noise, although only at the level of 20 dB. Noise removal technique is only capable of producing a system with sufficient accuracy (77.1%) up to 20 dB noise level. For larger noise, this technique can not work properly.
2. Bispectrum able to capture the characteristics of voice signals without adding noise or with the addition of noise, and visually it still looks up to levels above 0 dB. For noise level 0 dB, the shape of bispectrum has undergone significant changes compared with the one from original signal
3. In 2D-MFCC, the value bispektrum grouped on some channel that is formed by following bispektrum data distribution. Afterwards is the process of wrapping and cosine transformation. This technique is capable of providing accuracy to the original signal, the original signal plus noise 20 dB, 10 dB, 5 dB and 0 dB are respectively 89%, 87%, 76%, 48% and 26%.

From the experiments we have done, seen that the filter that is used for wrapping process contributes significantly to the level of accuracy. Therefore, further research is necessary to experiment using various forms of filters, such as those developed by Slaney (filter has a constant area, so the higher the filter is not fixed, but follow its width), also from the aspect of the number of filters (linear and logarithmic filters). In our research, we are just experimenting with the bispectrum (third order HOS), so we need further experiments using the HOS with higher order.There are Some disadvantages, (Farbod & Teshnehlab, 2005) with Gaussian HMM, especially in its assumptions, ie normality and independently, and constraints due to limited training data. Therefore it needs to do experiments that integrate 2D-MFCC (HOS-based) with the HMM model is not based on the assumption of normality, and do not ignore the fact that there is dependencies between observable variables.

6. References

Buono, A., Jatmiko, W. & Kusumoputro, B. (2008). Development of 2D Mel-Frequency Cepstrum Coefficients Method for Processing Bispectrum Data as Feature Extraction Technique in Speaker Identification System. *Proceeding of the International Conference on Artificial Intelegence and Its Applications* (*ICACIA*), Depok, September 2008

Nikeas, C. L. & Petropulu, A. P. (1993). *Higher Order Spectra Analysis : A Nonlinear Signal Processing Framework*, Prentice-Hall, Inc., 0-13-097619-9, New Jersey

Fanany, M.I. & Kusumoputro, B. (1998). *Bispectrum Pattern Analysis and Quantization to Speaker Identification*, Master Thesis in Computer Science, Faculty of Computer Science, University of Indonesia, Depok, Indonesia

Ganchev, T. D., (2005). *Speaker Recognition*. PhD Dissertation, Wire Communications Laboratory, Department of Computer and Electrical Engineering, University of Patras Greece.

Dugad, R., & Desai, U. B., (1996). *A Tutorial on Hidden Markov Model*. Technical Report, Departement of Electrical Engineering, Indian Institute of Technology, Bombay, 1996.

Rabiner, L., (1989). A Tutorial on Hidden Markov Model and Selected Applications in Speech Recognition. *Proceeding IEEE*, Vol 77 No. 2., pp. 257-286, 0018-9219, , Pebruari 1989

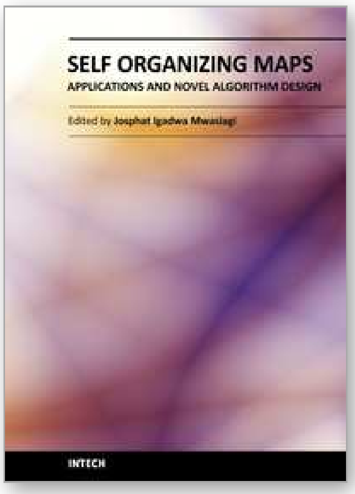
Boll, S. F., (1979). Suppression of Acoustic Noise in Speech Using Spectral Substraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-27, No. 2, April 1979, pp. 113-120, 0096-3518

Widrow, B. et. al., (1975). Adaptive Noise Canceling : Principles and Applications. *Proceeding of the IEEE*, Vol. 63. No. 12. pp. 1691-1716

Nilsson, M & Ejnarsson, M., (2002). *Speech Recognition using Hidden Markov Model : Performance Evaluation in Noisy Environment*. Master Thesis, Departement of Telecommunications and Signal Processing, Blekinge Institute of Technology

Reynolds, D., (2002). *Automatic Speaker Recognition Acoustics and Beyond*. Tutorial note, MIT Lincoln Laboratory, 2002

Farbod H. & M. Teshnehlab. (2005). Phoneme Classification and Phonetic Transcription Using a New Fuzzy Hidden Markov Model. *WSEAS Transactions on Computers*. Issue 6, Vol. 4.



Self Organizing Maps - Applications and Novel Algorithm Design
Edited by Dr Josphat Igadwa Mwasiagi

ISBN 978-953-307-546-4
Hard cover, 702 pages
Publisher InTech
Published online 21, January, 2011
Published in print edition January, 2011

Kohonen Self Organizing Maps (SOM) has found application in practical all fields, especially those which tend to handle high dimensional data. SOM can be used for the clustering of genes in the medical field, the study of multi-media and web based contents and in the transportation industry, just to name a few. Apart from the aforementioned areas this book also covers the study of complex data found in meteorological and remotely sensed images acquired using satellite sensing. Data management and envelopment analysis has also been covered. The application of SOM in mechanical and manufacturing engineering forms another important area of this book. The final section of this book, addresses the design and application of novel variants of SOM algorithms.