View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by IntechOpen

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

6

Modelling and Understanding of Speech and Speaker Recognition

Tilendra Shishir Sinha[1] and Gautam Sanyal[2]
[1]*Computer Science & Engineering Department, DIMAT*
Raipur, Chhattisgarh State
[2]*Computer Science & Engineering Department, NIT*
Durgapur, West Bengal
India

1. Introduction

The main goal of automatic speech and speaker recognition (ASSR) is to transcribe natural speech and recognize its speaker. Recognizing a spoken sentence is obviously a knowledge-intensive process, which must take into account all variable information about the speech communication process. Out of several approaches, artificial intelligence approach has been observed to give remarkable results by visualizing, analysing and finally making a decision on the acoustic patterns. The present chapter deals with two basic processes for ASSR: *modelling* process and *understanding* process. The modelling process involves three stages: enhancement, segmentation and pre-processing. The understanding phase involves the recognition of the speech and the speaker through the known knowledge-based model. It has been observed from the literature that, in any speech processing system, because of channel coupling, many source signals mix together. Thus there may be a chance of variability in speech signals affecting the performance of the speech systems due to some factors like: background and channel noise, electrical noise from different sources, meaningless sounds (a sneeze) or filler words ('uh' or 'um') between words, different speaking rate, mood and styles of speakers. Due to this some signals may not be observed and may result to out-of-vocabulary (OOV) word problem during recognition or understanding process. Thus speech enhancement has played a vital role in the development of a perfect acoustic model or noise – free artificial word model (AWM) and vowel diphthong model (VDM). To process for speech enhancement in frequency – domain, the speech signal has to be split into frames using blind signal separation (BSS) method (Yilmaz and Richard 2004; Yamashita and Hirai 2004; Araki et al 2002; Murata et al 2001; G. J. Jang and Lee 2003; F. Bach and Jordan 2005; B. A. Pearl Mutter and Olsson 2006). Further the spectrum of the background noise has to be estimated by subtracting the noise spectrum from the spectrum of the frame (M. Berouli et al 1979; Yasser Ghanbari and Md. Reza 2004). It has been observed that some of the noise remains in the spectrum when the value of noise is greater than its mean. At the same time, some of the speech spectrum gets removed when noise is greater than the actual value of noise. In the spectrum this produces negative values and have to be set to zero. The overall effect puts a noise in the output signal known as

residual noise. To reduce the level of residual noise further subtraction has to be carried out by adjusting the 'over-subtraction' or 'over-estimation' factor with respect to signal to noise ratio as suggested by Berouli et al (1979). Before processing the speech any further, the wavelet (a real valued function of time) in a noisy signal (if any) has to be detected, using discrete wavelet transform (DWT) method. Further, analysis has to be done for the speech segmentation, which means partitioning an entire speech into isolated sub-words with optimal boundaries (Shriberg 2000; Abdulla 2002; Delacourt and Wellekens 2000; Shafron and Rose 2003). In segmental modelling, speech parameters are represented by trajectories, that is, sequences of points in the parameter space. The speech trajectories have to be characterized using the mean, varian and shape of the particular segment. The shape of the signal has to be obtained using the wavelet coefficient as estimated earlier. Constructing a probability density function and using adaptive vector quantization over the training vector data set speech segmentation has to be done. Later on, loss-less compression methods: discrete cosine transform (DCT) and principle component analysis (PCA) have to be employed. Next the pre-processing (that is extraction of speech features) has to be done using hybrid approach of soft-computing techniques. Here the hybrid approach resemblances to artificial neural network (ANN) and genetic algorithm (GA). The hybrid approach has to be applied in a well-defined way in the present chapter that has been illustrated in the subsequent paragraphs.

2. Modelling of AWM and VDM

For the formation of AWM and VDM (as shown in figure 1), good distortion-free features have to be extracted. If this fails then the recognition process also degrades or suffers.

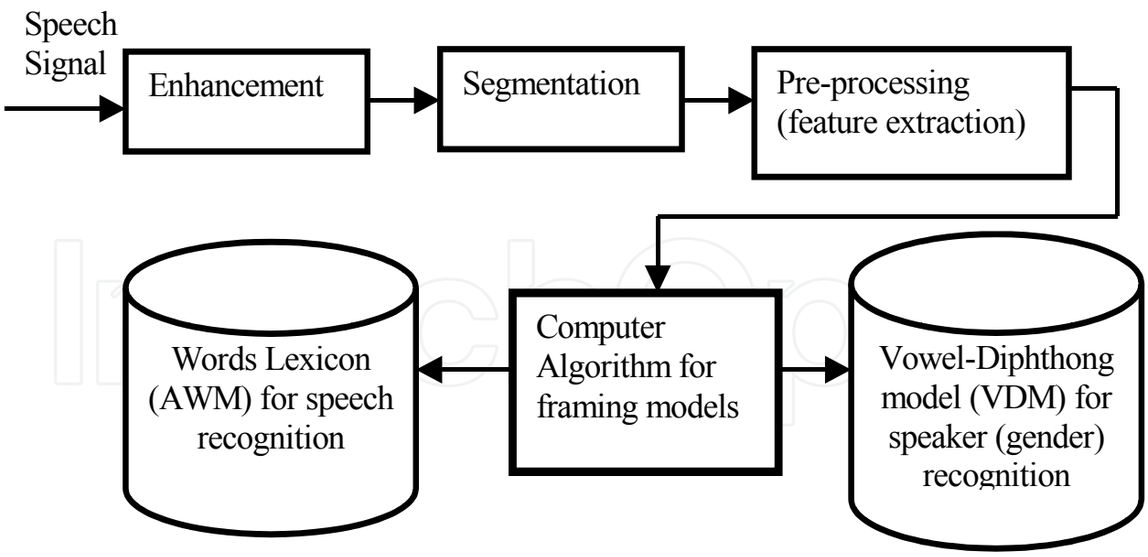


Fig. 1. General outline for framing models (AWM and VDM)

The most important speech features that have to be extracted are: pitch or fundamental frequency, formant frequencies or energy values, speaking rate, speech duration and so on, for the formation of models (that is AWM and VDM). Based on these features, twenty-two parameters have to be extracted for further analysis. The pitch related parameters that have

to be extracted are: mean, median, standard deviation, minimum and maximum range of pitch. The energy related parameters that have to be extracted are: mean, median, standard deviation, minimum and maximum range of loudness (energy). The duration related features that have to be extracted are: the ratio of voiced and unvoiced speech, speech rate. The speech rate has to be calculated by taking the ratio of duration of voiced speech to the total number of words uttered. All the above features have to be extracted using cepstral analysis, particle-filtering (Monte-Carlo) method from the spectrogram. The spectrogram based pitch and formant detection, have to be carried out (Hue et al 2001; Gustafson 2002; Arulampalam 2002; Vermaak 2002; Welling et al 1998). The formant frequency plays a vital role in giving the details about the vocal tract shape and its movements in various pronunciations. More illustrations have been done in the next subsequent subsections of this chapter.

2.1 Extraction of speech features

Before the extraction of speech features and relevant parameters, first, frequency bands corresponding to the analysed formants have to be extracted using forward-backward dynamic programming (FBDP) method. Next a particle-filtering method has to be applied to locate formants in every formant area based on the posterior probability density function (pdf) described by a set of support points with associated weights. As per the work done by Acero (1999) and Watanabe (2001), it has been found that capturing and tracking formants accurately from natural speech are very difficult task because of the variety of speech sounds. Typically, formant-tracking algorithms have three phases: *pre-emphasis*, *frame-dependent formant candidates*, and *generation-and-tracking*. For the first two phases, hypothesis testing and cepstral analysis has to be adopted. The third phase has to be carried out using the spectrogram-based particle-filtering method, because it is well known that the horizontal bands in grey-scale spectrogram with higher energy show the formant positions, which can be easily tracked in the spectrum. The mathematical analysis for this has been illustrated in the subsequent sections of this chapter. The main idea behind an acoustic model, that has been depicted in figure 2, with a trained data set, arranged in word-map, has to be categorized to estimate the best parameters that define the distribution (namely the mean, variance, shape and so on), and are represented as 'w%' where % = 1 to maximum size of the vocabulary.

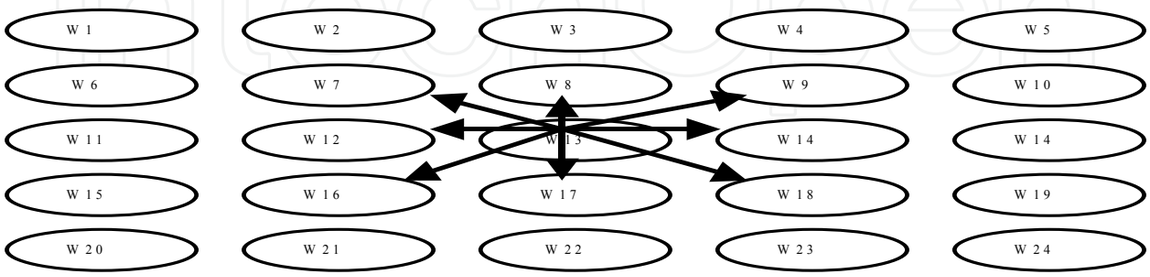


Fig. 2. A noise-free artificial word model (AWM)

The arrows are being designated as directions of constraints. In this model each phrase or word has to be linked with eight other corresponding word cell. Each word cell has to be trained and stored using forward-backward dynamic programming (FBDP) method. The

mathematical analysis has been also illustrated for three constraints in the subsequent sections of this chapter.

2.1.1 Formation of AWM and VDM with mathematical analysis

The speech production process is the initial step in the human speech communication system. This process is very complex and involves many components. During normal conversation, speaker stops or pauses for a while, because that time his or her brain might be busy in searching for an appropriate word from the vocabulary. If the brain has to be trained properly, every word cell in the brain gets activated and the speaker speaks continuously. If the training has not been done properly then a little amount of word cell gets activated and the speaker speaks with some pause. In a similar manner, the present experimental set-up has to be done, by forming an intermediate transient speech (ITRANS) table or master table.

To explain this formation of AWM in more illustrative way, initially the speech signal has to be captured through the microphone and background noise has to be removed successfully, using blind signal separation and spectral subtraction method. Then word boundaries have to be computed not only by using traditional zero-crossing measurement (ZCM) but also by using discrete cosine transform (DCT), because it has a strong energy compaction property. The main property is that it considers real-values and provides better approximation of a signal with fewer coefficients. With reference to figure 2, the model is being composed of many simple non-linear processors called neurons connected in parallel. Each neuron has an input and output characteristics and performs a computation or function of the form:

$$O_i = f(S_i) \text{ and } S_i = W^T X \quad (1)$$

where $X = (x_1, x_2, x_3, \ldots, x_m)$ is the input vector to the neuron and W is the weight matrix with w_{ij} being the weight (connection strength) of the connection between the j^{th} element of the input vector and i^{th} neuron. The f (.) is an activation or nonlinear function (usually a sigmoid), O_i is the output of the i^{th} neuron and S_i is the weighted sum of the inputs. A single neuron, as shown in figure 3, by itself is not a very useful tool for AWM formation.

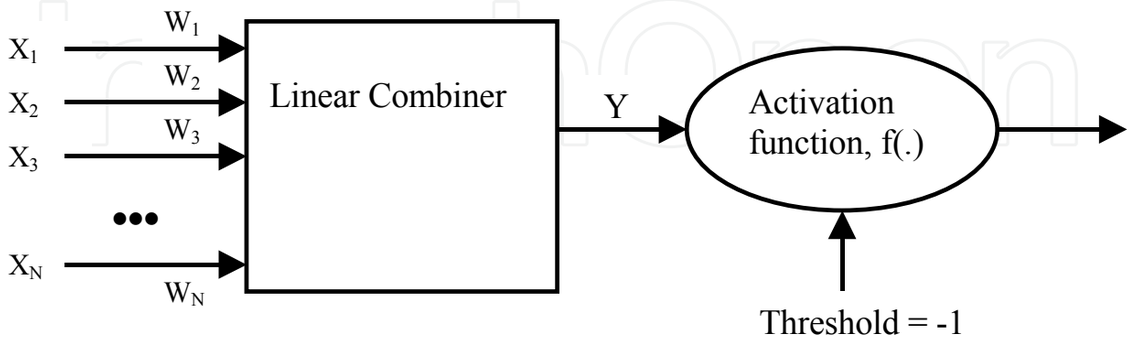


Fig. 3. An artificial neuron

The real power comes when a single neuron is combined into a multi-layer structure called neural networks (as shown in figure 4). The neuron has a set of nodes that connect it to the inputs, output or other neurons called synapses. A linear combiner is a function that takes

all inputs and produces a single value. Let the input sequence be $\{X_1,X_2,\ldots,X_N\}$ and the synaptic weight be $\{W_1,W_2,W_3,\ldots,W_N\}$, so the output of the linear combiner, Y, yields,

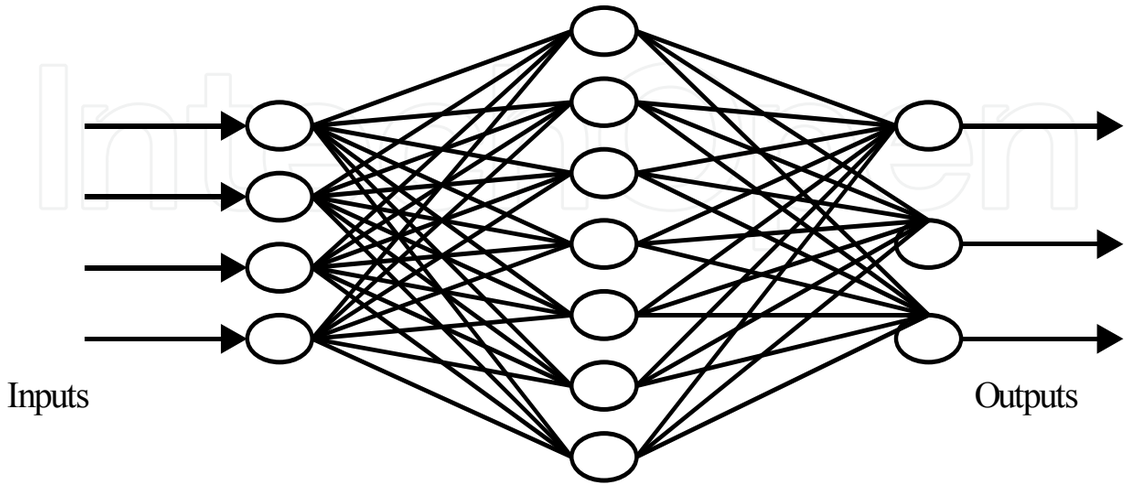


Fig. 4. A simple neural network

$$Y = \sum_{i=1}^{N} X_i W_i \tag{2}$$

An activation function will take any input from minus infinity to infinity and squeeze it into the range –1 to +1 or between 0 to 1 intervals. Usually an activation function being treated as a sigmoid function that relates as:

$$f(Y) = \frac{1}{1+e^{-Y}} \tag{3}$$

The threshold defines the internal activity of the neuron. This has to be kept fixed to –1. In general, for the neuron to fire or activate the sum should be greater than the threshold value. The learning capability is a result of the ability of the network to modify the weights through usage of a learning rule. Here, feed-forward network has to be used as a topology and backpropagation as a learning rule. A simple neural-network has been shown in figure 4, with inputs as speech feature values and a hidden and output layer.

The extracted speech feature values of each of the training sets have fed as input to the neural network. If 'ℓ' features have to be fed as input nodes then '2ℓ' nodes have to be used for the hidden layer. Each output neuron represents a word, thus only one output node has been treated. Here, twenty-two speech parameters have been extracted, hence twenty-two nodes in the input layer and forty-four nodes at the hidden layer and a single node at the output layer have been used. In order to minimize the error between the inputs and outputs of the neural-network, the weights of the threshold input has been adjusted using backpropagation algorithm (Wang et al 1999; Cybenko 1989; Hornik et al 1989; Ooyen et al 1992; Ismail et al 2004), which has also been depicted below (algorithm – 1):

Algorithm – 1: Backpropagation algorithm

Initialization: Initial weights w_i set to small random values;

Learning rate, $\eta = 0.1$

Repeat

For each training data set (x,y)

Calculate the outputs using the sigmoid function,

$O_j = \sigma(S_j) = \frac{1}{1+e^{-S_j}}$, where $S_j = \sum_{i=0}^{d} w_{ij} O_j$

$O_k = \sigma(S_k) = \frac{1}{1+e^{-S_k}}$, where $S_k = \sum_{i=0}^{d} w_{ik} O_k$

Compute the benefit β_k at the nodes 'k' in the output layer:

$\beta_k = O_k(1 - O_k)\,[Y_k - O_k]$

Compute the changes for weights j → k on connections to nodes in the output layer:

$\Delta w_{jk} = \eta \beta_k O_j$

$\Delta w_{ok} = \eta \beta_k$

Compute the benefit β_j for the hidden layer 'j' with the formula:

$\beta_j = O_j(1 - O_j)\,[\sum_k \beta_k w_{jk}]$

Compute the changes for the weights I → j on connections to nodes in the hidden layer:

$\Delta w_{ij} = \eta \beta_j O_i$

$\Delta w_{ok} = \eta \beta_j$

Update the weights by the computed changes: $w = w + \Delta w$,

Until termination condition is satisfied

Based on the assumption that the original spectral is additive with noise. To compute the approximate shape of the wavelet (i.e., Any real valued function of time possessing some structure), in a noisy signal and also to estimate its time of occurrence, two methods are available, first one is a simple structural analysis and the second one is the template matching technique. For the detection of wavelets in noisy signal, assume a class of wavelets, $S_i(t)$, I = 0,2,…..N-1, all having some common structure. Based on this assumption, consider a speech signal s(n) has to be corrupted by stationary additive noise d(n), to produce a noisy speech signal x(n) and has been modeled by the equation

$$x(n) = s(n) + G\,d(n) \tag{4}$$

where s(n) is the clean speech signal, d(n) is the noise and G is the term for signal-to-noise ratio control. Next windowing the signal and assuming G = 1, equation (4) becomes:

$$x_w(n) = s_w(n) + d_w(n) \tag{5}$$

Fourier transform of both sides of equation (5), yields:

$$X_w(e^{j\omega}) = S_w(e^{j\omega}) + D_w(e^{j\omega}) \tag{6}$$

Where $X_w(e^{j\omega})$, $S_w(e^{j\omega})$ and $D_w(e^{j\omega})$ are the Fourier transforms of windowed noisy, speech and noise signals respectively. To simply further, the notation the 'w' subscript has been dropped and multiplying both sides by their complex conjugates, it yields:

$$\left|X(e^{j\omega})\right|^2 = \left|S(e^{j\omega})\right|^2 + \left|D(e^{j\omega})\right|^2 + 2\left|S(e^{j\omega})\right|\left|D(e^{j\omega})\right|\cos(\Delta\theta) \tag{7}$$

where $\Delta\theta$ is the phase difference between speech and noise. Thus $\Delta\theta = \Delta\theta_S - \Delta\theta_D$.
The histograms for $\Delta\theta_S$, $\Delta\theta_D$, $\Delta\theta$ and cosine ($\Delta\theta$) has been shown in figure 5, figure 6, figure 7 and figure 8 respectively. For further analysis some assumptions have to be made: noise and speech magnitude spectrum values are independent of each other and also the phase of noise and speech are independent of each other. Taking the expected value of both sides of equation (7), and substituting $E\{\cos(\Delta\theta)\} = 0$, a power spectrum of the speech has been obtained which has been given in equation (8):

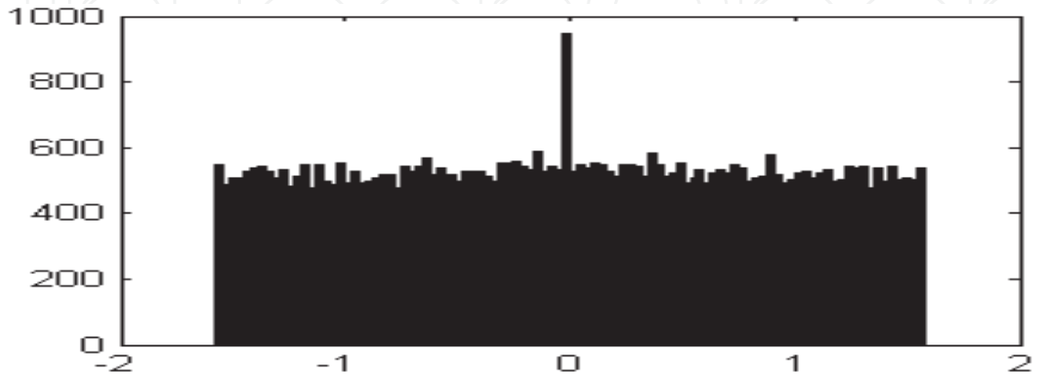


Fig. 5. Histogram of $\Delta\theta_S$ (speech magnitude spectrogram)

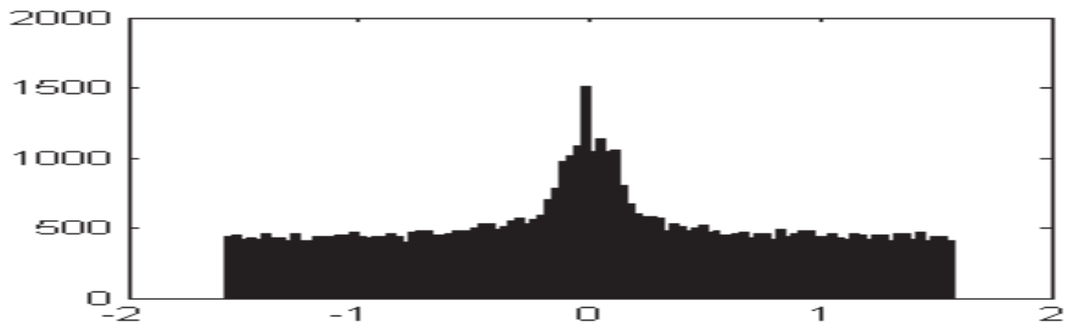


Fig. 6. Histogram of $\Delta\theta_D$ (noise magnitude spectrogram)

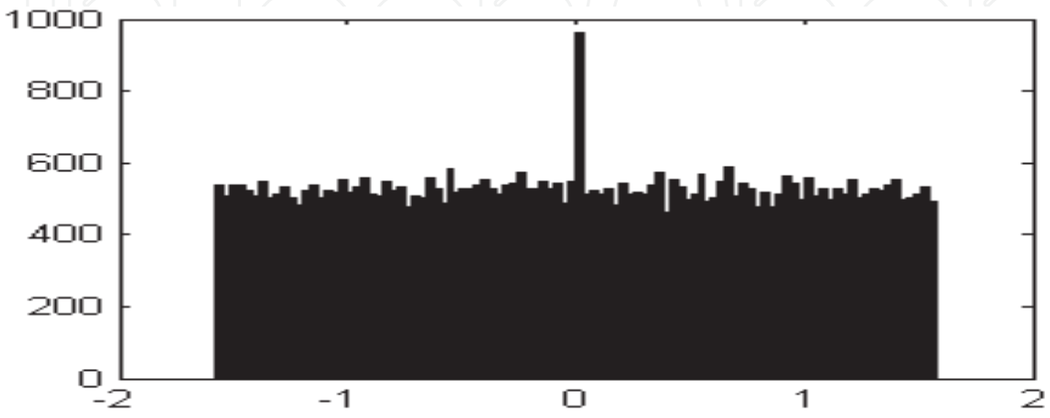


Fig. 7. Histogram of $\Delta\theta = \Delta\theta_S - \Delta\theta_D$

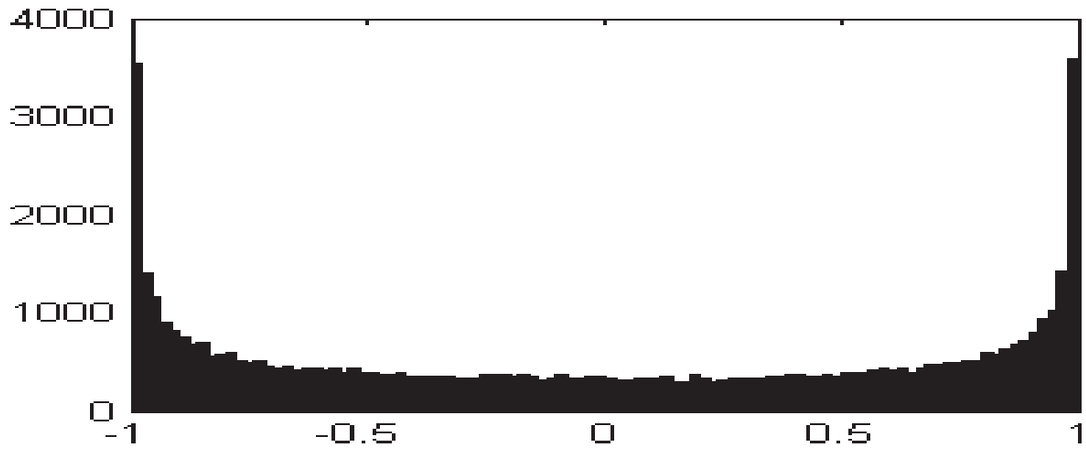


Fig. 8. Histogram of cosine ($\Delta\theta$) = cosine ($\Delta\theta_S$ - $\Delta\theta_D$)

$$\left|S(e^{j\omega})\right|^2 = \left|X(e^{j\omega})\right|^2 - E\left\{\left|D(e^{j\omega})\right|^2\right\} \tag{8}$$

Similarly, the magnitude spectrum of the speech, has to be estimated by substituting E {cos ($\Delta\theta$)} = 1, in equation (7) and hence computing the expected value of it. Thus it yields:

$$\left|S(e^{j\omega})\right| = \left|X(e^{j\omega})\right| - E\left\{\left|D(e^{j\omega})\right|\right\} \tag{9}$$

The histogram has been plotted for the equations (8) and (9) and has been depicted in figure 9. It has been observed from figure 9, that there remains some noise in the spectrum, as shown in narrow bands. This occurs due to the presence of negative values in the spectrum that are to be removed, hence these negative values has to be set to zero (Berouti et al. (1979)). To reduce the level of residual noise further subtraction has to be done. Thus equation (9) has been further simplified and it yields:

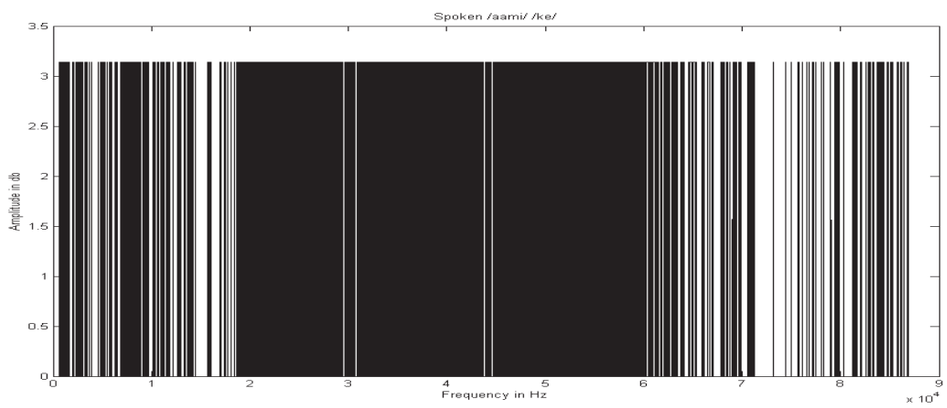


Fig. 9. Fluctuations of noise spectrogram for a speech uttered through microphone

$$\left|R(e^{j\omega})\right| = \left|X(e^{j\omega})\right| - \alpha E\left\{\left|D(e^{j\omega})\right|\right\} \tag{10}$$

where 'α' in the over-subtraction factor whose value should be more than unity. The residual noise has to be practically categorized and has been shown in figure 10.

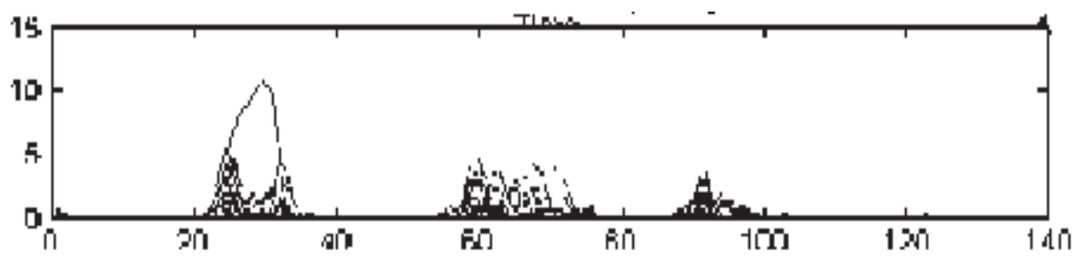


Fig. 10. Noise spectrum without negative values

To de-noise further this noisy speech signal, discrete wavelet transform has to be applied. Let the mother wavelet or basic wavelet be ψ(t), which yields:

$$\psi(t) = \exp(j2\pi ft - t^2/2) \tag{11}$$

Further as per the definition of Continuous wavelet transform CWT (a,τ), the relation yields to:

$$\text{CWT}(a, \tau) = (1/\sqrt{a}) \int x(t)\, \psi\{(t-\tau)/a\}\, dt \tag{12}$$

The parameters obtained in equation (12) has to be discretized, using discrete parameter wavelet transform, DPWT (m, n), by substituting a = a_0^m, τ = n $\tau_0 a_0^m$. Thus equation (12) in discrete form results to equation (13):

$$\text{DPWT}(m, n) = 2^{-m/2} \sum_k x(k)\, \psi(2^{-m}k - n) \tag{13}$$

where 'm' and 'n' are the integers, a_0 and τ_0 are the sampling intervals for 'a' and 'τ', x(k) is the speech signal. The wavelet coefficient has been computed from equation (13) by substituting $a_0 = 2$ and $\tau_0 = 1$.

Once the residual noise has been lowered, further analysis has to be done for the segmentation stage of speech processing. In classification problems with two or more classes, it is often required to choose a subset of 'd' speech features out of the given 'n' speech features (d < n). To do this, a measure of class separability has to be done. In the present work, scatter matrices have been used to form a separability criterion. A criterion for separability can be any criterion which is proportional to the between scatter matrix and also proportional to the inverse of the within scatter matrix. Maximization of such a criterion will ensure that while maximizing the distance between classes, there is no sufficient amplification in the scatter of the classes, thus causing no improvement to the separability. This scatter matrix has to be computed by finding the covariance matrix of the speech features in a given class. This has to be computed on the application of Fishers linear discriminant analysis (FLDA) which is a transformation that reduces the dimensionality of the feature vector from 'n' into d = M – 1 (where M is the number of classes involved), while

optimally preserving the separability between classes. The idea behind the Fisher's linear discriminant is the projection of n dimensional feature vectors onto a lower dimensional surface. The surface has to be chosen in such a way that the separation between classes must be kept at a minimum distance of the regression line. In order to find the optimal surface to project onto, a measure of separability must be done.
Consider a two-class problem with 'N' known samples, 'X_i'. 'N_1' of which belong to class 'w_1' and 'N_2' of which belong to class 'w_2'. Consider 'Y_i' be a linear combination of the features 'X_i' :

$$Y_i = \rho^T X_i \tag{14}$$

The 'n' dimensional vector, ρ, can be considered a line in the n dimensional space, then Y_i is the projection of X_i on this line (scaled by $\|\rho\|$). Let 'μ_I' be the mean of the 'N_i' samples of class 'w_i' in the 'n' dimensional space:

$$\mu_i = \frac{1}{N_i}\sum X_i \tag{15}$$

and the mean of the projected points Y_i on the line ρ, $\overline{\mu_i}$, is the projection of μ_i :

$$\overline{\mu_i} = \frac{1}{N_i}\sum Y_i = \frac{1}{N_i}\sum \rho^T X_i = \rho^T\mu_i \tag{16}$$

The separation of the means on ρ is given by,

$$\left|\overline{\mu_1} - \overline{\mu_2}\right| = \left|\rho^T(\mu_1 - \mu_2)\right| \tag{17}$$

Since the separation of the two classes must include the variance of three samples. Defining the n x n scatter matrix,

$$w_i = \sum_{\beta \in w_i} (X - \mu_i)(X - \overline{\mu_i})^T \tag{18}$$

where w_i is the estimation of the covariance of the i^{th} class in the n-dimensional feature space. It represents a measure of the dispersion of the signals belonging to w_i. Thus the total matrix is defined as:

$$W = w_1 + w_2 \tag{19}$$

Consider now the variance between the means of the various classes. Denote the matrix, B, represents the dispersion between the means of the various classes.

$$B = (\mu_1 - \mu_2)\left(\overline{\mu_1} - \overline{\mu_2}\right)^T \tag{20}$$

The auto-covariance of the speech signal has also to be computed using the relation:

$$C_{xx}(g) = E\{[x(nT) - x'(nT)][x(nT) - x'(nT)]\} \tag{21}$$

Then the power spectrum density has to be calculated from equation (21) and it yields to,

$$P_E(f) = \sum_{m=0}^{N-1} C_{xx}(m) W \exp(-j2\pi fm) \tag{22}$$

where $C_{xx}(m)$ is the auto-covariance function with 'm' sample. The data compression has to be performed using discrete cosine transform as shown below,

$$X_c(k)=\mathrm{Re}[X(k)]= \sum_{n=0}^{N-1} x(n)\cos\left(\frac{k2\pi n}{N}\right), k=0,1,..,N-1 \tag{23}$$

Further reduction in the dimensionality of the feature vector has to be carried out using principal component analysis. For this first discrete fourier transformation (DFT) method has to be employed, it yields to the equation of the form:

$$X(k)=F_D[x(nT)]= \sum_{n=0}^{N-1} x(nT)\exp\ (-jk(2\pi / N)n) \tag{24}$$

where k = 0, 1, 2, ……..,N-1. If W_N = exp (-j2π/N), then equation (24) becomes,

$$X(k) = F_D[x(nT)] = \sum_{n=0}^{N-1} x(nT) W_N^{kn} \tag{25}$$

Further for the computation of principal components (i.e., eigen values and the corresponding eigen vectors), a pattern vector $\overline{p_n}$, which can be represented by another vector $\overline{q_n}$ of lower dimension, has to be formulated using (25) by linear transformation. Thus,

$$\overline{p_n} = [M]\ \overline{q_n} \tag{26}$$

where $[M] = [X(k)]$ for k = 0 to N-1 and $\overline{q_n}$ = min([M]), such that $\overline{q_n} > 0$

Taking the covariance of equation (26), it yields, the corresponding eigen vector,

$$\overline{P} = \mathrm{cov}(\overline{p_n}) \tag{27}$$

And thus,

$$\overline{P}\ .\ M_i = \lambda_i\ .\ M_i \tag{28}$$

where 'λ_i' are the corresponding eigen values.

One of the fundamental problems that arise when computing two patterns is that of time scaling. Most of the researchers assume that both pattern and template (reference) to be compared share the same time base. This is not always correct, especially in speech analysis. It has been found that when a speaker utters the same word several times, generally each utterance he / she does so with different time bases. Each word is spoken such that parts of it are uttered faster, and parts are uttered slower. The human brain, it seems, can easily overcome these differences and recognize the word. But machine finds this a severe difficulty for the recognition. Due to this there may be a chance of occurring out-of-vocabulary (OOV) word problem. In order to sort out such problem, dynamic time warping

(DTW) method has to be adopted. Mathematically, this has been analyzed in the subsequent paragraphs.

Assume two speech signals, say $x(t_i)$ and $x(t_j)$ are defined, each with its own time base, t_i and t_j. Also assume that the beginning and end of the speech signal are known, denoted as (t_{is}, t_{if}) and (t_{js}, t_{jf}) respectively. If both the signals are sampled at the same rate, then both signals begin t sample i = j = 1, that occurs without any loss of generality. Thus, the mapping function, i = j . (i / j), is linearly related. Since speech signals are non-linear, so non-linear time warping functions must be calculated, with several assumptions. Let the warping function, w(k), be defined as a sequence of points: c(1), c(2), …..,c(k), where c(k) = (i(k), j(k)) is the matching of the point i(k) on the first time-base and the point j(k) on the second time-base. This has been summarized in figure 11, below. From figure 11, the warping, w(k), only allows to compare the appropriate parts of $x(t_i)$ with that of $x(t_j)$. Setting the monotonic and continuity conditions on the warping function, it restricts to the relations between two consecutive warping points, c(k) and c(k-1).

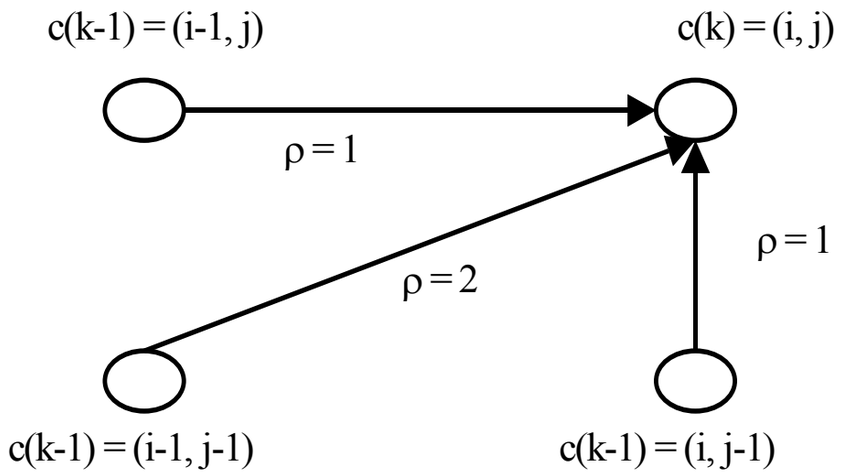


Fig. 11. Constraints on the dynamic time warping (DTW)

Thus from figure 11, there are three ways to get to the point c(i,j), which is given below,

$$c(k) = c(i,j) \tag{29}$$

and

$$c(k-1) = \begin{cases} (i(k), j(k)-1) \\ (i(k)-1, j(k)-1) \\ (i(k)-1, j(k)) \end{cases} \tag{30}$$

Thus the boundary condition is,

$$c(k) = (i,j) \tag{31}$$

By the boundary condition, matching of the beginning and end of the speech signal has to be done using dynamic programming method. As the measures have not been stored then it is difficult to trace the optimal route in an artificial word model (AWM) for the best match of the word. To overcome such problem, forward-backward dynamic programming (FBDP) method has to be adopted.

Next for the formation of AWM, the extracted speech feature values have to be fed as input to the neural network. Let the input sequences are $\{x_1, x_2, \ldots, x_n\}$, which takes real values within the range (-n, n). The weights $w_1, w_2, \ldots, w_n$, correspond to the synaptic strengths of the neuron. They serve to increase or decrease the effects of the corresponding 'x_i' input values. The sum of the products $x_i * w_i$, i = 1 to n, serve as the total combined input to the node.

So to perform the computation of the weights, assume the training input vector be 'G_i' and the testing vector be 'H_i' for i = 1 to n. The weights of the network have to be re-calculated iteratively comparing both the training and testing data sets so that the error is minimized.

The weight matrix $W = \{w_{ij}\}$ has to be computed through the relation,

$$w_{ij} = \sum_{r=1}^{n} G^T(r)H(r) \tag{32}$$

To compute the net input to the output units, the delta rule for pattern association is employed, which is given by the relation,

$$y_{\text{-inj}} = \sum_{i,j=1}^{n} x_i w_{ij} \tag{33}$$

where '$y_{\text{-inj}}$' is the output pattern for the input pattern 'x_i' and j = 1 to n.

Thus the weight matrix for the auto-associative memory neural network has to be calculated from equation (33) and the responses have to be checked by the trained input patterns. The output vector 'y' gives the pattern associated with the input vector 'x'. An activation function (usually a unipolar sigmoid) will take any input from minus infinity to infinity and squeeze it into the range –1 to +1 or between 0 to 1 intervals. A unipolar sigmoid function relates as:

$$f(y_{\text{-inj}}) = \frac{1}{1+e^{-y}} \tag{34}$$

Thus the neural network model can learn from the input / output training data pairs. Once the training has been done, it can be used as a function simulator. Similarly for framing vowel-diphthong model (VDM), the same (above discussed) concept for AWM formation, has to be followed. The constraint is that only vowel sounds are to be captured through microphone from male and female subjects. An algorithm called SCB_AWM_VDM (soft-computing based artificial word model and vowel – diphthong model) has been developed, which has been depicted in algorithm – 2 below.

Algorithm - 2: SCB_AWM_VDM

1. Record a speech through a microphone and store it in a file with extension wav.
2. Find the length of the speech signal, say N.
3. Create the row vectors 'n' and 'k' such that $0 \le n, k \le N - 1$.

4. Employ BSS and SS methods
5. Employ DFT and compute the auto-correlation coefficients. Find the minimum, maximum, mean value of the amplitude and pitch of the analyzed signal, and PSD function and probability distribution function. Compress the data further by employing DCT and also find the principal component values.
6. Find the length of the compressed speech signal. Apply AVQ technique and create a token of words
7. Count the number of frames. Pass it to a file, say atm.dat with the fields of the database as frame-1, frame-2,……,frame-N.
8. Compute the range of parameters using the relation as,
   UB = upper bound = (((mmax – mmean) / 2)*A) + mmean
   LB = lower bound = (((mmean – mmin) / 2)*A) + mmin
   where 'A' is the pre-emphasis coefficient
9. Using each of the frames extract the parameters and store in a master file as a template, thus forming a noise-free artificial word model and vowel-diphthong model.

2.2 Performance measures of developed algorithm for modelling: A case study

The number of training samples per word has to be kept sufficient for improving the accuracy of pattern matching and hence increases the performance factor. With this small condition, the developed algorithm called SCB_AWM_VDM has to be applied for the formation of a noise-free artificial word model (AWM) taking into consideration 22 speakers of varying age groups. The vocabulary has to be limited to 215 Bengali (Indian Language) words. In this work, nine male and six female adults of age group 30 – 40 years, five male adults of age group 40 – 50 years and two male adults above 50 years of age have to be selected for the testing of developed algorithm. Each phrase has to be uttered five times by each speaker leading to a total size of (22 x 215 x 5) 23650 Bengali words. Some of the speech samples have to be collected through e-mail through attachments and tested with the proposed algorithm for the formation a large Bengali vocabulary. Here, Bengali (an Indian) language has to be adopted as a case study. The developed algorithm has to be used as a tool for the formation of AWM and can also be used independently with some natural language. The worst-case time complexity is Oh (N * log(N)) and the worst-case space complexity is Oh (N*P) of the developed algorithm, where 'N' means total number of words and 'P' means the number of speech features (here P = 22). The complexities of the developed algorithm have been shown in figure 12.

From figure 12, it has been observed that accuracy increases almost exponentially. Similarly, vowel-diphthong model (VDM) has to be formed using the developed algorithm taking into consideration 10 speakers of varying age groups. Here, for VDM formation in Bengali (an Indian) language, six male and four female adults of age groups 30 – 40 years have to be selected for the testing of developed algorithm.

More discussions based on the practical implementation of the developed algorithm for the formation of a noise-free AWM and VDM can be made. Figure 13a, shows the original speech signal uttered /bondo/ /koro/ in Bengali means /close/ /it/ in English, has to be segmented using adaptive vector quantization (AVQ) method. The segmented result has been shown in figure 13b. After segmenting the words uttered, the features have to be extracted, using discrete cosine transform (DCT) and principal component analysis (PCA).

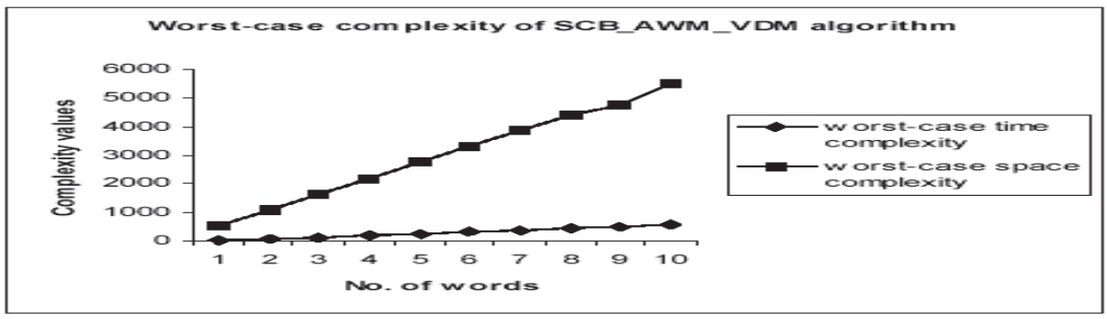


Fig. 12. Complexities of developed SCB_AWM_VDM algorithm

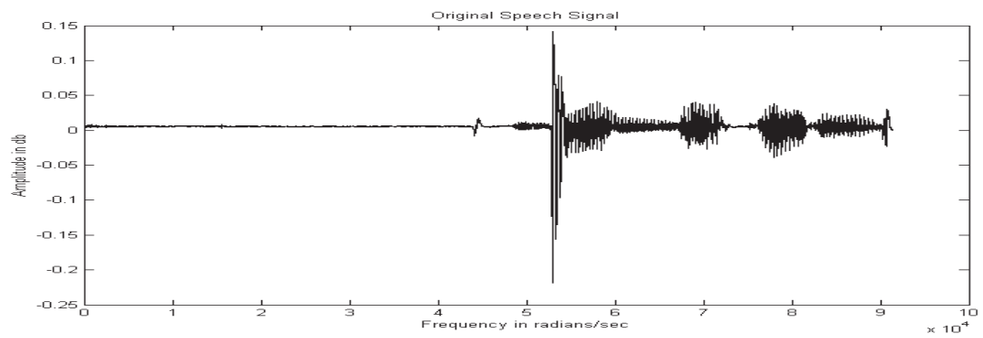


Fig. 13a. Original speech signal spoken /bondo/ /koro/

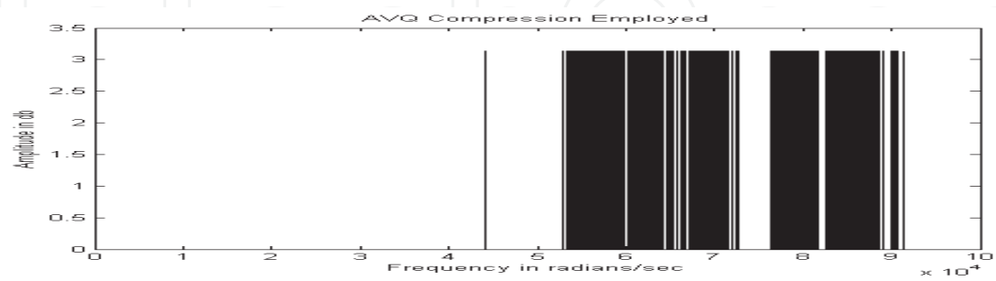


Fig. 13b. Segmentation using adaptive vector quantization (AVQ) method

The vocal tract model frequency response correspond to the position of the formants of the speech signal along frequency axis has been shown in figure 14.

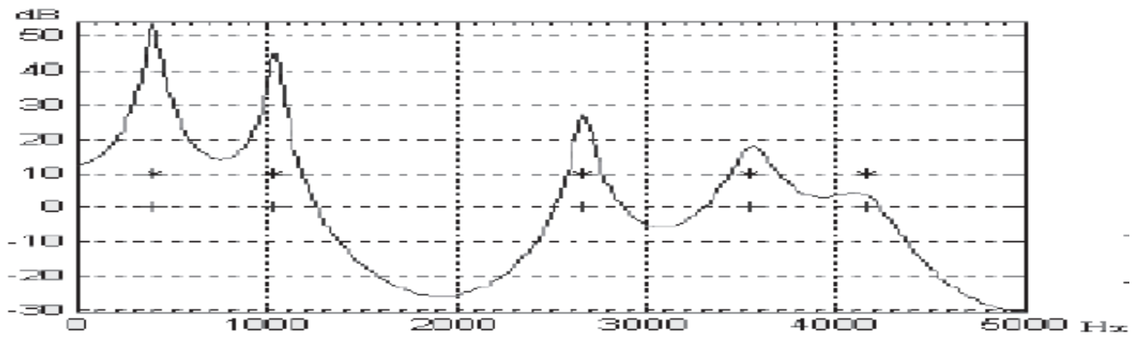


Fig. 14. Vocal tract model frequency response where + and * correspond to the position of the formants of the speech signal along frequency axis.

In figure 14, + and * shows the positions of the formants of the speech signal. Some patterns has been shown related to segmentation of the voiced pitch of the speech signal using AVQ method. Figure 15 and figure 16 shows the proper voiced pitch features using AVQ technique. It has been observed that segmentation is more accurate using AVQ technique as compared to traditional usage of zero-crossing measurement (ZCM).

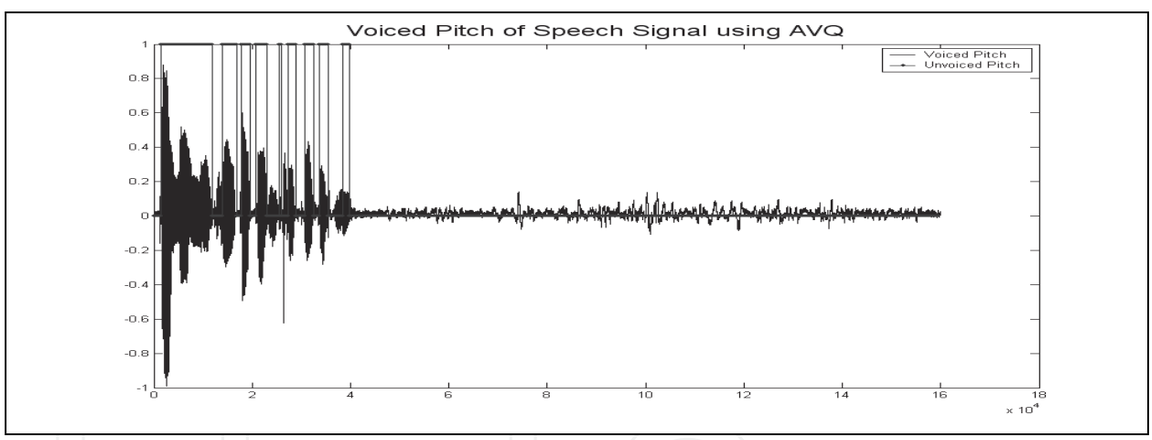


Fig. 15. Voiced pitch of the speech signal segmented using AVQ method

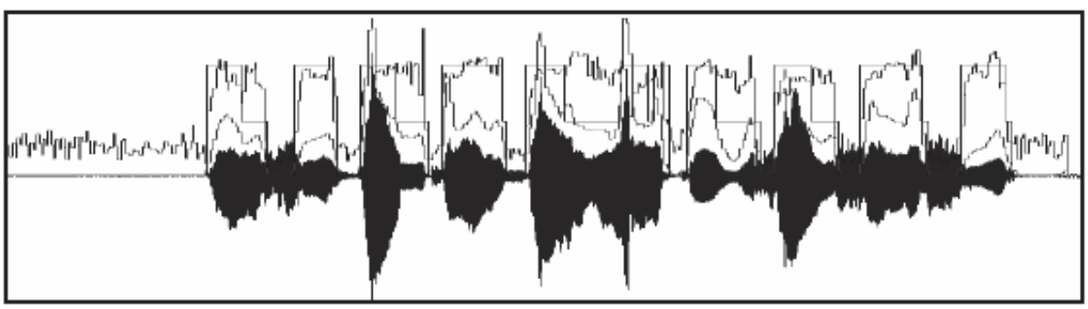


Fig. 16. Proper voiced pitch segmentation of the speech signal using AVQ method

To track the formats of the speech signal, in the present work, five formants (F_1, F_2, F_3, F_4 and F_5) have to be tracked that has been shown in figure 17.

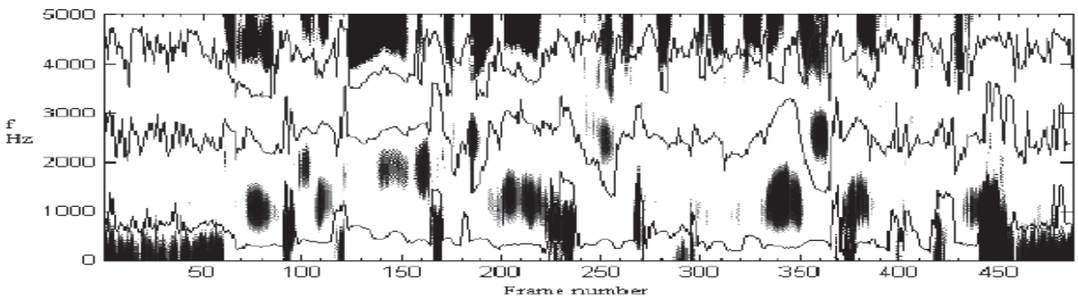


Fig. 17. Five formants have to be tracked on uttering /aakashe/ /pakhi/ /oodche/ in Bengali by a male speaker means /A/ /bird/ /is/ /flying/ /in/ /the/ /sky/ in English.

The twenty-two speech features related to pitch, energy or loudness, duration, formants and speaking rate have been depicted in table 1 and table 2.

Features	lb	μ	ub	m	σ
F_0 or pitch	-41.64	-5.71	30.23	-5.69	0.3966
Energy	-0.07	0.03	0.10	0.95	0.3962
Duration	0.00	1.16	3.19	--	--
Formants	0.0164	0.0167	0.0171	0.0128	0.4376
Speaking rate	0.00	0.91	0.83	0.88	--

Table 1. Enhanced parameters extracted from a speech signal (sample #1)

Features	lb	μ	ub	m	σ
F_0 or pitch	-41.64	-5.71	30.23	-5.69	0.3966
Energy	-0.07	0.03	0.10	0.95	0.3962
Duration	0.00	1.16	3.19	--	--
Formants	0.0164	0.0167	0.0171	0.0128	0.4376
Speaking rate	0.00	0.91	0.83	0.88	--

Table 2. Enhanced parameters extracted from a speech signal (sample #2)

3. Understanding of AWM and VDM

In this section of the chapter, hybrid approach of soft-computing techniques has to be used for the understanding of AWM and VDM for the recognition of speech and speaker. The

important speech features $K_a = \{k_1,k_2,\ldots, k_m\}$ extracted from speech signal has to be modeled as a function of the super-flame information using a neural network of soft computing techniques. This model has to be used then in conjunction with a genetic algorithm to obtain optimized super-flame information resulting in low K_a values from the model (AWM). This has to be carried out by estimating, the mapping-function between inputs and outputs of the model. Such functions are usually highly non-linear and have to be computed using adaptive vector quantization (AVQ) (Tolvi (2004); Hongwei et al (2005)) based unidirectional temporary associative memory (UTAM) of neural-network, as depicted in figure 18.

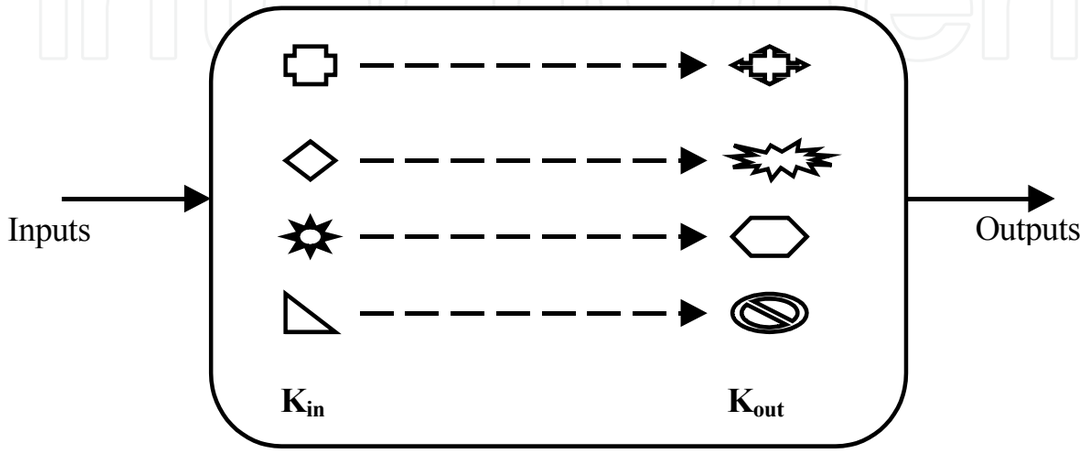


Fig. 18. Unidirectional temporary associative memory (UTAM)

In figure 18, the inputs K_{in} and the outputs K_{out} components have been shown. The term unidirectional has to be used because each K_{in} component is mapped with K_{out} component with one-to-one relationship. Each component has to be designated with a unique codeword. The set of codewords is called a codebook. The concept of UTAM has to be employed in the present work, as mapping-function for two different cases:

1. Distortion measure between unknown and known speech signals
2. Locating codeword between unknown and known speech feature

To illustrate these cases, Let $K_{in} = \{I_1,I_2,\ldots,I_n\}$ and $K_{out} = \{O_1,O_2,\ldots\ldots,O_m\}$ consisting of 'n' and 'm' input and output codeword respectively. The values of 'n' and 'm' are the maximum size of the vocabulary set. In the recognition stage, an unknown speech signal, represented by a sequence of feature vector, $U = \{U_1,U_2,\ldots,U_u\}$, has to be compared with a known speech signal stored in the form of model (AWM), represented by a sequence of feature vector, $K_{database} = \{K_1,K_2,\ldots,K_q\}$. Hence to satisfy the unidirectional associatively condition, i.e., $K_{out} = K_{in}$, an artificial word model (AWM) has to be utilized for proper matching of words. The matching of words, have to be performed on computing the distortion measure. The word with lowest distortion has to be chosen. This yield to, the relation,

$$C_{found} = \underset{1<=q<=n}{\arg\min}\left\{S(U_u,K_q)\right\} \tag{35}$$

The distortion measure has to be computed by taking the average of the Euclidean distance

$$S(U,K_i) = \frac{1}{Q}\sum_{i=1}^{Q} d(u_i,C_{\min}^{i,q}) \tag{36}$$

where $C_{\min}^{i,q}$ denotes the nearest word in the template or AWM and d(.) is the Euclidean distance. Thus, each feature vector in the sequence 'U' has to be compared with the codeword in AWM, and the minimum average distance has to be chosen as the best-match codeword. If the unknown vector is far from the other vectors, then it is very difficult to find the word from the AWM, resulting to out-of-vocabulary (OOV) word problem. Assigning weights to all the codewords in the database (called weighting method) has eliminated the OOV word problem. So instead of using a distortion measure a similarity measure that should be maximized are considered. Thus it yields,

$$S_w(U,K_i) = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{d(u_i,C_{\min}^{i,q})}w(C_{\min}^{i,q}) \tag{37}$$

Dividing equation (36) by equation (37), it yields,

$$\gamma = \text{recognition rate} = \frac{S(U,K_i)}{S_w(U,K_i)} = \frac{unweighted}{weighted} \tag{38}$$

The procedure for computing the weights, has been depicted in an algorithm – 3 below:

Algorithm – 3: procedure to compute weight (S)

```
for each C_I in S do
    for each C_J in C_I do
            sum = 0
            for each C_K and K != I, in S do
               d_min = distancetonearest(C_J,C_K);
                    sum = sum + 1 / d_min;
             endfor;
        w(C_IJ) = 1 / sum;
    endfor
endfor
return weights = w(C_IJ)
```

Next for locating the codeword, hybrid approach of soft computing has to be applied in the well-defined way in the present chapter. The hybrid approach of soft computing techniques utilizes some bit of concepts from forward-backward dynamic programming and some bit of neural-networks. The work done by Fernando Bacao et al (2005), S. H. Ling et al (2007), H. Sakoe et al (1997), R. S. Chang et al (1978), and C. Y. Chang et al (1973), have been extended by considering eight constraints for searching the AWM using concepts of genetic algorithm (GA), for the best match of the uttered phrase. In general, for an optimal solution, GA is the best search algorithm based on the mechanics of natural selection, crossover and mutation. It combines survival of the fittest among string structures with a structured yet randomized information exchange. In every generation, new sets of artificial strings are created and hence tried for a new measure. It efficiently exploits historical information to speculate on new search points with expected improved performance. In other words genetic algorithms are theoretically and computationally simple and thus provide robust and optimized search methods in complex spaces. The selection operation has to be performed by selecting the speech signal as chromosomes from the population with respect to some probability

distribution based on fitness values. The crossover operation has to be performed by combining the information of the selected chromosomes (speech signal) and generates the offspring. The mutation operation has to be utilized by modifying the offspring values after selection and crossover for the optimal solution. Here in the present chapter, an acoustic model signifies the population of genes or speech parameters. Using genetic algorithm along with associative memory technique a similar type of work has been done by Tilendra Shishir Sinha et al (2006) for the recognition of speech and speaker using proposed GSAMTSS (genetic search with associative memory technique for the speech and speaker) algorithm. The methodology adopted was different in classification and recognition process and the work has been further modified by them and has been highlighted in the present part of the book using soft computing techniques of genetic and artificial neural network.
In many applications, speech signals have been either stored for later use or transmitted over some media. In both cases, interest lies in reducing the size of the signal because of cost, time and certain other benefits. As an application of the previous work done by Tilendra Shishir Sinha et al (2006), it has been discussed properly how the speech parameters can be embedded within images for promoting global cyber security through steganalysis on a standalone machine. Further some of the results reported by Yuexi Ren et al (2004), have described the semantic analysis for proper speech user interface in an intelligent tutoring system. In the work carried out earlier by Tilendra Shishir Sinha et al (2006), regarding the recognition of the gender of the speaker, the threshold values of the genders of the speakers have to be assumed which creates a void and the present work has attempted to fill in this void using mathematical analysis and employing known algorithms like polar algorithm. For the recognition of a speech and speaker, experimental setup has been summarized in figure 19.

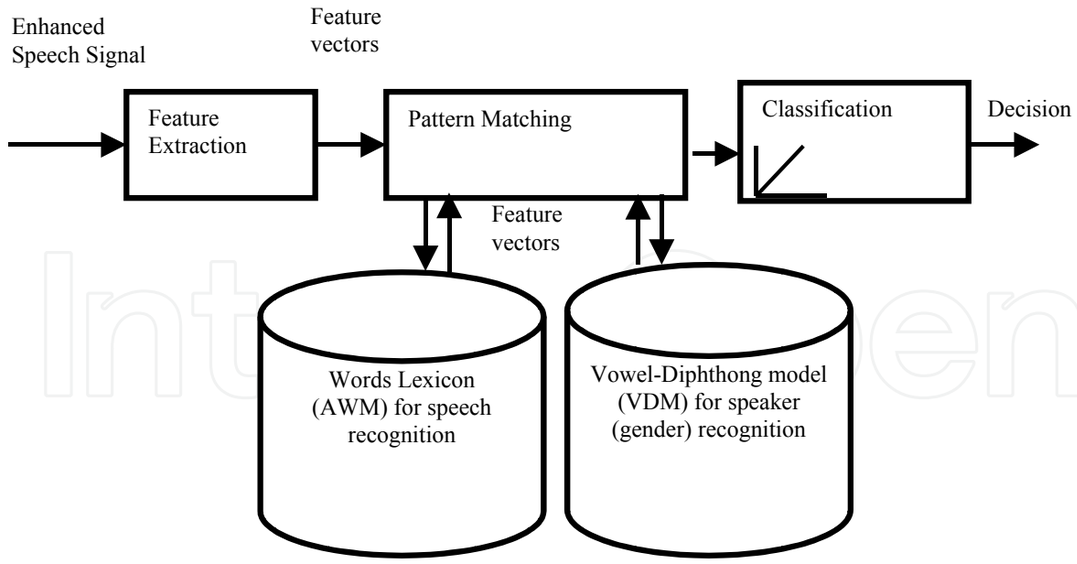


Fig. 19. General framework for recognition process

For the recognition of speech, first the speech signals (test signal) have to be made free from noise using blind signal separation (BSS) and spectral subtraction (SS) method. Hence segmented and features have to be extracted and computed the distance measures between trained data set and test data set. Hence searched for the codeword from the Bengali

vocabulary (master database or template) depending upon the minimum distance measures. If test and trained pattern matches then the decision has to be taken with: *acceptance* or *rejection*.
Generally, gender or speaker recognition is being carried out with prior knowledge of vowel uttered by the speakers (Doddington 2001). Using two vowels {o, e} the present work has to be carried out, because most of the Bengali sentences end either with 'o' or with 'e'. Extracting formant based speech features from the word model has carried this part of recognition for the gender. Depending upon the best matches decision has to be taken whether the speaker is 'male' or 'female'. The analysis has to be done using a two-class (male and female) problem. The above theoretical study has to be implemented experimentally and mathematical analysis related to the work has been also discussed.
For the best match of the words spoken in Bengali language, the developed algorithm called RCGSTSS (real-coded genetic search technique for the speech and speaker recognition) has been employed (Tilendra Shishir Sinha and Gautam Sanyal 2009).

Algorithm – 4: RCGSTSS algorithm

1. Capture a test speech signal through microphone and store it in a file with extension wav. Find the length of the test speech signal, say, M. Create the row vectors 'n' and 'k' such that $0 \leq n, k \leq M-1$
2. Assign spch_wer = 0, spch_acc = 0
3. Randomly generate an initial population $X(o) = (X_1, X_2, \ldots, X_M)$
4. Read the size of the trained model, say, Q, and set the counter, say, q = 0
5. Do while ($q \leq Q$)
   Compute the fitness $f(X_i)$ of each individual X_i of the current population.
   Generate an intermediate population $X_r(t)$ applying the reproduction operator.
   Generate X(t+1) applying other operators to $X_r(t)$.
   Increment t = t + 1
   Increment the counter, q = q + 1
   If matched then
   spch_text = popup_from_UTAM(spch_code_word)
   spch_text = concatenate(spch_text)+concatenate(space(2))
   else
   spch_wer = spch_wer + 1
   End if
   If q = Q then
   spch_vowel = popup_from_UTAM(spch_code_word)
   spch_vowel = max(spch_vowel)
   if spch_vowel >= threshold then
   speaker = 'Female'
   else
   speaker = 'Male'
   End if
   else
   Display 'Not matched'
   End if
   End do
1. Display 'Spoken words in: ' + spch_text + 'Word error rate: '+spch_wer
2. Display 'Gender of Speaker: '+speaker + 'Recognition accuracy:'+ spch_acc

4. Conclusion

In order to recognize certain Bengali phrases spoken in speaker independent environment a noise-free artificial word model and vowel diphthong model has to be developed using blind signal separation method, spectral subtraction method, and adaptive vector quantization. It has been also observed that the residual noise occurs because of the channel coupling, which has to be minimized using spectral subtraction method by further adjusting the 'over subtraction' coefficient 'α'. Due to channel coupling most of the valuable information in speech signals goes un-observed. This problem has to be also rectified using blind signal separation method and filtered further using loss-less compression technique called discrete cosine transform and also employed principal component analysis for further analysis. The developed algorithm called SCB_AWM_VDM (soft computing based artificial word model and vowel-diphthong model) has to be implemented to train the system considering 22 speakers of varying age groups for AWM formation and 10 speakers of varying age groups for VDM formation. The vocabulary has to be limited to 215 Bengali words, considering, nine male and six female adults of age group 30 – 40 years, five male adults of age group 40 – 50 years and two male adults above 50 years of age. Each phrase has to be uttered five times by each speaker leading to a total size of (22 x 215 x 5) 23650 Bengali words for overcoming the out-of-vocabulary (OOV) word problem during recognition process of the speech and the speaker of Bengali language. Hence for simultaneous automatic speech and speaker recognition (ASSR) in combination with AWM another model called VDM has to be used. These models have to be used properly by adopting the methods forward-backward dynamic programming (FBDP) and genetic algorithm (GA). The divergence has to be calculated and also comparison has to be made with the observation that GA converges with optimal solution, thus improving the performance of the complete dialogue system. The developed algorithm (RCGSTSS) has to be successfully tested with necessary experimental data.

5. References

Yilmaz O, Rickard S., 2004, *Blind separation of speech mixtures via time-frequency masking*, IEEE Transactions on Signal Processing, 52(7): 1830-1847.

Yamashita J, Hirai Y, 2004, *Blind source separation using orientation histograms in joint mixture distributions*, In the Proceedings of Neural Network Computing and Intelligence, pp 152-157.

Araki S, Makino S, Mukai R, Hinamoto Y, Nishikawa T, Saruwatari H, 2002, *Equivalence between frequency domain blind source separation and frequency domain adaptive beam forming*, In *the* Proceedings of ICASSP, vol II, pp 1785-1788.

Murata N, Ikeda S, Ziehe A, 2001, *An approach to blind source separation based on temporal structure of speech signals*, Neuro Computing 41: 1-24.

Jang, G. J. and Lee, T.W., 2003, *A maximum likelihood approach to single channel source separation*, in the Proceedings of JMLR, vol. 4, pp. 1365-1392.

Bach, F. and Jordan, M.I., 2005, *Blind one-microphone speech separation: A spectral learning approach*, in the Proceedings of NIPS, pp. 65-72.

Pearl Mutter, B. A. and Olsson, R. K., 2006, *Algorithms differentiation of linear programs for single-channel source separation*, in the Proceedings of MLSP.

Berouti, M., Schwartz, R., and Makhoul, J., 1979, *Enhancement of speech corrupted by acoustic noise*, in *the* Proceedings of the IEEE Conference ASSP, April, pp. no. 208-211.

Yasser Ghanbari and Mohammad Reza Karami, 2004, *Spectral subtraction in the wavelet domain for speech enhancement*, International Journal of Softwares and Information Technologies, vol 1, no.1, August, pp. no. 26-29.

Shriberg, E., Stoleke, A., Hakkani-Tur and Tur, G., 2000, *Prosody-based automatic segmentation of speech into sentences and topics*, Speech Communication, vol 32 (1-2), pp 127-154.

Abdulla, W.H., 2002, *HMM-based techniques for speech segments extraction*, in the Journal of Scientific Programming, vol 10(3), pp. 221-239.

Delacourt, P., and Wellekens, C. J., 2000, *DISTBIC: A speaker based segmentation for audio data indexing*, in the Proceedings of Speech Communication, vol. 32, pp. 111-126.

Shafran I., and Rose, R., 2003, *Robust speech detection and segmentation for real-time ASR applications*, in Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP'03), vol. 1, pp. 432-435, Hong Kong, April.

Hue, C., Cadre, J., and Pervez, P., 2001, *A particle filter to track multiple objects*, IEEE Workshop on Multi Object Tracking.

Gustafson, F., 2002, *Particle filters for positioning, navigation and tracking*, IEEE Transaction on Signal Processing, pp. 425-437.

Arularnpalam, M., 2002, *A tutorial on particle filters for online nonlinear / non-Gaussian Bayesian tracking*, IEEE Transaction on Signal Processing, pp. 174-188.

Vermaak, J., 2002, *Particle methods for Bayesian modelling and enhancement of speech signals*, IEEE Transaction on Speech and Audio Processing, pp. 173-185.

Welling, L., and Ney, H., 1998, *Formant estimation for speech recognition*, IEEE Transaction Speech and Audio Processing, pp. 36-48.

Acero, A., 1999, *Formant analysis and synthesis using HMMs*, in the Proceedings of Eurospeech.

Watanabe, A., 2001, *Formant estimation method using inverse-filter control*, IEEE Transaction on Speech and Audio Processing, pp. 317-326.

Wang, C., and Principe, J. C., 1999, *Training Neural Networks with additive noise in the desired Signal,* IEEE Transaction on Neural Network., vol. 10, no. 6, pp. 1511 – 1517.

Cybenko, G., 1989, *Approximation by Superposition of a Sigmoid function,* Mathematics of Controls, Signals and Systems, vol. 2, pp. 303 – 314.

Hornik, K., Stinchcombe, M., and White, H., 1989, *Multilayer feed forward networks are universal approximators,* Neural Networks, vol. 2, pp. 359 – 366.

Ooyen, V., and Nienhuis, 1992, *Improving the convergence of the Backpropogation algorithm*, Neural Networks, vol. 5, pp. 465 – 471,.

Ismail, S., Abdul Manan Bin Ahmad, *2004, Recurrent Neural Network with Backpropogation through time algorithm for Arabic Recognition,* in the Proceedings of 18th European Simulation Multiconference, Europe.

Tolvi, J., 2004, *Genetic algorithms for outlier detection and variable selection in linear regression models*, Soft Computing, 8: 527-533, Springer-Verlag, London

Hongwei Sun, Kwok-Yan Lam, Siu-Leung Chung, Weiming Dong, Ming Gu, Jiaguang Sun, 2005, *Efficient vector quantization using genetic algorithm*, *Neural – Computing and applications*, 14 : 203 – 211, Springer – Verlag London

Fernando Bacao, Victor Lobo, Marco Painbo, 2005, *Applying genetic algorithms to zone design*, *Soft Computing*, 9 :341 – 348, Springer – Verlag London

Ling, S. H., and Leung, F.H.F., 2007, *An improved genetic algorithm with average-bound crossover and wavelet mutation operations*, Soft computing, 11 : 7-31, *Springer – Verlag London*

Sakoe, H., and Chiba, S., 1997, *Dynamic programming optimization for spoken word recognition*, in the proceedings of ICASSP, vol. 26, no. 1, pp. 43-49.

Chang, R. S and Eisentein, B. A., 1978, *Feature selection via dynamic programming for text-independent speaker identification*, IEEE Transaction Acoustic, Speech Signal Processing, vol. 26, no. 397

Chang, C. Y., 1973, *Dynamic programming as applied to feature subset selection in pattern recognition system*, IEEE transaction system man cybernetics, vol. 3, no. 166.

Sinha, Tilendra Shishir, Sanyal, Gautam and Mukherji, Abhijit, 2006, *Some aspects of modelling and simulation for the recognition of speech and speaker of Bengali language using proposed GSAMTSS Algorithm*, International Journal of Systemics, Cybernetics and Informatics, pp. no. 69 – 75.

Sinha, Tilendra Shishir and Sanyal, Gautam, 2006, *Neuro – Genetic based speech processing for promoting global cyber security using steganography technique*, *Proceedings of IEEE INDICON, September 15-17, 2006 Conference on Emerging Trends in ICT*, New Delhi, India.

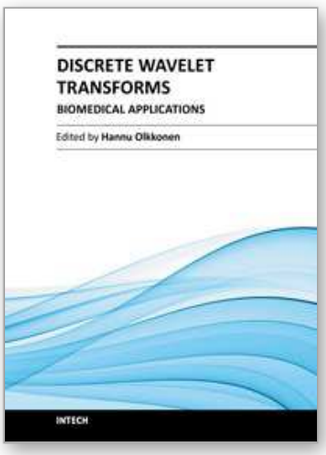
Yuexi Ren, Mark Hasegawa Johnson, Stephen E. Levinson, 2004, *Semantic analysis for a speech user interface in an intelligent tutoring system*, ACM, *pp. 13 – 16*.

Doddington, G., 2001, *Speaker recognition based on idiolectal differences between speakers*, In the proceedings of EUROSPEECH, Aalborg, Denmark, pp. no. 2521-2524.

Sinha, Tilendra Shishir., and Sanyal, Gautam, 2008, *Modelling and Simulation of Speech and Speaker Recognition of a Language,* International Journal of Tomography and Statistics (IJTS), Fall 2009, Vol. 12, No. F09, pp. 19-38.

Sinha, Tilendra Shishir., and Sanyal, Gautam, 2008, *Understanding of speech and speaker model for recognition of a language,* International Journal of Artificial Intelligence (IJAI), Autum 2009, Vol. 2, No. A09, pp. 107-125.

IntechOpen



Discrete Wavelet Transforms - Biomedical Applications

Edited by Prof. Hannu Olkkonen

ISBN 978-953-307-654-6

Hard cover, 366 pages

Publisher InTech

Published online 12, September, 2011

Published in print edition September, 2011

The discrete wavelet transform (DWT) algorithms have a firm position in processing of signals in several areas of research and industry. As DWT provides both octave-scale frequency and spatial timing of the analyzed signal, it is constantly used to solve and treat more and more advanced problems. The present book: Discrete Wavelet Transforms - Biomedical Applications reviews the recent progress in discrete wavelet transform algorithms and applications. The book reviews the recent progress in DWT algorithms for biomedical applications. The book covers a wide range of architectures (e.g. lifting, shift invariance, multi-scale analysis) for constructing DWTs. The book chapters are organized into four major parts. Part I describes the progress in implementations of the DWT algorithms in biomedical signal analysis. Applications include compression and filtering of biomedical signals, DWT based selection of salient EEG frequency band, shift invariant DWTs for multiscale analysis and DWT assisted heart sound analysis. Part II addresses speech analysis, modeling and understanding of speech and speaker recognition. Part III focuses biosensor applications such as calibration of enzymatic sensors, multiscale analysis of wireless capsule endoscopy recordings, DWT assisted electronic nose analysis and optical fibre sensor analyses. Finally, Part IV describes DWT algorithms for tools in identification and diagnostics: identification based on hand geometry, identification of species groupings, object detection and tracking, DWT signatures and diagnostics for assessment of ICU agitation-sedation controllers and DWT based diagnostics of power transformers.The chapters of the present book consist of both tutorial and highly advanced material. Therefore, the book is intended to be a reference text for graduate students and researchers to obtain state-of-the-art knowledge on specific applications.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Tilendra Shishir Sinha and Gautam Sanyal (2011). Modelling and Understanding of Speech and Speaker Recognition, Discrete Wavelet Transforms - Biomedical Applications, Prof. Hannu Olkkonen (Ed.), ISBN: 978-953-307-654-6, InTech, Available from: http://www.intechopen.com/books/discrete-wavelet-transforms-biomedical-applications/modelling-and-understanding-of-speech-and-speaker-recognition

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China

www.intechopen.com

51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen