International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:5, No:9, 2011

Assamese Numeral Speech Recognition using Multiple Features and Cooperative LVQ - Architectures

Manash Pratim Sarma and Kandarpa Kumar Sarma, *Member, IEEE*

Abstract—A set of Artificial Neural Network (ANN) based methods for the design of an effective system of speech recognition of numerals of Assamese language captured under varied recording conditions and moods is presented here. The work is related to the formulation of several ANN models configured to use Linear Predictive Code (LPC), Principal Component Analysis (PCA) and other features to tackle mood and gender variations uttering numbers as part of an Automatic Speech Recognition (ASR) system in Assamese. The ANN models are designed using a combination of Self Organizing Map (SOM) and Multi Layer Perceptron (MLP) constituting a Learning Vector Quantization (LVQ) block trained in a cooperative environment to handle male and female speech samples of numerals of Assamese- a language spoken by a sizable population in the North-Eastern part of India. The work provides a comparative evaluation of several such combinations while subjected to handle speech samples with gender based differences captured by a microphone in four different conditions viz. noiseless, noise mixed, stressed and stress-free.

Keywords—Assamese, Recognition, LPC, Spectral, ANN.

I. Introduction

Speech recognition is a method that uses an audio input for data entry to a computer or a digital system allowing it to derive some meaning out of it. Speech contains emotions and feelings and is generated by precisely coordinated muscle actions in the head, neck, chest, and abdomen. Speech results after a gradual process involving years of learning and practice [1]. Speech analysis and synthesis is one of the most thrust areas of research with applications to communication systems, process control, automation etc. Various related applications are also possible: speech enhancement, speech synthesis, speech coding, storage, retrieval etc. Speech corpus can be generated by extracting carefully chosen features from the speech signal [1]. Feature extraction involves transforming the input data into the set of values that best describes the input under consideration [2]. This work focuses on the design of a Speech Recognition System to handle numerals of Assamese language. The work considers the the extraction of Linear Predictive Code (LPC) and Principal Component Analysis (PCA) features of the captured speech samples recorded under varied conditions with gender and mood variations. A hybrid feature set is formed using LPC and PCA features. Further, the spectrum of the captured speeches are also considered as features. These extracted feature types are applied to classifiers formed using Learning Vector Quantization (LVQ) blocks. The LVQ - block is formed by using a combination of Self Organizing Map (SOM) and Multi Layer Perceptron (MLP). LVQ blocks are further arranged in a cooperative architecture to minimize the classification error and maximize the prediction rate. The system also been tested for speaker - dependant and speaker - independent cases. The success-rates vary but the cooperative architectures provide better results at the cost of increased times required for training.

Manash Pratim Sarma and Kandarpa Kumar Sarma are with the Department of Electronics and Communication Technology, Gauhati University, Guwahati - 781014, Assam, India. e-mail: (manashpelsc@gmail.com and kandarpaks@gmail.com)

Several work have been reported with regards to LVQ based speech recognition. Yet research has continued in area and more and more works are being reported which have explored innovative means of tackling speech recognition and related aspects. Some of the relevant works can be enumerated as below:

1) An extension of a self-organizing map called self-organizing multilayer perceptron (SOMLP) whose purpose is to achieve quantization of spaces of functions has been presented in *B. Gas* [3]. Possible use of the commonly used vector quantization algorithms (LVQ algorithms) to build new algorithms called functional quantization algorithms (LFQ algorithms) has also been demonstrated. The SOMLP algorithm allows quantization of function with high dimensional input space and as a consequence, classical FDA methods can be outperformed by increasing the dimensionality of the input space of the functions under analysis.
2) A novel Learning Vector Quantization (LVQ) based speech recognition method with the use of MFCC (Mel Frequency Cepstral Coefficient) and Differential MFCC has been presented in [4]. The work used a normal Kohonen LVQ network and then an improved LVQ scheme and simulation result has been reported. The improved LVQ network is based on two nearest neurons (winning and next winning) by virtue of which it is capable of classifying two nearest input classifier vectors.
3) A Hidden Markov Model (HMM) and LVQ based recognition scheme has been proposed in [5]. The work introduced MFCC, ∇ MFCC and $\nabla\nabla$ MFCC extraction algorithms are introduced, then these coefficients are normalized by HMM-based Viterbi method and then the resulted feature set is used to make learn coarsely by the first LVQ network and then finely by the improved LVQ network.
4) An interactive and incremental learning algorithm based

on entropy guided LVQ method for robot speech learning has been reported in [6].

5) A discriminative training procedure based on Learning Vector Quantization (LVQ) where the codebook is expressed in terms of probabilistic models has been proposed in [7].
6) An development and analysis of a learning vector quantization (LVQ) based algorithm for combined compression and classification and the testing of the convergence using the ODE method from stochastic approximation has been reported by *Baras, and Dey* [8].
7) An axiomatic approach to soft learning vector quantization (LVQ) and clustering based on reformulation has been presented by *Karayiannis* [9]. The reformulation of the fuzzy c-means (FCM) algorithm provides the basis for reformulating entropy-constrained fuzzy clustering (ECFC) algorithms which indicates minimization of admissible reformulation functions using gradient descent leads to a broad variety of soft learning vector quantization and clustering algorithms.
8) A novel speech recognition system based on the use of the Fuzzy Neural Network for 2-D phoneme sequence pattern recognition has been presented by *Kwan*[10]. The Self- Organizing Map (SOM) and then the Learning Vector Quantization (LVQ) are used to organize the phoneme feature vectors of short and long phonemes segmented from speech samples to obtain their phoneme maps. The 2-D phoneme response sequences of the speech samples are formed optimally on the phoneme maps by the Viterbi search algorithm. These 2-D phoneme response sequence curves are used as inputs to the Fuzzy Neural Network for training and recognition of speech utterances.
9) The automatic assessment of voice quality is addressed by means of short-term Mel Cepstral parameters (MFCC), and Learning Vector Quantization (LVQ) in a pattern recognition stage has been presented by *Lechn, Godino-Llorente, Osma-Ruiz, Blanco-Velasco and Cruz-Roldn* [11].
10) Two neural-network based classification approaches applied to the automatic detection of voice disorders has been studied by *Godino-Llorente and Gmez-Vilda* [12]. Structures studied Multilayer perceptron and learning vector quantization fed using short-term vectors calculated accordingly to the well-known Mel Frequency Coefficient cepstral parameterization.

The use of speech as an input to a computer or digital system is important because it provides a direct interaction between man and machine which always has been a challenge to designers engaged in the development of knowledge - based systems. A system with speech mode input is useful for a country like India and a state like Assam where a vast population is semi-literate. It will contribute towards automation, increase in reach of technology for a large section of the population and accelerated decision making. A speech oriented input system will be more robust and is likely to be cheap as it shall require only a microphone as an input device instead of a keyboard. Indian languages contain huge diversity of phonetic content in the languages spoken. It is more so for a language like Assamese which has evolved over the years from its original Indo-European roots with modification, alterations and strong influences of the local dialects and Indo-Chinese culture abundant in the Brahmaputra valley in the state of Assam in the north eastern part of India. This group of the original Indo-European family is classified as the easternmost member of this New Indo-Aryan (NIA) subfamily spoken in the Brahmaputra Valley of Assam by a population of over twenty million and acts as a link-language to people from neighbouring states. A unique feature of the Assamese language is a total absence of any retroflex sounds. Instead the language has a whole series of alveolar sounds, which include oral and nasal stops, fricatives, laterals, approximants, flaps and trills, unlike other Indo-Aryan and Dravidian languages [13]. Therefore, there exists a requirement to carry out extensive studies for design of speech processing and recognition systems exclusively in Assamese.

Work on speech technology in Indian languages have been going on over the last few decades. Several work are reported in languages like Hindi, Marathi, Gurmukhi, Tamil, Telegu, Bengali etc.

A work in Hindi conducts phoneme categorization experiments for Indian languages. In this direction a major effort was made to categorize Hindi phonemes using a time delay neural network (TDNN), and compare the recognition scores with other languages [14]. Another work in Hindi is related to the application of hybrid features using linear prediction in combination with multi-resolution capabilities of wavelet transform. Wavelet-Based Linear Prediction Coefficients (WBLPC) are obtained by applying 3 and 4-level wavelet decomposition and then having linear prediction of each sub-bands to get total 13 features. These features have been tested using a linear discriminant function and Hidden Markov Model (HMM) based classifier for speaker dependent and independent isolated Hindi digits recognition [15]. Another work is related to Hindi Paired Word Recognition (HPWR). It has been examined with the help of intelligent hybrid computing scheme based on wavelet transform and Probabilistic Neural Network (PNN) [16].

A few works have also been reported in Assamese. Some recent works are as [17] to [20].

II. Raw speech signals

As mentioned earlier, since the work is a part of Assamese Speech Recognition System, speech signals with gender and mood variation, uttering Assamese numerals from zero to nine,are captured. This results in a total of following broad sets of speech signals:

1) *Girl mood1*
2) *Girl mood2*
3) *Boy mood1*
4) *Boy mood2*
5) *Girl Reference*
6) *Boy Reference*

The waveforms of speech signal uttering *Xunya* (Zero in Assamese)in various mood and gender is shown in Figure 1

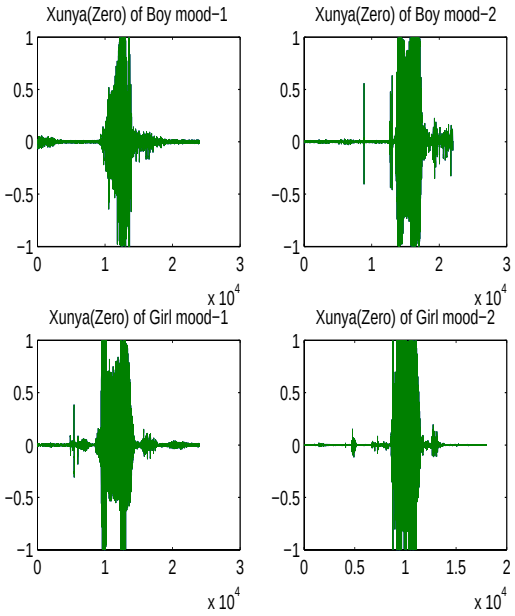

Fig. 1. Waveform of *xunya*(Zero in Assamese)with gender and mood variations

The gender distinction is brought into the samples to ascertain the ability of the proposed model to deal with such variations. In reality male and female voice and speech samples are different. Male vocal folds tend to be longer and thicker than female vocal folds causing them to vibrate more slowly. Male speakers have an average F_0 upto 200 Hertz (for speakers of German and English language it is about 100120 Hz while for females this values is twice. Female vocal folds are shorter and lighter and vibrate at approximately twice the male frequency [21].

The spectrum of a speech signal uttering *Xunya* (Zero in Assamese) is shown in Figure 2 . For recording the speech signal, a PC headset and a sound recording software, Gold Wave, is used. GoldWave's Monitor recording option helps to adjust the volume level before recording. While recording, the sampling rate taken is 8000 Hz in mono channel mode.

III. Feature Extraction of the Speech Signals

Selecting appropriate features from a speech signal is an important issue to achieve high accuracy in speech recognition. Feature extraction involves analysis of speech signal. Broadly the feature extraction methods are classified as temporal analysis and spectral analysis techniques. In temporal analysis the speech waveform itself is used for analysis. In spectral analysis spectral representation of speech signal is used for analysis. The extracted feature vector must contain information that is useful to identify and differentiate speech sounds insensitive external noise and other irrelevant factors [22]. The feature types considered are Linear Predictive Code (LPC) and Principal Component Analysis (PCA) features of the captured speech samples. A hybrid feature set is formed using LPC and PCA features. Further, the spectrum of the captured speeches are also considered as features which are derived taking Discrete Fourier Transform (DFT). A brief description of each of the feature extraction process is provided below. Before feature extraction, a series of operations are carried out which constitutes pre-processing of the speech samples.

1) *Pre-emphasis:* The speech signal (here, also refereed as *word*), s(n), is filtered with a first-order FIR filter to spectrally flatten the signal. We used one of the most widely used pre-emphasis filters of the form

$$H(Z) = 1 - az^{-1} \tag{1}$$

where a=15/16. For pre-emphasizing one mask pattern is used ie [1-0.97] which signifies a Z-transform at 97/z.

2) *Normalization:* After pre-emphasis, each word has its energy normalized. Based on the energy distribution along the temporal axis, it is computed the center of gravity, and this information is used as reference for temporal alignment of the words. The energy of each word was computed using 60 non overlapping windows.

3) *Frame Blocking:* The pre-emphasized speech signal, $s[n]$, is blocked into frames of N samples, with adjacent frames being separated by M samples. If we denote the l^{st} frame of speech by $xl[n]$, and there are L frames, then

$$x_l[n] = s[M_l + n] \tag{2}$$

where n=0, 1,. N-1, and l=0,1,...,L-1.

4) *Windowing:* Each individual frame is windowed to minimize the signal discontinuities at the borders of each frame. If the window is defined as w[n], $0 < n < N-1$, then the windowed signal is

$$x_w[n] = x_l[n]w[n] \tag{3}$$

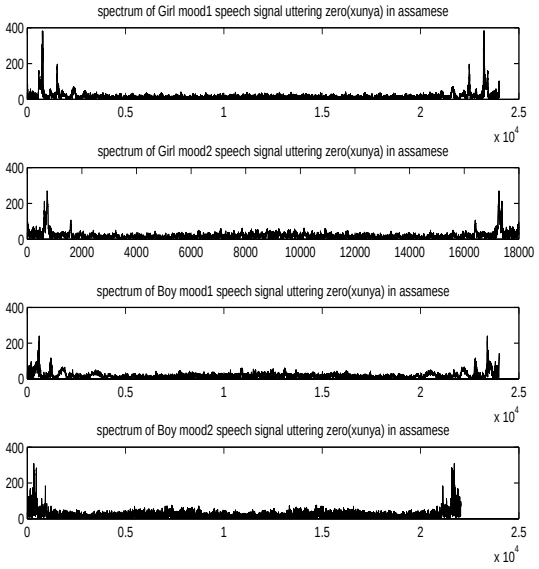

Fig. 2. Spectral representation of *xunya*(Zero in Assamese)with gender and mood variations

where $0 < n < N - 1$. We used a Hamming window, a typical window used for the autocorrelation method of LPC.

A. LPC Feature Extraction:

Linear predictive analysis of speech has become the predominant technique for estimating the basic parameters of speech. Linear predictive analysis provides an accurate estimate of the speech parameters and also an efficient computational model of speech. The basic idea behind linear predictive analysis is that a specific speech sample at the current time can be approximated as a linear combination of past speech samples. Through minimizing the sum of squared differences, over a finite interval between the actual speech samples and linear predicted values a unique set of parameters or predictor coefficients can be determined [23].

1) *LPC Analysis using the Autocorrelation Method:* The processing steps involved in the LPC analysis using the autocorrelation method of order p is described below:
In matrix form

$$\mathbf{Ra} = \mathbf{r} \tag{4}$$

where $\mathbf{r} = [r(1)r(2)...r(3)]^T$ is the autocorrelation vector, $\mathbf{a} = [r_1 r_2 ... r_p]^T$ is the filter co-efficient vector, and

$$R = \begin{pmatrix} r(0) & r(1) & ... & r(\text{p-1}) \\ r(1) & r(0) & ... & r(\text{p-2}) \\ r(2) & r(1) & ... & r(\text{p-3}) \\ . & . & . & . \\ r(\text{p-1}) & r(\text{p-2}) & ... & r(0) \end{pmatrix} \tag{5}$$

is the Toeplitz autocorrelation matrix. This matrix is nonsingular and gives the solution

$$a = R^{-1} r \tag{6}$$

These coefficients form the basis for linear predictive analysis of speech. The coefficients of a p^{th}-order linear predictor (FIR filter) that predicts the current value of the real-valued time series x is based on past samples as shown in eq. 7

$$\hat{x(n)} = -a(2)x(n-1) - a(3)x(n-2) - .. - a(p+1)x(n-p) \tag{7}$$

p is the order of the prediction filter polynomial, a = [1 a(2) ... a(p+1)].

2) *Cepstral co-efficient:* Almost all speech recognizers use cepstral analysis techniques because it operates in a domain in which the excitation function and vocal tract filter function are separable. This indicates that the characteristics of vocal tract and excitation are well represented separately in the cepstral coefficients. There are two types of cepstral approaches: FFT cepstrum and LPC cepstrum. In the FFT cepstral analysis, the real cepstrum $c(n)$ is defined as the inverse FFT transform of the logarithm of the speech magnitude spectrum [24] [25].
The actual predictor coefficients obtained as mentioned above can never be used in speech recognition, since they typically show high variance. Hence it is required to transform the predictor coefficient to a more robust set of parameters known as cepstral coefficients. They can be directly derived from the set of LPC co-efficients using the recursion
c_0 = r(0),

$$c_m = a_m + \sum_{k=1}^{M-1} \frac{k}{m} c_k a_{m-k}, \tag{8}$$

where $1 < m < p$, and

$$c_m = \sum_{k=1}^{M-1} \frac{k}{m} c_k a_{m-k}, \tag{9}$$

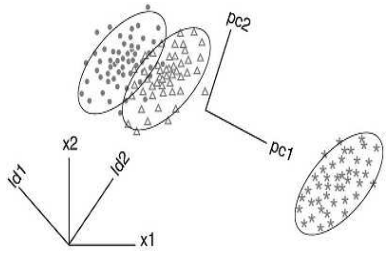

Fig. 3. Dimension reduction using PCA

where $m > p$. The cepstral co-efficient are the coefficients of the Fourier transform representation of the log magnitude of the spectrum [26]. The size of the LPC feature vector taken is 20 as per the requirements described in [17].

B. PCA Feature Extraction:

The feature set next is completely changed and a Principal Component Analysis (PCA) set extracted for all the samples. PCA involves a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible [22]. PCA is a common technique for finding patterns and size reduction in data of high dimension (Figure 3). The LPC and PCA features are mixed to form the third set which is called the hybrid feature set. The hybrid feature set, formed with inputs from two different approaches, contains dissimilar sample components which helps the ANN blocks to carry on the learning process and generate lower convergence rates.

C. Spectral Feature Extraction:

An important aspect of speech recognition is the ability of the system to track the variation in the input speech sample. Speech is dynamic in nature with multiple sources of acoustic interference. A system with the ability to tackle the temporal variations of speech is better equipped to provide higher success rates of a recognition [27].
Any DFT coefficient of a speech frame, considered as a function of frame index with the discrete frequency fixed, can be interpreted as the output of a linear time-invariant filter with a narrow-bandpass frequency response. Therefore, taking a second DFT of a given spectral band, across frame index, with discrete frequency fixed, will capture the spectral changes in that band with different rates. This effectively extracts the modulation frequency response of the spectral band [28].
Let X[n, k] be the DFT of a speech signal x[m], windowed by a sequence w[m]. Rearranging, the DFT operation maybe expressed as,

$$X[n,k] = x[n] * h_k[n] \tag{10}$$

where $*$ denotes convolution and,

$$h_k[n] = w[-n]e^{\frac{j2\pi kn}{M}} \tag{11}$$

From eq.s 10 and 11, it is seen that the k^{th} DFT coefficient X[n, k], as a function of frame index n, and with discrete frequency k fixed, is equivalent to the output of a linear time invariant filter with impulse response $h_k[n]$.
Let a sequence be taken as $y_k[n] = X[n,k]$. Then taking a second DFT of this sequence over P points, gives

$$Y_k[q] = \sum_{p=0}^{P-1} y_k[n+p]e^{\frac{j2\pi qp}{P}} \tag{12}$$

$$Y_k[q] = \sum_{p=0}^{P-1} X[n+p,k]e^{\frac{j2\pi qp}{P}} \tag{13}$$

where $Y_k(q)$ is the q^{th} modulation frequency coefficient of k^{th} primary DFT coefficient. Lower values of q's tackle slower spectral changes and higher q's deal with faster spectral changes of the speech signal.
The features thus extracted are applied to the system formed by a LVQ- and a Cooperative LVQ-Architecture classifier (Figures 4 and 5).

IV. LVQ- and a Cooperative LVQ-Architecture using SOM - MLP combination

Learning Vector Quantisation (LVQ) is a supervised version of vector quantisation, similar to Self Organising Maps (SOM). It can be applied to multi-class pattern classification in speech recognition. As in supervised method, LVQ uses known target output classifications for each input pattern of the form [29]. The LVQ training algorithm is faster than backpropagation due to the fact that only the weights leading to the winning neuron are updated and the network does not attempt to minimize some target value constraints for neurons which are not interesting for the current training pattern. The consideration of the winning neuron is also the reason for the efficiency of the training algorithm: It does not try to change some weights which do not contribute to the classification process and therefore, the complete emphasis is on the minimization of the classification error of the training data [30]. The entire system has two distinct parts for dealing with two classes of input classified into male and female clusters. The first block is formed by a Generalized Feed Forward Artificial Neural Network (GFFANN) which acts like a class mapper network. It categories the inputs into two gender based clusters. It contains one hidden layer with tan-sigmoid activation function and one input and the other the output layer each fitted with log-sigmoid activation functions. The classification is just in terms of two classes male and female. The decision of this network is placed as a class code into the input sample vector and then passed on to two LVQ - blocks formed by SOM - MLP ANNs as shown in Figure 5. The first LVQ block tackles all female speech inputs while the second LVQ block handles the male sample.

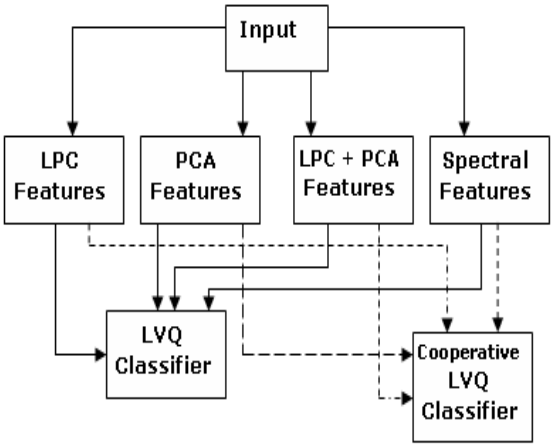

Fig. 4. Block diagram of the complete work

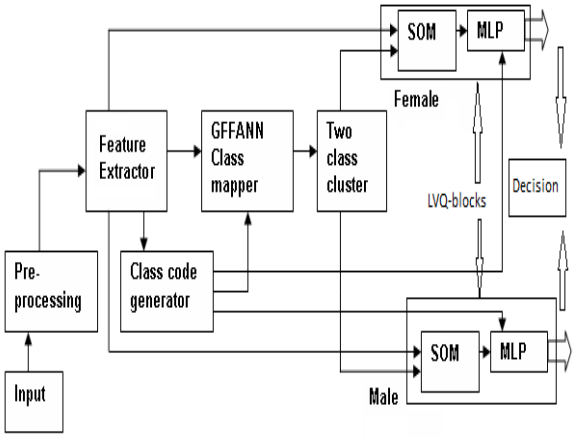

Fig. 5. LVQ- and a Cooperative LVQ-Architecture classifier

A. Training the SOM:

The SOM uses an unsupervised learning method to map high dimensional data into a 1-D, 2-D, or 3-D data space, subject to a topological ordering constraint. A major advantage is that the clustering produced by the SOM retains the underlying structure of the input space, while the dimensionality of the space is is reduced. As a result, a neuron map is obtained with weights encoding the stationary probability density function p(X) of the input pattern vectors [29].

- **Network Topology:**The SOM consists of two main parts, the input layer and the output map. There is no hidden layer. The dimensionality of the input layer is not restricted, while the output map has dimensionality 1-D, 2-D, or 3-D. An example of a SOM is shown in Figure-6 and a planar array of neurons with the neighborhoods is shown in Figure-7

The winning output node is determined by a similarity measure. It can either be the Euclidean distance or the dot product between the two vectors. Here Euclidean distance is

considered. The norm adopted is

$$\|x - w_m\| = min_i \|x - w_i\| \tag{14}$$

where w_i is the winning neuron. For the weight vector of unit ***i*** the SOM weight update rule is given as

$$w_i(t+1) = w_i(t) + N_{mi}(t)[x(t) - w_i(t)] \tag{15}$$

where t is the time, x(t) is an input vector, and $N_{mi}(t)$ is the neighborhood kernal around the winner unit m. The neighborhood function in this case is a Gaussian given by

$$N_{mi}(t) = \alpha(t) exp[\frac{-\|r_i - r_m\|}{\sigma^2(t)}] \tag{16}$$

where r_m and r_i are the position vectors of the winning node and of the winning neighborhood nodes, respectively. The learning rate factor $0 < \alpha(t) < 1$, decreases monotonically with time, and $\frac{3}{4}(t)$ corresponds to the width of the neighborhood function, also decreasing monotonically with time. Thus, the winning node undergoes the most change, while the neighborhood nodes furthest away from the winner undergo the least change.

- **Batch training:** The incremental process defined above can be replaced by a batch computation version which is significantly faster[5]. The batch training algorithm is also iterative, but instead of presenting a single data vector to the map at a time, the whole data set is given to the map before any adjustments are made. In each training step, the data set is partitioned such that each data vector belongs to the neighborhood set of the map unit to which it is closest, the Voronoi set [31]. The sum of the vectors in each Voronoi set are calculated as:

$$S_i(t) = \sum_{j=1}^{{}^nV_i} X_j \tag{17}$$

  where ${}^nV_i$ is the number of samples in the Voronoi set unit i. The new values of the weight vectora are then calculated as:

$$w_i(t+1) = \frac{\sum_{j=1}^{m} N_{ij}(t) S_j(t)}{\sum_{j=1}^{m} ({}^nV_i) h_{ij}(t)} \tag{18}$$

  where m is the number of map units an ${}^nV_j$ is the number of sample falling into Voronoi set V_i.
- **Quality Measures:** Although the issue of SOM quality is not a simple one, two evaluation criteria, resolution and topology preservation, are commonly used [32]. Other methods are available in [32] and [33].

The following are among some of methods used for training and weight updating of SOMs as part of competitive learning [30]:

1) **Inner Product** and
2) **Euclidean Distance Based Competition**.

A detailed account of the two methods maybe given as below [29] [30]:

1) **Inner Product and Euclidean Distance Based Competition:** The training process of the SOM is linked with the creation of a code-book which provides the account of finding the best match among the set of input patterns given to it. There are two ways in which the best matching codebook vector can be found. The first way is to employ an inner product criterion–select the best reference vector by choosing the neuron in the competitive layer that receives the maximum activation. This means that for the current input vector X_k, we compute all neuron activations

$$y_j = X_k^T W_j \qquad j = 1, ..., m \tag{19}$$

and the winning neuron index, J, satisfies

$$y_J = max_j \{X_k^T W_j\} \tag{20}$$

Alternatively one might select the winner based on a Euclidean distance measure. Here the distance is measured between the present input X_k and the weight vectors W_j, and the winning neuron index J, satisfies

$$\|X_k - W_J\| = min_j \{\|X_k - W_j\|\} \tag{21}$$

It is misleading to think that these methods of selecting the winning neuron are entirely distinct[323]. To see this,

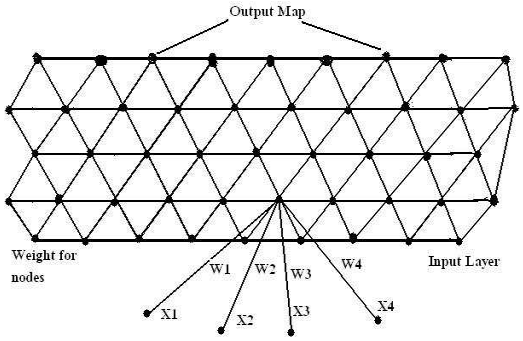

Fig. 6. Self Organizing Feature Map

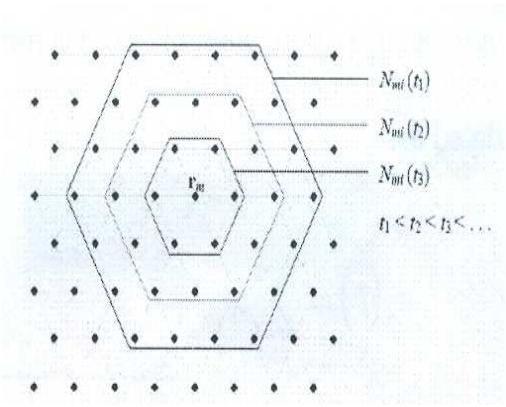

Fig. 7. Topological neighborhood of Self Organizing Feature Map

assume that the weight *equinorm* property holds for all weight vectors:

$$\|W_1\| = \|W_2\| = ... = \|W_m\| \quad (22)$$

We now rework the condition given in Eq. 21 as follows:

$$\|X_k - W_J\|^2 = min_j\{\|X_k - W_j\|^2\} \quad (23)$$

$$\Rightarrow (X_k - W_J)^T(X_k - W_J) = min_j\{(X_k - W_j)^T(X_k - W_j)\} \quad (24)$$

Let

$$X_{1k} = \{\|X_k\|^2 - 2X_k^T W_j + \|W_j\|^2\} \quad (25)$$

$$\Rightarrow \|X_k\|^2 - 2X_k^T W_J + \|W_J\|^2 = min_j\|X_{1k}\| \quad (26)$$

$$\Rightarrow -2X_k^T W_J + \|W_J\|^2 = min_j\{(-2X_k^T W_j + \|W_j\|^2)\} \quad (27)$$

We now subtract $\|W_J\|^2$ from both side of Eq. 27 and invoke the weight equinorm assumption. This yields:

$$-2X_k^T W_J = min_j\{-2X_k^T W_j\} \quad (28)$$

or,

$$X_k^T W_J = max_j\{2X_k^T W_j\} \quad (29)$$

which is the identical to criterion given in Eq. 21. Neuron J wins if its weight vector correlates maximally with the impinging input.

In the case of **ART 1** F_2 neurons we have seen that a binary choice network requires neuronal excitatory self feed-back, lateral inhibition and a faster-than-linear signal function. In computer simulations, however, one may simply choose the neuron index with maximum activity or minimum distance.

2) **A Generalized Competitive Learning Law**: Competitive learning requires that the weight vector of the winning neuron be made to correlate more with the input vector. This is done by perturbation of only the winning weight vector $W_J = (w_{1J}, ..., w_{nJ})^T$ towards the input vector. the scalar implementation of this learning law in difference form is presented below:

$$w_{iJ}^{k+1} = w_{iJ}^k + \eta x_i^k \qquad i = 1, ..., n \quad (30)$$

However, as we have seen in such forms of learning, the weight grow without bound and some form of normalization is required. By normalization we mean that the sum of the instar weight should add up to unity: $\sum_{i=1}^n = 1$. This can be ensured by normalizing the inputs within the equation and incorporating an extra weight subtraction term:

$$w_{iJ}^{k+1} = w_{iJ}^k + \eta(\frac{x_i^k}{\sum_j x_j^k} - w_{iJ}^k) \qquad i = 1, ..., n \quad (31)$$

This equation leads to total weight normalization. To see this, we simply add all n equations (Eq. 31) to get the total weight update equation:

$$\tilde{W}_{J(k+1)} = \tilde{W}_{J(k)} + \eta(1 - \tilde{W}_{J(k)}) \quad (32)$$

where $\tilde{W}_{J(k)} = \sum_{i=1}^n w_{iJ}^k$. It is clear from Eq. 32, that the weight sum eventually approaches a point on the unit hyperspher, S^n . If the inputs are already normalized then normalization is not required within the law and weight update is direct:

$$w_{iJ}^{k+1} = w_{iJ}^k + \eta(x_i^k - w_{iJ}^k) \qquad i = 1, ..., n \quad (33)$$

which effectively moves the Jth instar W_J, in the direction of the input vector X_k.

We have written this equation for a winning neuron with index J. In general one may re-write this equation for all neurons in the field as :

$$w_{iJ}^{k+1} = \begin{cases} w_{ij}^k + \eta(x_i^k - w_{iJ}^k), & i = 1, ..., nj = J \\ w_{ij}^k & i = 1, ..., nj \neq J \end{cases} \quad (34)$$

where $J = argmax_j\{y_j^k\}$. Based on this, one can generalize further and write out the following standard competitive form in discrete time:

$$w_{ij}^{k+1} = w_{ij}^k + \eta s_j^k(x_i^k - w_{ij}^k) \qquad i = 1, ..., n \quad j = 1, ..., m \quad (35)$$

where $s_j^k = 1$ only for j=J and is zero otherwise, for a hard competitive field.

With these ideas in hand we now study three important classes of neural network models that have their foundation in competitive learning.

The MLPs, as parts of the LVQ blocks, are trained as per the considerations of back-propagation algorithm.

B. Cooperative LVQ Architecture:

The fundamental considerations governing the working and parameter selection of the cooperative ANNs or committee machines can be explained using the following analysis [34] [29]:

Let a training set of m input - output pairs be $(x^1, t_1), (x^2, t_2), ...(x^m, t_m)$ be given and N networks are trained using this set of data. For simplicity, let for n-dimensional input there be a single output. Let for network functions f_i for a number of networks represented by indices $i = 1, 2, ...N$, the cooperative or committee network formed generates as output given as

$$f = \frac{1}{N}\sum_{i=1}^N f_i \quad (36)$$

The rationale behind the use of the averaging in the output of the cooperative or committee network as given by eq. 36 is the fact that if one of the constituent networks in the ensemble is biased to some part of the input samples, the ensemble average can scale down the prediction error considerably [34]. A quadratic error function can be computed from each of the error vectors e_i using the ensemble function f as

$$Q = \sum_{i=1}^m [t_i - \frac{1}{N}\sum_{i=1}^N f_i]^2 \quad (37)$$

Using matrix notation, the quadratic error can be expressed as

$$Q = |\frac{1}{N}(1,1,....)E|^2 = \frac{1}{N^2}(1,1,....1)EE^T(1,1,.....1)^T \quad (38)$$

EE^T is the correlation matrix representing the error residuals. If each function approximation produces uncorrelated error vectors, the matrix EE^T is diagonal and the i^{th} diagonal element Q_i is the sum of quadratic deviations for each functional approximation, i.e $Q_i = \|e^i\|^2$. Thus,

$$Q = \frac{1}{N}(\frac{1}{N}(Q_1 + Q_2 + ... + Q_N)) \quad (39)$$

It implies that the total quadratic error of the ensemble is less by a factor $\frac{1}{N}$ than the average of the quadratic errors of the total computed approximations. This holds only if N is not very large. If the quadratic errors are not uncorrelated, i.e if EE^T is not symmetric, a weighted combination of N functions f_i can be approximated as

$$f = \sum_{i=1}^{N} w_i f_i \quad (40)$$

The weights w_i must be computed in such a way as to minimize the expected quadratic deviation of the function f for the given training set. With the constraint with the constraint $w_1 + + w_N = 1$, eq. 38 transforms to

$$Q = \frac{1}{N^2}(w_1, w_2, ...w_N)EE^T(w_1, w_2, ...w_N)^T \quad (41)$$

Differentiating the above eq. 41 w. r. t $w_1, , w_N$, and using a Lagrangian multiplier λ for the constraint $w_1 + + w_N = 1$, the above functional modifies to

$$Q' = \frac{1}{N^2}wEE^T + \lambda(1,1,.....1)w^T \quad (42)$$

$$= \frac{1}{N^2}wEE^T + \lambda 1 w^T \quad (43)$$

where 1 is a row vector with all its N components equal to 1. Setting the partial derivative of Q' with respect to w to zero this leads to

$$\frac{1}{N^2}wEE^T + \lambda 1 = 0 \quad (44)$$

With simplification,

$$\lambda = \frac{1}{N^2 1(EE^T)^{-1}1^T} \quad (45)$$

The optimal weight set can be calculated as

$$w = \frac{1(EE^T)^{-1}}{1(EE^T)^{-1}1^T} \quad (46)$$

assuming that the denominator does not vanish. This method, however, is dependent on the constraint that EE^T is not ill-conditioned.

V. Experimental Results and Discussion

The SOM and the MLPs take different times to train. Some of the average results for 50 samples of each of the 10 sets of noise-free, noise mixed, stress free and stressed samples applied to SOM and MLP blocks are shown in Tables I and II. A similar set of results are shown by Table III.

TABLE I
Variation of the average training time of a SOM

Sl Num	Epoch	Time is sec.s
1	250	47.3
2	500	78.2
3	750	108.3
4	1000	133.1

TABLE II
Variation of the average training time of a MLP

Sl Num	Epoch	Time is sec.s
1	250	51.1
2	500	79.8
3	750	114.2
4	1000	144.3

The LPC features are extracted from the pre-processed samples. Next, the LPC feature vector is applied to the SOM which produces the code analogous to a vector quantization (VQ) code form. At the end of 1000 training epochs the code generated by the SOM is applied to the MLP which compares it with the desired output. The output is a class-code which is known to the trained MLP. The MLP is trained to recognize the class codes. The MLP block is trained with error back propagation with Levenberg-Marquardt optimization. At the end of 2000 epochs, on an average for 50 samples of each of the 10 sets of noise-free, noise mixed, stress free and stressed samples, the MSE reaches the 10^{-4} and recognition performance around 93% mark. Table IV shows the training time and success rate variation with respect to the frame size of the samples.

The 28-frame LPC vector size produces the best results though it takes the longest time to train. This combination is used to perform the speech recognition task.
Other features are also extracted as per the considerations given in Sections III-B to III-C. Two sets are created- one for training and the other for validation. The validation samples are used to perform the recognition part with the selected and trained architecture.

TABLE III
Variation of the average training time of a LVQ block

Sl Num	Epoch	Time is sec.s
1	250	81.1
2	500	104.2
3	750	124.1
4	1000	164.2

TABLE IV
Variation of the average training time with LPC-VQ length of the samples applied to the Composite LVQ block

Sl Num	Input Length	Time is sec.s	Success in %
1	7	78.9	90.3
2	14	132.3	91.8
3	21	203.1	93.2
4	28	254.2	94.8

TABLE V
AVERAGE RECOGNITION RATES IN % GENERATED BY DIFFERENT FEATURE TYPES WHEN APPLIED TO A LVQ BLOCK

Sl Num	Input Sample	LPC	PCA	Hybrid	Spectral
1	Male- noise-less	94.1	94.4	95.2	96.2
2	Male- noise-mixed	92.7	93.1	94.1	95.3
3	Male- stressed	92.6	92.8	93.5	95.1
4	Male- stress free	94.1	94.4	95.2	96.1
5	Female- noise less	92.3	92.6	93.5	96.1
6	Female- noise-mixed	92.4	92.8	93.5	95.6
7	Female- stressed	93.1	93.3	93.8	95.5
8	Female- stress free	94.5	95.1	95.7	96.5

TABLE VI
AVERAGE RECOGNITION RATES IN % GENERATED BY DIFFERENT FEATURE TYPES WHEN APPLIED TO THE COOPERATIVE LVQ ARCHITECTURE

Sl Num	Input Sample	LPC	PCA	Hybrid	Spectral
1	Male- noise-less	95.2	96.2	97.1	98.6
2	Male- noise-mixed	93.8	94.7	95.3	97.2
3	Male- stressed	93.5	94.8	95.4	97.2
4	Male- stress free	95.2	96.3	97.1	97.8
5	Female- noise less	94.1	96.1	96.8	97.2
6	Female- noise-mixed	93.6	95.1	95.5	96.5
7	Female- stressed	94.3	95.3	95.8	96.3
8	Female- stress free	95.6	96.2	97.1	98.7

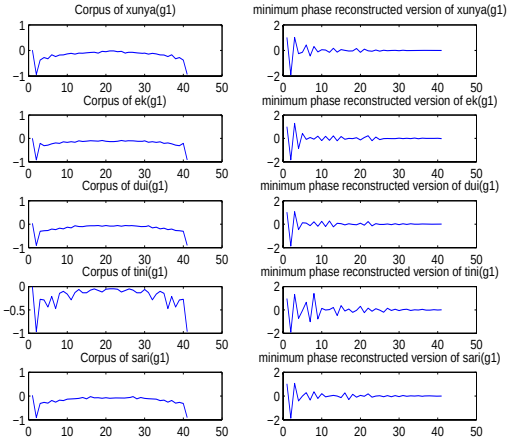

Fig. 8. Assamese speech corpus of a girl voice for zero to four

A. Performance of the LVQ System using Mulitple Features:

The results obtained, on an average, from 50 samples of each of the 10 sets of noise-free, noise mixed, stress free and stresses samples, are shown in Table V. The hybrid and spectral features provide success rates around the 95 - 96.5 % range. These features capute the details of the input samples better than the LPC and the PCA features.

B. Performance of the Cooperative LVQ System using Mulitple Features:

The results obtained, on an average, from 50 samples of each of the 10 sets of noise-free, noise mixed, stress free and stresses samples, show a marked improvement then the case when the features are applied separately to the Cooperative LVQ architecture. The results are shown in in Table VI. The advantage using the hybrid and spectral features can be clearly observed as the success rates shown by these features are around 95 - 98.7 %. Such success rates are also due to the Cooperative LVQ architecture. The feature and classifier combination thus developed shows the ability to tackle Assamese numeral speech inputs with mood, gender and recording condition variation.

VI. CONCLUSION

The LVQ system used for speech recognition of ten Assamese numerals recorded with gender and mood variation show a performance characteristics though quiet satisfactory yet has margins of errors which indicate that further improvement will be required.

Among the features considered, the LPC is the most popular feature set that can be readily used for a speech recognition system. The LPC predictor length should be properly selected to achieve higher success-rates. The PCA features capture the sample details in a manner whcih give better success rates then the PCA and also contribute towards dimensionality reduction. It helps to minimize computational complexity. The hybrid feature set formed by the LPC and PCA combination improves success rates further as they capture sample from two different considerations, hence contain more sample details. The spectral features contain slowly and fast varying signal components of the speech samples. Speech inputs with mood and gender variations have time varying properties. Therefore, the spectral features obtained by taking DFT twice provide superior content, hence have generated the highest success rates.

Classifier design is an important aspect. For a language like Assamese which reach phonetic variations, though LVQ is an acceptable classifier, a Composite LVQ- Architecture shows that by distributing the work load, success rates of recognition can be improved even further. However, training time is an issue which is nearly one and a half times more than the unitary LVQ-block. It can be tackled by use of superior hardware.

The system, though, remains speaker - dependant, can be converted to a speaker - independent form by the use of hybrid classifiers. It can also be extended to include all types of speech inputs with phoneme based recognition considerations.

REFERENCES

[1] "National Institute on Deafness and Other Communication Disorders", (www.nidcd.nih.gov/health/voice/whatisvsl.htm)
[2] A. Saxena and A. Singh: "A Microprocessor based Speech Recognizer for Isolated Hindi Digits", Department of Electrical Engineering, Indian Institute of Technology Kanpur, India (www.stanford.edu/ asaxena/research/speechrecognizer.shtml)
[3] B. Gas: "Self-Organizing MultiLayer Perceptron", *IEEE Transactions on Neural Networks*, Vol.: 1(99), pp: 1 - 1, 2010.
[4] L. Shuling, W. Chaoli, D. Jiaming: "Nonspecific Speech Recognition Method Based on Composite LVQ1 and LVQ2 Network", *IEEE Conference on Control and Decision Conference, 2009 (CCDC '09)*, pp: 2304 - 2308, 2009.
[5] L. Shuling, W. Chaoli, D. Jiaming: "Nonspecific Speech Recognition based on HMM / LVQ Hybrid Network", *Second IEEE International Conference on Intelligent Computation Technology and Automation, 2009 (ICICTA '09)*, Vol: 1, pp: 645 - 648, 2009.

[6] L. Qiong, L. Stephen, W. Ying and H. Thomas: "Robot Speech Learning via Entropy Guided LVQ and Memory Association", *Proceedings of IEEE International Joint Conference on Neural Networks, 2001 (IJCNN '01)*, Vol: 3, pp: 2176 - 2181, 2001.
[7] H. Jaakko, T. Volker and S. Olli: "A Learning Vector Quantization Algorithm For Probabilistic Models", *X European Signal Processing Conference (EUSIPCO 2000)*, Vol. II, pp: 721-724, 2000.
[8] J. S. Baras, and S. Dey: "Combined Compression and Classification with Learning Vector Quantization" *IEEE Transactions on Information Theory*, Vol: 45 (6), pp: 1911 - 1920, 1999.
[9] N. B. Karayiannis: "An Axiomatic Approach to Soft Learning Vector Quantization and Clustering", *IEEE Transactions on Neural Networks*, Vol: 10 (5), pp: 1153 - 1165, 1999.
[10] H. K. Kwan: "Fuzzy Neural Network For Phoneme Sequence Recognition", *IEEE International Symposium on Circuits and Systems (ISCAS 2002)*, Volume: 2, pp: II- 847 -850, 2002.
[11] N. S. Lechn, J. I. Godino-Llorente, V. Osma-Ruiz, M. Blanco-Velasco and F. Cruz-Roldn: "Automatic Assessment of Voice Quality According to the GRBAS Scale", *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, pp: 2478 - 2481, 2006.
[12] J. I. Godino-Llorente, P. Gomez-Vilda: "Automatic detection of voice impairments by means of short-term cepstral parameters and neural network based detectors", *IEEE Transactions on Biomedical Engineering*, Vol: 51 (2), pp: 380 - 384, 2004.
[13] tdil.mit.gov.in/assamesecodechartoct02.pdf (courtesy: Prof. Gautam Baruah, Dept. of CSE, IIT Guwahati, Guwahati, Assam, India.)
[14] A. Dev, S. S. Agrawal and D. R. Choudhury: "Categorization of Hindi phonemes by neural networks", *Spinger Journal of AI and Society*, vol. 17 (3-4), pp. 375-382, 2003.
[15] A. Sharma, M. C. Shrotriya, O. Farooq and Z. A. Abbasi: "Hybrid wavelet based LPC features for Hindi speech recognition", *International Journal of Information and Communication Technology*, vol. 1 (3-4), pp. 373-381 (9), 2009.
[16] D. K. Rajoriya, R. S. Anand and R. P. Maheshwari: "Hindi paired word recognition using probabilistic neural network", *International Journal of Computational Intelligence Studies (IJCISTUDIES)*, Vol. 1, No. 3, pp. 291 - 308, 2010.
[17] M. Sarma, K. Dutta and K. K. Sarma: "Assamese Numeral Corpus for Speech Recognition using Cooperative ANN Architecture", *International Journal of International Journal of Electrical and Electronics Engineering*, vol.3:8, pp. 458 - 468, 2009.
[18] M. Sarma, K. Dutta and K. K. Sarma: "Speech Corpus of Assamese Numerals Extracted using an Adaptive Pre-emphasis Filter for Speech Recognition", *Proceedings of IEEE International Conference on Computer and Communication Technology (ICCCT-2010), Allahabad, India*, 2010.
[19] M. Sarma, K. Dutta and K. K. Sarma: "LPC-Cepstrum Corpus of Assamese Numerals for Speech Recognition Using Recurrent Neural Network", *Proceedings of IEEE Communications Society Sponsored Conference International Conference on Advances in Communication, Network and Computing (CNC 2010), Calicut, India*, 2010.
[20] M. P. Sarma and K. K. Sarma: "Speech Recognition of Assamese Numerals using combinations of LPC - features and heterogenous ANNs", *Proceedings of International Conference on Advances in Information and Communication Technologies (ICT 2010), Kochi, Kerala, India*, 2010.
[21] A. P. Simpson, "Phonetic differences between male and female speech", Language and Linguistics Compass 3/2, pp.: 621 640, 2009.
[22] B. Yegnanarayana, *Artificial Neural Networks*, 1^{st} Ed., PHI, New Delhi, 2003.
[23] *Feature Extraction*, cslu.cse.ogi.edu /toolkit /old /old /version $2.0a$ /.../ $node5.html$.
[24] B. Atal, "Efficient coding of LPC parameters by temporal decomposition", *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '83)*, Vol. 8, pp: 81 - 84, 1983.
[25] K. Y. Lee, A. M. Kondoz, and B. G. Evans: "Speaker adaptive vector quantisation of LPC parameters of speech", *Electronics Letters*, Vol. 24 (22), pp: 1392 - 1393, 1988.
[26] M. P. Kesarkar, *Feature Extraction for Speech Recogntion*, M.Tech. Credit Seminar Report, Electronic Systems Group, EE. Dept, IIT Bombay, November, 2003.
[27] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, 1^{st} Ed., Prentice Hall, 1978.
[28] V. Tyagi, I. McCowan, H. Misra and H. Bourlard: " Mel-Cepstrum Modulation Spectrum (MCMS) Feature for Robust ASR", *Dalle Molle Institute for Perceptual Artificial Intelligence (IDIAP)*, P.O. Box 592, CH-1920, Martigny, Switzerland.
[29] S. Haykin, *Neural Networks A Comprehensive Foundation*, Pearson Education, 2^{nd} edition, 2003.
[30] S. Kumar, *Neural Networks A Classroom Approach*, Tata McGraw Hill, 8^{th} Reprint, 2009.
[31] E. Alhoneiemi, J. Hollmn, O. Simula and J. Vesanto: *"Process monitoring and modeling using the self-organizing map"*, Integrated Computer Aided Engineering, Vol. 6 (1), pp. 3-14, 1999.
[32] S. Kaski and K. Lagus: *"Comparing self-organizing maps"* , Proceeding of International Conference on Neural Networks, pp. 809-814, 1997.
[33] H. U. Bauer and K. Pawelzik: *"Quantifying the neighborhood preservation of self-organizing feature maps"*, IEEE Transactions on Neural Networks, Vol. 3, no. 4, pp. 570-579, 1992.
[34] R. Rojas, *Neural Networks-A Systematic Introduction*, Springer, Berlin, 1996.

Manash Pratim Sarma , completed MSc in Electronics from Gauhati University, Guwahati Assam, India in 2008. He also completed MTech from Tezpur University, Assam, India. Presently he is a research associate at Department of Electronics and Communication Technology, Gauhati University, Assam, India. His field of interest includes Speech Processing, Ultra-Wide Band Communication and ANN applications.

Kandarpa Kumar Sarma , presently with the Department of Electronics and Communication Technology, Gauhati University, Assam, India, completed MSc in Electronics from Gauhati University in 1997 and MTech in Digital Signal Processing from IIT Guwahati, Guwahati, India in 2005 where he further continued his research work. His areas of interest include Applications of ANNs and Neuro-Computing, Document Image Analysis, 3-G Mobile Communication and Smart Antenna.