A Novel Pattern Recognition Model for Real-Time Voice Data Input

Yogesh Kumar Sen
Dept. of Electronics and Telecommunication
National Institute of Technology
Raipur, India
kumarsen.yogesh@gmail.com

R. K. Chaurasiya
Dept. of Electronics and Telecommunication
National Institute of Technology
Raipur, India
rkchaurasiya@nitrr.ac.in

Dr. ShrishVerma
Dept. of Electronics and Telecommunication
National Institute of Technology
Raipur, India
shrishverma@nitrr.ac.in

Abstract— **The classical front end analysis in speech recognition is a spectral analysis which parameterizes the speech signal into feature vectors. This paper proposes a voice recognition model that is able to automatically classify and recognize a voice signal with background noise. The model uses the concept of spectrogram, pitch period, short time energy, zero crossing rate, mel frequency scale and cepestral coefficient in order to calculate feature vectors. The k-Nearest Neighbor (k-NN) classification is used for classification and recognition of real-time input signal. Analytical hierarchical process is used for deciding the weightage of different features.**

Index Terms — **k-nearest neighbor, AHP, pitch-period, pattern classification.**

I. Introduction

The voice recognition is the ability of a machine to recognize the spoken words and convert them to any desired form. In the current scenario when we are moving towards the automated world, the applications of real-time voice recognition are increasing day by day. The voice recognition system is a good choice to give a voice command for any device which requires user inputs to operate. Lifts, television, gaming-stations, smart-phones and medical instruments are the few of the many such examples.

The real time voice recognition system first requires some training and then is ready to recognize the real time voice data input. For a new incoming voice command, the system tries to match its features from the existing data set. The command is then classified into the 'best-matched' command from the existing data set. The technological advancement in the field of pattern recognition has made the voice recognition more reliable and user friendly.

In this paper we propose a real-time voice recognition system based on K-Nearest neighbor classifier. Short Time Energy (STE), Zero Crossing Rate (ZCR), Power Spectral Density (PSD) and Pitch Period (PP) are used as the features to be used by the classifier. Instead of giving flat or random weightages to the features, we have applied Analytical Hierarchical Process (AHP) to decide weightages of the four features.

II. Related Work

Different methods have been used in the field of voice recognition [1][2][3]. Common methods use one or two features from zero crossing rate, short time energy, pitch period, autocorrelation function and cepestral coefficient. P. Khunarsal [4] came up with a new idea of using PSD as a feature for voice signal. Using one or two such features does not represent the complete information of the data, and hence results in the poor accuracy of classification.

Usually the voiced/unvoiced analysis is performed in conjunction with pitch analysis. Rabiner et al [5] proposed a pitch independent voiced and unvoiced classification using short time energy, zero crossing rate and linear predictive coding coefficient analysis. The method is very sensitive to the chosen parameter values and requires an exhaustive training.

B. Jounghoon et al[3] proposed a voice recognition model using mel frequency scale and cepestral coefficients. The accuracy of this model depends on the degree of voice periodicity. But in reality, voice is not period as sudden change in our vocal cord articulation can produce non periodic voice signal.

Rabiner et al [6] proposed a highly efficient hidden Markov model for voice recognition, but in this model large number of training set is required to obtain average Statistical spectral characteristics.

We are using ZCR, STE, PSD and PP as our features for classification. Additional we are combining mel frequency scale, LPC and autocorrelation function in order to determining more accurate pitch period. In other words, we are using almost all important features related to voice signal, and hence the feature vector represents the data set more accurately.

III. Algorithm

The proposed algorithm takes a voice signal input and provides the output class. Figure 1 shows the flowchart of the algorithm. First we take a voice input signal, then the silence part of the signal is removed, i.e. we extract the real input.

The four important features viz. STE, ZCR, PSD and Pitch Period are extracted in the next step. A brief definition and mathematical formulation of these features are discussed below:

A. Short time energy (STE)

STE for each frame is the energy in a sound at a specific instance in time and it provides amplitude variation with frames. STE for the unvoiced part of the signal is very high

978-1-4799-4236-7/14/$31.00 ©2014 IEEE

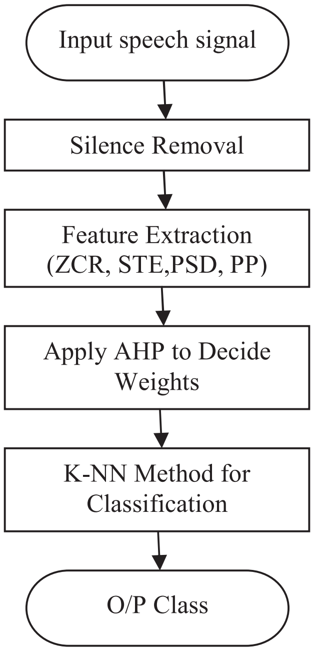
Input speech signal
Silence Removal
Feature Extraction (ZCR, STE,PSD, PP)
Apply AHP to Decide Weights
K-NN Method for Classification
O/P Class

Figure1:- Flow chart of our Algorithm

compared to the STE for voiced part. Using this fact, we can separate the voiced and the unvoiced part of signal.

We calculate STE E_n for each frame independently using Eq. 1.

$$E_n = \sum_{m=-\infty}^{\infty} (x[n]\, w[n-m])^2 \tag{1}$$

B. Zero Crossing Rate (ZCR)

ZCR for a frame is the measure of number of times the signal changes it's sign. Practically, it provides the frequency content of the signal which is useful for separation of voiced and unvoiced part of the signal. ZCR for high speech signal is unvoiced and for low speech signal is voiced. Mathematically ZCR $Z_n$is calculated as in Eq. 2.

$$Z_n = \sum_{m=-\infty}^{\infty} |sgn(x[m]) - sgn(x[m-1])| w[n-m] \tag{2}$$

Where

$$sgn(x[m]) = \begin{cases} 1, & x[m] \geq 0 \\ -1, & x[m] < 0 \end{cases}$$

Thus, Z_n provide the number of zero crossing frame by frame.

C. Power Spectral Density (PSD)

PSD represents the power distribution of signal at different frequency components. PSD can be calculated by taking the Fourier transform of the autocorrelation function of a signal. Autocorrelation function r (τ) for signal x[n] is given by Eq. 3.

$$\mathrm{r}(\tau) = \sum_{n=0}^{N-1-|i-j|} x[n] * x[n+|i-j|] = \mathrm{r}(|i-j|) \tag{3}$$

PSD from autocorrelation can be determined by Equation 4:

$$PSD = \sum_{\tau=-\infty}^{\infty} \mathrm{r}(\tau) e^{-j2\pi f\tau} \tag{4}$$

In order to get better performance, we use the log of PSD as a feature instead of PSD.

D. Pitch Period

Period for any signal can be defined the time required to complete one cycle. From human articulatory system we understand that voice generated due to the vibration in the vocal cord and these vibrations are constant short duration practically for 10 - 20 milliseconds. Thus, for speech signal the pitch period can be though as the period of vocal cord vibration that occurs during the production of voiced speech.

In recent years, various types of pitch period extraction methods have been applied. We have developed and implemented a new technique for determining the pitch using the short time cepestral coefficient .Cepestral coefficient is the log of Fourier transforms of speech signal calculated frame by frame. Here we are assuming that a female & male speaker speaks with the frequency of 50Hz-300Hz and 200Hz-450Hz respectively thus our region of interest is 50Hz-450HZ. Differential of cepestral coefficient gives us a constant for fixed spectral slope and peaks in the spectrum get well preserved, these peaks with respect to 50 Hz for each frame gives us the relative pitch frequency for that frame.

The value of ω obtained by solving Eq. 5 gives us the relative pitch frequency.

$$\left(\frac{\partial}{\partial\omega}\left[\log\left|X(e^{j\omega})\right|\right]\right) = 0 \tag{5}$$

Where, $X(e^{j\omega})$ is the Fourier transform of x(t).

Finally, we are normalizing the whole pitch to make sure that the relative pitch doesn't make any difference.

After extracting the features we take some samples as trainer with their known classes. Then we apply the k-NN algorithm for every new coming data point to find it's class.

E. Analytical hierarchical Process (AHP)

AHP is a structured technique for organizing and analyzing complex decisions [7]. We are applying this method to determine the weights to be assigned to different features based on their relative importance. For this purpose we prepare a pair-wise comparison matrix using a scale of relative importance. If a feature is compared to itself then it is assigned a value 1. Importance 3, 5, 7, and 9 verbally means 'moderate importance', 'strong importance', 'very strong importance', and 'absolute importance' respectably.

We also need to do the consistency check on the prepared matrix and if Consistence Ratio (CR) is less than0.1 then we can accept the calculated weights. [7]

We are using AHP for deciding weight for these feature vector based on their importance. Experimentally we found that pitch period has the highest priority, then comes PSD and

ZCR and STE has the same priority. Depending upon their priority we are giving equal importance to ZCR and STE, moderate importance to PSD over ZCR and STE and strong importance to PP over ZCR and STE. PP is considered moderately important over PSD. The resultant Importance matrix I is given by Equation 6.

$$I = \begin{array}{ccccc} - & \text{ZCR} & \text{STE} & \text{PSD} & \text{PP} \\ \text{ZCR} & 1 & 1 & 1/3 & 1/5 \\ \text{STE} & 1 & 1 & 1/3 & 1/5 \\ \text{PSD} & 3 & 3 & 1 & 1/3 \\ \text{PP} & 5 & 5 & 3 & 1 \end{array} \quad (6)$$

From I matrix we calculate weight for each feature using the AHP and got $W_{ZCR}= 0.0993$,$W_{STE} = 0.0963$,$W_{PSD}= 0.2495$, $W_{\text{Pitch Period}}= 0.5579$.

We also went thru the consistency check and found the Consistency Ratio (CR) = 0.0163 which is acceptable and hence we can move forward with above calculated weights.

F. K Nearest Neighbor (K-NN) Algorithm

For efficient pattern classification we are using k Nearest Neighbor algorithm. K-NN algorithm measures the distance between real time voice sample $X$and the set of stored voiced sample. Euclidian distance d of a new voice sample X from a stored sample X_1 is calculated as in the equations 7.

$$d = \sqrt{\begin{array}{c} W_{ZCR}\left(X_{ZRC} - X_{1,ZRC}\right)^2 + W_{STE}\left(X_{STE} - X_{1,STE}\right)^2 + \\ W_{PSD}(X_{PSD} - X_{1,PSD})^2 + W_{PP}(X_{PP} - X_{1,PP})^2 \end{array}} \quad (7)$$

Cluster formation occurs depending upon the closeness of real time sample with different set of data samples. The class of the new data point is decided using majority voting of k-nearest neighbors.

IV. EXPERIMENTS

Although the proposed algorithm can be used to recognize any number of real-time voice inputs, but in the current experimental work we have trained the algorithm for 10 different voice input “zero”, “one”, “two”, “three”, “four”, five”, “six”, “seven”, “eight”, “nine”.

The algorithm first takes a 1-second input voice signal. Normally a person takes less than a second time to speak any of the above mentioned voice inputs, so we will have to remove the part of the signal which consists of silence.

After the silence removal stage, we sample the voice input with a frequency of 8000 Hz, then divide the whole voice sample in a frame of 12.5 ms, so that each frame will have N= 100 samples. To maintain continuity between the frames, we convolute them with a N point rectangular hamming window w(n) given by Eq. 8.

$$w[\,n\,] = 0.54 - 0.46\cos\left(2\pi \frac{n}{N}\right), \quad 0 \le n \le N \qquad (8)$$

For each frames we calculate STE, ZRC and pitch period independently. The PSD is calculated for the complete silence time plot for the signal of the word “one”. The signal is of 1 second duration and is divided into 8000 samples. We have taken 100 samples of each word “zero”, “one”,…”nine” for training purpose. Three different users (consisting of two male and one female) have given the inputs for training sample. For each of the samples we have determined all the features and stored them in our training data-base. Figure 3, 4, 5 and 6 shows STE,ZRC, PSD and Pitch Period calculated for the word “one”.

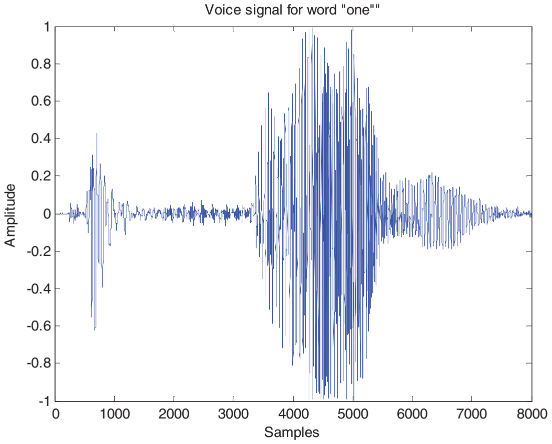

Figure 2:- Voice signal for the word "one". The 1 second duration of the time axis is divided into 8000 samples. We can clearly observe the silence in the signal till around 3200-3400 samples starting from sample 0.

Now for each incoming real time data input we apply k-NN classification algorithm. The weighted Euclidian distance is calculated as in Eq. 7, where the weights are decided scientifically by applying AHP. The class of the sample is decided by majority voting from 10 nearest samples (k=10 in k-NN). The maximum accuracy for 200 real-time input samples in this case was found to be near 83%.

But in any real world scenario the model with this kind of poor accuracy will not be accepted, so we decided to adapt a different training policy. For this we have taken the 100 training inputs (for each word) from a single user and applied our algorithm to recognize his/her own real time voice input
We were able to get a maximum accuracy of around 90% in this case. We further extended the training samples from 100 to 200 and observed the maximum accuracy of around 92 %.

V. RESULTS

We have tested the algorithm for recognition of ten digits (zero to nine). The experimentation is done with different kind of training and test inputs. The accuracy obtained in different cases is listed in Table 1.

In Table 1, Multiuser case is a 3 user input case with two male and one female users. The test samples are also randomly chosen from any of these three users. In the single user case the training samples and test sample are taken from a single user.

I. CONCLUSION

In this paper we have proposed a novel pattern recognition model for recognizing real time voice data input. The proposed

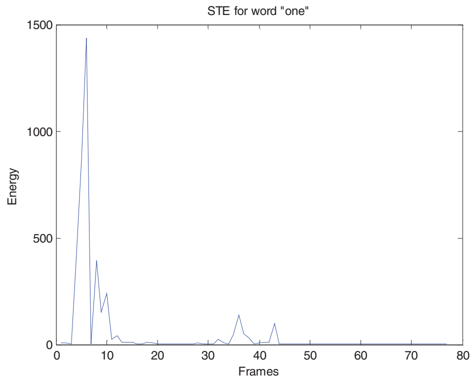

Figure 3:- STE for the word "one".

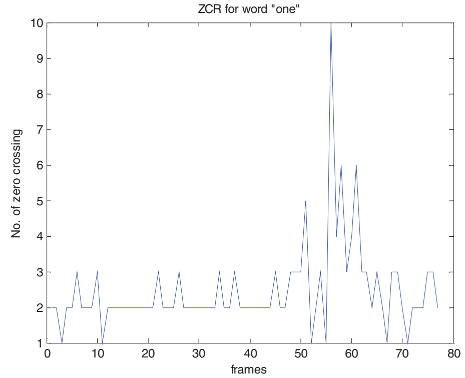

Figure 4:- ZCR for the word "one".

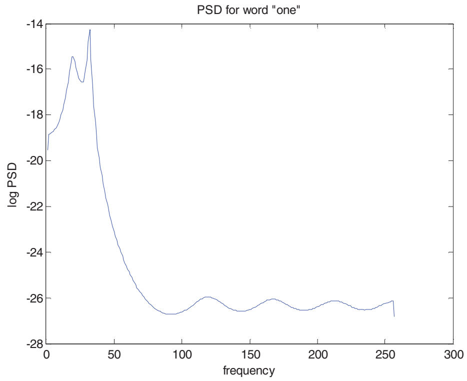

Figure 5:- PSD for the word "one".

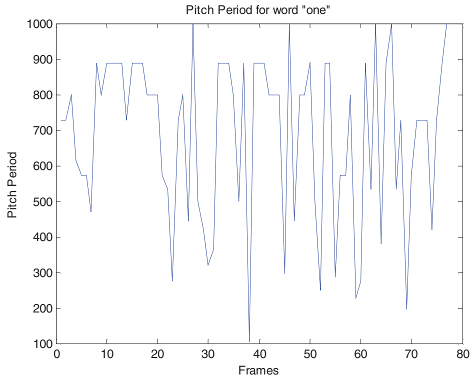

Figure 6:- Pitch Period for the word "one".

Training Case	No. of Training Samples (per word)	No. of Test Samples	Percentage Accuracy
Multi user input	100	100	81.0%
Multi user input	100	200	83.0%
Single user input	100	100	87.0%
Single user input	100	200	89.5%
Single user input	200	100	90.0%
Single user input	200	200	92.0%

Table 1: Different training cases, size of training data and accuracy obtained for different number of test samples.

model applies k-NN algorithm for classification by using AHP to scientifically decide the weights of different features.

We can conclude from the results section that our model is performing well if it is trained for a single user, and can be used in various commercial situations. But for multiuser case the accuracy is not very high. This can be considered as a limitation of the proposed model and provides the scope for the future work. The combination of more than one different classifier can be used for increasing the accuracy in case of multiple users.

II. References

[1] L. Rabiner and B. H. Juang, "Fundamental of speech recognition", Prentice Hall.

[2] M. Radmard, M. Hadavi and M. M. Nayebi, " A New Method of Voiced /Unvoiced Classification Based on Clustering", Journal of Signal and Information Processing , 2011,2, pp 336 - 347 .

[3] B. Jounghoon and K. Hanseok, “Spectral Subtraction Using Spectral Harmonics for Robust Speech Recognition in Car Environments”, ICCS 2003, LNCS 2660, pp.1109-1116 .

[4] P. Khunarsal, C. Lursinsap, and T. Raicharoen, " Singing Voice Recognition based on Matching Spectrogram Pattern", Proceeding of International Joint conference on Neural Networks, Atlanta, Georgia, USA , June 14-19, 2009, pp 1595-1599.

[5] L. Rabiner, and B. S. Atal, "A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with application of speech recognition ", IEEE translation on Acoustic, speech and signal processing , Vol. ASSP-24, NO.3, June 1976, pp 201-212.

[6] B.H Juang and L.R. Raibner, " Hidden markov model for speech recognition", TECHNOMETRICS, August 1991, Vol.-33, No.3

[7] R. V. Rao, “Decision Making in the Manufacturing Environment: Using Graph Theory and Fuzzy Multiple Attribute Decision Making Methods”, Springer Series in Advanced Manufacturing. 2013.