Voice Authorization using Voice Recognition and Verification Method

Yimon Hlaing, Soe Soe Aye

University of Computer Studies, Kyainge Tong

yimon.ucsvoice2009@gmail.com, ucsktg2008@gmail.com

Abstract

A method for enabling to obtain access via a microphone without typing password in login page is described. Voice recognition is the task of determining the identity of a person from his or her voice. Voice verification is defined as deciding if a speaker is who he claims to be. After entry and recognition of password, the set of parameters are used to generate speaker verification for password has a predetermined relationship, the user is accepted. If not, the user is rejected to allow access. A voice recognizer performs voice recognition functions on the speech. A frame mapper marks the beginning and ending of each digit, if the password utterance is numbers, within the utterance. Verification method generates a verification signal in response to verifying a minimum number of the passwords within the utterance being matched with the verification templates.

Keywords: Voice recognition, Voice verification, parameters, matching.

1. Introduction

Of the many types of biometric technologies available today, voice identification and authentication solution have a unique edge over much of competition. Verifying people identities is one of the immense demands in many applications that require access. Using driver's license or other identity cards proved very ineffective. Thus, alternative ways have to be investigated. One of the most successful ways is identifying people through their biometric attributes. A successful biometric is human voice which is the foundation of speaker recognition research field. In many situations, matching the voice requires the entry of numerous voice samples. Furthermore, processing the numerous samples each time access is necessary takes a large amount of computing power and time. This is especially unnecessary when most of the voice content will be numbers, names, codes, phrases, etc.

Voice recognition (Speaker recognition) is the task of determining the identity of a person from his/her voice. It is an object of the present invention to recognize alphanumeric strings spoken over a microphone. It is another object of the invention a method for recognizing alphanumeric strings where recognition occurs on the basis of an ensemble of alphanumeric characters as opposed to individual character recognition. This is provided in a method for enabling a user to obtain access to services via a microphone by entering a spoken password. Each spoken password is then recognized using voice recognition algorithm.

Voice verification biometric system verifies an individual identify by comparing an individual's live voice to stored voice samples. This type of system is different from voice identification technology, which is designed to identify the person who is speaking out of database of known individuals. Furthermore, this system differs from speech recognition technology which is designed to understand the words spoken, regardless of the speaker. The biometric features used in comparing different voice prints can include the base tones, nasal tones, larynx vibrations, cadence, pitch, tone, frequency and duration of the voice. Because of the variety of features in the human voice, recordings and reproductions cannot deceive sophisticated voice verification systems.

2. Related Work

Linear predictive coding (LPC) is a tool used mostly in audio signal processing and speech processing for representing the spectral envelope of a digital signal of speech in compressed form, using the information of a linear predictive model[3,4]. It is one of the most powerful speech analysis techniques, and one of the most useful methods for encoding good quality speech analysis techniques, and one of the most useful methods for encoding good quality speech at a low bit rate and provides extremely accurate estimates of speech parameters [10].

Waveform ROM in digital sample-based music synthesizers made by Yamaha Corporation is compressed using LPC algorithm. 0-to-32nd order LPC predictors are used in FLAC audio codec [9]. Because speech signals vary with time, this process is done on short chunks of the speech signal, which are called frames; generally 30 to 50 frames per second give intelligible with good compression [5]. The basic techniques of recognizers (without

intelligence) are following wide varieties of approaches with different claims of success by each group of authors who put their faith in their favorite way. However, the sole technique that gains the acceptance of the researchers to be the state of the art is Hidden Markov Model (HMM) technique [1, 2, 7]. HMM is agreed to be the most promising one. It might be used successfully with other techniques to improve the performance, such as hybriding the HMM with Artificial Neural Networks (ANN) algorithms [7, 8].

A caller can say a phone number to be dialed by, for example, a long distance service provider or a cellular phone service provider and the present invention is used to verify that the caller is authorized to use the account [6]. Gaussian HMM approach is to be preferred for real world speaker verification when only limited amounts of training data are available. Results will be discussed for both text-dependent and text-independent speaker verification [5].

Many spectral-analysis methods have been proposed for speech-signal modeling. These include such standard methods as measurement of the discrete (fast) Fourier transform (FFT), all-pole minimum-phase linear prediction (LPC) methods, and autoregressive/moving average models (Allen and Rabiner 1977; Atal and Hanauer 1971; Cadzow 1982; Makhoul 1975; Markel and Gray 1976; Schafer and Rabiner 1971). Even the more traditional filter-bank method of spectral analysis is still used in some systems (Dautrich, Rabiner, and Martin 1983), particularly in hardware implementations. To emphasize spectral properties that are known to be important to a human listener, auditory models can be incorporated in the overall spectral representation (Cohen 1985; Ghitza 1986). In speech modeling, we often call this short-time spectral vector an observation vector or simply an observation. In Figure 2, where the analysis mechanism is illustrated, we use a 30-msec Hamming window (frame) with successive spectral frames spaced 15 msec apart. FFT spectra of the first four frames of the signal are plotted in the figure, each being fitted with a 10th order LPC all-pole smoothed model spectrum.

3. Proposed Method

There are two major steps in the proposed voice recognition and verification algorithm. First, the signal is represented using LPC analysis. These representations are then recognized by Hidden Markov Model (HMM). Finally, the recognized parameter vectors are matched with prestored authorized user's vectors. Authorization / non-authorization are detected from this utilizing a decision scheme. Details of the algorithms are explained in the following sections.

3.1. Voice recognition

The input signal x (n) is multiplied by a Hann window to yield successive windowed segments of $x_s(n)$. These windowed segments are then transformed into spectral domain by using LPC.

3.1.1 LPC Feature Analysis

One way to obtain observation vectors O from speech samples s is to perform a front end spectral analysis. The type of spectral analysis that is often used is called Linear Predictive Coding (LPC) and a block diagram of the steps that are carried out is given in Fig.1.The overall system is a block processing model in which a frame of N_A samples is processed and a vector of features O_t is computed. The steps in the processing are as follows:

(1) Preemphasis: The digitized speech signal is processed by a first-order digital network in order to spectrally flatten the signal.

(2) Blocking into Frames: Sections of N_A consecutative speech samples are used as a single frame. Consecutive frames are spaced M_A samples apart (we use M_A=100 corresponding to 15ms frame spacing, or 30ms frame overlap).

(3) Frame Windowing: Each frame is multiplied by an NA sample window (we use hanning window) w (n) as to minimize the adverse effects of chopping an N_A samples section out of the running speech signal.

(4) Autocorrelation Analysis: Each windowed set of speech samples is auto correlated to give set of (p + 1) coefficients where p is the order of the desired LPC analysis.

(5) LPC/Cepstral Analysis: For each Frame, a vector of LPC coefficients is computed from the autocorrelation vector.

(6) Cepstral Weighting: The Q-coefficient cepstral vector c_t (m) at time frame t is weighted by a window w_c (n) of the form,

$$w_c(n)=1+\frac{Q}{2}\sin\left(\frac{\pi M}{Q}\right), 1\leq m\leq Q \quad (1)$$

to give

$$\hat{c}_t(m)=c_t(m).w_c(n) \quad (2)$$

(7) Delta Cepstrum: The time derivative of the sequence of weighted cepstral vectors is approximated by a first-order orthogonal polynomial over a finite length window of (2k+1)

frames, centered around the current vector. The cepstral derivative is computed as

$$\Delta c\hat{}_t(m)=[\sum_{k=-K}^{K} kc\hat{}_{t-k}(m)].G_t, 1\leq m\leq Q \quad (3)$$

Where G is a gain term chosen to make the variances of $c\hat{}_t(m)$ and $\Delta c\hat{}_t(m)$ equal.

3.1.2 Hidden Markov Model

In order to do isolated voice recognition, we must perform the following:

(1) For each word v in the vocabulary, we must build an HMM λ^{v}, i.e.; we must estimate the model parameters (A, B, π) that optimize the likelihood of the training set observation vectors of the v_{th} word.

(2) For each unknown word which is to be recognized, the processing of Fig.2 must be carried out, namely measurement of the observation sequence O= {O_1, O_2,...O_T}, via a feature analysis of the speech corresponding to the word; followed by calculation of model likelihoods for all possible models, P (O/ λ^{v}), 1<v<V; followed by selection of the word whose model likelihood is highest, i.e.;

$$v^*=\text{argmax}_{1\leq v\leq V} [P(O/\lambda^{v})] \quad (4)$$

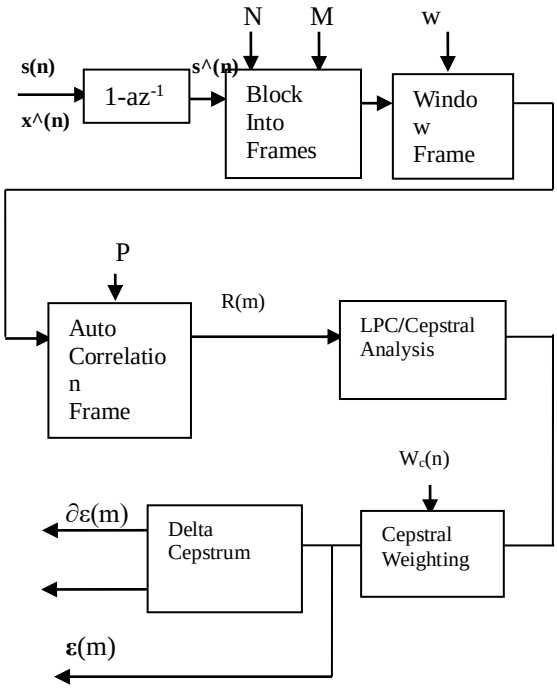

Figure 1. Block diagram of the computations required in the front end feature analysis of the HMM recognizer.

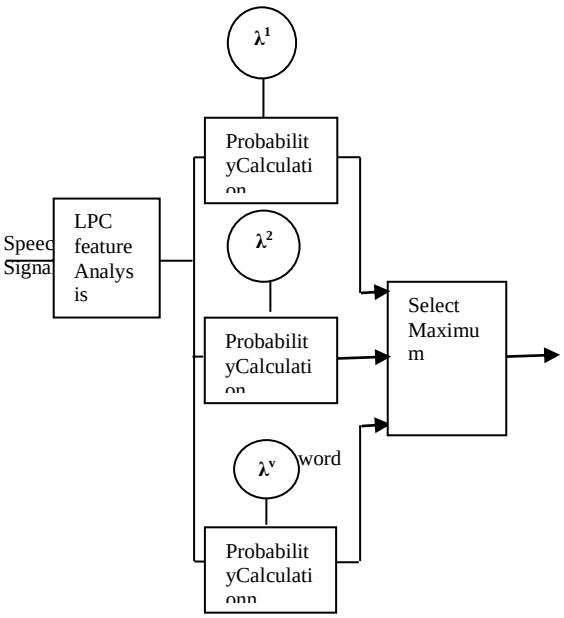

Figure 2.Block diagram of an isolated HMM recognizer

3.2 Voice Verification

To effect voice verification, voice verification parameter data vector p is then input to a verifier routine which also receives the voice verification class reference data for user. Specifically, the voice verification class reference data is provided from the voice verification reference database.

Verifier routine generates one of two different outputs: ACCEPT, REJECT. By way of background, the routine begins after the determination, preferably by the voice recognition algorithm, that the password is valid. Although in the preferred embodiment each voice verification parameter vector is generated as each digit is recognized, it is equally possible to refrain from generating the voice verification parameter vector until after a test is perform to determine whether the password is valid. In particular, the N_p element voice verification parameter vectors for each digit of the spoken password are compared with the previously-generated voice verification class reference data vectors stored in the voice verification reference database. This verification routine is calculated with Equation 5.

D= sum ((X-Y).^2).^0.5 (5)

where,

D=Euclidean distance
X=prestored voice codes
Y=user's live voice password codes.

3.3. Sample Entropy

Entropy is the rate of information production. Methods for estimation of entropy of a system represented by a time series are not, however, well suited to analysis of the short and noisy data sets encountered in cardiovascular and other biological studies. Entropy provides a complexity measure of a time series, such as discredited biological signal.

SampEn (m,r,N) is precisely the negative natural logarithm of the conditional probability that two sequences similar for m points remain similar at the next point, where self-matches are not included in calculating the probability. Thus a lower value of SampEn also indicates more self-similarity in the time series.

For an input signal u of length N, {u(j): $1 \le j \le N$} forms the N-m+1 vectors. $X_m(i)$ for { i / $1 \le i \le$ N-m+1} where $X_m(i)$={u(i+k) :$0 \le k \le$m-1} is avector of m data points from u(i) to u(i+m-1). In this context, only the first N-m vector of length m are considered to ensure that, $X_m(i)$ and $X_{m+1}(i)$ are defined for $1 \le i \le$N-m. Let $B^m(r)$ be the probability that two sequence will match for m points and $A^m(r)$ be the probability that two sequences will mtch for m+1 points.

B(r) is defined as $(N-m-1)^{-1}$ times the numbers of vectors $X_m(j)$ within r of $X_m(i)$, where $i \le j \le$N-m, and $j \neq I$ to exclude self matches. Then $B^m(r)$ is defined as

$$B^m(r) = (N_M)^{-1} \sum_{i=1}^{N-1} B_i^m(r) \qquad (6)$$

Similarity, A is defined as $(N-m-1)^{-1}$ times the numbers of vectors $X_{m+1}(i)$, where $i \le j \le$N-m and $j \neq i$. Then set $A^m(r)$ as

$$A^m(r) = (N-m)^{-1} \sum_{i=1}^{N-1} A_i^m(r) \qquad (7)$$

Finally, Sample Entropy (SampEn) is calculated by

$$\text{SampEn}(m,r,N) = -\ln \frac{A^m(r)}{B^m(r)} \qquad (8)$$

4. System implementation

For Each word v in the vocabulary, we must build an HMM λ^v, i.e., we must estimate the model parameter (A,B,π) that optimize the likelihood of the training set observation vectors for v^{th} word.

T=Length of the sequence of observations (training set)

N=number of states (we either know or guess this number)

M=number of possible observations (from the training set)

Omega_X= {q_1,…,q_N} (Finite set of possible states)

Omega_O= {v_1,…,v_M} (Finite set of possible observations)

X_t random variable denoting the state at time t (State variable)

O_t random variable denoting the observations at time t (output variable)

Sigma=o_1,…,o_T (sequence of actual observations)

Distributional parameters are:

A= $\{a_{ij}\}$,a_{ij}=Pr(X_t+1=q_i) (trasition probabilities)

B= {b_i},b_i(k)= Pr (O_t=v_k /X_t=q_i_t) (observation probabilities)

Pi={pi_i},pi_i=Pr(X_O=q_i) (initial state distribution)

The probability computation step is generally performed using the Viterbi Algorithm(i.e., the maximum likelihood path is used) and requires on the order of $V.N^2.T$ computations.

V=number of words,

N=number of states,

T=observations.

In Viterbi Algorithm,

Input : prior (i)=Pr (Q(1)=i)
transmat(i,j)=Pr(Q(t+1)=j/Q(t)=i)
obslik (i, t)=Pr (y(t)/Q(t)=i)

Outputs :path(t)=q(t), where $q_1,\ldots,q_t$ is the argmax of the expression.

Delta(j,t)= probabilities of the best sequence of length t-1 and then going to state j and O (1:t)

Psi (j, t)= the best predecessor state given that we ended up in state j at t.

The voice recognition algorithm determines the closet match. Each has its own matching score. The lowest value of matching score is first choice. And the difference between the lowest value and the highest value is second choice. The first choice and second choice are the predetermined values for voice verification. When all words of the password have been recognized, the voice recognition portion of the method is complete.

For verification of authorization, we use Euclidean distance to calculate the matching of voice codes from HMM and from prestored voice codes' database. First, a weighted Euclidean distance d(i) is computed for each digit where: p(i, j) is the j^{th} component of the length N_p vector generated from the i^{th} digit in the length N_d current password entry sequence, p(i, j)is the j^{th} component of the reference vector of the i^{th} digit for the alleged enrolled caller, w_1 is a constant weighting vector, precalculated to yield optimum system performance, and d(i) is the resultant

weighted Euclidean distance measure for the i^{th} digit in the current password entry sequence.

The distance vector d is then sorted in ascending order. An ensemble distance is then calculated as a weighted combination of these sorted distances, where d is the sorted distance vector w_2 is another constant weighting vector, precalculated to yield optimum system performance and D is the resultant ensemble distance measure for the entire current password entry sequence, with respect to the alleged enrolled caller. The ensemble distance is compared to acceptance thresholds. If the ensemble distance is below the acceptance threshold, the test is positive and the user is given the authorized message. If the distance is greater than the threshold, the user's access is denied and the method terminates.

And then,we calculated the sample entropy for time complexity. SampEn measures the regularity of data sequence. A low value of SampEn indicates that the sequence is regular. With increasing is regularity a larger value of SapEn is defined.

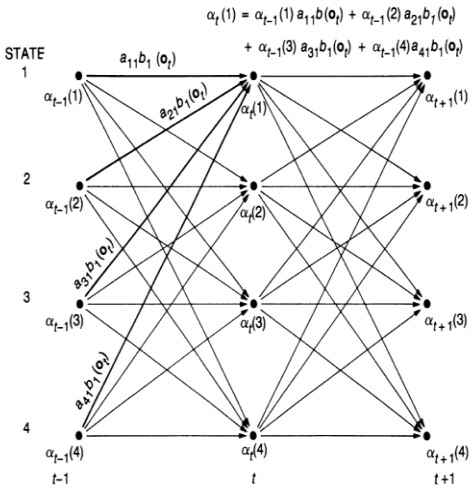

Figure 3. State structure for HMM probabilities calculation

5. Experimental Results

In this paper, the performance of proposed method is evaluated using the speech signals passwords from prestored database .We tested the algorithm with 100 speech signals spoken by 10 speakers (5 males and 5 females) using a Pentium 4 processor and we obtained an average of correct recognition rate of up to 95% and rejection rate is 0.001%. Matching Ratio is calculated with Equation 6.In each of the above databases there were variable length password strings with from 1 to 7 words per string and allowing time for speaking password is 10s. The performance of the HMM recognizer and verification, is given in Table.1, where the entries in the table are average string error rates for cases in which the string length was unknown apriority(UL) and for cases in which the string length was known apriority (KL). Results are given both for the training and recognition set (from which word models were derived), and for the verification test.

$$MR = \frac{N_z * 100}{T_n} \quad (9)$$

where, MR=matching ratio

N_z =number of equal bits of voice codes

T_n =number of total bits.

Table 1. Performances of the HMM Recognition and Verification

	Training and Reconition Set		**Verification test**	
Mode	**UL**	**KL**	**UL**	**KL**
Speaker Trained (10 talkers)	**0.39**	**0.16**	**0.78**	**0.35**
Multi Speaker	**1.74**	**0.98**	**2.85**	**1.65**
Speaker Independent (20 talkers)	**1.24**	**0.36**	**2.94**	**1.75**

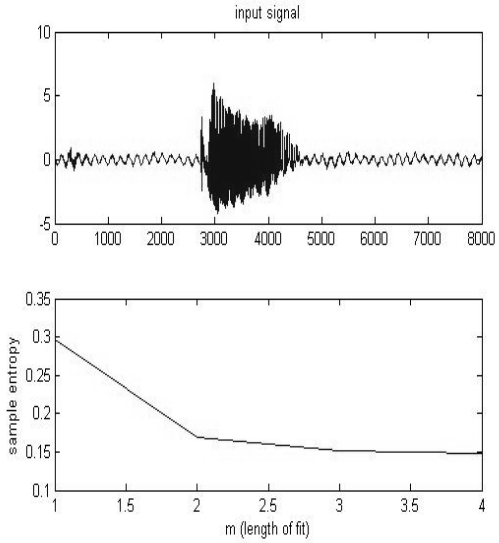

Figure 4. Real password of authorized person

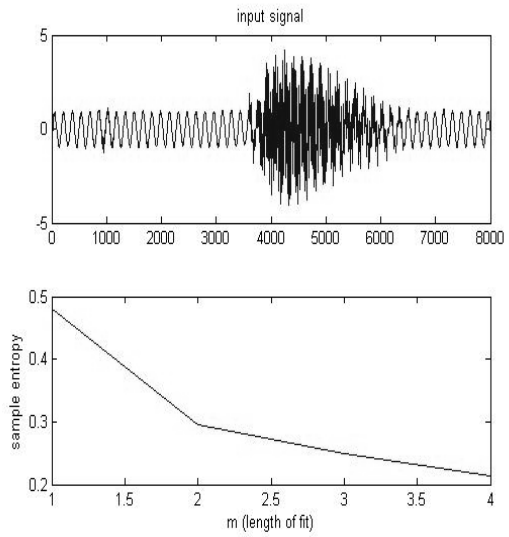

Figure 5. Same password of authorized person spoken by unauthorized person

6. Conclusion

The proposed algorithm is developed using MATLAB. This relates generally to voice processing and more particularly to a method and system for performing speaker verification on a spoken utterance. System verifies an individual's identify by comparing an individual's live voice to store voice samples. This thesis employs the users to login without typing password and evaluate the performance of voice recognition and verification.

7. References

[1] L.E. Baum and T.Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," Ann. Math. stat., Vol.37, pp.1557- 1563, 1996.

[2] L.E. Baum and J.A.Egon, "An inequality with applications to statistical estimation for probabilistic functions of a Markov PROCESS AND TO MODEL FOR ECOLOGY," Bull, Armor. Meteorol. Soc, Vol.73, pp.360-363, 1967.

[3] J.K Baker, "The dragon system. An overview" IEEE Trans. Acoustic Speech Signal Processing, vol. ASSP-2 no.1, pp.24, Feb, 1975.

[4] F.Jetinek, "Continuous speech recognition by statistical methods", Proc.IEEE, vol.64, pp.532-536, Apr.1976.

[5] R.Bakis, "Continuous speech word recognition via cent second acoustic states," in Proc.ASA Meeting (Washington, DC), Apr.1976.

[6] B.H Juang, "On the hidden Markov model and dynamic time warps for speech recognition-A unified view", AT & T Tech's: vol.63, no.7, pp.1213-1243, Sept, 1984.

[7] D.Slezad, P.Synak, A.Wieezorkowska and J.Wroblewski (July 2003), Application of temporal descriptors to musical instrument sound recognition. Journal of Intelligent Information Systems.81 (1):71-93.

[8] E.Scheirer and M.Slancy (April 1997), Construction and Evaluation of a Robust Multi-Features speech/music discriminator. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing Proceedings.2p 1334.

[9] M.Abe and M.Nishiguchi (August 2002), Self optimized spectral correlation method for background music identification. Proceedings 2002 IEEE International Conference on Multimedia and Expo (Cat.No.02TH8604) 1. (1):333336.

[10] D.Li, J.K.Sethi, N.Dhnitrova and T.McGee (April 2001), Classification of general audio data for content-based retrieval. Pattern Recognition Letters.22 (5):533-544.