Speaker Verification System

*Speaker recognition* is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. This technique makes it possible to use the speaker's voice to verify their identity and control access to services such as voice dialing, banking by telephone, telephone shopping, database access services, information services, voice mail, security control for confidential information areas, and remote access to computers.

Speaker recognition can be classified into identification and verification. *Speaker identification* is the process of determining which registered speaker provides a given utterance. *Speaker verification*, on the other hand, is the process of accepting or rejecting the identity claim of a speaker.

All speaker recognition systems contain two main modules

1)*feature extraction* 2)*feature matching*.

Feature extraction is the process that extracts a small amount of data from the voice signal that can later be used to represent each speaker.

Feature matching involves the actual procedure to identify the unknown speaker by comparing extracted features from his/her voice input with the ones from a set of known speakers.

Features:

We have used two type of features in our project.They are MFCC and Wavelet based.

MFCC:

First the silence frames are removed from the speech signal using the threshold energy value.

MFCCs are commonly derived as follows:

-The short term fourier transform is performed on the speech signal frame.

-For each frame the STFT magnitude spectrum is computed and further processes by using the 24 triangular shaped mel-filter banks to find out the filter bank energies.

-Then discrete cosine transform (DCT) is taken on the spectral energies, to obtain MFCC.

-13-dimensional MFCC is considered (Excluding Co)

Wavelet

The new features are named subband based cepstral parameters (SBC).

This coefficients are to be compared with the standard MFCC.

There are two parameters in MFCC

1. The first stage is the computational filter bank energies and
2. The second stage is the decorrelation of log filter bank energies with a DCT to get MFCC.

Even the SBC parameters derivation follows the same process expect the filterbank energies are derived from the wavelet packet transform rather than short time fourier transform. Here these features outperform MFCCs.

Subband energy computation

SBCCC

SBC

DCT

Perceptual Wavelet packet trandsform

Windowing

TRAINING PHASE

The basic for both identification and verification is GMM which is used to represent speakers. The distribution of feature vectors extracted from a person’s speech is moduled by Gaussian mixture density.

For a D-dimensional feature vector denoted as x, the mixture density for speaker is defined as

The density is a linear combination of M component uni-modal gaussian densities, each parameterized by a mean vector and covariance matrix

IDENTIFICATION SYSTEM

The identification is a straight forward maximum likelihood classifier. For a group of S speakers represented by models

The main objective Is to find the speaker model which has the maximum posterior probability for the input feature vector sequence, X= { }, The minimum error bayes’ decision rule is

Assuming equal prior probabilities of speakers the term p() and p() are constant. Using logarithmic and the assumed independence between observations, the decision rule becomes

VERIFICATION SYSTEM

Here the verification task is very though than identification task. The system must decide if the input voice came from the claimed speaker, with a well defined, or not the claimed speaker which is ill-defined. Cast in hypothesis testing frame work, for a given input utterances X and a claimed identity the choice is between and ;

: X is from the claimed speaker

: X is not from the claimed speaker

To perform the optimum likelihood ratio test to decide between and then requires model of the universe of possible non –claimant speakers. The application of this hypothesis testing approach is first described, followed by a discussion of a techniques for selecting speakers for modelling the non claimant alternative hypothesis.

Sbc:

Add\_noise\_to\_files.m ->adds the noise

Create folder sbc\_train\_noise\_10 ,sbc\_test\_noise\_10 where noise can be white , car , babble ,factory

Create spkr\_verifydata\_car\_10\_2+3- means we are creating speaker verification data with car noise of 10 db which includes 2 positive test cases and 3 negative test cases.

10db denoted snr value.

Sbtrain->train sbc features

Sbctest-> create sbc test features

Gmmtraining.m

testingwithGMMmodel 🡪spkr recognition percentage

Findingthreshold.m

Final\_spkr\_verification.m

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | mfcc |  | Sbc |  |  |
| NOISE(10 db) | ACC | PRECISE | %spk ID | ACC | Precise | %spk ID |
| WHITE | 40 | 42.3 | 9.1 | 45.6 | 39.7 | 33.3 |
| CAR | 42.6 | 45.8 | 49.4 | 44.1 | 47.6 | 57.8 |
| BUBBLE | 34.3 | 35.4 | 36.7 | 35.5 | 37.9 | 39.6 |
| FACTORY | 44.2 | 46.9 | 46.4 | 76.3 | 78.06 | 44.55 |

Finally we also developed gui model for speaker verification system .The function is as follows:

function mygui()

figure('Menubar','none','Name','spkr\_verify\_system','NumberTitle','off','Position',[100,100,600,600]);

b1 = uicontrol('Style','PushButton','String','load speaker model','Position',[150,400,300,50],...

'CallBack',@b1Pressed);

function b1Pressed(h, eventdata)

[filename,pathname] = uigetfile('\*.mat','Select the MATLAB code file');

assignin('base','f1',filename);

assignin('base','p1',pathname);

uiimport(fullfile(pathname,filename));

b2 = uicontrol('Style','PushButton','String','load test file','Position',[20,300,200,50],...

'CallBack',@b2Pressed);

function b2Pressed(h, eventdata)

[filename,pathname] = uigetfile('\*.wav','Select the MATLAB file');

assignin('base','f2',filename);

assignin('base','p2',pathname);

uiimport(fullfile(pathname,filename));

b3 = uicontrol('Style','PushButton','String','extract features','Position',[380,300,200,50],...

'CallBack',@b3Pressed);

function b3Pressed(h, eventdata)

disp('extracting features.................');

y2 = evalin('base', 'data');

sig=y2.\*y2;

E=mean(sig);

Threshold=0.04\*E;

k=1;

dest = 0;

for b=1:100:(length(sig)-100)

if((sum(sig(b:b+100)))/100 > Threshold)

dest(k:k+100)=y2(b:b+100);

k=k+100;

end;

end;

dest=dest';

y3=mfcc\_rasta\_delta\_pkm\_v1(dest,8000,13,26,20,10,0,0,1);

assignin('base','y3',y3);

disp('feature extraction done.....');

b4 = uicontrol('Style','PushButton','String','check claimed speaker','Position',[150,200,300,50],...

'CallBack',@b4Pressed);

function b4Pressed(h, eventdata)

threshold = 4218.2;

p1 = evalin('base','p1');

f1 = evalin('base','f1');

a = dir(p1);

mix1 = evalin('base','MIX');

y1 = evalin('base','y3');

verification = zeros(2,1);

verification(1) = mean(log(gmmprob(mix1,y1))/length(y1));

for k=3:length(a)

if(~strcmp(f1,a(k).name))

assignin('base','mix1',load(fullfile(p1,a(k).name)));

verification(2) = verification(2) + mean(log(gmmprob(mix1,y1)/(length(a)-3)));

end

end

tilda = verification(1)-verification(2)

if(tilda < threshold)

msgbox('claimed spkr is true');

else

msgbox('claimed spkr is false');

end

end;

end

end

end

end