Speaker Recognition using Mel Frequency Cepstral Coefficients with Spectral Subband Centroids Features

Roy Amante Salvador and Isabelle Tingzon
Department of Computer Science
College of Engineering, University of the Philippines Diliman
Quezon City Philippines, 1101
(632) 434 3877

***Abstract*—Speaker recognition is the process of recognizing a speaker's identity by his or her voice. Humans sound differently and there are features in our speaking voice which differentiate us from other people. In this paper, we show an end-to-end implementation of a Speaker Recognition System. One of the most popular features used in Speaker Recognition which is based on the human Auditory System is the Mel Frequency Cepstral Coefficients (MFCC). MFCC yields high performing results however is quite sensitive to additive noise distortion. When supplemented with feature robust to noise such as the Spectral Subband Centroids (SSC), speaker recognition performance increases. We also show an array of binary SVM per speaker outperforms other out-of-the-box multi-class classifiers in determining the speaker in audio frames.**

***Keywords*—Speaker Recognition, MFCC, SSC, SVM**

I. Introduction

The human voice contains acoustic patterns and characteristics that differ from person to person depending on one's anatomical structure of the vocal tract, speaking style, and accents. These distinctive features can be used to identify a speaker. *Speaker recognition* is the process of automatically extracting, characterizing, and recognizing a speaker's identity based on the information available in his or her speech signal. Automatic speech recognition systems allow for users to verify their identity which has potential value in a range of applications, especially in security and identity management. Many devices today, for example, employ security features involving the human voice such as voice activated commands and voice biometrics as a means for access control and remote authentication. In terms of e-commerce, telephone banking which involves higher levels of customer verification can benefit from the advantages of an automated speaker recognition system. Speaker recognition can also be useful in law enforcement and criminal investigations (e.g. comparing the voice of an assailant against a database of suspects to find the closest match).

As in most pattern recognition systems, speaker recognition involves several steps including data acquisition, data pre-processing, feature extraction, and feature matching. In our implementation of the speaker recognition system, we develop an interface for capturing audio signals through a microphone, which we use to record audio signals of nine (9) different users. These audio signals were then segmented and pre-processed using the Hamming window function, and a threshold is applied to eliminate silent frames. The two more important phases of the project are the *feature extraction* and *feature matching*, or classification. In our implementation, we extract the Mel Frequency Cepstral Coefficients (MFCC) and Spectral Subband Centroids (SSC) features which are widely used in automatic speech and speaker recognition systems. For feature matching, we apply an array of classification algorithms including SVM, decision trees, and Naive Bayes and compare their performances against each other in terms of speed and accuracy. The code is developed in Python and performs speaker recognition successfully.

II. Literature Review

There exist various research describing an implementation of an end-to-end automatic speaker recognition system [7], [10], [17], [9], [18], [12]. In this section, we describe the techniques and methods used in related research for the selection and extraction of features as well as classification.

A. Features

Feature extraction allows us to identify the components of the audio signal that are good for identifying the linguistic content while discarding unnecessary information. Some of the features largely used in speaker recognition are the following [1]:

- Mel Frequency Cepstral Coefficents (MFCC)
- Spectral Subband Centroids (SCC)
- Log Filterbank Energies
- Filterbank Energies

In this paper, we will focus primarily on MFCC and SCC as these are the most widely used.

B. Mel-Frequency Cepstrum Coefficients (MFCC)

First introduced by Davis and Mermelstein in the 1980's, Mel-Frequency Cepstrum Coefficients (MFCC) is recognized as one of the best parametric representation of acoustic signal and has become widely used feature used in automatic speech and speaker recognition [1], [6]. MFCCs are described as follows [17]:

"MFCC is based on the human peripheral auditory system. Human perception of frequency contents of sounds for speech signal does not follow a linear scale. For each tone with an actual frequency t measured in Hz, a subjective pitch is measured on a scale called the Mel Scale".

To derive the MFCCs, the windowed audio signal is converted into frequency representation using Fast Fourier Transform. Powers of the resulting spectrum are converted to the Mel scale.

The Mel scale is used to relate perceived frequency (pitch) to the actual measured frequency and is used to make the features extracted match more closely what humans hear. The formula for converting from frequency to Mel scale is

$$M(f) = 1125 \ln \left(1 + \frac{f}{700}\right)$$

We take Discrete Cosine Transform (DCT) of the logarithm of the powers of each Mel frequency. The resulting amplitude of this spectrum are the Mel Frequency Cepstral Coefficients.

C. Spectral Subband Centroids (SSC)

In [14], Spectral Subband Centroids (SSC) was proposed as new features to supplement cepstral features for speech recognition. These features are similar to formant frequencies and can be easily extracted without any estimation errors from the power spectrum of the speech signal. The SSC is computed by dividing the frequency band into a fixed number of subbands and getting the centroid for each subband using the power spectrum of the speech signal.

D. Classification

The classification procedure, or *feature matching*, in most literature involves the application of supervised classification algorithms. The goal of a supervised training algorithm is to produce a learning model from a labeled training set. In supervised learning, a label, which in this case is the *speaker ID*, is associated with each feature vector and is used to determine the class to which the vector belongs. Many of these algorithms have been proposed to solve problems in binary class case but can be easily extended to multi-class case [2].

Recently, several supervised classification algorithms have been evaluated for speaker recognition. These include support vector machines (SVM) [12], [13], multilayer perceptrons (MLP) [9], vector quantization (VQ) [7], [9], decision trees [9], artificial neural networks [10], auto-associative neural network (AANN) [18], and modified neural tree network (MNTN) [9]. Recent advanced techniques have also been developed, one example being Soni and Nijhawan's implementation of text-dependent speaker identification system which takes into consideration what the speaker is saying [13]. In another research by Kari and Muthulakshmi, a real time implementation of speaker recognition is presented where MFCC is used with neural networks to achieve a satisfactory performance of 80% [10]. Related research also includes emotion identification from voice which applies similar methodologies [12]. The results show that the use of the extracted feature MFCC with SVM can achieve high levels of accuracy (up to 95%).

III. Architecture

A. System Diagram

As in any pattern recognition systems, speaker recognition involves several steps: data acquisition, data pre-processing, feature extraction, and feature matching. Figure 1 shows an overview of the system and its components.

Two important phases are feature extraction and feature matching. Feature extraction in the context of a speaker recognition system involves identifying the components of the audio signal that are good for recognizing linguistic content while discarding all other unnecessary information. Feature matching on the other hand involves using classification algorithms, which in this project, include binary SVM, multi-class SVM, decision trees and Naive Bayes classifier.

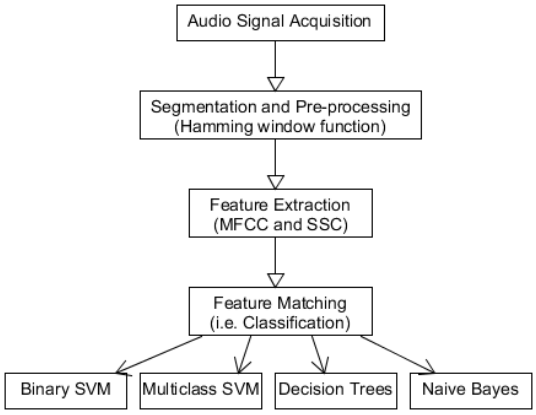

Fig. 1. Overview of the System

B. Audio Signal Acquisition

The system has an interface for capturing the voice of the speaker. Nine (9) users were asked to record their voices via the microphone and the said system interface. Each user was asked to pronounce two sets of words (one for the training set and the other for the test set) with a varied range of vowel sounds as described in Tables I and II.

TABLE I. Training Set

A	E	I	O	U
bag	egg	piece	on	use
cat	set	fit	pawn	book
par	mend	bin	ton	root
fan	fence	int	law	hoop
ant	ebb	reek	ore	use

TABLE II. Test Set

A	E	I	O	U
sad	end	ill	boss	hue
pal	zen	mint	ode	soot
cab	sense	feel	mode	cute
wham	well	pick	lawn	swoon
arg	lent	wig	ohm	loom

The audio signal recording is sampled at 44100 Hz using stereo channels and saved in signed 16 bit wave file format.

C. Segmentation and Pre processing

After obtaining the audio signal, we divide it into several overlapping frames. We used 10 ms overlap or skip size between 30 ms frames. Each frame is applied with a Hamming window function H(N). Hamming window is defined as

$$H(t) = 0.54 - 0.46\cos\left(\frac{2\pi t}{N-1}\right) \qquad 0 \leq t \leq N-1$$

where N is the frame size. Our signal then becomes

$$w(N) = x(N)H(N)$$

where $x(N)$ is the raw audio signal of the frame.

Since we only expect the audio signal to contain the voice of the speaker, we employ a voice activity threshold θ to determine which frames the user is speaking. We define voice activity α in the frame as

$$\alpha = \frac{\sum_{i=1}^{N} |w(i)|}{N}$$

If the windowed frame doesn't meet the threshold i.e. $\alpha \leq \theta$, it is considered a silent frame and is ignored. We find $\theta = 200$ to be a suitable threshold value.

D. Feature Extraction

For frames meeting the voice activity threshold described in the previous section, we extract the Mel Frequency Cepstral Coefficients (MFCC) and Spectral Subband Centroids (SSC) features using an available open source library Python Speech Features. We used its default number of features - 13 for MFCC and 26 for SSC.

E. Classification

In this paper, we use several classification algorithms including SVM, Decisions Trees, and Naive Bayes Classifier for the feature matching phase of our speaker recognition system. We then analyze and compare their performances in terms of accuracy and speed.

Support vector machines (SVM) was our first choice for classification since they are known to be among the most robust of classification algorithms [5]. SVM involve mapping input vectors to higher dimensional feature space where an optimal hyperplane can be computed by finding the maximal margin. In our speaker recognition system, we decompose the multi-class classification problem into a set of binary classification tasks to be solved by an array of binary SVMs, a technique described in [2]. Using binary SVM for K classes, we require one SVM classifier per speaker/class k such that each classifier is trained on positive examples for the class k and negative examples for all other $K-1$ classes (i.e. label is +1 if class is k, -1 otherwise). In testing an unknown sample, the SVM classifier producing the best output (or with the highest rank) is considered the winner, and this class label is subsequently assigned to that example. For comparison, we also employ a multi-class SVM classifier and test its performance against the binary SVM classification. LibSVM, a Python library for support vector machines [4] was used in the construction of the binary SVM and scikit-learn [15] in multi-class classification.

Decision trees were also chosen as a preference for classification due to their natural capability to handle multi-class classification problems with ease. The most widely known algorithms for building decision trees include ID3/C4.5/C5.0 [16], and Classification and Regression Trees (CART) [3]. Decision trees are constructed using features and threshold that yield the largest information gain at each node[16], [2], [3]. New data is tested by starting at the root node and following down to the leaf nodes, and at internal nodes a decision is performed based on a certain extracted feature. The overall objective of constructing a decision tree is to produce a good generalization of the data. In our speaker recognition system, we constructed decision trees with the aid of scikit-learn [15], an open source machine learning tool in Python, which uses an optimized version of the CART algorithm.

Lastly, we used the Gaussian Naive Bayes classifier which computes the likelihood of features using the following formula [15].

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{\left(-\frac{(x_i-\mu_y)}{2\pi\sigma_y^2}\right)}$$

where μ_y and σ_y are estimated using maximum likelihood. Similar to decision trees, the Naive Bayes classifier was employed with the aid of the open source machine learning tool in Python, scikit-learn [15].

IV. Results

In measuring the performance of the system, we consider two metrics. The primary metric is the rank. Obtaining a rank 1 means the system is able to correctly identify the speaker. In case this is not achieved ($rank > 1$), it still gives us an idea how near (or far) the system is able to recognize the speaker. The nearer the average rank is from 1 the higher its performance. The secondary metric is the Score or Accuracy. This is defined as the ratio of the number of correctly identified frames for the speaker and the number of the total frames meeting the voice activity threshold of an audio clip. Of course we want this metric to be as high as possible but obtaining a low score doesn't necessarily mean poor overall system performance. If the score of the speaker is the highest among all the other speakers in the database despite having a low score, it is still considered a correct classification.

A. Features

Initially the system was built to use only the Mel Frequency Cepstral Coefficients (MFCC). The system was performing very well always ranking the Speakers at number 1. This is until it was tested inside a room with other people in it and one of the test subject's training clip had audible background noise of other people. This prompted us to find a way to minimize the effect of background noise and we went with adding the Spectral Subband Centroids (SSC) feature which is said to be robust when it comes to noise distortion. Table III shows the performance of the system using MFCC only, SSC only and the combination of the two (MFCC + SSC) of 7 speakers.

TABLE III. TRAINING AND TEST SET PERFORMANCE USING MEL FREQUENCY CEPSTRAL COEFFICIENTS (MFCC) AND SPECTRAL SUBBAND CENTROIDS (SSC) FEATURES

	MFCC		SSC		MFCC + SSC	
	Rank	Score	Rank	Score	Rank	Score
Training Set						
Speaker 1	1	89.93%	1	99.71%	1	100%
Speaker 2	1	94.33%	1	99.89%	1	100%
Speaker 3	1	90.34%	1	99.78%	1	100%
Speaker 4	1	93.00%	1	99.79%	1	100%
Speaker 5	1	91.20%	1	99.60%	1	100%
Speaker 6*	1	85.59%	1	99.46%	1	100%
Speaker 7	1	92.14%	1	99.79%	1	100%
Average	**1**	**90.93%**	**1**	**99.72%**	**1**	**100%**
Test Set						
Speaker 1	1	70.34%	1	67.84%	1	73.09%
Speaker 2	1	68.44%	1	64.29%	1	62.54%
Speaker 3	1	61.44%	1	60.36%	1	62.32%
Speaker 4	1	66.31%	1	59.59%	1	67.01%
Speaker 5	1	79.68%	1	68.96%	1	73.49%
Speaker 6*	5	24.26%	1**	51.31%	1**	43.87%
Speaker 7	1	84.03%	1	74.36%	1	79.96%
Average	**1.57**	**64.93%**	**1**	**63.82%**	**1**	**66.04%**

* had noisy background

** near miss. Rank 2 within 5% distance

For the training set, MFCC scored high at around 90% average and the SSC feature scored very high at 99%. Combination of the two allowed the classifier to completely memorize the training data, categorizing all of the patterns correctly.

For the test set, it can be seen that MFCC generally outperforms SSC in terms of score with the exception of Speaker 6 who has audible background noise in his training audio clip. MFCC ranked and scored poorly with Speaker 6 but SSC was able to rank the Speaker at number 1. MFCC with SCC outperformed both individual MFCC and SSC in terms of average score and was able to rank all the Speakers at number 1. We conclude the combination of the two features provided a synergic effect on system performance.

B. Classifier

We've also compared performance of having a binary SVM classifier for every speaker setup against some multi-class supervised learning algorithms for validation. At this point, we decided to use MFCC + SSC and add 2 more speakers in the system. Table IV compares the total training time and classification performance of Binary SVM per speaker, One vs Rest Multiclass SVM, Decision Tree and Naive Bayes Classifier.

Although our chosen classifier architecture took the longest time to train, it outperformed the other three classifiers significantly in terms of average score. All classifiers performed well in ranking. Only Naive Bayes Classifier missed recognizing 1 speaker which is the one with a lot of background noise. The other three correctly classified all the speakers in both training and test set. This shows that our chosen set of features, Mel Frequency Cepstral Coefficients (MFCC) with Spectral Subband Centroids (SSC) is strong.

V. CONCLUSION

This paper presents a successful implementation of an automatic speaker recognition system using features MFCC and SCC trained on binary SVM, multi-class SVM, decision trees, and Naive Bayes classifier. Overall, all the classifiers worked well, ranking the correct speaker as number 1 for all test samples except for the Naive Bayes classifier which misclassifies speaker 6. We believe that this misclassification can be attributed to the noisy quality of the audio signal of this particular speaker. Moreover, our findings show that the binary SVM achieved the highest average score (or accuracy) of 75.39%. Results also show that the combination of MFCC and SSC yields the best performance compared to using MFCC or SSC alone.

Future work includes using an audio noise reduction system for better quality audio recordings. In the real world setting, capturing background noises is inevitable; thus noise reduction methods as described in [8], [11] will help reduce instances of misclassification (such that in the case of speaker 6) and will lead to an overall better classification performance.

ACKNOWLEDGMENT

The authors would like to thank Dr. Prospero Naval for allowing us to work on this topic. This is done as a Mini Project Requirement for CS 280 Intelligent Systems course.

REFERENCES

[1] Mel Frequency Cepstral Coefficents, 2015. Last accessed 4 November 2015.

[2] Mohamed Aly. Survey on multiclass classification methods. 2005.

[3] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

TABLE IV. TRAINING TIME AND PERFORMANCE OF DIFFERENT CLASSIFIERS USING MEL FREQUENCY CEPSTRAL COEFFICIENTS (MFCC) WITH SPECTRAL SUBBAND CENTROIDS (SSC) FEATURES

Training Time	Binary SVM per Speaker 2-24 seconds per SVM		One vs Rest Multiclass SVM 12 seconds		CART Decision Tree 4 seconds		Naive Bayes Classifier $<$ 1 second	
	Rank	Score	Rank	Score	Rank	Score	Rank	Score
Training Set								
Speaker 1	1	100%	1	51.93%	1	100%	1	41.74%
Speaker 2	1	100%	1	46.56%	1	100%	1	71.02%
Speaker 3	1	100%	1	55.57%	1	100%	1	49.78%
Speaker 4	1	100%	1	36.08%	1	100%	1	42.13%
Speaker 5	1	100%	1	53.30%	1	100%	1	26.56%
Speaker 6*	1	100%	1	59.72%	1	100%	1**	29.83%
Speaker 7	1	100%	1	51.01%	1	100%	1	38.49%
Speaker 8	1	100%	1	86.84%	1	100%	1	72.75%
Speaker 9	1	100%	1	83.82%	1	100%	1	91.73%
Average	**1**	**100%**	**1**	**58.31%**	**1**	**100%**	**1**	**51.56%**
Test Set								
Speaker 1	1	77.31%	1	52.38%	1	43.63%	1	40.69%
Speaker 2	1	68.71%	1	48.65%	1	35.75%	1	75.79%
Speaker 3	1	68.48%	1	40.55%	1	34.63%	1	33.82%
Speaker 4	1	70.45%	1	33.48%	1	31.52%	1	40.58%
Speaker 5	1	76.81%	1	51.67%	1	45.27%	1**	22.93%
Speaker 6*	1**	49.14%	1**	31.44%	1**	26.20%	3	7.06%
Speaker 7	1	82.16%	1	60.21%	1	45.71%	1	37.66%
Speaker 8	1	90.13%	1	75.55%	1	76.51%	1	60.66%
Speaker 9	1	95.33%	1	80.59%	1	75.23%	1	92.77%
Average	**1**	**75.39%**	**1**	**52.72%**	**1**	**46.05%**	**1.22**	**45.77%**

* had noisy background

** near miss. Rank 2 within 5% distance

[4] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[5] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[6] Steven B Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357–366, 1980.

[7] Minh Do. DSP Mini-Project: An Automatic Speaker Recognition System. Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign.

[8] Ray Dolby. An audio noise reduction system. *Journal of the Audio Engineering Society*, 15(4):383–388, 1967.

[9] Kevin R Farrell, Richard J Mammone, and Khaled T Assaleh. Speaker recognition using neural networks and conventional classifiers. *Speech and Audio Processing IEEE Transactions on*, 2(1):194–205, 1994.

[10] Bhanuprathap Kari and S Muthulakshmi. Real Time Implementation of Speaker Recognition System with MFCC and Neural Networks on FPGA. *Indian Journal of Science and Technology*, 8(19), 2015.

[11] Klaus Linhard. Noise-reduction method for noise-affected voice channels, March 21 1995. US Patent 5,400,409.

[12] A Milton, S Sharmy Roy, and S Selvi. Svm scheme for speech emotion recognition using mfcc feature. *International Journal of Computer Applications*, 69(9):34–39, 2013.

[13] Geeta Nijhawan and MK Soni. Speaker recognition using support vector machine. *International Journal of Computer Applications*, 87(2), 2014.

[14] Kuldip K Paliwal. Spectral subband centroid features for speech recognition. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 2, pages 617–620. IEEE, 1998.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[16] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[17] Vibha Tiwari. MFCC and its applications in speaker recognition. *International Journal on Emerging Technologies*, 2010.

[18] B Yegnanarayana, K Sharat Reddy, and S Prahallad Kishore. Source and system features for speaker recognition using aann models. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 1, pages 409–412. IEEE, 2001.