Text and Language Independent Speaker Identification By Using Short-Time Low Quality Signals

Maurizio Bocca*, Reino Virrankoski**, Heikki Koivo*

* Control Engineering Group
Faculty of Electronics, Communications and Automation
Helsinki University of Technology (TKK)
P.O.Box 5500, FI-02015 TKK, Finland
Tel. +358-9-451-5215, Fax +358-9-451-5208
{maurizio.bocca, heikki.koivo}@tkk.fi
** Telecommunication Engineering Group
Department of Computer Science
University of Vaasa
P.O.Box 700, FI-65101 Vaasa, Finland
Tel. +358-6-324-8694, Fax. +358-6-324-8467
reino.virrankoski@uwasa.fi

Abstract—*Several speaker identification applications that exploit voice signals recorded by using wireless networks of small, low-power acoustic sensors are becoming feasible. However, the acoustic signals provided by these devices have typically lower signal-to-noise ratio compared to wired microphone systems. In this paper, we present a text and language independent speaker identification algorithm based on a cepstral speech parameterization method. We analyze the robustness of the algorithm when the quality of the recorded voice signals is decreased. We also investigate how the number of cepstral coefficients considered in the extracted feature vector, and the resolution of the Discrete Fourier Transform affect the algorithm performance. To make the application as close to real-time as possible, we propose a light-weight classification technique based on a simple –yet effective– similarity measure.*

1. INTRODUCTION

It is nowadays possible to supply several personal items such as mobile phones, laptops, magnetic keys, electronic wallets, or guns, with voice sensing capability by using miniaturized acoustic sensors. By exploiting the uniqueness of the human voice, the access to such personal items can be limited only to their owners. Furthermore, in high-security applications, speaker identification can be part of the person biometric detection.

If we target to create a model of the ongoing situation inside an unknown building, a wireless network of nodes equipped with acoustic sensors can provide useful information, e.g. in military, police, and rescue operations. The acoustic signals provided by the network can be exploited in many ways, including speaker identification. The voice signals collected by small and unnoticeable acoustic sensors can be matched against already existing databases to detect the presence of those potentially dangerous individuals who have already been classified by the authorities. The speaker identification algorithm must also be able to point out if the person whose voice has just been recorded is not already present in the database. This would allow the authorities expanding the number of records included in their database for possible future critical situations.

The above mentioned indoor situation modeling system must be rapidly deployable to an unknown building interior, and must also operate in real-time. This forces us to minimize the delays caused by communications and computation. To fulfill these strict real-time requirements, we ignore methods that are computation intensive or require a priori information about the features of the environment. Instead, we propose a light-weight algorithm based on Mel-Frequency Cepstral Coefficients (MFCCs).

However, in wireless sensor networks (WSN), the applicable sampling frequency as well as the length of the sampling period is strictly limited by the scarce resources, in terms of computational power and memory size, respectively, of the sensor nodes. In this case, a speaker identification algorithm has to operate with noisy and short-time signals. Therefore, an important question concerns the minimum requirements for the quality of the recorded signals to perform the speaker identification task with a significant accuracy.

In this paper, we present a computationally light-weight speaker identification algorithm. Next, we analyze how its performance is affected by the quality of the recorded voice signals, in terms of applied sampling frequency and length of the sampling period. The algorithm is based on a frequency-plane analysis by using MFCCs. We also investigate how the number of considered mel-cepstral coefficients, and the number of bins used in the Discrete Fourier Transform (DFT) affect the algorithm performance. The number of available MFCCs is upper limited by the number of bins used in the DFT. Finally, we introduce a light-weight –yet effective– threshold-based method to determine if the voice under investigations does not refer to a person present in the database. We study how the applied value of the threshold affects the overall algorithm performance.

The paper is organized as follows. In the next section, we discuss the related work. Section 3 describes the proposed speaker identification algorithm, while simulation setup and results are presented in section 4. Conclusions and directions for future work are given in section 5.

2. Related Work

Different types of features, such as fingerprints, face traits, iris, and voice, have been used in biometric identification systems. Speaker identification algorithms are composed of two parts: the first extracts one or more feature vectors from the voice signal, while the second computes some similarity measure between the feature vector extracted from the signal under investigations and the ones stored in the database. The decision about the identification is based on the computed similarity [1] [2] [3].

An optimal characterizing feature must have maximal inter-speaker (signals of different individuals) and minimal intra-speaker (signals of the same person) variation. Also, it must be robust against voice disguise and mimicry, and against distortion and noise in the signal. The variability of the channel and of the environment is one of the most important factors affecting the accuracy of speaker identification algorithms. Several techniques, such as feature warping [4] and feature mapping [5], have been proposed to contrast it.

MFCCs have been extensively used in speech recognition, speaker identification and music-related applications. Seddik et al. [6] fed a neural network classifier with the MFCCs extracted from the speaker phonemes. A method to reduce the training time of the neural network is presented in [7]. In [8], MFCCs are used for the identification of singers: the singing introduces much larger variability compared to the normal speech, and it also includes much higher frequency components. MFCCs are also used by Eronen and Klapuri [9] in a musical instrument recognition application. In [10], Eronen analyzes and compares the effectiveness of several types of features to recognize different musical instruments. The best results are obtained with two sets of MFCCs.

Gaussian mixtures models (GMMs) [11] have been the state-of-the-art text independent speaker identification algorithm for many years. Support Vector Machines (SVMs) have also been used in speaker identification applications [12].

We introduce a light-weight speaker identification algorithm and evaluate how the quality of the recorded signals affects its accuracy. The feature vector characterizing the speaker is composed by the MFCCs and by their first and second order temporal derivatives. We analyze the effect of the number of considered cepstral coefficients and of the resolution of the DFT. Our results define the minimum requirements for the wireless acoustic sensors to collect voice signals that enable a successful identification.

3. Cepstral Parameterization Process

The applied speaker parameterization method is based on cepstral analysis as described in [1] [3]. In (7), we propose a light-weight method to separate the MFCCs vectors related to speech portions of the signal from the ones corresponding to silence or background noise.

A speech signal of N samples is first collected to vector $\boldsymbol{x}$ = [x(1),..., x(N)]. The high frequencies of the spectrum, which are reduced by the human speech production process, are enhanced by applying a filter to each element $x(i)$ of $\boldsymbol{x}$:

$$x_p\left(i\right) = x\left(i\right) - \alpha x\left(i-1\right), \quad i = 2,\ldots,N. \qquad (1)$$

The enhanced speech signal vector is called $\boldsymbol{x_p}$. The pre-defined parameter α usually belongs to range [0.95*,* 0.98] [3]. The signal is then windowed with a Hamming window of $L_w = t_w.f_s$ points, where t_w is the time length of the window (30 msec), and f_s is the sampling frequency of the signal. The shift between two consecutive windows is set to 2/3 of the window length.

The DFT is applied to each window of the signal. The results are then collected to matrix **T**. Each column of **T** contains N_{bins} elements, where N_{bins} is the number of bins applied in the DFT. Since this transform provides a symmetric spectrum, only the first half of each column of **T** is preserved. We get a matrix **F**, which contains only the first $N_{bins}/2$ rows of **T**.

The power spectrum, which represents the portion of the signal power included within given frequency bins, is computed by squaring the norm of each element in **F**:

$$\mathbf{P_w} = \left[\left|\mathrm{F}\left(i,j\right)\right|^2\right] \quad i = 1,\ldots,\frac{N_{bins}}{2} \quad j = 1,\ldots N_w. \qquad (2)$$

The frequencies located in the range of human speech are further on enhanced by multiplying the power spectrum matrix $\mathbf{P_w}$ by a filterbank matrix $\mathbf{B_f}$. Thus, we get a smoothened power spectrum matrix $\mathbf{P_s} = \mathbf{P_w}\,\mathbf{B_f}$.

$\mathbf{B_f}$ represents a filterbank of triangular filters whose central frequencies are located at regular intervals in the so-called mel-scale. The conversion from the mel-scale to the normal frequency one is done according to [13]:

$$f_{Hz} = 700\left(10^{\frac{Fmelscale}{2595}} - 1\right). \tag{3}$$

The mel-scale filterbank reduces the random variation in the high-frequency region of the spectrum by progressively increasing the bandwidth of the mel-filters.

After having transformed $\mathbf{P_s}$ into decibels ($\mathbf{P_{db}}$), the MFCCs are computed by applying the Discrete Cosine Transform (DCT) to each column vector in $\mathbf{P_{db}}$. The main advantage of this transform is that it converts statistically dependent spectral coefficients into statistically independent cepstral coefficients [14] [15] [16]. The elements of the mel-cepstral matrix $\mathbf{C_p}$ are calculated as:

$$\mathbf{C_p}(k,l) = a(k)\sum_{i=1}^{\frac{N_{bins}}{2}} \mathrm{P_{db}}(i,l)\cos\left(\frac{\pi(2i-1)(k-1)}{N_{bins}}\right) \tag{4}$$

where $1 \le k \le N_{cep}$, $1 \le l \le N_w$, and

$$a(k) = \begin{cases} \sqrt{\dfrac{N_{bins}}{2}} & ,k = 1 \\ \sqrt{\dfrac{4}{N_{bins}}} & ,2 \le k \le N_{cep} \le \dfrac{N_{bins}}{2} \end{cases} \tag{5}$$

In (4), N_{cep} is the number of cepstral coefficients that are considered. The number of elements of each column of $\mathbf{P_{db}}$, $N_{bins}/2$, represents the upper limit for the number of available MFCCs ($N_{cep} \le N_{bins}/2$).

The first cepstral coefficient of each window is ignored since it represents only the overall average energy contained in the spectrum. The rest of the coefficients are centered by subtracting the mean of the respective mel-cepstral vector. We get the centered mel-cepstral matrix $\mathbf{C}$. The lowest and highest order coefficients are de-emphasized by multiplying each column of $\mathbf{C}$ by a smoothening vector $\mathbf{M}$. By doing so, we get a smoothened mel-cepstral matrix $\mathbf{C_s}$. The elements of $\mathbf{M}$ are computed according to:

$$M(i) = 1 + \frac{N_{cep}-1}{2}\sin\left(\frac{\pi i}{N_{cep}-1}\right), \tag{6}$$

where $i = 1,\ldots, N_{cep} - 1$ [17].

We compute a normalized average vector of $\mathbf{C_s}$, such that each value $C_N(i)$ in the vector $\mathbf{C_N} = [C_N(1) \ \ldots \ C_N(N_w)]$ is the mean of the respective column in $\mathbf{C_s}$, normalized to range [0,1]. We are able to separate the windowed mel-cepstral vectors related to speech portions of the signal in $\mathbf{C_s}$ from the ones corresponding to silence or background noise by using the overall mean of $\mathbf{C_N}$ as a criterion. Thus, the matrix $\mathbf{C_{sp}}$, containing only the useful mel-cepstral vectors, is:

$$\mathbf{C_{sp}} = \left[C_s(j) \mid C_N(j) \ge \mu(\mathbf{C_N})\right] \quad j = 1,\ldots,N_w, \tag{7}$$

where j denotes the jth mel-cepstral vector of matrix $\mathbf{C_s}$ and $\mu(\mathbf{C_N})$ is the overall average of $\mathbf{C_N}$.

The final mel-cepstral coefficients $\mathbf{C_{cep}}$ are computed by taking the row-wise average of $\mathbf{C_{sp}}$:

$$\mathbf{C_{cep}} = \begin{bmatrix} \mu\left\{C_{sp}(1,1), \ \ldots \ C_{sp}(1,n)\right\} \\ \vdots \\ \mu\left\{C_{sp}(N_{cep}-1,1), \ \ldots \ C_{sp}(N_{cep}-1,n)\right\} \end{bmatrix}, \tag{8}$$

where n (with $n \le N_w$) is the number of mel-cepstral vectors selected from $\mathbf{C_s}$ into $\mathbf{C_{sp}}$ according to (7).

The information carried by $\mathbf{C_{cep}}$ is extended to capture the dynamics of the speech by including the temporal first and second order derivatives of the smoothened mel-cepstral matrix $\mathbf{C_s}$. The elements included in the first order temporal derivative matrix $\mathbf{\Delta C_s}$ are computed as:

$$\Delta C_s(i,j) = \frac{\sum_{k=-\Theta}^{\Theta} k C_s(i,j+k)}{\sum_{k=-\Theta}^{\Theta} k^2}, \tag{9}$$

where $1 + \Theta \le j + k \le N_w - \Theta$ and $1 \le i \le N_{cep} - 1$. As in (9), the second order temporal derivative $\mathbf{\Delta\Delta C_s}$ is obtained by computing the first order temporal derivative of $\mathbf{\Delta C_s}$ [3] [18]. $\mathbf{\Delta C_{cep}}$ and $\mathbf{\Delta\Delta C_{cep}}$ are computed from the matrices $\mathbf{\Delta C_s}$ and $\mathbf{\Delta\Delta C_s}$ by following the same procedure as in (7)-(8).

Finally, the MFCCs and their first- and second order temporal derivatives are collected into the feature vector $\mathbf{F_s}$:

$$\mathbf{F}_s = \left[\mathbf{C_{cep}}^T \quad \mathbf{\Delta C_{cep}}^T \quad \mathbf{\Delta\Delta C_{cep}}^T\right]. \tag{10}$$

$\mathbf{F_s}$ has $3\cdot(Ncep - 1)$ elements and characterizes the speaker.

4. SIMULATIONS AND RESULTS

4.1. Setup

The simulations are performed in Matlab. Our self-collected database includes 15 languages and 60 individuals (45 men, 15 women), for a total of 190 signals, with length varying between 8 and 10 seconds. Each signal was recorded with a commercially available wired microphone (Labtec desk mic 534). To guarantee the text and language independency of the algorithm, each person was recorded for a minimum of two times while speaking freely, and possibly using different languages. The signals were recorded in different indoor environments (e.g. office rooms, corridors, halls): this fact introduces variability in the recorded level of background noise and in the space reverberation, conditions known as channel variability.

The whole database was divided into two parts: the first (15 languages, 45 individuals, 36 men, 9 women, 140 samples) was used to study the accuracy of the algorithm in assigning the correct identity to the sample under investigations. The second part of the database (10 languages, 15 individuals, 10 men, 5 women, 50 samples) was exploited to analyze the performance of the algorithm in determining if the signal under investigations did not refer to a person included in the database. In simulations, each signal was matched against all the other signals of the database. Given the presence of 2-4 signals related to the same person, we were able to estimate the accuracy of the algorithm.

As similarity measure between the extracted feature vectors (10) of the voice signals, we chose the Euclidean distance. In our simulations, this similarity measure differentiated the feature vectors better than others, such as the Manhattan and Chebyshev distance, or the Pearson correlation coefficient.

4.2. The Effect of N_{bins} and N_{cep}

In the first group of simulations, realized with the first part of our database, we set the length of the sampling period to 8 seconds and the sampling frequency to 8 kHz: these values represent the best available quality of the recorded voice signals. Next, we varied the number of bins (N_{bins}) used in the DFT from 128 to 2048, and the number of cepstral coefficients (N_{cep}) considered in the computation from 10 to 1024 (with $N_{cep} \leq N_{bins}/2$). The 78% maximum accuracy in the identification was reached with N_{bins} = 512 and N_{cep} = 100.

We observed that the accuracy of the algorithm is marginally affected by the value of N_{bins}, while N_{cep} plays a big role. As shown in Figure 1, for any value of N_{bins}, the best accuracy is obtained when N_{cep} = 100. The performance rapidly decreases when N_{cep} is further on reduced. In fact, the lower order MFCCs are heavily affected by the random spectral variations and slowly varying additive noise distortion. On the contrary, when N_{cep} is increased, the performance of the algorithm first slightly decreases, and then levels off. This is explained by the fact that the higher order MFCCs carry less informative content than the lower order ones, and they tend to overlearn the spectral features of the voice signal.

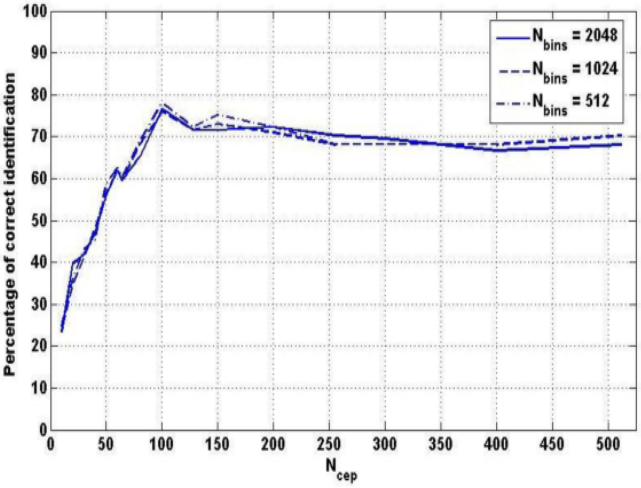

Figure 1 – The effect of N_{cep} on the algorithm accuracy (L = 8 seconds, f_s = 8 kHz).

4.3. The Effect of L and f_s

In the second group of simulations, we kept constant N_{bins} = 512 and N_{cep} = 100 (best configuration), and we varied the length of the sampling period (L) from 8 to 2 seconds, and the sampling frequency (f_s) from 8 kHz to 200 Hz. In this way we were able to test the robustness of the algorithm with short-time low quality signals, such as the ones typically recorded by wireless sensor nodes.

Figure 2 shows the results of this second set of simulations. The accuracy of the identification weakens linearly when f_s is reduced from 8 kHz to 2 kHz (for L = 8 seconds and f_s = 2 kHz, we still get 62.5%). When f_s is further on reduced, the accuracy of the identification rapidly collapses.

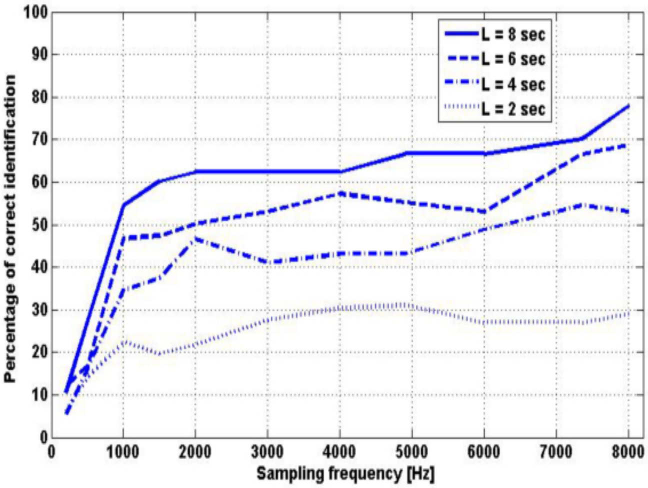

Figure 2 – The effect of f_s on the algorithm accuracy (N_{bins} = 512, N_{cep} = 100).

Finally, the algorithm performance weakens linearly when L is shortened from 8 to 2 seconds. With L = 6 seconds and f_s = 8 kHz, the accuracy is still 70%. The combined effect of the two parameters, f_s and L, is shown in Figure 3.

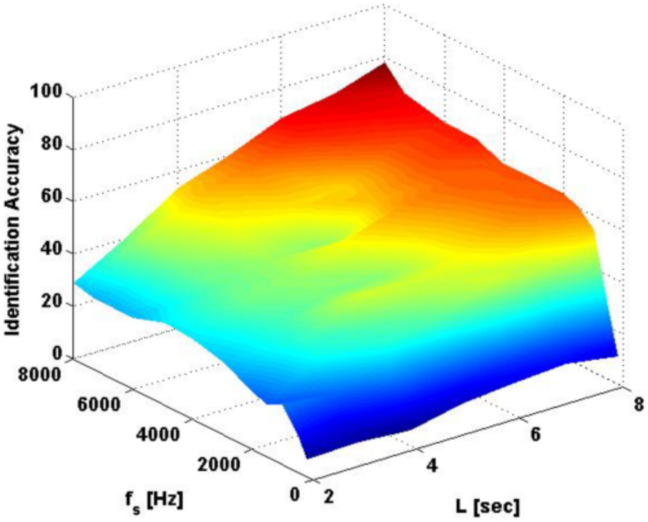

Figure 3 – The combined effect of f_s and L on the algorithm accuracy (N_{bins} = 512, N_{cep} = 100).

4.4. The Detection of Signals Related to Individuals not Included in the Database

We used the second part of the database to evaluate the capability of our speaker identification algorithm to detect those voice signals related to individuals not yet included in the database.

The light-weight method we propose is based on a threshold value (T_{hr}), calculated from the mean (μ_{cor}) and the standard deviation (σ_{cor}) of the distances of the correct identifications registered in the first two sets of simulations, adjusted with a pre-defined parameter (m):

$$T_{hr} = \mu_{cor} + m \cdot \sigma_{cor} \tag{11}$$

If the minimum distance found between the feature vector extracted from the signal under investigations and the ones extracted from the signals included in the first group of the database (known identities) is larger than the threshold, then the voice signal is classified as referring to a new person not yet included in the database.

In the end, we evaluated the accuracy of the algorithm both in correctly classifying the signals related to individuals already included in the database (P_{DB}), and in detecting those signals corresponding to individuals not yet included in the database (P_{NotDB}). The results are shown in Figure 4.

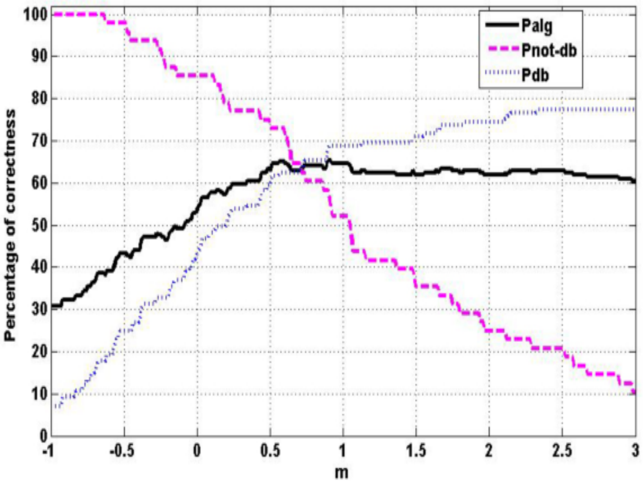

Figure 4 – The effect of m on the algorithm accuracy

The parameter m defines the value of the threshold. When T_{hr} is considerably smaller than μ_{cor} (negative values of m), the algorithm misclassifies as related to individuals not yet included in the database most of the voice signals (high P_{NotDB}, low P_{DB}). On the contrary, when T_{hr} is considerably larger than μ_{cor} (positive values of m), the algorithm is not able to recognize those signals corresponding to individuals not yet included in the database (low P_{NotDB}, high P_{DB}). The maximum overall accuracy (P_{ALG} = [65%,70%]) is reached when m ranges in the interval [0.5,1].

5. CONCLUSIONS AND FUTURE WORK

The proposed speaker identification algorithm is based on speech parameterization by using cepstral analysis. In the feature vector extraction process, we introduced in (7) a light-weight method to separate the portions of the signal related to speech from the ones corresponding to silence or background noise.

The algorithm was first tested to evaluate its accuracy in correctly classifying the voice signals included in a database of known identities. We discovered that with signals having a maximum length of 8 seconds and sampling frequency of 8 kHz, the best accuracy (78%) is obtained with N_{bins} = 512 and N_{cep} = 100. The use of more MFCCs in the computation rather weakens than improves the accuracy. This result does not improve consistently when the applied resolution of the DFT is increased.

With N_{bins} = 512 and N_{cep} = 100 (optimal configuration), the accuracy of the identification stays above 60% with signals 8 seconds long and with sampling frequency ranging from 1.5 to 8 kHz. With f_s between 7 and 8 kHz, the accuracy is in the range between 70 and 80%.

Then, we introduced in (11) a light-weight threshold-based method to determine if the voice under investigations does not refer to any person present in the database. We analyzed how the applied value of the threshold affects the overall algorithm accuracy, which remains in the range between 65 and 70%.

In future work, we will study how the algorithm accuracy can be improved by modifying the feature vector extraction process. In case of mixed signals (two or more individuals talking simultaneously), we will first separate the different components (individuals) by using a Blind Signal Separation technique based on Independent Component Analysis. Then, we will process the separated signals with the identification algorithm-

Finally, we will collect voice signals with wireless acoustic sensors, both in the single-speaker and multi-speaker case, and we will again evaluate the accuracy of our algorithm.

6. REFERENCES

[1] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-29, NO. 2, pp. 254–272, April 1981.

[2] S. Furui, "Comparison of speaker recognition methods using statistical features and dynamic features," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-29, NO. 3, pp. 343–350, June 1981.

[3] F. Bimbot, J-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrvovska-Delacretaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 430–451, 2004.

[4] J. Pelecanos, and S. Sridharan, "Feature Warping for Robust Speaker Verification", *in ODYSSEY-2001*, Crete, Greece, pp. 213-218, June 18-22, 2001.

[5] D. Reynolds, "Channel Robust Speaker Verification via Feature Mapping", *in Proc. ICASSP 2003*, Hong-Kong, pp. II-53-56, April 6-10, 2003.

[6] H. Seddik, A. Rahmouni, and M. Sayadi, "Text independent speaker recognition using the mel frequency cepstral coefficients and a neural network classifier," *in Proc. of ISCCSP 2004*, pp. 631–634, 2004.

[7] L. Rudasi and S. A. Zahorian, "Text independent talker identification using neural networks," *in Proc. of ICASSP*, vol. 1, pp. 389–392, 1991.

[8] A. Mesaros and J. Astola, "The mel-frequency cepstral coefficients in the context of singer identification," *in Proc. of ISMIR 2005*, London, UK, September 11-15, 2005.

[9] A. Eronen and A. Klapuri, "Musical instrument recognition using cepstral coefficients and temporal features," *in Proc. ICASSP 2000, Istanbul*, June 5-9, 2000.

[10] A. Eronen, "Comparison of features for musical instrument recognition," *in Proc. of WASPAA'01*, New Platz, NY, USA, pp. 19–22, October 21-24, 2001.

[11] D. Reynolds, T. Quatieri, R. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models", *Digital Signal Processing*, vol. 10, no. 1-3, 2000.

[12] V. Wan, and S. Renals, "Speaker Verification Using Sequence Discriminant Support Vector Machines", *in IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 2, March 2005.

[13] S. Stevens, J. Volkman, and E. B. Newman, "A scale for the measurement of the psychological magnitude of pitch," *Journal of the Acoustical Society of America*, vol. 8, pp. 185–190, 1937.

[14] B. P. Bogert, M. J. R. Healy, and J. W. Tukey, "The quefrency analysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking," *in the Proceedings of the Symposium on Time Series Analysis, New York, USA*, pp. 209–243, 1963.

[15] A. V. Oppenheim and R. W. Schafer, "Homomorphic analysis of speech," *IEEE Transactions on Audio and Electroacoustics*, vol. 16 no. 2, pp. 221–226, 1968.

[16] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*, Prentice Hall, Englewood Cliffs, NJ, USA, 1989.

[17] B. H. Juan, L. R. Rabiner, and J. G. Wilpon, "On the use of band-pass liftering in speech recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-35, no. 7, pp. 947–954, July 1987.

[18] L. Rabiner and B. H. Juang, *Fundamentals of speech recognition*, New Jersey, Prentice Hall, 1993.