15

Multidimensional Features Extraction Methods in Frequency Domain

Jesus Olivares-Mercado, Gualberto Aguilar-Torres, Karina Toscano-Medina, Gabriel Sanchez-Perez, Mariko Nakano-Miyatake and Hector Perez-Meana
National Polytechnic Institute
Mexico

1. Introduction

Pattern recognition have been a topic of active research during the 30 years, due to the high performance that these schemes presents, when they have been used in the solution of many practical problems in several fields of science, medicine and engineering. The efficiency of pattern recognition algorithms strongly depends in an accurate features extraction scheme that be able to represent the pattern under analysis using a number of parameters as small as possible, while keeping a large intra-pattern and very low inter-pattern similarities. These requirements have led to the development of several feature extraction methods, which can be divided in three groups. Feature extraction methods in time domain, spatial domain and frequency domain. In all cases the proposed feature extraction methods strongly depend of the specify applications. Thus the features extraction methods performing well in some applications, may do not perform well in others, for example the features extraction methods used for speech or speaker recognition are quite different to those used for fingerprints or face recognition. This chapter presents an analysis of some successful frequency domain feature extraction methods that have been proposed for applications involving audio, speech and images pattern recognition. Evaluation results are also provided to show the effectiveness of such feature extraction methods.

2. Feature extraction in speaker recognition

Speech is a widely used biometric feature for person recognition, where a person is recognized through his voice. To develop this kind of systems, several frequency domain methods have been proposed such as LPCepstral, described below, the LPCESPTRAL combined with dynamic features like the Cepstral Mean Normalization (CMN). In this section an analysis of this feature extraction methods is provided together with some evaluation results to show recognition performance.

2.1 Speaker recognition system

A general speaker recognition system, shown in Fig 1, consists mainly, of three stages: the feature extraction stage, where appropriate information is estimated in a suitable form and size, from the speech signal to obtain a good representation of the speaker features, the classifier stage, where the speaker models are adapted using the feature vectors, and the decision stage, where the recognition decision is taken. The SRS system under analysis firstly extracts the features vector from the speaker voice. To this end, firstly estimates the LPC-Cepstral (Ganchev et al., 2002) coefficients using only the voiced parts of speech signals. Then using the estimated LPC-Cepstral coefficients, the dynamic features are estimated to enhance the speaker features vector. Next the estimated dynamical features vector is feed to a Gaussian Mixture Model, GMM, which is used to obtain a representative model for each speaker. Taking into account that a SRS can be improved taking the voiced part of the speech signal because this contains the main information relative to the speaker identity (El-Solh, 2007; Markov & Nakagawa, 1999; Plumper et al., 1999). For this reason in this book the features vector will be derived from the LPC-cepstral extracted only from the voiced part speech signal.

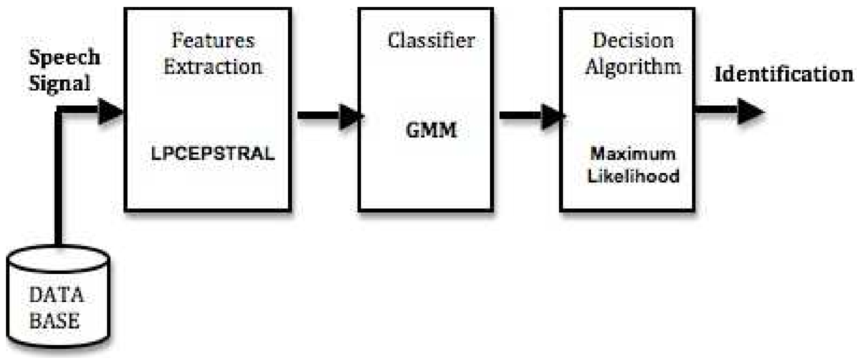


Fig. 1. General Speaker Recognition System

2.2 Feature vector extraction

A good performance of any pattern recognition system strongly depends on the extraction of a suitable feature vector that allow unambiguous representation of the pattern under analysis, with a number of parameters as small as possible. A simple way to estimate the speaker characteristics is the use of the linear prediction coefficients (LPC) of speech signal. The main reason about it is the fact that using these parameters can satisfactorily represent the structure of the vocal tract. However, it has been reported that better performance can be obtained if the LPC are combined with some frequency domain representation. One of these representations are the LPC features combined with the cepstral analysis, which allows to get a robust speaker characterization with low sensitivity to the distortion introduced in the signal transmitted through conventional communication channels (**?**).

The features vectors extracted from the whole speech signal provide a fairly good performance. However, when the LPCepstral coefficients are obtained from the LPC analysis, useful information of the speaker is still ignored or not taken in account, such as the pitch that is a specific feature of the individual speaker identity widely used to represent the glottal flow information.The performance of SRS can be seriously degraded when the SRS uses speech signal transmitted though some communication channel, such as a telephone one, due to the frequency response of the communication channel as well as the environment or the microphone characteristics. The LPCepstral coefficients have shown to be robust for reducing the problem of low speech quality. In most SRS even if they have a good performance using the same data training (closed test), their performance considerably degrades when the systems are used with different data set (open test), because the data for closed and open test for each speaker may have different acoustic conditions. Thus, channel normalization techniques may have to be used to reduce the speaker features distortion, keeping in such way a good recognition performance. Among the channel normalization technique we have the Cepstral Mean Normalization (CMN), which can provide a considerable environmental robustness at a negligible computational cost (Liu et al., 1993). On the other hand, the use of more than one speaker feature is proposed as well as combination of them to get a more robust feature vector. To improve the SRS performance the LPCepstral coefficients obtained using the CMN can be combined with the pitch information because the pitch is a very important speaker feature.

2.2.1 Features vector derived form LPCepstral

To estimate the LPCepstral coefficients, firstly the speech signal is divided in segments of 20 ms length with 50% overlap using a Hamming window. Next, the LPC coefficients are estimated using the Levinson algorithm such that the mean square value of prediction error given by.

$$E[e(n)] = E\left[S(n) - \sum_{i=1}^{P} a_i S(n-i)\right] \tag{1}$$

becomes a minimum, where $E[.]$ is the expectation operator, P is the predictor order and a_i is the $i-th$ linear prediction coefficient (LPC). Next, once the LPC vector has been estimated, the LPCepstral coefficients can be obtained in a recursive way as follows (Simancas-Acevedo et al., 2001).

$$c_n = -a_n + \frac{1}{n}\sum_{i=1}^{n}(n-i)a_i c_{n-i}, \qquad n > 0 \tag{2}$$

where c_n is the $n-th$ LPCepstral coefficient. Thus the SRS feature vector becomes

$$X_t = \left[c_{1,t}, c_{2,t}, c_{3,t}, \cdots, c_{d,t}\right] \tag{3}$$

where t denotes the frame number.

2.2.2 Features vector derived from LPCepstral of voiced segments

The pitch and voiced part detection plays a very important roll in the speaker recognition systems because the pitch value and the voiced segments of speech signals contain the most important information about the speaker identity. Then the feature vector could be extracted only from the voiced segments of speech signal (Rabiner & Gold, 1975). To this end, firstly the pitch period is detected using the autocorrelation method (Campbell, 1997) as follows: Initially, the speech signal is segmented in frames of 20 ms with 10 ms overlap using a Hamming window; next the center clipper method (Ganchev et al., 2002) is applied to the windowed frame to reduce the effect of the additive noise intrinsic within of the speech signal. Subsequently, the autocorrelation of the center clipped segment is obtained. Finally the pitch value is estimate as the distance between two consecutive positions in which the normalized autocorrelation sequence is larger than a given threshold, as proposed in

(Seung-Jin et al., 2007). Using the pitch information, the speech segment is then classified as voiced or unvoiced, because the pitch only appears in the voiced segments. Thus, if the pitch does not exist the speech segment is considered as a unvoiced segment (Campbell, 1997; **?**); and the pitch exists then speech segment is classified as a voiced speech segment. The Fig. 2 shows clearly the result of this procedure that proves that the detection of the voiced part is correctly done.

If only the voiced segments of speech signal are taken in account for features extraction, the original speech signal is transformed into a new speech signal containing only voiced parts, neglecting in such way the unvoiced and noisy silence parts, as shown in Fig. 3. This may improve the feature extraction because the unvoiced and silence parts provide non-useful information (Pool & Preez, 1999).

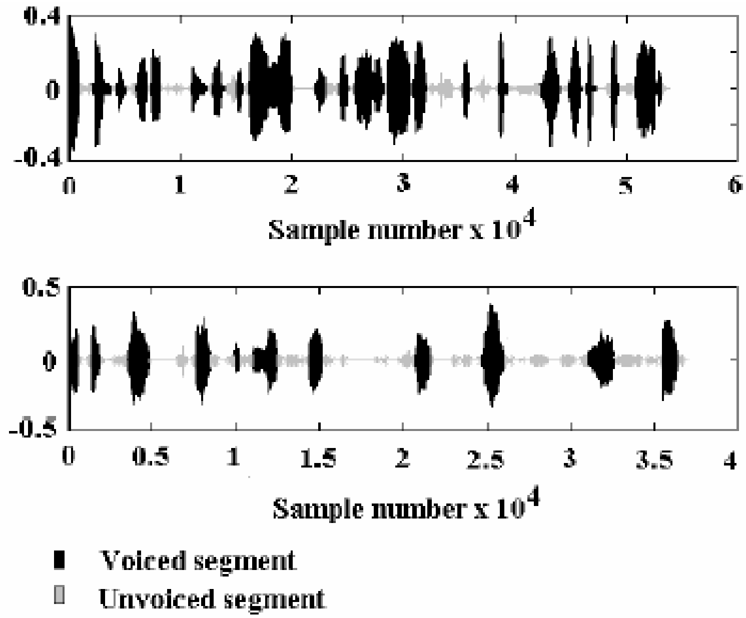


Fig. 2. Pitch period detection in two speech signals

The new signal, with only the voiced parts, has less samples number than the original one, but contains the essential parts required to estimate the principal features that identify the speaker and, as shown in the Fig. 3, the number of data is reduced, in many cases, as far as 50%. Once the new signal is constructed only with the voiced parts, it is divided in segments of 20 ms length with 50% overlap using a Hamming window. Next, the LPC coefficients are estimated using the Levinson algorithm as mentioned above, and then the LPCepstral coefficients are estimated using the eq. (2). Here two different features vectors can be estimated. The first one consists only of the LPCepstral of voiced segment.

$$X_t^v = \left[c_{1,t}^v, c_{2,t}^v, c_{3,t}^v, \cdots, c_{d,t}^v\right] \tag{4}$$

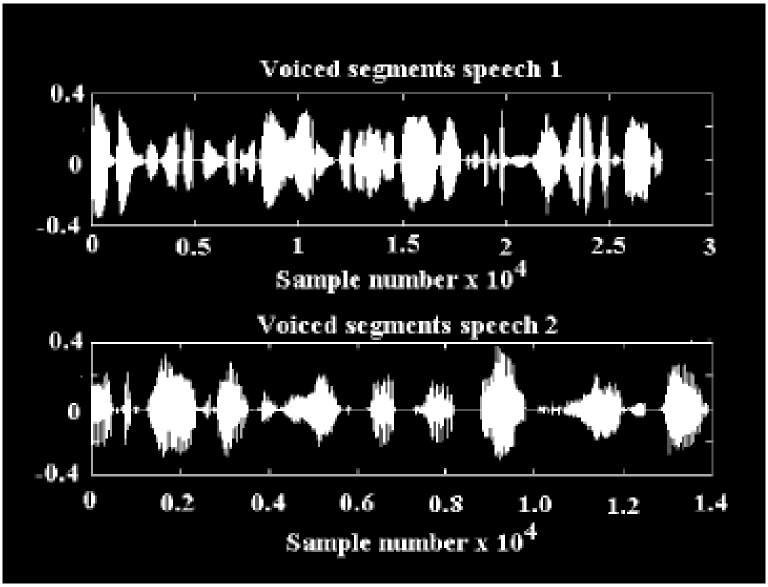


Fig. 3. Speech signal reduction taking only voiced part

and the second one consists of LPCepstral coefficients and pitch information

$$X_t^p = \left[c_{1,t}^v, c_{2,t}^v, c_{3,t}^v, \cdots, c_{d,t}^v, log_{10}F_{0,t}\right] \tag{5}$$

where $c_{n,t}^v$ is the $n - th$ LPCepstral coefficient and $F_{0.t}$ is the inverse of the pitch period at the block t. Here $log_{10}F_{0,t}$ is used instead of the pitch period, because the probability distribution of the $log_{10}F_{0,t}$ is close to the normal distribution.

2.2.3 Reinforcing and enhancing feature vectors

In long distance speaker recognition, the speech signal is transmitted through a channel communication and then is processed by the SRS. However, the speech signal suffers some distortion or variation due to the communication channel effects, noise environment, etc. Because these distortions are added to the principal components of the speech signal, it is necessary to remove the undesirable information before to proceed with the recognition process. To this end it would be convenient to enhance the estimated feature vector. Thus we can subtract the global average vector from all feature vector components. In this process, known as Cepstral Mean Normalization (CMN) (Murthy et al., 1999; Reynolds, 1995; Hardt & Fellbaum, 1997), it is assumed that the mean values of LPCepstral coefficients of clean speech is zero, so that if the mean value is subtracted from the feature vector components its mean value becomes zero. This avoids the distortion introduced by the additive noise when the signal passes through the communication channel. This technique is equivalent to a high-pass filtering of LPCepstral coefficients, because the CMN estimates the mean value of the LPCepstral vector coefficients and subtracts it from each component, as shown in eq. (6):

$$CMN_{n,t} = c_{n,t} - \frac{1}{T}\sum_{t=1}^{T} c_{n,t}, \qquad 1 \le n \le d \tag{6}$$

where $c_{n,t}$ is $n-th$ LPCepstral coefficient at block t and T is the total number of frames in which the speech signal was divided to extract the feature vectors. Fig. 4a and Fig. 4b show the effect produced in the feature vector when the Cepstral Mean Normalization (CMM) technique is applied to one of the LPCepstral coefficients extracted from the only the voiced part of the speech signal. In this situation the features vector becomes

$$CMN_t = \left[CMN_{1,t}, CMN_{2,t}, CMN_{2,t}, \cdots, CMN_{d,t}\right] \tag{7}$$

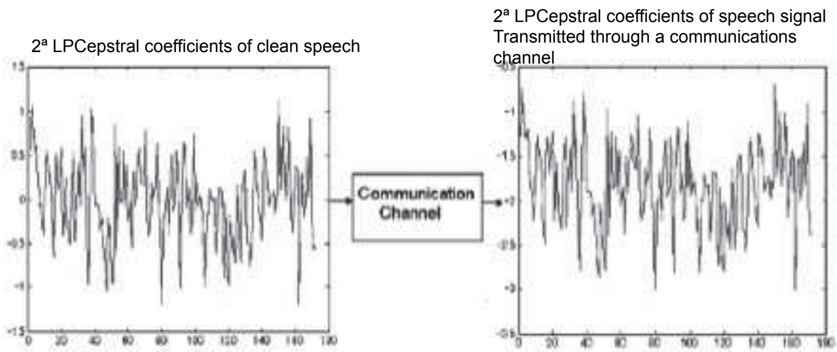


Fig. 4. The second LPC-Cepstral coefficient extracted from the speech signal

2.3 Results

Here present some results of a SRS that were evaluated using a feature vector with 16 LPCepstral coefficients extracted from only voiced part of the speech signals. The first evaluation of the baseline system is using these 16 LPCepstral coefficients extracted from only voiced part. The second evaluation is using the CMN technique to enhance the feature vector quality which has been affected by the channel communication and the environment noise. The third evaluation is using a combination of LPCepstral and pitch information and the forth evaluation is using the combination of LPCepstral coefficients and pitch information applying the CMN technique. All system evaluations, which are discussed in each respective section, are presented in the Table 1 and Table 2 where for system evaluation in close test, the same data for training was used and for system evaluation in open test, 2 different repetitions were used which were stored in different times, giving a total of 3658 phrases..

3. Feature extraction in face recognition

Face recognition has a large amount of biometric applications. The intra-person variations of the face image derive mainly from changes in facial expressions, illumination conditions as well as because the use of some accessories such as eyeglasses and muffler, etc. These

Feature Vector	LPCepstral From whole speech signal	LPCepstral From voiced part
Evaluation(Close test) 7147 phrases	96.61%	97.13%
Evaluation(Open test) 3658 phrases	82.34%	83.57%

Table 1. Results using whole and voiced part speech signal

Feature Vector	LPCepstral From voiced part	LPCepstral From voiced parts using CMN	LPCepstral From voiced and pitch information	LPCepstral From voiced parts using pitch and CMN
Closed test 6581 phrases	93.31%	80.72%	99.18%	97.29%
Open test 3282 phrases	76.97%	70.88%	80.29%	77.57%

Table 2. Results with different features vectors

variations make the face recognition a very difficult task. Most approaches to face recognition are in the image domain whereas we believe that there are more advantages to work directly in the spatial frequency domain. By going to the spatial frequency domain, image information gets distributed across frequencies providing tolerance to reasonable deviations and also providing graceful degradation against distortions to images (e.g., occlusions) in the spatial domain.

3.1 Face recognition system

This section provides a detailed description of a general face recognition system which consists in three stages each one. Figure 5 shows the block diagram of this system. Firstly in the pre-processing the input image is normalized, equalized or some other method for enhance a image. In the feature extraction the phase spectrum is extracted, after that, the Principal Components Analysis (PCA) (Kriegman et al., 1997; Hager et al., 1999) is applied to the phase spectrum to obtain a dominant feature of the faces. Next, the features in principal components space are fed into classifier and the image will be classified to the class given maximum likelihood.

3.2 Feature vector extraction

In the method proposed by Savvides (Savvides et al., 2004) were considering the combination of principal component analysis and phase spectrum of an image. Oppenheim (Oppenheim et al., 1980; Lim & Oppenheim, 1981) show that the image phase spectrum contains the most important information required for face image recognition, doing less relevant the use of the magnitude spectrum. His research also shows that getting only the phase spectrum of an image, can reconstruct the original image to a scale factor, therefore, information phase is the most important in the representation of a 2D signal in the Fourier domain. This is also demonstrated by a simple experiment, this is shown in Figure 6, in which the face of A (a) is reconstructed using the phase spectrum of A (c) and the magnitude spectrum of face B (f);

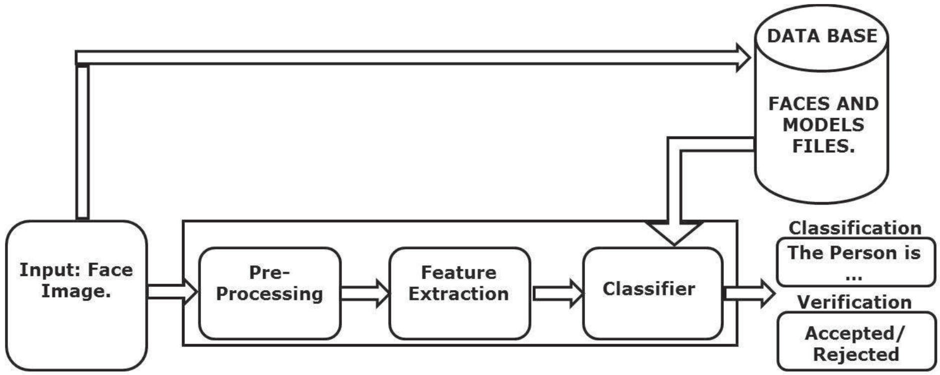


Fig. 5. General Face Recognition System

and the face of B (e) is reconstructed using the phase spectrum of B (g) and the magnitude spectrum of A (b). The reconstructed images, figures 2(d) and 2(h), show that the synthesized face image clearly resemble A (a) and B (e), respectively.

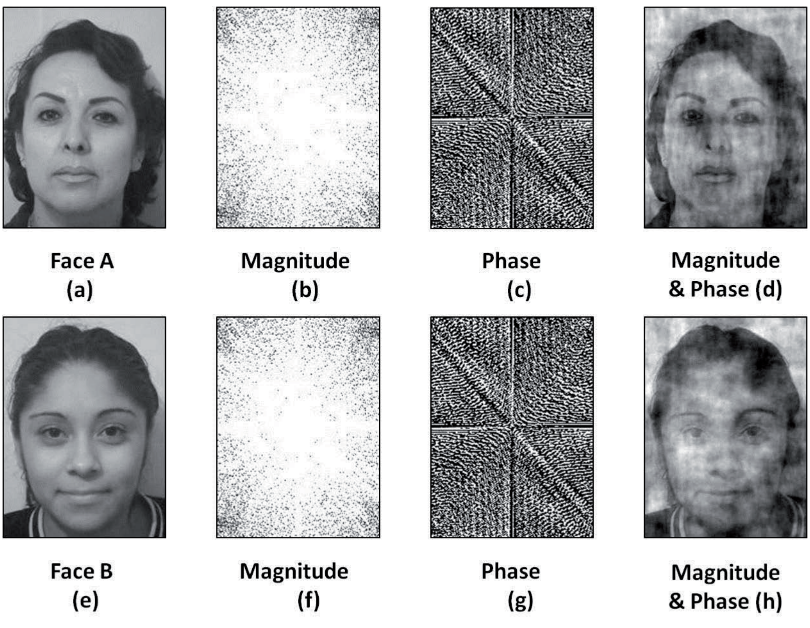


Fig. 6. Oppenheim Experiment

However, the performance of PCA in the frequency domain alone does not constitute any progress, this is because the eigenvectors obtained in the frequency domain are simply the Fourier transform of spatial domain. We begin this derivation by defining the standard 2-D discrete Fourier transform (DFT). Given an input 2-D discrete signal $x[m,n]$ of size $M \times N$

denote its Fourier transform as $X[k,l]$ whose Fourier transform pair is defined as follows:

$$x[m,n] \rightleftarrows X[k,l] \tag{8}$$

$$X[k,l] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] exp \frac{-i2\pi km}{M} exp \frac{-i2\pi ln}{N} \tag{9}$$

$$X[m,n] = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X[k,l] exp \frac{-i2\pi km}{M} exp \frac{-i2\pi ln}{N} \tag{10}$$

where $i = \sqrt{-1}$

If we have N training images then the covariance matrix of the Fourier transform is obtained by:

$$\widehat{\Sigma_f} = \frac{1}{N} \sum_{i=1}^{N} \{F(x_i - \widehat{\mu}\}\{F(x_i - \widehat{\mu}\}^T = F\widehat{\Sigma_s}F^{-1} \tag{11}$$

once the PCA was normalized, the eigenvectors w_f de $\widehat{\Sigma_f}$ are obtained by:

$$F\widehat{\Sigma_s}F^{-1}w_f = \lambda w_f \tag{12}$$

multiplying both sides by F^{-1} have:

$$\widehat{\Sigma_s}F^{-1}w_f = \lambda F_{-1}w_f \tag{13}$$

now formulate the space domain PCA and it shows that the space in the domain of the covariance matrix is:

$$C_s = \widehat{\Sigma_s} \tag{14}$$

$$C_s v_s = \lambda v_s \tag{15}$$

$$\widehat{\Sigma_s} v_s = \lambda v_s \tag{16}$$

where v_s is the space in the domain of the eigenvectors.

Comparing equation 13 with equation 16 we can see that is a relationship between space and frequency domain of the eigenvectors $(v_s and v_f)$ related by an inverse Fourier transform as follows:

$$v_s = F^{-1}v_f \tag{17}$$

3.3 Results

Here present some results, where the Face Recognition System was evaluated using the "AR Face Database", which has a total of 9,360 face images of 120 people (65 men and 55 women). This database includes 78 face images of each people with different illuminations, facial expression and partial occluded face images with sunglasses and scarf. To evaluate the performance of the proposed methods under several illumination and occlusion conditions, two different training sets are used. The first one consists of 1200 images, 10 images of each person, with different illumination and expression conditions; while the second set consists of

1200 face images, 10 per each person, with different illumination conditions, expressions and occlusions, which are the result of using sunglasses and scarf. The remaining images of the AR face database are used for testing. The table 3 show some results of face identification and table 4 some results of face verification.

	Training set 1	Training set 2
Eigenphases	80.63	96.28

Table 3. Results using Eigenphases for identification.

	False acceptance	False reject
Training set 1	0.5	19.07
Training set 2	0.5	3.80

Table 4. Results using Eigenphases for verification

4. Feature extraction in sound recognition

Sound recognition (Goldhor, 1993) has a large amount of application civil as well as military applications such as engine diagnostic, airplane or ship recognition, etc. This applications also depends on the above mentioned feature extraction methods. This section describes a sound recognition system using a frequency domain feature extraction method like LPCepstral.

4.1 Sound recognition system

Figure 7 shows the proposed system. This system consist of four sequential processes: first a common database of environmental sounds is obtained, after this a segmentation algorithm is applied to each token (file) of this database; third LPC-Cepstral features are extracted from each segmented file and the DFT is computed from these coefficients, finally the DFT magnitude is computed and a training strategy is adopted. The decision is taken at the final process and a recognition percentage is computed.

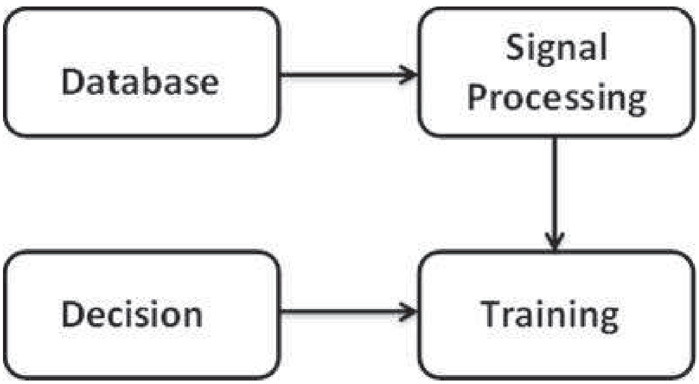


Fig. 7. General Sound Recognition System

4.2 Feature vector extraction

Figure 8 shows the applied processes in the signal analysis. With this signal analysis a high efficiency of feature extraction is obtained, this facilitates to the neural network the recognition process, this means that higher percentages of verification and identification can be obtained.

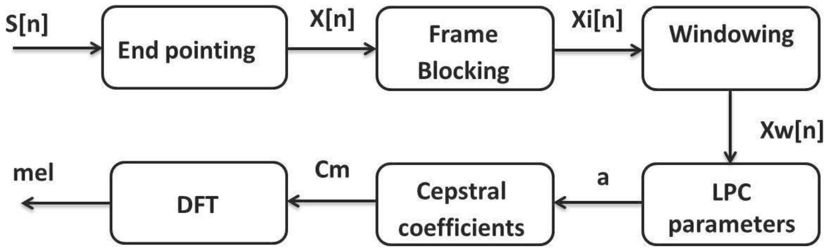


Fig. 8. Methodology used in the signal processing

4.2.1 End-pointing algorithm

In time domain, magnitude, energy, power, maximums and minimums can be computed from which, the energy is used. Once the energy was calculated, a reference is obtained, with this reference the signal can be limited. In the discrete case the energy is defined as:

$$E[n] = \sum_{n=-\infty}^{\infty} s^2[n] \tag{18}$$

Now, a gamma constant is defined, this constant indicates the number of samples taken from the signal. The following step is to make a relationship between the sound signal and the gamma constant:

$$E[n] = [(1-\gamma) * E_{n-1}] * [\gamma * y_n^2] \tag{19}$$

To each file stored in the database an end-pointing algorithm was applied. In order to limit the signal two thresholds of 20 and 10 the maximum energy must be defined, this corresponds to the percentage taken form the signal. This algorithm compares the thresholds with each sample of the energy until a sample is greater or equal to these thresholds, indicating the signal is beginning.

4.2.2 Frame blocking and windowing

The sound signal, $\widehat{S}[n]$, is blocked into frames of 240 samples that corresponds to 30 msec, in which voice is considered stationary (Kitamura & Hayahara, 1988), with adjacent frames being separated by 120 samples. The use of frames implies three parameters: frame size, frame increment and frame overlapping:

$$S_f = I_f + O_f \tag{20}$$

where I_f represents the frame beginning and O_f the frame overlapping.

To the sequence of analysis frames generated from each end-pointed file was applied a windowing algorithm, Hamming window was used:

$$\widehat{S}_w[n] = \widehat{S}[n]W[n] \tag{21}$$

where $0 < n < N-1$, N is the number of samples in the analysis frame and $W[n]$ is a Hamming window.

4.2.3 LPC and LPC-Cepstral parameters

In each window 17 LPC coefficients were calculated with Levinson-Durbin recursion. LPC-Cepstral coefficients can be derived directly from the set of LPC coefficients using the algorithm:

$$C[n] = -a[n] - \frac{1}{n} \sum_{k=1} n = 1kC[k]a[n-k] \tag{22}$$

where $n > 0$, $C_0 = a_0 = 1$, $k > p$ and $a[n]$ represents the linear prediction coefficients, estimated, using a linear filter, as it is shown in Figure 9 , where a given sound sample cam be approximated or predicted as a linear combination of its past p samples, as shown in eq. 22. The number of frames generated for each signal was of 64. The result in effect was that each signal was represented by a 17 by 64 array of the Cepstral coefficients, with the 64 rows representing time and the 17 columns representing frequency.

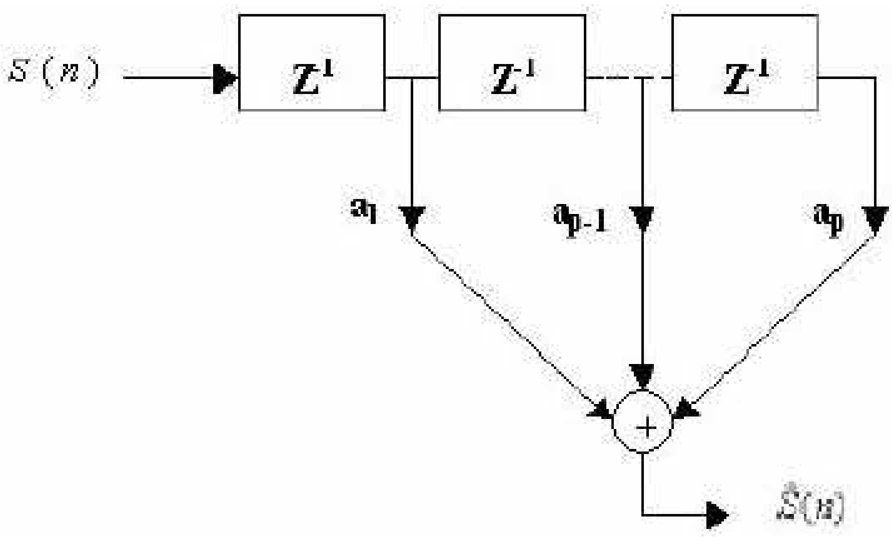


Fig. 9. Linear Prediction Filter used for the LPC-Cepstral Estimation

$$\widehat{S}(n) = \sum_{i=1}^{P} a(i)S(n-i) \tag{23}$$

where $S(n)$ is the estimated signal in time n, P is the filter order and a is the filter coefficients vector.

A 64-point DFT was then calculated for each column in the matrix and the first 32 points of this symmetrical transform retained. The resulting square matrix is a two dimensional Cepstral representation of the input signal. Each column corresponds to a particular spectral frequency, and ach row corresponds to a temporal frequency. The first column contains the DFT of the power envelope of the signal. The first row contains the DFT of the average signal spectrum. The first element of the first column contains the average signal power level. It Šs typical of a two-dimensional Cepstral representations of acoustic signals, and certainly for our signals, that this corner element is the largest component and the size of the components in the first row and first column are larger than the size of interior matrix components. After this the DFT magnitude for each column in the matrix is computed. The LPC-Cepstral coefficients, which are the Fourier transform representation of the spectrum, have been shown to be more robust for speech recognition than the LPC coefficients, in this case we applied this method. The Cepstral transforms were calculated for two variants of the spectrum within each frame: a linear variant and one in which the frequency scale was warped using mel frequency transformation.

4.3 Results

The used model is an artificial neural network backpropagation. The traditional error backpropagation algorithm is used. For each sound pattern, 50 sound files were used in the network training process. The sound samples are first normalized so that the average magnitude becomes zero and the standard deviation is one. Clusters, or classes, were formed by grouping the feature vectors for each type of sound. For the network training, the ideal number of hidden-layer neurons was chosen from the experimental work. The hope, of course, is that the samples of each sound will cluster together in that space and that cluster for different sounds will be rejected.

Two neural networks, per sound-source, were trained. Four stages (one per neural network) were necessary for the network training and each stage corresponds to each file stored in the database. 32 input-layer neurons were necessary for the neural network training, 10, 15 and 20 hidden-layer neurons were used in this neural network and the best results were obtained with 20 neurons; 1 output-layer neuron, per network, was necessary for verify the source-sounds and 4 output-layer neurons in the identification process.

5. Feature extraction in fingerprint recognition

Fingerprint Identification (Aguilar et al., 1994) is one of the most reliable and popular personal identification methods. The performance of minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the input fingerprint images quality. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. In such situations, the ridges can be easily detected and the minutiae can be accurately located in the image. However, in practice, because of skin conditions (e.g., wet or dry, cuts, and bruises), sensor noise, incorrect finger pressure; and inherently low-quality fingers (e.g., elderly people, manual workers), etc. a significant amount of fingerprints are poor quality images.

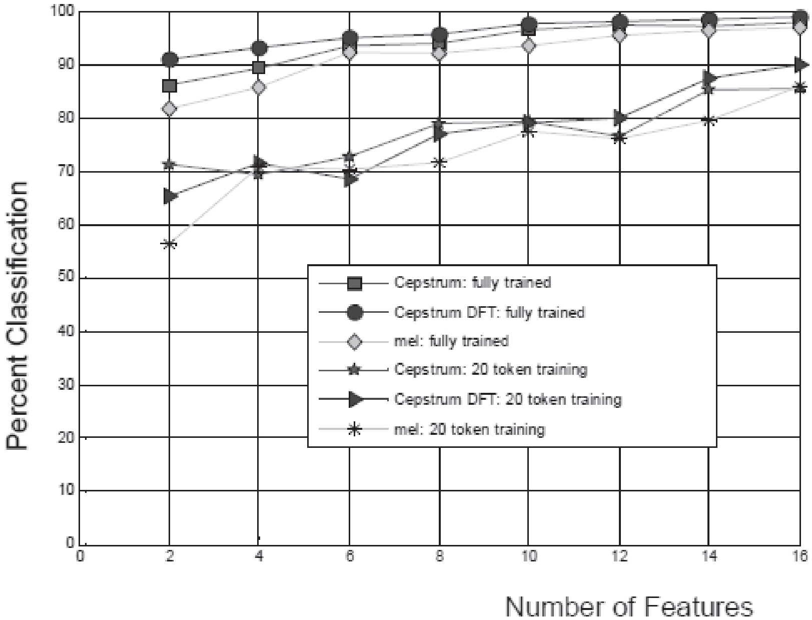


Fig. 10. Percent of correct classification of test.

5.1 Fingerprint recognition system

Figure 10 shows the main stages of a biometric system. As mentioned above, a good quality of the input image ensures that the recognition will be higher. Therefore, we can use Fourier to enhance the quality of input images of a biometric system.

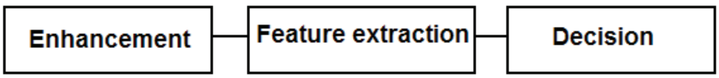


Fig. 11. General Fingerprint Recognition System

5.2 Feature vector extraction

A good fingerprint image will have high contrast and well-defined ridges and valleys, and a poor quality fingerprint will have low contrast and ridges and valleys poorly defined. In a biometric system we can have images with different qualities, as shown in Figure 11. Thus the goal of an enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and classify the unrecoverable regions as too hazy for further processing. Because of the regularity and continuity properties of the fingerprint images, the occluded and corrupted regions can be recovered using the contextual information from the surrounding neighborhood. This section describes a contextual filtering completely in the frequency domain, where each image is convolved with a precomputed filter whose size is equal to the image size. Therefore the algorithm does not use the full contextual information provided by the fingerprint image.

Fig. 12. Set of Fingerprints

An advantage of this approach is that, for its operation, it does not require the computation of intrinsic images; which has the effect of increasing the dominant spectral components while attenuating the weak ones. However, in order to preserve the phase, the enhancement also retains the original spectrum F(u,v). In this algorithm the fingerprint enhancement are transformed from spatial domain to frequency domain by Fourier transforming.

5.2.1 Fourier domain filtering

Sherlock (Sherlock et al., 1994) perform contextual filtering completely in the frequency domain, where each image is convolved with a precomputed filter whose size is equal to the image size. However, the algorithm assumes that the ridge frequency is constant through out the image in order to prevent having a large number of precomputed filters. Therefore the algorithm does not use the full contextual information provided by the fingerprint image. Watson (Watson, Candela & Grother) proposed another approach for performing image enhancement completely in the frequency domain, which is based on the root filtering technique. In this approach the image is divided into overlapping blocks, where in each block the enhanced image is obtained as:

$$I_{enh}(x,y) = FFT^{-1}\left[F(u,v)|F(u,v)|^{k}\right] \tag{24}$$

$$F(u,v) = FFT(I(x,y)) \tag{25}$$

where the k in formula (24) is an experimentally determined constant, which we choose equal to 0.45. Here, while a higher "k" improves the appearance of the ridges, filling up small holes in ridges, a too large value of "k" may result in false joining of ridges, such that a termination might become a bifurcation. Another advantage of this approach is that, for its operation, it does not require the computation of intrinsic images; which has the effect of increasing the dominant spectral components while attenuating the weak ones. However, in order to preserve the phase, the enhancement also retains the original spectrum $F(u,v)$. Figure 12 shows fingerprint images after of the FFT.

5.3 Results

In this section a fingerprint recognition algorithm using FFT was evaluated. The tests consisted of the recognition of 125 people. Table 5 shows the recognition results obtained using FFT. We show that it results in 11.3% improvement in recognition rate over a set of 1000

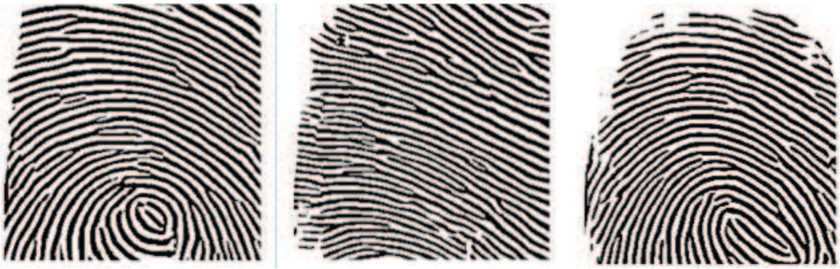


Fig. 13. Set of Fingerprints

images. True recognition means that the person was effectively recognized, the rest is divided into two: false recognition and without recognition. False recognition occurs when a person is confused and without recognition when the system does not deliver any possible identified person.

Total percentage	True recognition	False recognition	without recognition
Without FFT	81.8%	9.4%	8.8%
Using FFT	93.1%	3.8%	3.1%

Table 5. Test results made to 1000 images

The ROC curves before and after enhancement are as shown in the Figure 13.

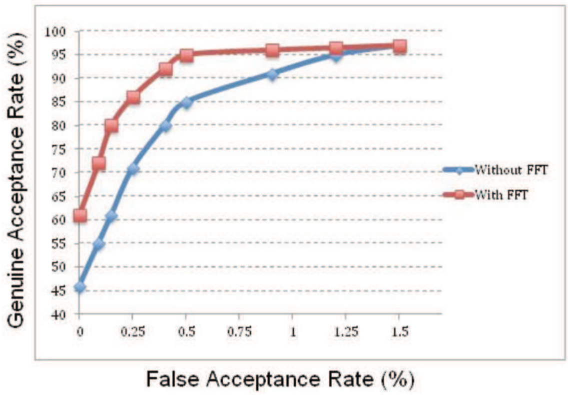


Fig. 14. Set of Fingerprints

6. References

Ganchev T., Tsopanoglou A., Fakotakis N., Kokkinakis G. (2002). Probabilistic Neural Networks Combined with GMMs For Speaker Recognition over Telephone Channels. 14-th International Conference On Digital Signal Processing (DSP 2002), Santorini, Greece, Volume II, 1081Ű1084.

El-Solh A., Cuhadar A., Goubran R. A., (2007). Evaluation of Speech Enhancement Techniques for Speaker Identification in Noisy Environments. Multimedia Workshops, 2007. ISMW '07. Ninth IEEE International Symposium on, page(s): 235 - 239 Print ISBN: 9780-7695-3084-0.

Markov K. P., Nakagawa S. (1999). Integrating Pitch and LPC-Residual Information with LPC-Cepstral for Text-independent Speaker Recognition. J. Acoustic Society of Japan (E), 20, 4, 281-291.

Plumper M. D., Quatieri T. F., Reynolds D. A. (1999). Modeling of the Glottal Flow Derivative Waveform with Application to Speaker Identification. IEEE Transactions on Speech and Audio Processing, vol. 7, Nř5.

Liu F., Stern R. M., Huang X., Acero A. (1993). Efficient Cepstral Normalization For Robust Speech recognition. Proceeding HLT '93 Proceedings of the workshop on Human Language Technology. ISBN:1-55860-324-7, Stroudsburg, PA, USA.

Simancas-Acevedo E., Kurematsu A., Nakano-Miyatake M., Perez-Meana H. (2001). Speaker Recognition Using Gaussian Mixtures Model. Lecture Notes in Computer Science, Bio- Inspired Applications of Connectionism, Springer Verlag, Berlin, 287Ű294.

Rabiner B., Gold B. (1975). Digital Processing of Speech Signals. Prentice Hall, Englewood Cliffd, NJ.

Campbell J. P. (1997). Speaker Recognition: A Tutorial. Proceedings of the IEEE, vol. 85, Nř9, 1437Ű1462.

Seung-Jin Jang, Seong-Hee Choi, Hyo-Min Kim, Hong-Shik Choi, Young-Ro Yoon, (2007). Evaluation of Performance of Several Established Pitch Detection Algorithms in Pathological Voices. Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE , page(s): 620 - 623, ISSN : 1557-170X, Print ISBN: 978-1-4244-0787-3 .

Pool J., du Preez J. A. (1999). HF Speaker Recognition. Thesis notes, Digital Signal Processing Group, Department of Electrical and Electronic Engineering, University of Stellenbosch, March 1999.

Murthy H. A., Beaufays F., Heck L. P., Weintraub M. (1999). Robust Text-Independent Speaker Identification over Telephone Channels. IEEE Transactions on Speech and Audio Processing, vol. 7, Nř5.

Reynolds D. A. (1995). Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. IEEE Transactions on Speech and Audio Processing, vol. 3, Nř1, 72Ű83.

Hardt D., Fellbaum K. (1997). Spectral Subtraction and Rasta Filtering in Text Dependent HMM-based Speaker Verification. Proc. of ICASSP, vol. 2, 867-870.

Kriegman D., Belhumeur P., Hespanha J. (1997). Eigenfaces vs fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), 711-720.

Hager J., Ekman P., Sejnowski T., Donato G., Bartlett M. (1999). Classifying facial actions. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, 974-989.

Savvides M., Kumar B.V.K.V., Khosla P.K. (2004). Eigenphases vs. eigen-faces. Pro- ceedings of the 17th International Conference on Pattern Recognition, Vol. 3, 810-813.

Oppenheim A. V., Hayes M. H., Lim J. S. (1980). Signal reconstruction from phase or magnitude. IEEE Trans. Acoust., Signal Processing, Vol. 28, pp. 672-680.

Lim J. S., Oppenheim A. V. (1981). The importance of phase in signals. Proc. IEEE, 69(5), 529-541.

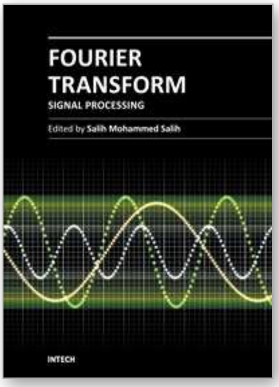
Goldhor, R.S., (1993), Recognition of Environmental Sounds, Proceedings of ICASSP, Vol. 1, 149-152.

Kitamura T., Hayahara, E. (1988), Word Recognition Using a Two-Dimensional Mel-Cepstrum in Noisy Enviroments, paper PPP6 presented at the 2nd Joint Meeting of the ASA and ASJ, Hawwaii.

Aguilar, G., Sanchez, G., Toscano, K., Salinas, M., Nakano, M., Perez, H., (2007), Fingerprint Recognition, Internet Monitoring and Protection, 2007. ICIMP 2007. Second International Conference on, page(s): 32, Print ISBN: 0-7695-2911-9 .

Sherlock B. G., Monro D. M., Millard K., (1994), Fingerprint enhancement by directional Fourier filtering, in: Visual Image Signal Processing, Vol. 141, pp. 87Ð94, 1994.

Watson C.I., Candela G.I., and Grother P.J., (1994), ÒComparison of FFT Fingerprint Filtering Methods for Neural Network Classification,Ó Tech. Report: NIST TR 5493, Sept. 1994.



Fourier Transform - Signal Processing
Edited by Dr Salih Salih

ISBN 978-953-51-0453-7
Hard cover, 354 pages
Publisher InTech
Published online 11, April, 2012
Published in print edition April, 2012

The field of signal processing has seen explosive growth during the past decades; almost all textbooks on signal processing have a section devoted to the Fourier transform theory. For this reason, this book focuses on the Fourier transform applications in signal processing techniques. The book chapters are related to DFT, FFT, OFDM, estimation techniques and the image processing techqniques. It is hoped that this book will provide the background, references and the incentive to encourage further research and results in this area as well as provide tools for practical applications. It provides an applications-oriented to signal processing written primarily for electrical engineers, communication engineers, signal processing engineers, mathematicians and graduate students will also find it useful as a reference for their research activities.