A New Nonlinear speaker parameterization algorithm for speaker identification

M. Chetouani, M. Faundez-Zanuy †, B. Gas, J.L. Zarader

Laboratoire des Instruments et Systèmes d’Ile-De-France
Université Paris VI, Paris, FRANCE
(†) Escola Universitària Politècnica de Mataró, Barcelona, ESPAGNE
mohamed.chetouani@lis.jussieu.fr faundez@eupmt.es
gas@ccr.jussieu.fr zarader@ccr.jussieu.fr

Abstract

In this paper we propose a new coding algorithm based on non-linear prediction: the Neural Predictive Coding model which is an extension of the classical LPC one. The features performances are estimated by two different methods: the Arithmetic-Harmonic Symmetry (AHS) and the Auto-Regressive Vectorial Models (ARVM). Two different methods are proposed for the coding method based on the Neural Predictive Coding (NPC): classical neural networks initialization and linear initialization. We applied these two parameters to speaker identification. The fist model obtained smaller rates. We show for the first model how it can be combined with the classical feature extractors (LPCC, MFCC, etc.) in order to improve the results of only one classical coding (MFCC provides 97.55% and MFCC+NPC 98.78%). For the linear initialization, we obtain 100% which is a great improvement. This study opens a new way towards different coding schemes that offer better accuracy on speaker recognition tasks.

1. Introduction

A key issue for implementing an accurate speaker recognition system is the set of acoustic features extracted from the speech signal. This set is required to convey as much speaker-dependent information as possible. The standard methodology to extract these features from the signal follows two trends: features are extracted through a filter bank processing or through linear prediction coding (LPC). Both the methods are to some extent linear procedures and are based on the underlying assumption that acoustic characteristics of human speech are mainly due to the vocal tract resonances, which form the basic spectral structure of the speech signal. However, human speech is a nonlinear phenomenon, which involves nonlinear biomechanical, aerodynamic, acoustic, and physiological factors, and LPC-derived parameters can only offer a sub-optimal description of the speech dynamics [1]. Therefore, in the last years there has been a growing interest for nonlinear models applied to speaker recognition applications.

In this paper we propose a new coding algorithm: The Neural Predictive Coding (NPC) model described in section 2. We propose an initialization method based on speech knowledge. Next, we present the database and the other coding methods. Then, we briefly introduce the methods used for speaker identification. Finally, we give some preliminary results.

This work has been supported by COST-277, FEDER & CICYT TIC-2003-08382-C05-02

2. Neural Predictive Coding

The Neural Predictive Coding (NPC) [2] model (cf. Fig. 1) is a non-linear extension of the well-known LPC encoder. Like in the LPC framework with the AR model, the vector code is estimated by prediction error minimization. The main difference relies in the fact that the model is non-linear and it is a connectionist model:

\[ \hat{y}_k = F(y_k) = \sum_j a_j \sigma(w^T y_k) \]  

(1)

Where \( F \) is the prediction function realized by the neural model. \( \hat{y}_k \) is the predicted sample and \( y_k \) the prediction context: \( y_k = [y_{k-\lambda}, y_{k-\lambda-1}, \ldots, y_{k-2\lambda}]^T \) and \( \lambda \) the length of the prediction window. \( w \) and \( a \) represent the first and the output layer weights. \( \sigma \) is a non-linear activation function, the sigmoid function in our case.

Figure 1: Neural Predictive Coding (NPC) architecture: a connectionist model used as a non-linear predictor.

The neural networks are known to have good approximation capabilities [3], and it is one of the reasons for their use in speech prediction. Multi-Layer Perceptron (MLP) [4], Radial Basis Functions [5], recurrent networks [4, 6] have been successfully applied to non-linear speech prediction. Non-linear prediction can be carried out by other models like Volterra filters [7], quadratic model [8], locally linear methods [9] or non-parametric methods [10].
The key idea is to use the NPC model as a non-linear auto-regressive model. As in the LPC framework for the predictor coefficients, the NPC weights are the vector code. It is well-known that the weights can be considered as a representation of the input vector. A drawback of this method is that non-linear models have no clear physical meaning [11]. The solution weights can be very different for the same minimum of the prediction error. In our approach, we impose constraints on weights.

2.1. Description

The NPC model is a Multi-Layer Perceptron (MLP) with one hidden layer. Only the output layer weights are used as coding vector instead of all the neural weights. For that we consider that the function $F$ realized by the model, under convergence assumptions, can be decomposed into two functions: $G_w$ (first layer weights) and $H_a$ (a output layer weights):

$$F_{w,a}(y_k) = H_a \circ G_w(y_k) \quad (2)$$

With $y_k = H_a(z_k)$ and $z_k = G_w(y_k)$.

As one can note the NPC structure allows, contrary to the LPC one, a different prediction window’s length independently to the coding vector size (cf. Fig. 1).

For that, the learning phase is realized in two times. First, the parameterization phase consists in the learning of all the weights by the prediction error minimization criterion:

$$Q = \sum_{k=1}^{K} (y_k - \hat{y_k})^2 = \sum_{k=1}^{K} (y_k - F(y_k))^2 \quad (3)$$

With $y$ the speech signal, $\hat{y}$ the predicted speech signal, $k$ the samples index and $K$ the number of samples.

In this phase, only the first layer weights $w$ which are the NPC encoder parameters are kept. Since the NPC encoder is set up by the parameters defined in the previous phase, the second phase, called the coding phase, consists in the computation of the output layer weights $a$. This is done also by prediction error minimization but only the output layer weights are updated.

One can note that the output function is linear (cf. equation 1), so it can be done by the Levinson algorithm as for the LPC model. Here, for consistency with the parameterization phase, it is done by the backpropagation algorithm. In the next section, we discuss on initialization of this phase.

2.2. A new coding phase

Once the NPC model is parameterized, the coding phase consists in the estimation of the second layer weights by minimizing the prediction error. Like for all optimization processes, the initial weights are randomly chosen. Within the neural networks framework, several studies [12, 13] have been carried out to try to improve the initialization phase. Generally, suggested methods consist in choosing randomly the weights with uniform or Gaussian distributions. Other methods consist in choosing the weights of the network in such way that activations are not in the saturating parts (near +1 and -1 for the sigmoid function) in order to start the minimization process effectively.

One of the major disadvantages of these methods is that they are not adapted to speech processing. Indeed, as the weights solutions minimizing the prediction error are multiple, one does not guarantee a continuity of their values in the frame-by-frame analysis specific to non-stationary signals like in speech processing. In order to guarantee this continuity but also in order to obtain a unique solution, we propose a new initialization method for the NPC coding phase.

We integrate speech knowledge for the initialization. We make the assumption that the NPC model is an extension of the LPC one. Consequently, one can consider as a linear initialization the NPC model initialized by the LPC model. This idea was used in [14] to initialize neuronal models (multi-layer perceptron, recurrent networks, etc.) by equivalent linear models: Auto-Regressive (AR), Auto-Regressive model with eXternal input (ARX). The method suggested is based on matrices decomposition methods (QR, SVD, etc.).

During the coding phase, the NPC model is parameterized, the first layer weights $w$ are known. The initialized second layer weights $a$ (vector code) are determined by equivalence between NPC and LPC models. By neglecting the bias and removing the non-linear activation functions (for linear approximation), one obtains the following equivalence:

$$\Theta = w \cdot a \quad (4)$$

With $\Theta$ LPC parameters of the speech signal, $w$ first layer weights (determined in the previous phase, the parameterization phase) and $a$ second layer weights (to determine).

If the NPC vector code dimension is $\beta$ and the prediction window is $\lambda$ then the LPC vector code dimension has to be set to $\lambda$. The second layer weights $a$ are given by:

$$a = w^+ \cdot \Theta \quad (5)$$

Where $w^+$ is the pseudo-inverse of $w$.

The initialized weights are determined by a simple LPC analysis with an order $\lambda$. Once initialization is carried out, the coding process (prediction error minimization) proceeds as well as the original NPC coding phase by backpropagation algorithm.

3. Feature extraction for speaker identification

Currently, in speaker recognition task, the speech feature extraction is carried out in a same way for all the speakers. Most of the efforts have been made in the second stage which is the model of the speaker (cf. Fig. 2).

In this paper, we propose another approach. The speaker is admittedly modeled by the second stage but the first stage is also specialized in the processing of this speaker. The speaker-dependent characteristics are extracted by the NPC model (cf. Fig. 2). Each NPC model is specialized in the processing of only one speaker.

3.1. Speaker-dependent feature extraction

Traditionally, the speaker recognition task is composed by two phases: the enrollment phase and test phase. Due to our specific approach, these two phases are modified.

3.1.1. Enrollment phase

During the enrollment phase or training phase, each speaker has to provide samples of their speech. Then, a reference model is computed from these samples. In our approach, contrary to other methods, a speaker model is composed by a feature extractor and a reference model. Consequently, the enrollment phase is carried out in two stages. First, the feature extractor is trained. This step is the NPC parameterization phase described previously. Secondly, from these features, a reference model is
computed. One can note that the reference model can be computed by various methods (Gaussian Mixtures Models, Neural Networks, etc.). Once this phase is accomplished, one obtains a NPC model and a reference model for each speaker.

A sentence of one minute is used for the enrollment phase. For NPC parameterization phase, we extract a part of the sentence (only 12 seconds). Once the NPC is parameterized, the whole sentence is coded. Then different methods are carried out in order to compute the reference model. This procedure is repeated for all the speakers belonging of the database.

The reference models are estimated by two methods: the Arithmetic-Harmonic Sphericity (AHS) [15] and the Auto-Regressive Vectorial Model (ARVM) [16, 17].

3.1.2. Test phase
During the test phase, the input speech samples are compared by a similarity measure with the reference models. The identity of the speaker is that whose reference model gives smallest similarity measurement.

The NPC model is parameterized during the enrollment phase and it is used as a speech encoder during the test phase. The speech input is encoded by all the feature extractors. The features resulting from this step are then compared with their appropriate reference models. The decisions are made in function of the used method (AHS or ARVM).

4. Experimental conditions
In order to evaluate the NPC performances, we apply it in a speaker identification task. This task is very interesting because it measures the modelization and the discrimination power of our approach.

4.1. Database
Our experiments have been computed over 49 speakers from the Gaudi database [18] that has been obtained with a microphone connected to a PC. The speech signal has been down-sampled to 8 kHz, pre-emphasized by a first order filter whose transfer function is \( H(z) = 1 - 0.95z^{-1} \). A 30ms Hamming window is used, and the overlapping between adjacent frames is 2/3. A parameterized vector of order 16 was computed. One minute of read text is used for training, and 5 sentences for testing (each sentence is about 2-3 seconds long).

4.2. Coding methods
We compare the proposed coding method based on Neural Predictive Coding (NPC) with the following classical ones:

4.2.1. LPC
The Linear Predictive Coding was a very commonly used method used in speech processing: recognition, synthesis or transmission. This method is based on a linear modelization of the vocal tract. The all-pole auto-regressive (AR) coefficients model of the spectrum captures the vocal tract properties. Indeed, this model is more adapted for voiced sounds.

4.2.2. LPCC
The Linear Predictive Cepstral Coding computes a LPC spectral envelope, before converting into cepstral coefficients. The LPCC are LP-derived cepstral coefficients. The LPCC is the most used coding method in speaker recognition.

4.2.3. MFCC
The Mel Frequency Cepstral Coding is the most used speech coding method in recognition systems. The MFCC is based on signal decomposition with the help of filter bank, which uses the Mel scale. The MFCC results of a discrete cosine transform of the real logarithm of the short-term energy expressed on a Mel-frequency scale. The MFCC has shown good performances in speech recognition [19] but also in speaker recognition [20].

4.2.4. PLP
The Perceptual Linear Predictive (PLP) [21] coding method is an example of knowledge integration resulting from psychoacoustics in the estimation of auto-regressive (AR) models. Indeed, this method integrates critical bands, equal loudness pre-emphasis and intensity-to-loudness compression. The PLP is based on the nonlinear Bark scale. It was originally designed to speech recognition with the removing of speaker dependent characteristics. However, it has been applied to speaker recognition and it has given good results [22].

4.3. Speaker identification algorithms
In order to measure the performances of each coding method, we used two different methods: the Arithmetic-Harmonic Sphericity (AHS) [15] and the Auto-Regressive Vector Model.
(ARVM). The AHS is a statistical method based on second order measures while the ARVM is a predictive based method.

4.3.1. The Arithmetic-Harmonic Sphericity (AHS)

A covariance matrix (CM) is computed for each speaker, and an Arithmetic-Harmonic Sphericity (AHS) measure is used in order to compare matrices [15]:

$$
\mu(C_j, C_{test}) = \log(\text{tr}(C_{test})^{-1} \text{tr}(C_j^{-1})) - 2\log(P)
$$

(6)

Where tr is the trace of the matrix, $P$ is the feature vector dimension ($P = 16$). The number of parameters for each speaker model is $P^2 + P$ (the covariance matrix is symmetric).

For the CM model, more parameters imply a higher dimensional feature vectors.

4.3.2. The Auto-regressive Vectorial Models (ARVM)

The Auto-Regressive Vectorial Models (ARVM) [16] is a vectorial predictive method. The speaker models are linear auto-regressive models, it is based on prediction of the q past parameters vector $\{x_{t-1}, x_{t-2}, \ldots, x_{t-q}\}$:

$$
\hat{y}_t = \sum_{i=0}^{q} A_i \cdot x_{t-i} + e_t
$$

(7)

Where $q$ is the model order, $\{A_i\}$ are $P \times P$ matrices. $e_t$ is a vectorial white noise.

The matrices $\{A_i\}$ are estimated by the help of the Levinson-Whittle-Robinson algorithm.

For the identification process, we use a symmetric distance.

4.4. Results

4.4.1. Identification by the AHS method

The identification rates are shown on table 1. One can see that, for the traditional methods, the best performances are obtained for the MFCC (97.55%) and the LPCC (96.73%) coding methods, which it is in agreement with the coding characteristics. Indeed, these methods try to model the phonetic context but also the speaker characteristics. The LPC model has a better score (90.61%) than the PLP (86.12%). This is due to the fact that the PLP method suppresses speaker dependent characteristics. It is why the PLP method allows comparable performances with the MFCC in speech recognition task.

Table 1: Experimental results for different speech coding methods with the AHS method.

| Speech coding method | Identification rate (%) |
|----------------------|-------------------------|
| LPC                  | 90.61                   |
| LPCC                 | 96.73                   |
| MFCC                 | 97.55                   |
| PLP                  | 86.12                   |
| NPC (random initialization) | 61.63               |
| NPC (linear initialization) | 100                 |

Depending on the initialization, the NPC behavior is very different. We obtain 61.63% for the random initialization while for the linear initialization we obtain 100%. This last initialization gives the best results. One of the reasons is that it allows a continuity on the values of the vector codes.

This proposal differs from [23, 24] that proposed the residual signal obtained by means of a nonlinear filtering as a distance measure. Thus, this paper presents an approach more similar to the classical coding schemes.

4.4.2. Identification by the ARVM method

In order to evaluate the coherence of the results, we use another method based: the Auto-Regressive Vectorial Models (ARVM). Table 2 shows the identification rates for different coding schemes. One can note that we obtain similar results for all the traditional coding methods: LPC, LPCC, MFCC and PLP. However, for the NPC random initialization, the results differ since we obtain 88.57% by the ARVM method and only 61.63% by the AHS one. One of the reasons for this difference is that the AHS method is a unimodal method which is restricted to model linear correlations [22]. Moreover, the NPC method is nonlinear consequently the behavior is different.

However, the NPC linear initialization gives also the best results (100%) with this method. Consequently, these results confirm the performances of the NPC linear initialization.

Table 2: Experimental results for different speech coding methods with the ARVM method.

| Speech coding method               | Identification rate (%) |
|------------------------------------|-------------------------|
| LPC                                | 90.61                   |
| LPCC                               | 93.06                   |
| MFCC                               | 93.69                   |
| PLP                                | 98.36                   |
| NPC (random initialization)         | 88.57                   |
| NPC (linear initialization)         | 100                     |

4.4.3. Fusion methods

The previous results are first tests for the new coding method in speaker recognition. However the NPC random initialization gives very different results from the linear initialization. Moreover, the results are better for the ARVM method. In this section, we study the behavior of the NPC random initialization by fusion methods. The fusion is only computed for the AHS method.

The fusion allows to evaluate the contribution of each coder but also it improves the results. We have evaluated the combination between classical coding methods (LPC, MFCC and PLP) and the NPC random initialization. The combination is done in similar way that in [23, 24]. This combination is known as opinion fusion [25].

For this purpose, we have followed this procedure:

1. Distance normalization [26]:

$$
\alpha_i' = \frac{1}{1 + e^{-k_i}}
$$

(8)

With $k = \frac{m_i - 2\sigma_i}{2\sigma_i}$. $\alpha_i'$ is the opinion of the classifier $i$. $\alpha_i' \in [0, 1]$ is the normalized opinion, $m_i$, $\sigma_i$ are the mean and the standard deviation of the opinions of classifier $i$ using the genuine speakers (intradistances).

2. Weighted sum combination with trained rule:

$$
O = \alpha_1 \alpha_1' + (1 - \alpha) \alpha_2
$$

(9)

Where $\alpha_1, \alpha_2$ are scores (distances) provided by each classifier. $\alpha$ is a weighting or combination factor.
By following this procedure, we compute several coding’s combinations. Table 3 shows the experimental results for the combinations.

Table 3: Experimental results for different combinations.

|        | LPCC | MFCC | PLP | NPC |
|--------|------|------|-----|-----|
| LPC    | 97.14| 99.18| 94.69| 95.47|
| LPCC   | 98.78| 97.96| 97.55|
| MFCC   | 97.55| 98.78|
| PLP    |      |      | 91.84|

For all the coding methods, the results are improved. For instance the MFCC-LPC combination gives very interesting rates (99.18%). An important parameter in this method is the weighting factor $\alpha$ which moderates the opinion of each classifier. Table 4 shows the value of this factor for each combination. For the MFCC-LPC combination: $\alpha = 0.63$ which demonstrates that it exists a complementarity between these encoders. For the PLP combinations, the results are better than those obtained with the AHS method (see table 1). However, the contribution of the PLP method is minor in this combination (PLP-LPC: $\alpha = 0.51$, PLP-LPCC: $\alpha = 0.27$, PLP-MFCC: $\alpha = 0$). Indeed, the best rate is obtained by the PLP-LPCC combination (97.96%), however the combination factor $\alpha$ is only 0.27.

The combinations allow also an improvement of the NPC rates (see table 3). The best results are obtained for a NPC-MFCC combination but the NPC contribution is also minor (NPC-LPC: $\alpha = 0.3$, NPC-LPCC: $\alpha = 0.15$, NPC-MFCC: $\alpha = 0.22$, NPC-PLP: $\alpha = 0.23$). Anyways, the combination of classical coding methods (LPCC, MFCC) and the NPC method improves the results and offers a new approach to the speaker recognition task. Indeed, the NPC model extracts speaker-dependent features.

Table 4: Selected combination factor $\alpha$ for the results shown in table 3.

|        | LPCC | MFCC | PLP | NPC |
|--------|------|------|-----|-----|
| LPC    | 0.59 | 0.63 | 0.51 | 0.3 |
| LPCC   | 0.38 | 0.27 | 0.15 |
| MFCC   | 0    | 0.22 |
| PLP    |      | 0.23 |

5. Conclusions

We have presented a new approach for feature extraction in speaker identification. The main idea is to extract speaker-dependent features. For that, we proposed a new feature extraction architecture. Traditionally, the feature extraction is carried out in a same way for all the speakers. In this paper, the feature extractor forms integral part of the speaker model as much as the reference model.

For that, we have applied the Neural Predictive Coding (NPC) model. This model is an extension of the well-known LPC model by a connectionist model. We have also proposed a new initialization method for the NPC coding phase. This initialization is referred as a linear initialization but it differs from conventional neural networks initializations. The proposed initialization exploits speech knowledge. We used the LPC coding method for the initialization of the nonlinear coding model. By this way, one can see the NPC as a nonlinear feature extractor initialized by a linear model.

The NPC model with a linear initialization gives significant improvements (100% with both AHS and ARVM) compared to the random initialization (AHS: 61.63%, ARVM: 88.57%). The tests have been carried out with two different methods. We used the Arithmetic-Harmonic Spericity (AHS) method and the Auto-Regressive Vectorial Models (ARVM). The results for the coding methods are coherent expect for the random initialization. We obtained 61.63% by the AHS method and 88.57% by the ARVM.

In order to improve but also to better understand the behavior of our new coding method, we computed speaker identification by fusion methods. The fusion is carried out in the classification level. This procedure makes it possible to evaluate the performances of each feature extractor by combinations with others methods. Consequently, the NPC results (random initialization) are improved but its contribution is minor.

6. Future works

Our future studies are devoted to the application of the NPC linear initialization to other tasks like speaker verification (SV). One of the goals of the SV process is to discriminate between speakers. The proposed speaker modelization by a feature extractor and a reference model can be applied to speaker verification. When the speaker claims his identity rather than comparing to only the reference model, we make comparisons with the whole model (feature extractor + reference model). Concerning the NPC model, we are also investigating explicit discriminant feature extraction for speaker recognition.

For real applications, we are investigating robustness feature extraction by fusion methods. During this process, we extract several features (NPC, MFCC or LPCC) and the final decision is done by confidence measures.

7. References

[1] M. Faundez-Zanuy and A. Esposito, “Nonlinear speech processing applied to speaker recognition,” COST-275 workshop, pp. 93–98, 2002.
[2] B. Gas, J.L. Zarader, C. Chavy, and M. Chetouani, “Discriminant neural predictive coding applied to phoneme recognition,” Neurocomputing, vol. 56, pp. 141–166, 2004.
[3] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” Mathematics of Control, Signals and Systems; vol. 2, 1989.
[4] M. Faundez-Zanuy, “Non-linear speech coding with rbf and elman based prediction,” IWANN, vol. II, pp. 671–678, 2003.
[5] M. Birgmeier, “Nonlinear prediction of speech signals using radial basis function network,” EUSIPCO, 1996.
[6] A. Hussain, “Locally-recurrent neural networks for real time adaptive nonlinear prediction of non-stationary signals,” Control and Intelligent Systems, vol. 28, pp. 64–71, 2000.
[7] J. Thyssen, H. Nielsen, and S.D. Hansen, “Non-linearities short-term prediction in speech coding,” Proc. ICASSP, vol. 1, pp. 185–188, 1994.
[8] M. Vandana, “M-ary predictive coding,” Proc. of ISCA Tutorial and Research Workshop on Non-Linear Speech Processing (NOLISP), 2003.
[9] A. Kumar and A. Gersho, “Ld-celp speech coding with nonlinear prediction,” *IEEE Signal Processing Letters*, vol. 4, no. 4, pp. 89–91, April 1997.

[10] S. Wang, E. Paksoy, and A. Gersho, “Performance of nonlinear prediction of speech,” *ICSLP*, vol. 1, pp. 29–32, November 1990.

[11] W. B. Kleijn, “Signal processing representations of speech,” *IEICE Trans. Inf. and Syst.*, vol. E86-D, no. 3, pp. 359–376, March 2003.

[12] L.F.A. Wessels and E. Barnard, “Avoiding false local minima by proper initialization of connections,” *IEEE Trans. on Neural Networks*, vol. 3, pp. 899–905, 1992.

[13] L. Bottou, *Une Approche Théorique de l’Apprentissage Connexioniste: Applications la Reconnaissance de la Parole*, Ph.D. thesis, Université Paris XI (Orsay), 1991.

[14] T. L. Burrows, *Speech Processing with Linear and Neural Networks Models*, Ph.D. thesis, Cambridge, 1996.

[15] B. Bimbot and L. Mathan, “Text-free speaker recognition using an arithmetic-harmonic sphericity measure,” *EUROSPEECH*, pp. 169–172, 1991.

[16] B. Bimbot, L. Mathan, A. De Lima, and G. Chollet, “Standard and target driven ar-vector models for speech analysis and speaker recognition,” *ICASSP*, vol. 2, pp. 5–8, 1992.

[17] C. Montacie and J.L. Le Floch, “Ar-vector models for free-text speaker recognition,” *ICLSP*, vol. 1, pp. 611–614, 1992.

[18] J. Ortega-Garcia and al., “Ahumada: a large speech corpus in spanish for speaker identification and verification,” *ICASSP*, vol. 2, pp. 773–776, 1998.

[19] L. Rabiner and B.J. Juang, *Fundamentals of speech recognition*, Prentice-Hall, 1993.

[20] H. Gish and M. Schmidt, “Text-independent speaker verification,” *IEEE Signal Processing Mag.*, vol. 11, pp. 19–32, 1994.

[21] H. Hermansky, “Perceptual linear predictive (plp) analysis of speech,” *The Journal of the Acoustical Society of America*, pp. 1738–1752, 1990.

[22] S. van Vuuren, “Comparison of text-independent speaker recognition methods on telephone speech with acoustic mismatch,” *ICLSP*, pp. 1784–1787, 1996.

[23] M. Faundez and D. Rodriguez, “Speaker recognition using residual signal of linear and nonlinear prediction models,” *ICSLP*, vol. 2, pp. 121–124, 1998.

[24] M. Fandez-Zanuy, “Speaker recognition by means of a combination of linear and nonlinear predictive models,” *EUROSPEECH*, vol. 2, pp. 763–766, 1999.

[25] M. Fandez-Zanuy, “Data fusion in biometrics,” *Submitted to the IEEE Aerospace and Electronic Systems*, 2004.

[26] C. Sanderson, “Information fusion and person verification using speech and face information,” *IDIAP Communication 02-33*, pp. 1–37, September 2002.