

Accepted Manuscript

Title: Evolutionary cepstral coefficients

Authors: Leandro D. Vignolo, Hugo L. Rufiner, Diego H. Milone, John C. Goddard

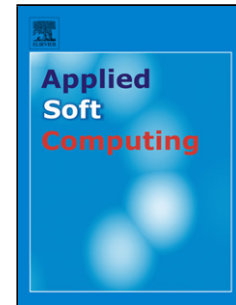
PII: S1568-4946(11)00022-6
DOI: doi:10.1016/j.asoc.2011.01.012
Reference: ASOC 1066

To appear in: *Applied Soft Computing*

Received date: 20-11-2009
Revised date: 3-8-2010
Accepted date: 3-1-2011

Please cite this article as: L.D. Vignolo, H.L. Rufiner, D.H. Milone, J.C. Goddard, Evolutionary cepstral coefficients, *Applied Soft Computing Journal* (2008), doi:10.1016/j.asoc.2011.01.012

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



Evolutionary cepstral coefficients

Leandro D. Vignolo*, Hugo L. Rufiner, Diego H. Milone

*Centro de Investigación y Desarrollo en Señales, Sistemas e Inteligencia Computacional,
Departamento de Informática, Facultad de Ingeniería y Ciencias Hídricas,
Universidad Nacional del Litoral, CONICET, Argentina*

John C. Goddard

*Departamento de Ingeniería Eléctrica, Iztapalapa,
Universidad Autónoma Metropolitana, México*

Abstract

Evolutionary algorithms provide flexibility and robustness required to find satisfactory solutions in complex search spaces. This is why they are successfully applied for solving real engineering problems. In this work we propose an algorithm to evolve a robust speech representation, using a dynamic data selection method for reducing the computational cost of the fitness computation while improving the generalisation capabilities. The most commonly used speech representation are the mel-frequency cepstral coefficients, which incorporate biologically inspired characteristics into artificial recognizers. Recent advances have been made with the introduction of alternatives to the classic mel scaled filterbank, improving the phoneme recognition performance in adverse conditions.

In order to find an optimal filterbank, filter parameters such as the central and side frequencies are optimised. A hidden Markov model is used as the classifier for the evaluation of the fitness for each individual. Experiments were conducted using real and synthetic phoneme databases, considering

*Corresponding author.

Centro de Investigación y Desarrollo en Señales, Sistemas e Inteligencia Computacional, Departamento de Informática, Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral, Ciudad Universitaria CC 217, Ruta Nacional No 168 Km 472.4, TE: +54(342)4575233 ext 125, FAX: +54(342)4575224, Santa Fe (3000), Argentina.

Email address: ldvignolo@fich.unl.edu.ar (Leandro D. Vignolo)

URL: <http://fich.unl.edu.ar/sinc> (Leandro D. Vignolo)

different additive noise levels. Classification results show that the method accomplishes the task of finding an optimised filterbank for phoneme recognition, which provides robustness in adverse conditions.

Keywords:

Automatic speech recognition, evolutionary computation, phoneme classification, cepstral coefficients

1. Introduction

Automatic speech recognition (ASR) systems require a preprocessing stage to emphasize the key features of phonemes, thereby allowing an improvement in classification results. This task is usually accomplished using one of several different signal processing techniques such as filterbanks, linear prediction or cepstrum analysis [1]. The most popular feature representation currently used for speech recognition is mel-frequency cepstral coefficients (MFCC) [2]. MFCC is based on a linear model of voice production together with the codification on a psychoacoustic scale.

However, due to the degradation of recognition performance in the presence of additive noise, many advances have been conducted in the development of alternative noise-robust feature extraction techniques. Moreover, some modifications to the biologically inspired representation were introduced in recent years [3, 4, 5, 6]. For instance, Slaney introduced an alternative [7] to the feature extraction procedure. Skowronski and Harris [8, 9] introduced the human factor cepstral coefficients (HFCC), consisting in a modification to the mel scaled filterbank. They reported results showing considerable improvements over the MFCC. The weighting of MFCC according to the signal-to-noise ratio (SNR) on each mel band was proposed in [10]. For the same purpose, the use of Linear Discriminant Analysis in order to optimise a filterbank has been studied in [11]. In other works the use of evolutive algorithms have been proposed to evolve features for the task of speaker verification [12, 13]. Similarly, in [14] an evolutive strategy was introduced in order to find an optimal wavelet packet decomposition.

Then, the question arises if any of these alternatives is really optimal for this task. In this work we employ an evolutionary algorithm (EA) to find a better speech representation. An EA is an heuristic search algorithm inspired in nature, with proven effectiveness on optimisation problems [15]. We propose a new approach, called evolved cepstral coefficients (ECC), in which

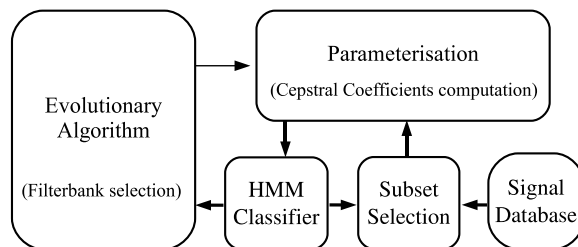


Figure 1: General scheme of the proposed method.

30 an EA is employed to optimise the filterbank used to calculate the cepstral
 31 coefficients (CC). The ECC approach is schematically outlined in Figure 1.
 32 To evaluate the fitness of each individual, we incorporate a hidden Markov
 33 model (HMM) based phoneme classifier. The proposed method aims to find
 34 an optimal filterbank, meaning that it results in a speech signal parameter-
 35 isation which improves standard MFCC on phoneme classification results.
 36 Prior to this work, we obtained some preliminary results, which have been
 37 reported in [16].

38 A problem arises in this kind of optimisation because over-training might
 39 occur and resulting filterbanks could highly depend on the training data
 40 set. This problem could be overcome by increasing the amount of data,
 41 though, much more time or computational power would be needed for each
 42 experiment. In this work, instead, we incorporate a training subset selection
 43 method similar to the one proposed in [17]. This strategy enables us to train
 44 filterbanks with more patterns, allowing generalisation without increasing
 45 computational cost.

46 This paper is organized as follows. First we introduce some basic con-
 47 cepts about EAs and give a brief description of mel-frequency cepstral coef-
 48 ficients. Subsequently, the details of the proposed method are described and
 49 its implementation is explained. In the last sections, the results of phoneme
 50 recognition experiments are provided and discussed. Finally, some general
 51 conclusions and proposals for future work are given.

52 1.1. Evolutionary algorithms

53 Evolutionary algorithms are search methods based on the Darwinian theo-
 54 ry of biological evolution [18]. This kind of algorithms present an implicit
 55 parallelism that may be implemented in a number of ways in order to increase
 56 the computational speed [14]. Usually an EA consists of three operations:

57 selection, variation and replacement [19]. Selection gives preference to bet-
 58 ter individuals, allowing them to continue to the next generation. The most
 59 common variation operators are crossover and mutation. Crossover com-
 60 bines information from two parent individuals into offspring, while mutation
 61 randomly modifies genes of chromosomes, according to some probability, in
 62 order to maintain diversity within the population. The replacement strat-
 63 egy determines which of the current members of the population, should be
 64 replaced by the new solutions. The population consists of a group of indi-
 65 viduals whose information is coded in the so-called chromosomes, and from
 66 which the candidates are selected for the solution of a problem. Each in-
 67 dividual performance is represented by its fitness. This value is measured
 68 by calculating the objective function on a decoded form of the individual
 69 chromosome (called the phenotype). This function simulates the selective
 70 pressure of the environment. A particular group of individuals (the parents)
 71 is selected from the population to generate the offspring by using the vari-
 72 ation operators. The present population is then replaced by the offspring.
 73 The EA cycle is repeated until a desired termination criterion is reached
 74 (for example, a predefined number of generations, a desired fitness value,
 75 etc.). After the evolution process the best individual in the population is the
 76 proposed solution for the problem [20].

77 1.2. Mel-frequency cepstral coefficients

78 Mel-frequency cepstral coefficients are the most commonly used alterna-
 79 tive to represent speech signals. This is mainly because the technique is
 80 well-suited for the assumptions of uncorrelated features used for the HMM
 81 parameter estimation. Moreover, MFCC provide superior noise robustness
 82 in comparison with the linear-prediction based feature extraction techniques
 83 [21].

84 The voice production model commonly used in ASR assumes that the
 85 speech signal is the output of a linear system. This means that the speech
 86 is the result of a convolution of an excitation signal, $x(t)$, with the impulse
 87 response of the vocal tract model, $h(t)$,

$$y(t) = x(t) * h(t), \quad (1)$$

88 where t stands for continuous time. In general only $y(t)$ is known, and it is
 89 frequently desirable to separate its components in order to study the features
 90 of the vocal tract response $h(t)$. Cepstral analysis solves this problem by

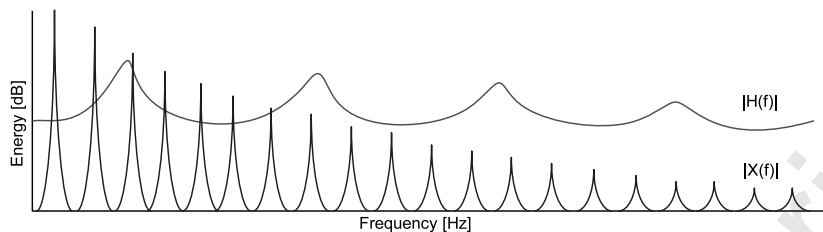


Figure 2: Magnitude spectrums of the excitation signal $X(f)$ and the vocal tract impulse response $H(f)$ from simulated voiced phonemes.

91 taking into account that if we compute the Fourier transform (FT) of (1)
 92 then the equation in the frequency domain is a product:

$$Y(f) = X(f)H(f), \quad (2)$$

93 where variable f stands for frequency, $X(f)$ is the excitation spectrum and
 94 $H(f)$ is the vocal tract frequency response. Then, by computing the loga-
 95 rithm from (2), this product is converted into a sum, and the real cepstrum
 96 $C(t)$ of a signal $y(t)$ is computed by:

$$C(t) = IFT\{\log_e |FT\{y(t)\}|\}, \quad (3)$$

97 where IFT is the inverse Fourier transform. This transformation has the
 98 property that its components, which were nonlinearly combined in time do-
 99 main, are linearly combined in the cepstral domain. This type of homomor-
 100 phic processing is useful in ASR because the rate of change of $X(f)$ and
 101 $H(f)$ are different from each other (Figure 2). Because of this property,
 102 the excitation and the vocal tract response are located at different places
 103 in the cepstral domain, allowing them to be separated. This is useful for
 104 classification because the information of phonemes is given only by $H(f)$.

105 In order to combine the properties of the cepstrum and the results about
 106 human perception of pure tones, the spectrum of the signal is decomposed
 107 into bands according to the mel scale. This scale was obtained through hu-
 108 man perception experiments and defines a mapping between the physical
 109 frequency of a tone and the perceived pitch [1]. The mel scaled filterbank
 110 (MFB) is comprised of a number of triangular filters whose center frequencies
 111 are determined by means of the mel scale. The magnitude spectrum of the
 112 signal is scaled by these filters, integrated and log compressed to obtain a log-
 113 energy coefficient for each frequency band. The MFCC are the amplitudes

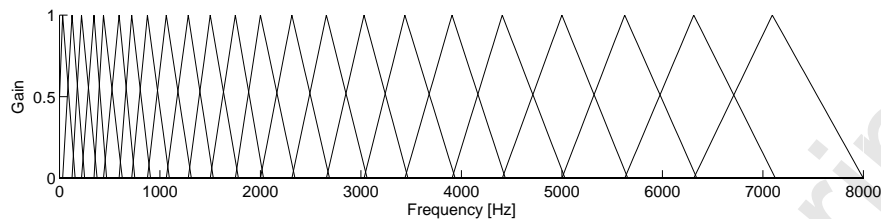


Figure 3: Mel scaled filterbank in the frequency range from 0 to 8kHz.

114 resulting from applying the IFT to the resulting sequence of log-energy co-
 115 efficients [22]. However, because the argument of the IFT is a real and even
 116 sequence, the computation is usually simplified with the cosine transform
 117 (CT). Figure 3 shows a MFB comprised of 26 filters in the frequency range
 118 from 0 to 8 kHz. As it can be seen, endpoints of each filter are defined
 119 by the central frequencies of adjacent filters. Bandwidths of the filters are
 120 determined by the spacing of filter central frequencies which depend on the
 121 sampling rate and the number of filters. That is, if the number of filters
 122 increases, the number of MFCC increases and the bandwidth of each filter
 123 decreases.

124 2. MATERIALS AND METHODS

125 This section describes the proposed evolutionary algorithm, the speech
 126 data and the preprocessing method. First, the details about the speech
 127 corpus are given and the ECC method is explained. In the next subsection
 128 some considerations about the HMM based classifier are discussed and finally
 129 the data selection method for resampling training is explained.

130 2.1. *Speech corpus and processing*

131 For the experimentation, both synthetic and real phoneme databases have
 132 been used. In the first case, five Spanish vowels were modelled using the clas-
 133 sical linear prediction coefficients [1], which were obtained from real utter-
 134 ances. We have generated different train, test and validation sets of signals
 135 which are 1200 samples in length and sampled at 8 kHz. Every synthetic
 136 utterance has a random fundamental frequency, uniformly distributed in the
 137 range from 80 to 250 Hz. In this way we simulate both male and female
 138 speakers. First and second resonant frequencies (formants) were randomly

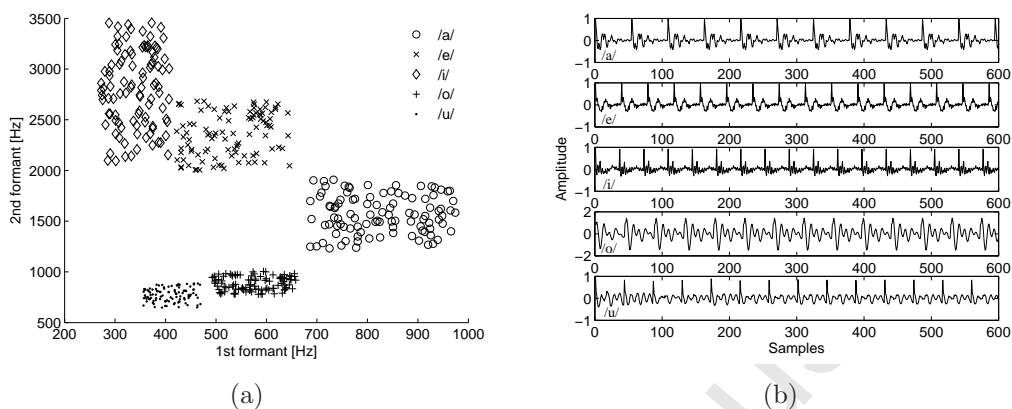


Figure 4: Synthetic phoneme database. a) First and second formant frequency distribution. b) Phoneme examples.

139 modified, within the corresponding ranges, in order to generate phoneme
 140 occurrences.

141 Our synthetic database included the five Spanish vowels /a/, /e/, /i/,
 142 /o/ and /u/, which can be simulated in a controlled manner.

143 Figure 4 shows the resulting formant distribution and some synthetic
 144 phoneme examples. White noise was generated and added to all these syn-
 145 thetic signals, so that the SNR of each signal is random and it varies uniformly
 146 from 2 dB to 10 dB. As these vowels are synthetic and sustained, the frames
 147 were extracted using a Hamming window of 50 milliseconds length (400 sam-
 148 ples). The use of a synthetic database allowed us to maintain controlled
 149 experimental conditions, in which we could focus on the evolutive method,
 150 designed to capture the frequency features of the signals while disregarding
 151 temporal variations.

152 Real phonetic data was extracted from the TIMIT speech database [23].
 153 Speech signals were selected randomly from all dialect regions, including both
 154 male and female speakers. Utterances were phonetically segmented to obtain
 155 individual files with the temporal signal of every phoneme occurrence. White
 156 noise was also added at different SNR levels. In this case, the sampling fre-
 157 quency was 16 kHz and the frames were extracted using a Hamming window
 158 of 25 milliseconds (400 samples) and a step-size of 200 samples. All possible
 159 frames within a phoneme occurrence were extracted and padded with zeros
 160 where necessary. The English phonemes /b/, /d/, /eh/, /ih/ and /jh/ were
 161 considered. The occlusive consonants /b/ and /d/ are included because they

162 are very difficult to distinguish in different contexts. Phoneme /jh/ presents
 163 special features of the fricative sounds. Vowels /eh/ and /ih/ are commonly
 164 chosen because they are close in the formant space. This group of phonemes
 165 was selected because they constitute a set of classes which is difficult to
 166 classify [24].

167 For simplicity we introduced the steps for the computation of CC in the
 168 continuous time and frequency domains. Although, in practice we use digital
 169 signals and the discrete versions of the transforms mentioned in Section 1.2.
 170 For both MFCC and ECC the procedure is as follows. First, the spectrum
 171 of the frame is normalised and integrated by the triangular filters, and every
 172 coefficient resulting from integration is then scaled by the inverse of the
 173 area of the corresponding filter. As in the case of Slaney's filterbank [7], we
 174 give equal weight to all coefficients because this is shown to improve results.
 175 Then the discrete cosine transform (DCT) is computed from the log energy
 176 coefficients. As the number of filters n_f in each filterbank is not fixed, we set
 177 the number of output DCT coefficients to $\lfloor n_f/2 \rfloor + 1$.

178 2.2. Evolutionary cepstral coefficients

179 The MFB shown in Figure 3, commonly used to compute cepstral coeffi-
 180 cients, reveals that the search for an optimal filterbank can involve adjusting
 181 several of its parameters, such as: shape, amplitude, position and size of each
 182 filter. However, trying to optimise all the parameters together is extremely
 183 complex, so we decided to maintain some of the parameters fixed.

184 We carried out this optimisation in two different ways. In the first case,
 185 we considered non-symmetrical triangular filters, determined by three param-
 186 eters each. These three parameters correspond to the frequency values where
 187 the triangle for the filter begins, where the triangle reaches its maximum, and
 188 where it ends. This is depicted in Figure 5, where the mentioned parameters
 189 are called a_i , b_i and c_i respectively. They are coded in the chromosome as
 190 integer values, indexing the frequency samples. The size and overlap between
 191 filters are left unrestricted in this first approach. The number of filters was
 192 also optimised by adding one more gene to the chromosome (n_f in Figure
 193 5). This last element in the chromosome indicates that the first n_f filters are
 194 currently active. Hence, the length of each chromosome is three times the
 195 maximum number of filters allowed in a filterbank, plus one.

196 In a second approach, we decided to reduce the number of optimisation
 197 parameters. Here, triangular filters were distributed along the frequency

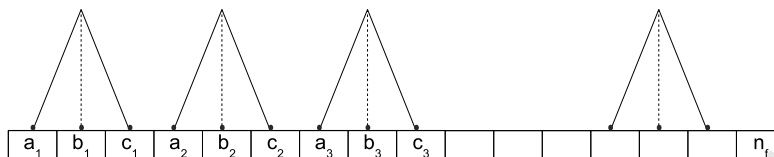


Figure 5: Scheme of the chromosome codification.

198 band, with the restriction of half overlapping. This means that only the central
 199 positions (parameters c_i in Figure 5) were optimised, and the bandwidth
 200 of each filter was adjusted by the preceding and following filters. In this case,
 201 the number of filters was optimised too.

202 In other approaches [13], polynomial functions were used to encode the
 203 parameters which were optimised. Here, in contrast, all the parameters are
 204 directly coded in the chromosome. In this way the search is simpler and the
 205 parameters are directly related to the features being optimised.

206 Each chromosome represents a different filterbank, and they are initialized
 207 with a random number of active filters. In the initialization, the position of
 208 the filters in a chromosome is also random and follows a discrete uniform
 209 distribution over the frequency bandwidth from 0 Hz to half the sampling
 210 frequency. The position, determined in this way, sets the frequency where
 211 the triangle of the filter reaches its maximum. Then, in the case of the three-
 212 parameter filters, a binomial distribution centred on this position is used to
 213 initialize the other two free parameters of the filter.

214 Before variation operators are applied, the filters in every chromosome
 215 are sorted by increasing order with respect to their central position. A chro-
 216 mosome is coded as a string of integers and the range of values is determined
 217 by the number of samples in the frequency domain.

218 The EA uses the roulette wheel selection method [25], and elitism is
 219 incorporated into the search due to its proven capabilities to enforce the
 220 algorithm's convergence under certain conditions [18]. The elitist strategy
 221 consists in maintaining the best individual from one generation to the next
 222 without any perturbation. The variation operators used in this EA are mu-
 223 tation and crossover, and they were implemented as follows. Mutation of a
 224 filter consists in the random displacement of one of its frequency parameters,
 225 and this modification is made using a binomial distribution. This mutation
 226 operator can also change, with the same probability, the number of filters in
 227 a filterbank. Our one-point crossover operator interchanges complete filters
 228 between different chromosomes. Suppose we are applying the crossover op-

229 erator on two parents, for instance A and B. Then, if parent B contains more
 230 active filters than parent A, the crossover point is a random value between 1
 231 and the n_f value of parent A. All genes (filters and n_f) beyond that point in
 232 either chromosome string are swapped between the two parents, resulting in
 233 an offspring with the same n_f of the first parent and an offspring with the
 234 same n_f of the second parent.

235 The selection of individuals is also conducted by considering the filterbank
 236 represented by a chromosome. The selection process should assign greater
 237 probability to the chromosomes providing the better signal representations,
 238 and these will be those that obtain better classification results. The proposed
 239 fitness function consists of a phoneme classifier, and the recognition rate will
 240 be the fitness value for the individual being evaluated.

241 *2.3. HMM based classifier*

242 In order to compare the results to those of state of the art speech recog-
 243 nition systems, we used a phoneme classifier based on HMM with Gaussian
 244 mixtures (GM). This fitness function uses tools from the HMM Toolkit [26]
 245 for building and manipulating hidden Markov models. These tools rely on
 246 the Baum-Welch algorithm [27] which is used to find the unknown paramete-
 247 rters of an HMM, and on the Viterbi algorithm [28] for finding the most likely
 248 state sequence given the observed events in the recognition process.

249 Conventionally, the energy coefficients obtained from the integration of
 250 the log magnitude spectrum are transformed by the DCT to the cepstral
 251 domain. Besides the theoretical basis given on Section 1.2, this has the effect
 252 of removing the correlation between adjacent coefficients. Moreover, it also
 253 reduces the feature dimension.

254 Even though DCT has a fixed kernel and cannot decorrelate the data as
 255 thoroughly as data-based transforms [29], MFCC are close to decorrelated.
 256 The DCT produces nearly uncorrelated coefficients [30], which is desirable for
 257 HMM based speech recognizers using GM observation densities with diagonal
 258 covariance matrices [31].

259 *2.4. Dynamic subset selection for training*

260 A problem in evolutionary optimisation is that it requires enormous com-
 261 putational time. Usually, fitness evaluation takes the most time since it re-
 262 quires the execution of some kind of program against problem specific data.
 263 In our case, for instance, we need to train and test an HMM based classifier
 264 using a phoneme database. This implies that the time for the evolution is

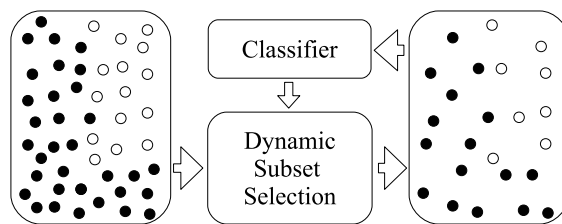


Figure 6: Scheme of the dynamic subset selection method.

265 proportional to the size of the data needed for fitness evaluation, as well as
 266 the population size and the number of generations. On the other hand, the
 267 data used for fitness evaluation dramatically influences the generalisation ca-
 268 pability of the optimised solution. Hence, there is a trade off between the
 269 generalisation capability and the computational time.

270 In this work we propose the reduction of computational costs and the
 271 improvement of generalisation capability by evolving filterbank parameters
 272 on a selected subset of train and test patterns, which is changed during
 273 each generation. The idea of active data selection in supervised learning was
 274 originally introduced by Zhang et al. for efficient training of neural networks
 275 [32, 33]. Motivated by this work, Gathercole et al. introduced some training
 276 subset selection methods for genetic programming [17]. These methods are
 277 also useful in evolutionary optimisation, allowing us to significantly reduce
 278 the computation time while improving generalisation capability.

279 While in [17] only one training data set was considered, our subset se-
 280 lection method consists in changing the test subset, as well as the training
 281 subset, in every generation of the EA. For the test set, the idea is to focus
 282 the EA attention onto the cases that were mostly misclassified in previous
 283 generations and the cases that were not used recently.

284 In order to illustrate this, an example with two classes of two-dimensional
 285 patterns is outlined in Figure 6. The subset is selected from the original data
 286 set according to the classification results. The algorithm randomly selects
 287 a number of cases from the whole training and test sets every generation,
 288 and a test case has more probability to be selected if it is difficult or has not
 289 been selected for several generations. Another difference with the method
 290 proposed in [17] is that the size of test and train subsets remains strictly the
 291 same for all generations. In the first generation the testing subset is selected
 292 assigning the same probability to all cases. Then, during generation g , a
 293 weight $W_i(g)$ is determined for each test case i . This weight is the sum of

294 the current difficulty of the case, $D_i(g)$, raised to the power d , and the age
 295 of the case, $A_i(g)$, raised to the power a ,

$$W_i(g) = D_i(g)^d + A_i(g)^a. \quad (4)$$

296 The difficulty of a test case is given by the number of times it was mis-
 297 classified and its age is the number of generations since it was last selected.
 298 Exponents d and a determine the importance given to *difficult* and *unse-*
 299 *lected* cases respectively. Given the sample size and other characteristics of
 300 the training data, these parameters are empirically determined. Each test
 301 case is given a probability $P_i(g)$ of being selected. This probability is given
 302 by its weight, multiplied by the size of the selected subset, S , and divided by
 303 the sum of the weights of all the test cases:

$$P_i(g) = \frac{W_i(g) * S}{\sum_j W_j(g)}. \quad (5)$$

304 When a test case i is selected, its age A_i is set to 1 and, if it is not selected,
 305 its age is incremented. While evaluating the EA population, difficulty D_i is
 306 incremented each time the case i is misclassified.

307 However, a problem arises when using an elitist strategy together with this
 308 method. As train and test subsets change, the best individual at a given time
 309 may no longer be the best one for the next generation. Although, probably it
 310 is still a good individual, we decided to maintain the best chromosome from
 311 the previous generation and assign the classification result from the current
 312 subset as its fitness.

313 3. Results and discussion

314 3.1. Synthetic Spanish phonemes

315 We conducted different EA runs and we found the best results when we
 316 evolved only the central filter positions and the number of filters, which we
 317 allowed to vary from 17 to 32. For the EA, the population size was set to 100
 318 individuals and crossover rate was set to 0.8. The mutation rate, meaning
 319 the probability of a filter to have one of its parameters changed, was set to
 320 0.1.

321 During the EA runs we used a set of 500 training signals and a different set
 322 of 500 test signals to compute the fitness for every individual. In this case,
 323 training and testing sets remained unchanged during the evolution. Each

Table 1: Average classification rates (percent) for synthetic phonemes.

FB	# filters	# coeff	Validation test	
			DCM	FCM
EFB 1	17	9	<i>95.20</i>	97.00
EFB 2	18	10	95.40	<i>96.80</i>
EFB 3	18	10	93.00	<i>96.40</i>
EFB 4	17	9	94.60	96.20
MFB	23	13	94.80	96.20
MFB	17	9	93.00	95.20

run was terminated after 100 generations without any fitness improvement. When a run was finished, we took the twenty best filterbanks according to their fitness, and we made a validation test with another set of 500 signals. From this validation test we selected the two best filterbanks, discarding those that were over-optimised (those with higher fitness but with lower validation result).

Table 1 summarizes the validation results for filterbanks from two different optimisations, and includes the classification results obtained using the standard MFB on the same data sets. The fourth column contains the classification results obtained when using an HMM with diagonal covariance matrices (DCM), and the fifth column contains the results obtained when using an HMM with full covariance matrices (FCM). Evolved filterbanks (EFB) 1 and 2 were obtained using HMM with DCM as fitness during the optimisation, while EFBs 3 and 4 were obtained using HMM with FCM. It can be observed that we obtained filterbanks that perform better than MFB when using FCM-HMM. Also, it is important to notice that MFB also performs better using FCM-HMM.

Figure 7 shows these four EFBs. One feature they all have in common is the high density of filters from approximately 500 to 1000 Hz, which could be related to the distribution of the first frequency formant (Figure 4). Moreover, considering the second formant frequency, it can be noticed that these groups of filters could distinguish phonemes /o/ and /u/ from the others. Another common trait in these four filterbanks is that the frequency range from 0 to 500 Hz is covered by only two filters, although, in EFB 3 there is a narrow filter from 0 to 40 Hz, besides these two. This narrow filter isolates the peaks at zero frequency from the phoneme information. Another likeness

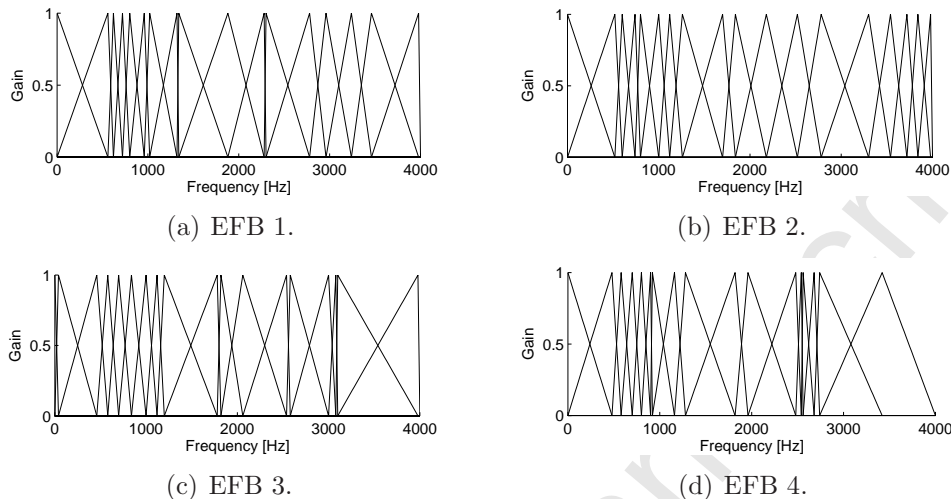


Figure 7: Filterbanks optimised for phonemes /a/, /e/, /i/, /o/ and /u/ from our synthetic database.

350 is that, in the band from approximately 1000 to 2500 Hz, the four filterbanks
 351 show similar filter distribution. On the other hand, a feature which is present
 352 only in the second filterbank is the attention given to high frequencies, as
 353 opposed to MFB, and taking higher formants into account.

354 3.2. Real English phonemes

355 In the second group of experiments the best results were obtained when
 356 considering non-symmetrical triangular filters, determined by three param-
 357 eters each. Also in this case, the number of filters in the filterbanks was
 358 allowed to vary from 17 to 32. For the fitness computation we used a dy-
 359 namic data partition of 1000 training signals and 400 test signals, and an
 360 HMM based classifier with FCM. The data partition used during the EA
 361 runs was changed every generation according to the strategy described in
 362 Section 2.4, and phoneme samples were dynamically selected from a total of
 363 6045 signals available for training and 1860 signals available for testing. As
 364 mentioned in Section 2.4, some preliminary experiments were carried out in
 365 order to set difficulty and age exponents (parameters d and a in equation
 366 4). Given the sample size and using different combinations, we found that a
 367 good choice is to set both parameters d and a to 1.0.

368 As in the experiments with synthetic phonemes, a EA run was ended

Table 2: Classification rates for English phonemes (percent). Average over ten train/test partitions. Filterbanks optimised at 0 dB SNR.

FB	# filters	# coeff	-5dB	0dB	20dB	clean	Diff
A0	32	17	24.76	32.62	58.26	65.54	0.44
A1	17	9	20.26	26.02	62.16	62.62	-9.68
A2	21	11	20.16	21.34	59.56	60.00	-19.68
A3	29	15	24.34	32.92	66.08	64.32	6.92
A4	19	10	20.38	26.32	63.64	61.22	-9.18
A5	19	10	20.52	26.24	60.62	60.26	-13.10
A6	21	11	31.10	35.78	61.52	60.80	8.46
A7	29	15	22.58	30.52	63.90	64.58	0.84
A8	25	13	22.94	30.76	62.10	62.08	-2.86
A9	22	12	23.60	31.54	63.54	66.14	4.08
MFB	23	13	20.00	23.18	68.40	69.16	

369 after 100 generations without any fitness improvement, and we took the ten
 370 best filterbanks according to their fitness. The settings for the parameters of
 371 the EA were also the same values given in Section 3.1. We made validation
 372 tests with ten different data partitions consisting of 2500 train patterns and
 373 500 test patterns each. Moreover, these validation tests were made using test
 374 sets at different SNR levels.

375 Here we show the classification results of filterbanks obtained from three
 376 EA runs which only differ in the noise level used for train and test sets for the
 377 fitness computation. Table 2 shows average classification results comparing
 378 filterbanks optimised for signals at 0 dB SNR against standard MFB, using
 379 DCM-HMM. We tested the best ten EFBs at different SNR, always training
 380 the classifier with clean signals. Each one of these results were obtained as
 381 the average of the classification with ten different data partitions. The last
 382 column gives the accumulated difference between each of the first ten rows
 383 and the last row, the higher values indicate the best filterbanks. For example,
 384 in Table 2, we obtain the value 0.44 in the first row by adding the difference
 385 of the values from column 4 to column 7 in the first row, from those in row
 386 11. As the number of filters is one of the optimised parameters, we compare
 387 all the EFBs against a MFB composed of 23 filters, which is a standard setup
 388 in speech recognition. It can be seen that when testing at -5 and 0 dB SNR
 389 the EFB A6 performs much better than MFB. From this we can assume that
 390 the distribution of filters in EFB A6 allows to distinguish better the formant

Table 3: Classification rates for English phonemes (percent). Average over ten train/test partitions. Filterbanks optimised at 20 dB SNR.

FB	# filters	# coeff	-5dB	0dB	20dB	clean	Diff
B0	20	11	20.04	22.24	62.30	63.06	-13.10
B1	19	10	22.18	30.06	53.76	64.12	-10.62
B2	22	12	22.44	30.24	60.68	64.96	-2.42
B3	19	10	21.38	27.84	68.08	67.80	4.36
B4	19	10	21.10	26.72	62.40	64.52	-6.00
B5	19	10	22.06	34.54	55.56	64.46	-4.12
B6	18	10	20.22	31.92	68.44	66.64	6.48
B7	19	10	22.88	31.98	64.44	67.26	5.82
B8	18	10	21.58	27.90	64.04	61.88	-5.34
B9	19	10	22.82	31.08	64.28	68.04	5.48
MFB	23	13	20.00	23.18	68.40	69.16	

391 frequencies from the noise frequency components. This means that the use
 392 of the evolved filterbank results in features which are more robust than the
 393 standard parameterisation.

394 The same comparison is made in Tables 3 and 4 for filterbanks optimised
 395 using signals at 20 dB SNR and clean signals respectively. Again, we can see
 396 that some EFBs perform considerably better than the MFB with noisy test
 397 signals, and there is even an improvement at 20 dB SNR in these cases.

398 From these three groups of EFBs we selected some of the best EFBs and
 399 further tested them at 5, 10, 15 and 30 dB SNR. The average results from ten
 400 data partitions can be found in Table 5, as well as the results for the MFB,
 401 HFCC and Slaney filterbanks. For the HFCC 30 filters were considered,
 402 one filter was added to the filterbank proposed in [34] because the sampling
 403 frequency used in our experiments is higher. The bandwidths of the filters
 404 in HFCC are controlled by a parameter called E-factor, which was set to 5,
 405 based on the recognition results shown in [34]. As suggested, the first 13
 406 cepstral coefficients were considered. The Slaney filterbank was comprised
 407 of 40 filters, as proposed in [7], and 20 cepstral coefficients were computed.

408 It can be seen that the EFBs perform better than the standard MFB
 409 when the SNR in testing signals is lower than the SNR in the training sig-
 410 nals. Moreover, EFB C4 and EFB B6 outperform the Slaney filterbank in all
 411 noise conditions considered except in the case of -5 dB SNR. On the other
 412 hand, the EFBs perform better than the HFCC filterbank at the lower SNRs,

Table 4: Classification rates for English phonemes (percent). Average over ten train/test partitions. Filterbanks optimised for clean signals.

FB	# filters	# coeff	-5dB	0dB	20dB	clean	Diff
C0	21	11	20.56	27.94	64.14	63.48	-4.62
C1	18	10	20.08	34.20	61.26	60.66	-4.54
C2	19	10	20.28	27.74	62.62	60.72	-9.38
C3	18	10	21.94	30.32	62.70	64.36	-1.42
C4	18	10	20.56	36.88	69.82	68.08	14.60
C5	18	10	22.26	30.42	65.14	63.40	0.48
C6	19	10	20.30	30.16	64.82	62.62	-2.84
C7	18	10	20.16	30.66	63.22	61.96	-4.74
C8	18	10	26.52	33.56	56.62	64.00	-0.04
C9	18	10	20.40	26.68	66.88	66.22	-0.56
MFB	23	13	20.00	23.18	68.40	69.16	

413 this is from -5 dB to 15 dB SNR. These improvements may be better visu-
 414 alized in Figure 8, where it is easy to appreciate that EFB C4 outperforms
 415 MFB in the range from 0 dB to 15 dB SNR. It can be seen that MFB is
 416 not outperformed for 30 dB SNR and clean signals, however this behaviour
 417 is common to most robust ASR methods [35]. For instance, the HFCC fil-
 418 terbank outperform MFB for noisiest cases, however, above 20 dB SNR the
 419 improvements are smaller. Moreover, the degradation of recognition perfor-
 420 mance is proportional to the mismatch between the SNR of the training set
 421 and the SNR of the test set [36, 4].

422 Figure 9 shows the selected EFBs from Table 5. As we stated before,
 423 one feature they all have in common is the wide bandwidth of most of the
 424 filters, compared with the MFB. This coincides with the study in [34] about
 425 the effect of wider filter bandwidth on noise robustness. In all the EFBs we
 426 can also see high overlapping between different filters, as there was not any
 427 constraint about this in the optimisation. However, this high overlapping
 428 which results in correlated CC could be beneficial for classification with full
 429 covariance matrix HMM. We can observe the grouping of a relatively high
 430 number of filters in the frequency band from 0 Hz to 4000 Hz in the case of
 431 EFB C4, which gives the best results for noisy test signals.

432 In order to analyse what information these representations are captur-
 433 ing, we recovered an estimate of the short-time magnitude spectrum using
 434 the method proposed in [37]. Which consists in scaling the spectrogram of

Table 5: Classification rates for English phonemes (percent). Average over ten train/test partitions.

FB	-5dB	0dB	5dB	10dB	15dB	20dB	30dB	clean
A3	24.34	32.92	37.68	46.36	52.98	66.08	65.04	64.32
A6	31.10	35.78	44.38	46.88	53.12	61.52	60.36	60.80
B6	20.22	31.92	55.12	67.20	68.84	68.44	67.20	66.64
B7	22.88	31.98	36.86	44.42	49.64	64.44	67.58	67.26
C4	20.56	36.88	60.30	68.32	68.70	69.82	67.42	68.08
C5	22.26	30.42	34.38	44.32	57.28	65.14	63.52	63.40
MFB	20.00	23.18	37.90	44.68	51.42	68.40	69.80	69.16
HFCC	20.24	25.98	47.26	62.78	67.68	70.54	69.42	70.36
Slaney	29.94	30.28	36.44	54.76	60.66	62.02	61.52	62.78

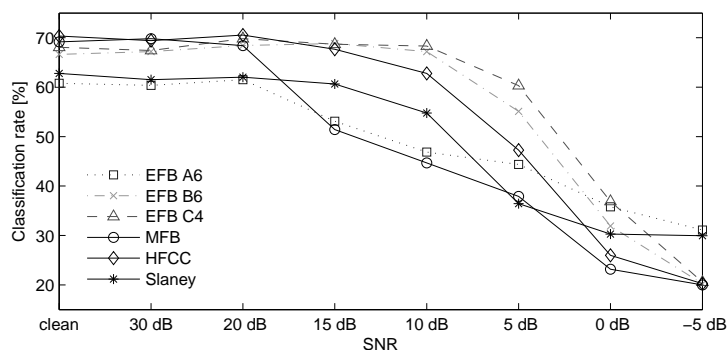


Figure 8: Performance of the best EFBs compared with MFB (English phonemes).

435 a white noise signal by the short-time magnitude spectrum recovered from
 436 the cepstral coefficients. Figures 10 and 11 show the spectrograms of sen-
 437 tence SI648 from TIMIT corpus, with additive noise at 50 dB and 10 dB
 438 SNR respectively. Figure 10 shows that wide filters of the EFB blur energy
 439 coefficients along the frequency axis, and it is more difficult to notice the
 440 formant frequencies, though this information is not lost. Moreover, phoneme
 441 classification is made easier by removing information related to pitch. On the
 442 other hand, from Figure 11 it can be seen that when the signal is noisy, the
 443 relevant information is clearer in the spectrogram reconstructed from ECC.
 444 This is because the filter distribution and bandwidths of EFB C4 allow the
 445 relevant information on higher frequencies to be conserved, which is hidden
 446 by noise when using MFCC.

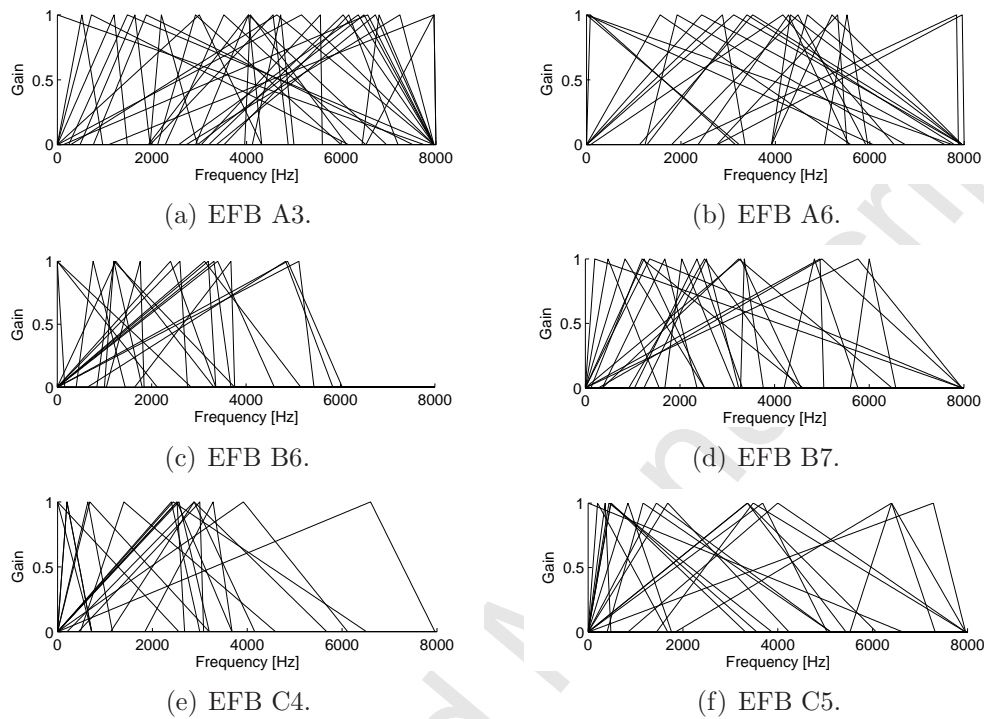


Figure 9: Filterbanks optimised for phonemes /b/, /d/, /eh/, /ih/ and /jh/ from TIMIT database.

447 Table 6 exhibits the confusion matrices for MFB and EFB C4, obtained
 448 when testing with signals at 10 and 15 dB SNR. From these matrices, it can
 449 be seen that phonemes /eh/ and /ih/ are mostly misclassified using MFB
 450 and they are frequently well classified using EFB C4. In fact, when the SNR
 451 is high, the performance in the classification of each of the five phonemes is
 452 similar for both MFB and EFB C4. However, the lower the SNR, the more
 453 MFB fails to classify phonemes /eh/ and /ih/. These are mostly confused
 454 with phonemes /b/ and /d/, while the success rate for phonemes /b/, /d/
 455 and /jh/ is barely affected. On the other hand, when using EFB C4 the effect
 456 of noise degrades the success rate for all phonemes uniformly, but none of
 457 them are as confused as in the case of MFB. That is, not only the average of
 458 success rate is higher, but also the variance between phonemes is lower. This
 459 means that the evolved filterbank provides a more robust parameterisation
 460 as it achieves better classification results in the presence of noise.

Table 6: Confusion matrices. Average classification rates (percent) from ten data partitions.

		MFB					EFB C4				
		/b/	/d/	/eh/	/ih/	/jh/	/b/	/d/	/eh/	/ih/	/jh/
15 dB	/b/	64.7	34.8	00.0	00.0	00.5	56.9	39.7	01.8	01.4	00.2
	/d/	11.7	83.2	00.0	00.1	5.00	14.1	79.9	00.6	00.9	04.5
	/eh/	33.1	51.0	05.0	07.1	03.8	03.9	04.5	73.5	18.1	00.0
	/ih/	21.8	45.3	04.7	18.9	09.3	12.6	09.9	18.2	59.3	00.0
	/jh/	00.1	14.6	00.0	00.0	85.3	00.3	25.3	00.2	00.3	73.9
						Avg: 51.42					Avg: 68.70
10 dB	/b/	55.4	44.0	00.0	00.0	00.6	48.8	48.6	01.5	00.5	00.6
	/d/	07.4	89.2	00.0	00.0	30.4	08.2	86.4	00.0	00.0	05.4
	/eh/	25.6	70.6	00.0	00.0	30.8	03.7	06.5	77.4	12.4	00.0
	/ih/	13.5	68.6	00.0	00.0	17.9	09.1	10.3	22.9	57.7	00.0
	/jh/	00.0	21.2	00.0	00.0	78.8	00.2	28.3	00.0	00.2	71.3
						Avg: 44.68					Avg: 68.32

461 3.3. Statistical dependence of ECC

462 As we mentioned in Section 2.3, MFCC are almost uncorrelated and are
463 suitable for the utilization of HMM. However, this assumption of weak sta-
464 tistical dependence may not be true for the ECC. As Figure 9 shows, filter
465 bandwidth and overlapping is usually higher for the optimised filterbanks
466 than MFB. This means that the energy coefficients contain highly redun-
467 dant information, and DCT may not be enough to obtain near decorrelated
468 coefficients in this case. In fact, we have studied and compared the statisti-
469 cal dependence of MFCC and ECC, and noticed that optimised coefficients
470 show, in general, higher correlation. Figure 12 shows the correlation matri-
471 ces of 10 cepstral coefficients computed over 1500 frames. In order to make
472 this comparison, we used a MFB consisting on 18 filters, the same num-
473 ber of filters in the optimised filterbank named C4. Correlation coefficients
474 corresponding to MFB are shown on top and those corresponding to the op-
475 timised filterbank C4 at the bottom. As can be seen, correlation matrices
476 show high statistical dependence between cepstral coefficients corresponding
477 to phonemes /eh/ and /ih/, and this is much more noticeable for the case
478 of the evolved filterbank. In order to obtain a measure of the statistical
479 dependence, the sum of the correlation coefficients for each phoneme was
480 obtained. These values can be seen on Table 7, and they were computed as
481 $\sum_i \sum_j |M_{i,j}| - \text{trace}(|M|)$, where M is the matrix of correlation coefficients.
482 From these values we can also see that ECC are more correlated than the

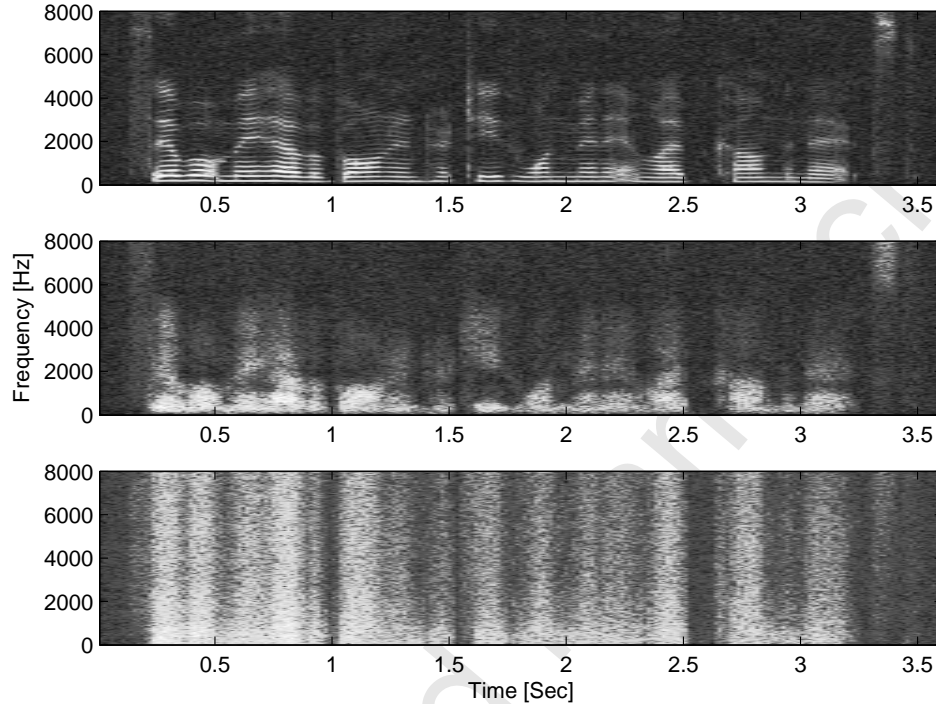


Figure 10: Spectrograms for sentence SI648 from TIMIT corpus at 50dB SNR. Computed from the original signal (top), reconstructed from MFCC (middle) and reconstructed from ECC (bottom).

483 MFCC for the set of phonemes we have considered.

484 The statistical dependence which is present in ECC implies that GM
 485 observation densities with diagonal covariance matrices (DCM) may not be
 486 the best option. Hence we decided to use full covariance matrices instead, to
 487 model the observation density functions during the optimisation. Moreover,
 488 as the MFCC are not completely decorrelated, they also allowed the classifier
 489 to perform better when using full covariance matrices (FCM) (See Table 1).

490 4. Conclusion and future work

491 A new method has been proposed for evolving a filterbank, in order to
 492 produce a cepstral representation that improves the classification of noisy
 493 speech signals. Our approach successfully exploits the advantages of evolu-
 494 tionary computation in the search for an optimal filterbank. Free parameters

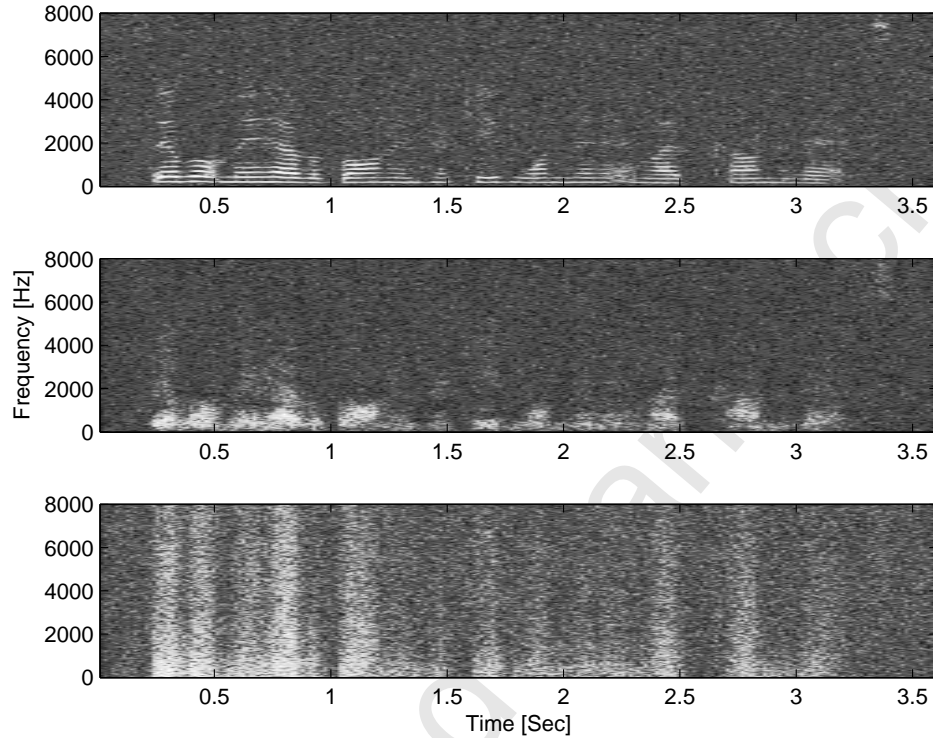


Figure 11: Spectrograms for sentence SI648 from TIMIT corpus at 10dB SNR. Computed from the original signal (top), reconstructed from MFCC (middle) and reconstructed from ECC (bottom).

495 and codification provided a wide search space, which was covered by the algo-
 496 rithm due to the design of adequate variation operators. Moreover, the data
 497 selection method for resampling prevented the overfitting without increasing
 498 computational cost.

499 The obtained representation provides a new alternative to classical ap-
 500 proaches, such as those based on a mel scaled filterbank or linear prediction,
 501 and may be useful in automatic speech recognition systems. Experimental re-
 502 sults show that the proposed approach meets the objective of finding a more
 503 robust signal representation. This approach facilitates the task of the classi-
 504 fier because it properly separates the phoneme classes, thereby improving the
 505 classification rate when the test noise conditions differ from the training noise
 506 conditions. Moreover, with the use of this optimal filterbank the robustness

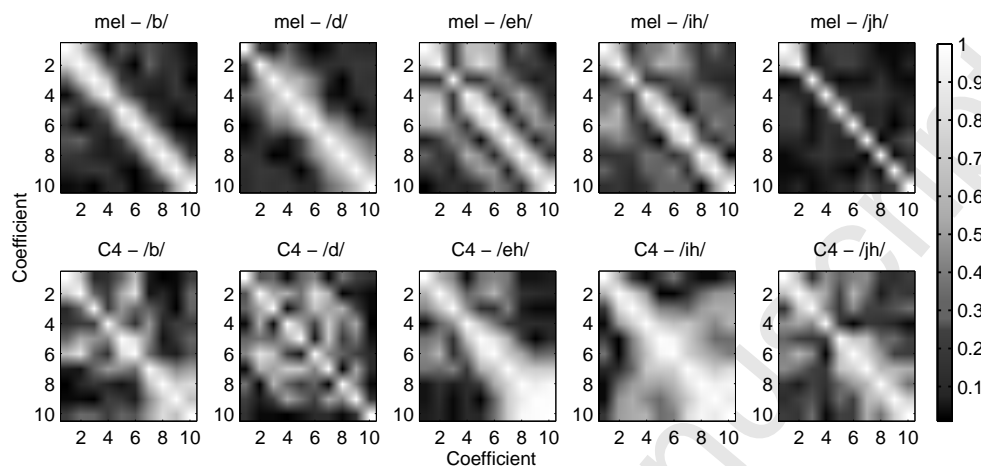


Figure 12: Correlation matrices of MFCC (top) and ECC (bottom).

Table 7: Sum of correlation coefficients.

	/b/	/d/	/eh/	/ih/	/jh/
MFB	20.9	24.9	30.4	27.2	11.2
C4	28.8	27.5	33.1	45.5	32.2

507 of an ASR system can be improved with no additional computational cost.
 508 These results also suggest that there is further room for improvement over
 509 the psychoacoustic scaled filterbank.

510 In future work, the utilisation of other search methods, such as particle
 511 swarm optimisation and scatter search will be studied. Different variation
 512 operators can also be considered as a way to improve the results of the
 513 EA. Moreover, the search for an optimal filterbank could be carried out by
 514 evolving different parameters. The possibility of replacing the HMM based
 515 classifier by another objective function, in order to reduce computational
 516 cost, will also be studied. In particular, we will consider fitness functions
 517 which incorporate information such as the gaussianity and the correlation of
 518 the coefficients, as well as the class separability.

519 References

- 520 [1] L. Rabiner, B.-H. Juang, Fundamentals of Speech Recognition, Prentice
 521 Hall PTR, 1993.

- 522 [2] S. V. Davis, P. Mermelstein, Comparison of parametric representations
523 for monosyllabic word recognition in continuously spoken sentences,
524 IEEE Transactions on Acoustics, Speech and Signal Processing 28 (1980)
525 57–366.
- 526 [3] B. Nasersharif, A. Akbari, SNR-dependent compression of enhanced Mel
527 sub-band energies for compensation of noise effects on MFCC features,
528 Pattern Recognition Letters 28 (11) (2007) 1320 – 1326, advances on
529 Pattern recognition for speech and audio processing.
- 530 [4] X. Zhou, Y. Fu, M. Liu, M. Hasegawa-Johnson, T. Huang, Robust Anal-
531 ysis and Weighting on MFCC Components for Speech Recognition and
532 Speaker Identification, in: Multimedia and Expo, 2007 IEEE Interna-
533 tional Conference on, 2007, pp. 188–191.
- 534 [5] H. Bõril and P. Fousek and P. Pollák, Data-Driven Design of Front-
535 End Filter Bank for Lombard Speech Recognition, in: Proc. of INTER-
536 SPEECH 2006 - ICSLP, Pittsburgh, Pennsylvania, 2006, pp. 381–384.
- 537 [6] Z. Wu, Z. Cao, Improved MFCC-Based Feature for Robust Speaker
538 Identification, Tsinghua Science & Technology 10 (2) (2005) 158 – 161.
- 539 [7] M. Slaney, Auditory Toolbox, Version 2, Technical Report 1998-010,
540 Interval Research Corporation, Apple Computer Inc. (1998).
- 541 [8] M. Skowronski, J. Harris, Increased MFCC filter bandwidth for noise-
542 robust phoneme recognition, Proceedings of the IEEE International
543 Conference on Acoustics, Speech, and Signal Processing (ICASSP) 1
544 (2002) 801–804.
- 545 [9] M. Skowronski, J. Harris, Improving the filter bank of a classic speech
546 feature extraction algorithm, in: Proceedings of the 2003 International
547 Symposium on Circuits and Systems (ISCAS), Vol. 4, 2003, pp. 281–284.
- 548 [10] H. Yeganeh, S. Ahadi, S. Mirrezaie, A. Ziaei, Weighting of Mel Sub-
549 bands Based on SNR/Entropy for Robust ASR, in: Signal Processing
550 and Information Technology, 2008. ISSPIT 2008. IEEE International
551 Symposium on, 2008, pp. 292–296.

- 552 [11] L. Burget, H. Heřmanský, Data Driven Design of Filter Bank for Speech
553 Recognition, in: Text, Speech and Dialogue, Lecture Notes in Computer
554 Science, Springer, 2001, pp. 299–304.
- 555 [12] C. Charbuillet, B. Gas, M. Chetouani, J. Zarader, Optimizing fea-
556 ture complementarity by evolution strategy: Application to automatic
557 speaker verification, *Speech Communication* 51 (9) (2009) 724 – 731,
558 special issue on non-linear and conventional speech processing.
- 559 [13] C. Charbuillet, B. Gas, M. Chetouani, J. Zarader, Multi Filter Bank
560 Approach for Speaker Verification Based on Genetic Algorithm, *Lecture
561 Notes in Computer Science*, 2007, pp. 105–113.
- 562 [14] L. Vignolo, D. Milone, H. Rufiner, E. Albornoz, Parallel implementation
563 for wavelet dictionary optimization applied to pattern recognition, in:
564 Proceedings of the 7th Argentine Symposium on Computing Technology,
565 Mendoza, Argentina, 2006.
- 566 [15] D. B. Fogel, *Evolutionary computation*, John Wiley and Sons, 2006.
- 567 [16] L. Vignolo, H. Rufiner, D. Milone, J. Goddard, Genetic optimization
568 of cepstrum filterbank for phoneme classification, in: Proceedings of
569 the Second International Conference on Bio-inspired Systems and Sig-
570 nal Processing (BIOSIGNALS 2009), INSTICC Press, Porto (Portugal),
571 2009, pp. 179–185.
- 572 [17] C. Gathercole, P. Ross, Dynamic training subset selection for supervised
573 learning in genetic programming, in: *Parallel Problem Solving from
574 Nature – PPSN III*, Lecture Notes in Computer Science, Springer, 1994,
575 pp. 312–321.
- 576 [18] T. Bäck, *Evolutionary algorithms in theory and practice: evolution
577 strategies, evolutionary programming, genetic algorithms*, Oxford Uni-
578 versity Press, Oxford, UK, 1996.
- 579 [19] T. Bäck, U. Hammel, H.-F. Schewfel, *Evolutionary computation: Com-
580 ments on history and current state*, *IEEE Trans. on Evolutionary Com-
581 putation* 1 (1) (1997) 3–17.
- 582 [20] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution
583 Programs*, Springer-Verlag, 1992.

- 584 [21] C. R. Jankowski, H. D. H. Vo, R. P. Lippmann, A comparison of signal
585 processing front ends for automatic word recognition, *IEEE Transactions*
586 *on Speech and Audio Processing* 4 (3) (1995) 251–266.
- 587 [22] J. R. Deller, J. G. Proakis, J. H. Hansen, *Discrete-Time Processing of*
588 *Speech Signals*, Macmillan Publishing, New York, 1993.
- 589 [23] J. S. Garofalo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett,
590 N. L. Dahlgren, DARPA TIMIT acoustic phonetic continuous speech
591 corpus CD-ROM, Tech. rep., U.S. Dept. of Commerce, NIST, Gaithers-
592 burg, MD (1993).
- 593 [24] K. N. Stevens, *Acoustic Phonetics*, Mit Press, 2000.
- 594 [25] A. E. Eiben, J. E. Smith, *Introduction to Evolutionary Computing*,
595 SpringerVerlag, 2003.
- 596 [26] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, P. Woodland,
597 HMM Toolkit, Cambridge University (2000).
598 URL <http://htk.eng.cam.ac.uk>
- 599 [27] F. Jelinek, *Statistical Methods for Speech Recognition*, MIT Press, Cam-
600 brige, Masachussets, 1999.
- 601 [28] X. D. Huang, Y. Ariki, M. A. Jack, *Hidden Markov Models for Speech*
602 *Recognition*, Edinburgh University Press, 1990.
- 603 [29] C. Wang, L. M. Hou, Y. Fang, Individual Dimension Gaussian Mix-
604 ture Model for Speaker Identification, in: *Advances in Biometric Person*
605 *Authentication*, 2005, pp. 172–179.
- 606 [30] O.-W. Kwon, T.-W. Lee, Phoneme recognition using ICA-based feature
607 extraction and transformation, *Signal Process.* 84 (6) (2004) 1005–1019.
- 608 [31] K. Demuynck, J. Duchateau, D. Van Compernelle, P. Wambacq, Im-
609 proved Feature Decorrelation for HMM-based Speech Recognition, in:
610 *Proceedings of the 5th International Conference on Spoken Language*
611 *Processing (ICSLP 98)*, Sydney, Australia, 1998.
- 612 [32] B.-T. Zhang, G. Veenker, Focused incremental learning for improved
613 generalization with reduced training sets, in: T. Kohonen (Ed.), *Proc.*

- 614 Int. Conf. Artificial Neural Networks, Vol. 1585, North-Holland, 1991,
615 pp. 227–232.
- 616 [33] B.-T. Zhang, D.-Y. Cho, Genetic Programming with Active Data Se-
617 lection, in: Lecture Notes in Computer Science, Vol. 1585, 1999, pp.
618 146–153.
- 619 [34] M. Skowronski, J. Harris, Exploiting independent filter bandwidth of
620 human factor cepstral coefficients in automatic speech recognition, The
621 Journal of the Acoustical Society of America 116 (3) (2004) 1774–1780.
- 622 [35] Y. Gong, Speech recognition in noisy environments: a survey, Speech
623 Commun. 16 (3) (1995) 261–291.
- 624 [36] G. M. Davis, Noise reduction in speech applications, CRC Press, 2002.
- 625 [37] D. P. W. Ellis, PLP and RASTA (and MFCC, and inversion) in Matlab,
626 online web resource (2005).
627 URL www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/



Figure 13: Leandro Daniel Vignolo.

628 **LEANDRO D. VIGNOLO** was born in San Genaro Norte (Santa Fe),
 629 Argentina, in 1981. In 2004 he joined the Laboratory for Signals and Compu-
 630 tational Intelligence, in the Department of Informatics, National University
 631 of Litoral (UNL), Argentina. He is a teaching assistant at UNL, and he
 632 received the Computer Engineer degree from UNL in 2006. He received a
 633 Scholarship from the Argentinean National Council of Scientific and Tech-
 634 nical Research, and he is currently pursuing the Ph.D. at the Faculty of
 635 Engineering and Water Sciences, UNL. His research interests include pat-
 636 tern recognition, signal processing, neural and evolutionary computing, with
 637 applications to speech recognition.

638 **HUGO L. RUFINER** was born in Buenos Aires, Argentina, in 1967.
 639 He received the Bioengineer degree (Hons.) from National University of
 640 Entre Ríos, in 1992, the M.Eng. degree (Hons.) from the Metropolitan
 641 Autonomous University, Mexico, in 1996 and the Dr.Eng. degree from the
 642 University of Buenos Aires in 2005. He is a Full Professor of the Department
 643 of Informatics, National University of Litoral and Adjunct Research Scientist
 644 at the National Council of Scientific and Technological Research. In 2006, he
 645 was awarded by the National Academy of Exact, Physical and Natural Sci-
 646 ences of Argentina. His research interests include signal processing, artificial
 647 intelligence and bioengineering.

648 **DIEGO H. MILONE** was born in Rufino (Santa Fe), Argentina, in
 649 1973. He received the Bioengineer degree (Hons.) from National University
 650 of Entre Rios, Argentina, in 1998, and the Ph.D. degree in Microelectronics
 651 and Computer Architectures from Granada University, Spain, in 2003. Cur-
 652 rently, he is Full Professor and Director of the Department of Informatics at



Figure 14: Hugo Leonardo Rufiner.



Figure 15: Diego Humberto Milone.



Figure 16: John C. Goddard.

653 National University of Litoral and Adjunct Research Scientist at the National
654 Council of Scientific and Technological Research. His research interests in-
655 clude statistical learning, pattern recognition, signal processing, neural and
656 evolutionary computing, with applications to speech recognition, computer
657 vision, biomedical signals and bioinformatics.

658 **JOHN C. GODDARD** received a B.Sc (1st Class Hons) from London
659 University and a Ph.D in Mathematics from the University of Cambridge. He
660 is a Professor in the Department of Electrical Engineering at the Universidad
661 Autónoma Metropolitana in Mexico City. His areas of interest include pat-
662 tern recognition and heuristic algorithms applied to optimization problems.