Journal of Computer Science 3 (8): 608-616, 2007
ISSN 1549-3636

Comparison of Speech Features on the Speech Recognition Task

Iosif Mporas, Todor Ganchev, Mihalis Siafarikas, Nikos Fakotakis
Department of Electrical and Computer Engineering, University of Patras
26500 Rion-Patras, Greece

Abstract: In the present work we overview some recently proposed discrete Fourier transform (DFT)- and discrete wavelet packet transform (DWPT)-based speech parameterization methods and evaluate their performance on the speech recognition task. Specifically, in order to assess the practical value of these less studied speech parameterization methods, we evaluate them in a common experimental setup and compare their performance against traditional techniques, such as the Mel-frequency cepstral coefficients (MFCC) and perceptual linear predictive (PLP) cepstral coefficients which presently dominate the speech recognition field. In particular, utilizing the well established TIMIT speech corpus and employing the Sphinx-III speech recognizer, we present comparative results of 8 different speech parameterization techniques.

Keywords: Speech parameterization, speech recognition, wavelet packets

INTRODUCTION

The contemporary speech recognition technology is based on the statistical analysis of speech performed through powerful pattern recognition techniques, such as the hidden Markov models (HMM)[1] and dynamic programming procedures, such as the Viterbi algorithm[2]. One problem that has not been solved yet with sufficient elegance is the speech parameterization, which has the task to present the information carried by the speech signal in a compact form, so that it can be efficiently utilized by the HMM classifier. Presently, it is well understood which of the speech properties the speech features need to preserve and which to suppress. Significant efforts have been made for devising transformations for post-processing of the speech feature vectors in order to reduce the effect of signal alteration due to adverse environmental conditions[3] or variability of speech related to differences in the vocal tract among different speakers[4-6]. However, there are numerous other practical difficulties[7] that render the accurate recognition of speech difficult. Ultimately, the task of designing speech features that would lead to reliable speech recognition has not been solved, yet.

The success of MFCC[8], combined with their robust and cost-effective computation, turned them into a standard choice in speech recognition applications. Later studies[9] have shown that the PLP features outperform MFCC in specific conditions, but generally no large gap in performance was observed between them. Other speech features, such as the Perceptual Linear Prediction Adaptive Component Weighting (ACW) cepstral coefficients[10], and various wavelet-based features, such as the SBC of Sarikaya and Hansen[11], WPF of Farooq and Datta[12], WPSR of Siafarikas et al.[13, 14], despite presenting reasonable solution for the same tasks, did not gain widespread practical use, due to their higher computational demands. Nowadays, many earlier computational limitations have been overcome, due to the significant performance boost of contemporary microprocessors. This opens possibilities for re-evaluation of the traditional solutions when speech features are selected for a specific application.

More often when new speech features are proposed they are contrasted either to MFCC or PLP and, rarely, to a larger number of other competitive parameters. Some exceptions are[13, 15, 16], etc, where the authors consider three or more of the previously mentioned speech features in addition to their proposed method. The lack of comparison to multiple known methods leads to a particular difficulty, which developers experience when they have to choose speech features for the needs of a speech recognizer. Usually, their first choice falls on the MFCC, since they are known to provide good performance and are straightforward to implement. The selection of alternative speech features is somehow complicated due to the lack of large-scale comparisons, especially as concerns the wavelet packets-based speech features. This raises the necessity

Corresponding Author: Iosif Mporas, Electrical and Computer Engineering Dept., University of Patras, GR-26500, Rion-Patras, Greece Tel: +302610996496, Fax: +302610997336

for a direct comparison of the traditional speech features against recent wavelet packet-based speech features in a common experimental setup, which is a time-demanding process.

In the present work, we employ the Sphinx-III speech recognizer[17] and the TIMIT speech database[18] to evaluate a large number of recent (DWPT)- and (DFT)-based speech parameterization approaches in a common experimental setup. We target at identifying the relative ranking of the evaluated alternative speech features and measuring the practical worth of replacing MFCC.

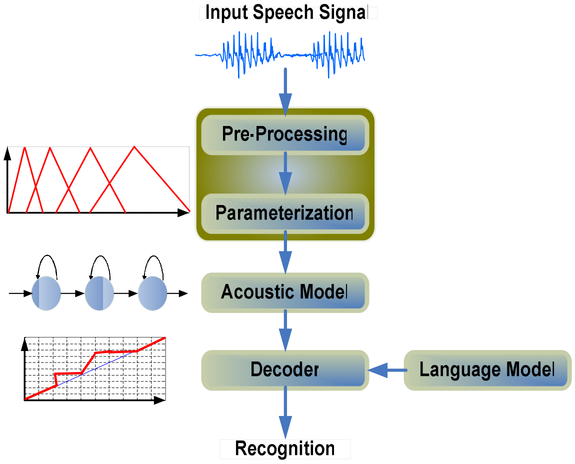

Fig. 1: Block diagram of the HMM-based speech recognition procedure

In the following section we offer a brief outline of the HMM-based speech recognition process. Next, we describe the speech parameterization schemes of interest, the experimental setup and then present comparative results for all feature extraction methods. This article ends with concluding remarks.

HMM-BASED SPEECH RECOGNITION

A speech recognition system operates in two modes: training (learning) and decoding (recognition). During learning it summarizes the characteristics of a set of sound units, e.g. phonemes, and builds models, which are afterwards utilized by defining the most probable sequence of sound units for a given speech sequence. During the recognition phase, these models are employed for decoding unknown input speech.

The state-of-the-art speech recognition technology is based on HMMs. HMMs are widely used in speech recognition because of their capability to model a sequence of discrete or continuous symbols. The speech signal can be approximated as a stationary signal in short-time intervals of about 10 milliseconds. Thus, speech is considered as a Markov model for many stochastic processes, known as states. The HMM tends to have, in each state, a statistical distribution called a mixture of Gaussians, which provides a likelihood for each observed vector. Each sound unit has a different output distribution. An HMM for a sequence of sound units is made by concatenating the separately trained hidden Markov models for the units. The most popular HMM used in speech recognition is the 3-state Bakis topology HMM with a non-emitting terminating state. In this topology it is assumed that each speech unit can be modelled by three distributions representing the beginning, middle and ending of it. The system can skip from state one to state three, bypassing state two completely. Such a topology implies that while the most general realization of the modelled sound has three distinct stages, some fulfilments may not have the middle stage.

A general structure of the decoding phase of an HMM-based system is illustrated in Fig. 1. As the figure presents, through pre-processing and parameterization steps the input speech is converted to a sequence of feature vectors, which is afterwards compared against an acoustic model, consisting of one HMM for every context-independent and context-dependent sound unit. The resulting acoustic score is combined with the score of a language model from the decoder. The language model consists of the probability of each word of a vocabulary to appear after a preceded word sequence. The sequence of words with the highest overall score is the recognized output.

SPEECH PARAMETERIZATION TECHNIQUES

Here, we consider the following relatively less studied speech parameterization techniques: SBC of Sarikaya & Hansen[11], WPF of Farooq & Datta[12], WPSR of Siafarikas et al.[13], OWPF of Siafarikas et al.[14] and HFCC-E of Skowronsky & Harris[16]. In addition, the well-known LFCC[8], MFCC[8] and PLP[9], whose performance is well studied, are employed as reference points. The last three speech features are well known and were used widely for speech recognition, while the HFCC-E scheme was recently proposed as a generalization of the MFCC that allows extended flexibility in the filter-bank design.

A general block diagram that summarizes the parameter estimation process for the speech parameterization methods under consideration is illustrated in Fig. 2. As the figure presents, the speech parameterization methods considered here share

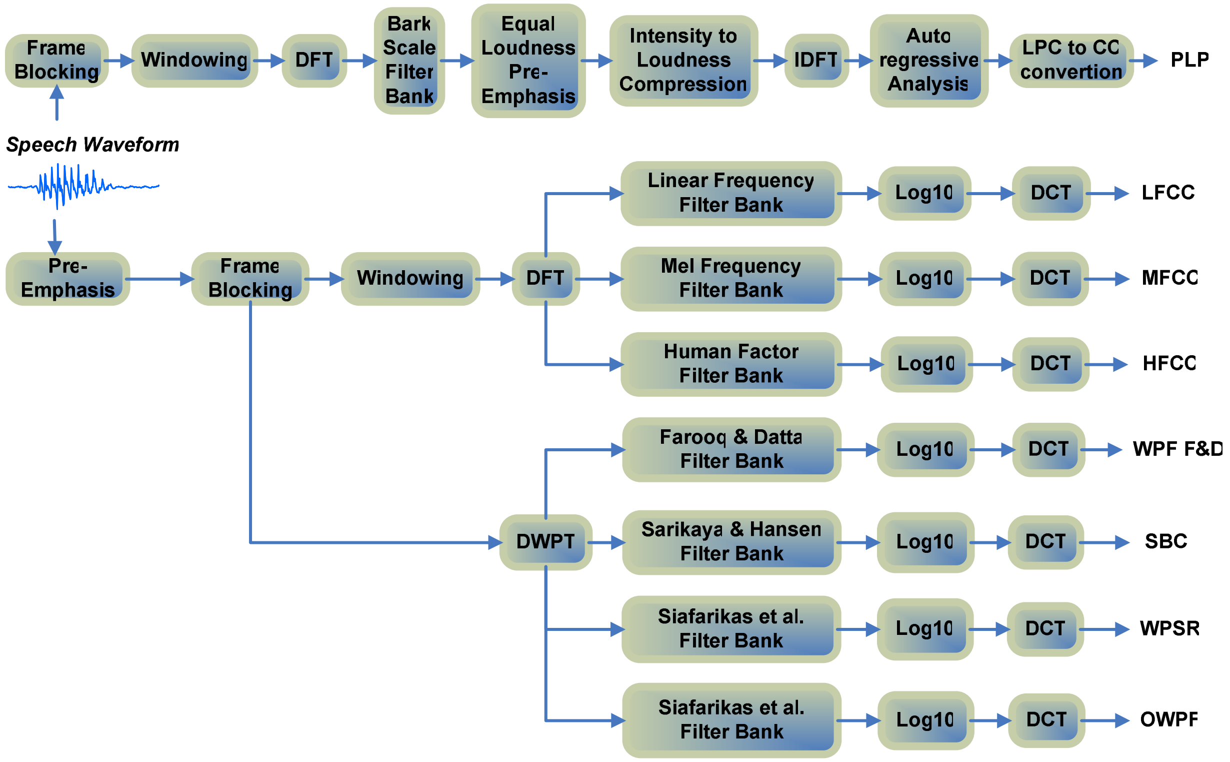

Fig. 2: Block diagram of extraction of the present evaluation's speech parameters

some common processing steps. Only the procedure of computing the PLP cepstral coefficients differs in a higher degree, when compared to the other speech parameterizations. In the following subsections we outline each method and describe how the originally proposed speech parameterizations were adjusted to unified frequency bandwidth and common settings.

DFT-BASED SPEECH PARAMETERIZATION

MFCC speech features (MFCC-FB40): Among various MFCC implementations discussed in the literature[16, 19], we rely on the one introduced by Slaney[20]. Due to its good performance, and to the fact that it is the default speech parameterization for the Sphinx-III speech recognizer, we consider it as a baseline in the comparative evaluation of speech parameterization methods.

In brief, assuming sampling frequency of 16 kHz, Slaney implemented a filter bank of 40 equal area filters, which cover the frequency range [133, 6855] Hz. The centre frequencies of the first 13 filters are linearly spaced in the range [200, 1000] Hz with a step of 66.67 Hz and the ones of the next 27 are logarithmically spaced in the range [1071, 6400] Hz with a step $logStep = 1.0711703$.

The MFCC computation starts with applying the N-point DFT:

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot \exp\left(\frac{-j2\pi nk}{N}\right), \; k = 0, \ldots, N-1 \qquad (1)$$

on the discrete input signal $x(n)$. Next, an equal area filter bank $H_i(k)$ is employed in the computation of the log-energy output:

$$X_i = \log_{10}\left(\sum_{k=0}^{N-1} |X(k)| \cdot H_i(k)\right), \; i = 1, \ldots, M. \qquad (2)$$

Finally, the DCT (3) provides the MFCC-FB40 parameters:

$$C_j = \sum_{i=1}^{M} X_i \cdot \cos\left(j \cdot (i - 1/2) \cdot \frac{\pi}{M}\right), \; j = 0, \ldots, J-1 \qquad (3)$$

where j is the serial index of the cepstral coefficient, M is the number of filters in the filter-bank and J is the number of MFCC that are needed. For speech recognition $J = 13$ is a widely accepted value.

For the purpose of fair comparison, we accepted the frequency range of MFCC [133, 6855] Hz as binding for all other speech parameterization methods under consideration in this study.

LFCC speech features (LFCC-FB40): The LFCC[8] are computed following the methodology of the MFCC-FB40 (as described in the previous section) with the only difference that the Mel-frequency warping step is skipped. Thus, the desired frequency range is implemented by a filter-bank of 40 equal-width and equal-height linearly spaced filters. The bandwidth of each filter is 164 Hz, and the whole filter-bank covers the frequency range [133, 6857] Hz. Obviously, the equal bandwidth of all filters renders unnecessary the effort for normalization of the area under each filter.

Analytically, the computation of the LFCC speech features is performed as follows: The N- point DFT (1) is applied on the discrete time domain input signal $x(n)$. Next, the filter bank is applied on the magnitude spectrum $\left|X(k)\right|$ and the logarithmically compressed filter-bank outputs X_i are computed (2). Finally, the DCT (3) is applied on the filter-bank outputs (2) to obtain the LFCC FB-40 parameters. Analogically to the MFCC FB-40 we compute only the first $J = 13$ parameters.

PLP speech features (PLP-FB19): The PLP parameters[9] rely on Bark-spaced filter-bank of 18 filters for covering the frequency range [0, 5000] Hz. Specifically, the PLP coefficients are computed as follows: Firstly the discrete time domain input signal $x(n)$ is subject to the N- point DFT (1), and then the critical-band power spectrum is computed through discrete convolution

$$\theta(B_i) = \sum_{B=-1.3}^{2.5} \left|X(B - B_i)\right|^2 \Psi(B) \qquad (4)$$

of the power spectrum with the piece-wise approximation of the critical-band curve[9, 19], where B is the Bark warped frequency obtained through the Hertz-to-Bark conversion. Equal loudness pre-emphasis is applied on the down-sampled $\theta(B)$ and then intensity-loudness compression is performed. To the result obtained so far an inverse DFT is performed to obtain the equivalent autocorrelation function. Finally, the PLP coefficients are computed after autoregressive modelling and conversion of the autoregressive coefficients to cepstral coefficients.

Here this filter-bank was adapted to the desired frequency range by discarding the first (lowest frequency) filter and all filters whose centre frequencies reside beyond 6855 Hz. This modification led to a filter-bank of 19 filters that cover the frequency range [100, 6400] Hz, which is the closest feasible implementation to the desired frequency range.

HFCC-E of Skowronsky & Harris: Skowronski & Harris[16] introduced the Human Factor Cepstral Coefficients (HFCC-E). In the HFCC-E scheme the filter bandwidth is decoupled from the filter spacing. This is in contrast to the earlier MFCC implementations[8, 20], where these were dependent variables. Another difference to the MFCC is that in HFCC-E the filter bandwidth is derived from the equivalent rectangular bandwidth (ERB), which is based on critical bands concept introduced by Moore and Glasberg[21] rather than on the Mel scale. Still, the centre frequency of the individual filters is computed by utilizing the Mel scale. Furthermore, in HFCC-E scheme the filter bandwidth is further scaled by a constant, which Skowronski and Harris labelled as E-factor. Larger values of the E-factor E={4, 5, 6} were reported[16] to contribute for improved noise robustness.

In brief, assuming sampling frequency of 12500 Hz Skowronski & Harris proposed the HFCC-E filter bank composed of 29 Bark-warped equal height filters, which cover the frequency range [0, 6250] Hz. The computation of the HFCC-E starts with N- point DFT (1) of the discrete input signal $x(n)$. Next, the HFCC-E filter-bank is applied on the magnitude spectrum $\left|X(k)\right|$ and the log-energy filter bank outputs are computed (2). Finally, the DCT (3) is applied to decorrelate the HFCC-E FB-29 parameters. For the purpose of fair comparison with the other speech parameterization schemes considered here, we compute only the first $J = 13$ cepstral coefficients.

In order to adapt the frequency range of the HFCC filter-bank to the one considered here we discarded the first two filters (these with lowest centre frequencies) and added a new one at the other end (highest centre frequency). This modification led to a filter-bank of 28 filters, which covers the frequency range [125, 6844] Hz. Furthermore in order to better understand the influence of the number of filters in the filter-bank on the speech recognition performance, we designed two other filter-banks that cover the same frequency range: (i) with 23 filters and (ii) with 40 filters. Their design was motivated by the MFCC implementations in HTK[22] and Slaney[20]. In the present work, we consider the E-factor $E = 1$ for all HFCC-E designs.

DWPT-BASED SPEECH PARAMETERIZATION

The SBC of Sarikaya & Hansen: Sarikaya & Hansen in[11] performed a wavelet packet decomposition of the frequency range [0, 4] kHz such that the 24 frequency subbands obtained follow the Mel scale for the task of stressed speech monophone recognition problem.

Following certain experimentations, they proposed a specific wavelet packet decomposition that provided the best overall result among a reasonable number of wavelet packet trees. The proposed analysis emphasizes low to mid frequencies assigning more subbands in these bands; overall, their decomposition preserves approximately a log-like distribution of the subbands across frequency. The wavelet packet decomposition is followed by the computation of the energy in each subband and the scaling by the number of transform coefficients in that subband. The corresponding subband signal energies for each frame are computed by the following relationship:

$$E_i = \frac{\sum_{m \in i} \left[\left(W_\psi x \right)(i), m \right]^2}{N_i}, \tag{5}$$

where $W_\psi x$ is the wavelet packet transform of signal x, i is the subband frequency index and N_i is the number of coefficients in the ith subband. Wavelet packet transform was implemented by using Daubechies' wavelet filter of order 32. The resulting speech features, which Sarikaya & Hansen named Subband Based Cepstral Coefficients (SBC), were derived with the application of DCT transformation on the subband energies:

$$SBC(j) = \sum_{i=1}^{M} \log E_i \cos\left(\frac{j(i-0.5)}{M} \pi \right)^2, \tag{6}$$

for $j = 1, ..., J$, where J is the number of SBC parameters and M is the total number of frequency bands.

To adjust the filter-bank of SBC to the desired frequency range we did the following two modifications: The initial two subbands were discarded and six new subbands with bandwidth of 500 Hz each were added at the end of the original frequency range. This kept the Mel-scale like frequency warping and led to the actual frequency range of [125, 7000] Hz that is covered by 28 frequency subbands, and that is the closest feasible implementation of the desired bandwidth.

The WPF of Farooq & Datta: Farooq and Datta[12] performed a wavelet packet decomposition of the frequency range [0, 8] kHz such that the obtained 24 frequency subbands closely follow the Mel scale for the task of phoneme recognition. Following their method, the phonemes were analyzed through 24 filters constituting a wavelet packet based filter-bank. Following the decomposition, the total energy E_p in each subband p was calculated as follows:

$$E_p = \sum_{j=1}^{N_p} \left(C_{j,p} \right)^2, \; p = 1, ..., M, \tag{7}$$

where $C_{j,p}$ is the jth coefficient in the pth subband, N_p is the number of wavelet packet coefficients in the pth subband and M is the number of subbands. The energies in each subband are further normalized with the number of wavelet packet coefficients in the corresponding subband as follows

$$F_p = E_p / N_p, \; p = 1, ..., M, \tag{8}$$

providing average energy per wavelet coefficients per subband F_p. The authors performed the wavelet packet decomposition using Daubechies' wavelet filter of order 12 in order to obtain features with emphasis on the lower frequency subbands. Subsequently, the normalized subband energies obtained at the output of the filter-bank were logarithmically compressed and subsequently decorrelated by applying the DCT. The feature set consisted of the first 13 coefficients of the resultant vector.

To adjust the filter-bank of WPF to the desired frequency range we discarded the first and the last subbands, which lead to 22 subbands that cover the range [125, 7000] Hz. This is the closest feasible fit to the desired frequency range.

WPSR of Siafarikas *et al.*: The wavelet packet features (WPSR) of Siafarikas et al.[13] were initially developed for the needs of speaker recognition, but here they are adapted to the speech recognition task. As discussed in the literature, wavelet packet analysis can be further enhanced and fine-tuned by carefully selecting a wavelet function (and consequently the corresponding wavelet and scaling filters) that is appropriate for the specific application in order to provide various time-frequency representations. The variety of existing wavelet families has been explored in order to augment the frequency localization abilities of the selected wavelet packet transform. The Battle-Lemarié polynomial spline wavelet of order 5 was found as the best choice for the basis function of the wavelet packet transform.

In contrast to the SBC and WPF F&D speech features, which are based on the Mel scale, the formulation of the WPSR wavelet packet features exploited the suitability of the various wavelet packet orthonormal transforms for the approximation of the psychoacoustic effect explained by the critical bands concept, which was introduced by Fletcher[23]. In their original design the authors used 66 filters to cover the frequency range [0, 4000] Hz. To adapt this filter-bank to the speech recognition task it was modified to have smoothly increasing frequency resolution as follows: resolution 31.25 Hz for the range [0, 1000] Hz, corresponding to 32 subbands; resolution 62.5 Hz for [1000, 2500] Hz, 24 subbands; resolution 125 Hz, for [2500, 4000] Hz, 12 subbands. These minor changes led to two extra subbands in the range [0, 4000] Hz, so that the totals become 68. The desired frequency range was implemented by discarding the first four subbands and adding a number of subbands at the end, in two different ways:

1) 23 new subbands with resolution 125 Hz each were added. A total of 87 subbands covering the frequency range [125, 6875] Hz was obtained.
2) 12 new subbands with resolution of 250 Hz each were added. This led to a total of 76 subbands covering the frequency range [125, 7000] Hz.

Consequently, these two versions of the WPSR differ only in the implementation of the upper part of the desired frequency range. In the following they are referred to as WPSR125 and WPSR250, respectively.

Overlapping WPF of Siafarikas *et al.*: Siafarikas et al.[14] introduced a generalization of the Wavelet Packet Transform refered to as Overlapping Wavelet Packet Transform (OWPT) that allows an effective utilization of specific frequency intervals of interest. Carefully selected basis vectors belonging to different levels of the OWPTs are grouped together in order to create an even larger collection of overlapping transforms. This is achieved by organizing all the OWPTs for levels $j = 0,\ldots,J$ into a tree structure, called Wavelet Packet (WP) tree.

Having constructed the WP tree, the coefficient vectors W_j^n can be collected together to form a set $S = \{W_j^n : j = 0,\ldots J, n = 0,\ldots 2^j - 1\}$, where each $W_j^n \in S$ is nominally associated with the frequency band I_j^n. Any subset $S_1 \subset S$ that provides a complete overlapping coverage of the interval $[0,1/2]$ with coefficient vectors W_j^n yields an OWPT. In this way, OWPT provides a flexible tiling of the time-frequency plane with various frequency resolutions in the corresponding time intervals along with emphasis in specific frequency subbands.

In[14] it has been reported that the following resolutions and overlapping areas provide the best speaker verification performance: resolution 31.25 Hz in the range [0, 1000] Hz; resolution 62.5 Hz for [875, 1500] Hz, [2000, 2625] Hz and [3000 3500] Hz; and resolution 125 Hz for [1500, 2000] Hz and [2375, 3000] Hz. Furthermore in order to adjust this WP tree to the frequency range of interest, we have excluded from the filterbank the subbands residing in the interval [0, 125] Hz and added extra frequency bands with resolution 125 Hz in the frequency range [4000, 6875] Hz. This resulted into a WP tree with a total of 92 frequency subbands, which cover the frequency range [125, 6875] Hz.

EXPERIMENTAL SETUP

The speech parameterizations of interest were evaluated on the TIMIT speech recognition corpus[18]. Its well-understood and widely-used experimental protocol facilitates the interpretation of results and provides the means for direct comparison to other speech features, which were not considered in the present work.

In brief, the audio material consists of single-channel 16-bit linear microphone recordings with sample rate of 16 kHz, representing 8 American-English dialects, subdivided into training and testing sets. The phonetic representation (pronunciation) of all the words in the TIMIT prompts was carried out with the exploitation of the lexicon provided with the database. In the lexicon, a phoneme set of 38 phonemes {*aa, ae, ah, ahr, aw, ay, b, ch, d, dh, eh, er, ey, f, g, hh, ih, iy, jh, k, l, m, n, ng, ow, oy, p, r, s, sh, t, th, uh, uw, v, w, y, z*} is utilized.

For each speech feature set, an acoustic model was trained in a two step procedure. Initially, an acoustic model was trained utilizing the speech material and the corresponding transcriptions. This acoustic model was force-aligned against the transcription of the training data in order to extract the phonetic representations of the words with multiple pronunciations. Next, the force-aligned transcriptions were used to train new acoustic models following the same procedure. Specifically, 13-dimensional feature vectors were utillized. For the speech parameterization methods that perform pre-emphasis in time domain, we used 1st order FIR filter with pre-emphasis factor equal to 0.97. Afterwards, the speech was segmented into frames of 16 or 25.625 milliseconds length with rate 100 frames per second. Furthermore, for the DFT-based features, each frame was weighted by a Hamming window. Due to the compact representation of the wavelets this is not necessary in the DWPT-based schemes. The delta and double-delta coefficients were appended to the base 13, which led to 39-dimentional feature vectors. No automatic gain control or variance normalization was applied.

The acoustic models used 3-state Bakis-topology HMMs with a non-emitting terminating state. One HMM model was built for each of the 38 monophones plus one model for silence. Means and variances of every model were initialized and, subsequently the *context-independent* (CI) phone models were trained through the Baum-Welch algorithm. During training the initial acoustic model, we carried out 15 iterations, while for the final one (which uses the force-aligned transcriptions) the number of iterations depended on the *convergence ratio*, i.e. on the likelihood of the current to the previous iteration ratio. The training process was terminated when the convergence ratio became less than 0.02. Next, *context-dependent* (CD) untied triphone models were trained for every triphone that had occurred at least 8 times in the training data. The CD models were initialized using the values of the parameters of the corresponding CI models and trained with the Baum-Welch algorithm. The training protocol was similar to the one used in the CI step. Subsequently, decision trees were built to determine which HMM states of all untied models present similarities to train global states (*senones*). In total 1000 senones were trained as derived from empirical rules[24]. Triphones that contributed to the training of a senone, also called tied-state, share that senone. At the next step, the decision trees were pruned, so as the number of leaves to become equal to the selected number of tied states (not including the CI states). Finally, the CD tied models were trained. The HMM states were modelled by a mixture of 8 Gaussian distributions by progressively splitting and retraining the models to 2, 4 and 8 Gaussians per state.

In the decoder, the HMM acoustic model is utilized in the recognition of the test speech data. The resultant acoustic scores are combined with weighted probabilities provided by a language model. Here, a tri-gram language model, built utilizing the CMU Language Model Toolkit[25] was used. We used all TIMIT sentences for the training of the language model.

The Sphinx-III decoder was set as follows: Beam selecting HMMs at each frame= 1.0e-55; Beam selecting word-final HMMs at each frame= 1.0e-55; Beam selecting HMMs transitioning to successors at each frame= 1.0e-55; Max # of histories to maintain at each frame= 120; Max # of active HMMs to maintain at each frame= 30000; Max # of distinct word exists to maintain at each frame= 25; Silence word probability =1.0; Language weight= 9.5. Settings that are not referred here have their default values, as specified in the Sphinx-III documentation[24].

EXPERIMENTAL RESULTS

All speech feature sets were processed in a uniform manner as described in the previous section. The MFCCs are considered as the baseline speech parameters. The word error rate (WER) and the sentence error rate (SER) in percentages are presented in Table 1. The total number of words in TIMIT test subset is 14553, and the number of sentences 1680. In all tables, the errors of word substitutions, deletions and insertions are designated as WS, WD and WI respectively. The number 16 in the brackets after the designation of the wavelet packet-based speech features denotes that these features in fact utilize only the first 16 milliseconds of the speech frame. This is forced by the requirement of the DWPT analysis that the number of input samples has to be exact power of 2.

Table 1. Results for window 25.625 milliseconds

Feature	WS	WD	WI	WER(%)	SER(%)
SBC (16)	597	194	117	6.2	21.3
WPSR125 (16)	596	212	113	6.3	21.8
OWPF (16)	586	221	120	6.4	22.1
WPSR250 (16)	592	218	128	6.5	21.6
WPF F&D (16)	619	207	161	6.8	22.9
LFCC-FB40	635	223	152	6.9	23.5
HFCC-FB23	799	162	231	8.2	27.3
HFCC-FB40	819	184	261	8.7	28.2
HFCC-FB28	844	157	266	8.7	28.9
PLP-FB19	868	150	295	9.0	29.4
MFCC-FB40	860	176	278	9.0	29.9

As presented in Table 1 all speech features evaluated here outperformed the baseline MFCCs. This was an expected outcome and confirms the results reported by the corresponding authors. However, from a practical point of view, it is more interesting to investigate the ordering and the actual improvement these speech features lead to when compared to the baseline. As the results show the lowest error rates were achieved for the SBC of Sarikaya & Hansen, followed by the WPSR125, OWPF and WPSR250 of Siafarikas et al., the WPF of Farooq & Datta, and afterwards by the DFT-based speech features. An interesting observation is that the LFCC-FB40, which uses a bank of equal-bandwidth filters with linear spacing of the central frequencies, outperformed the HFCC, PLP, and MFCC, which all possess frequency warping inspired by the human auditory system. The superior results for the DWPT-based speech features is due to: (i) the balanced time-frequency resolution these wavelet packet trees provide, when compared to the uniform frequency resolution of the DFT-based ones, and (ii) to the more suitable (for analysis of non-stationary speech signals) basis functions, which are more reasonable choice, when compared to the cosine functions.

Next, for the purpose of fair comparison, all experiments involving the DFT-based speech features were repeated for window size of 16 milliseconds, which corresponds to the effective frame size that the DWPT-derived speech features utilize. The results are presented in Table 2. As the table presents, the DWPT-derived speech features retained their superiority. With small exceptions in the ordering of the DFT-based speech features, the ranking remained the same as in Table 1.

Table 2. Results for window 16 milliseconds

Feature	WS	WD	WI	WER(%)	SER(%)
SBC	597	194	117	6.2	21.3
WPSR125	596	212	113	6.3	21.8
OWPF	586	221	120	6.4	22.1
WPSR250	592	218	128	6.5	21.6
WPF F&D	619	207	161	6.8	22.9
LFCC-FB40	635	223	152	6.9	23.5
HFCC-FB40	736	173	194	7.6	25.5
HFCC-FB28	759	176	183	7.7	26.0
HFCC-FB23	764	179	189	7.8	25.7
MFCC-FB40	733	167	247	7.9	27.1
PLP-FB19	868	150	295	9.0	29.4

In order to assess the statistical significance of obtained results, the T-test was performed for every pair of results (see Table 3). The grey cells in the table correspond to pairs, which are not statistically different, i.e. to pairs for which the T-test has produced absolute value smaller than the significance threshold 1.98.

Table 3. T-test for the 16 millisecond window results

T-test	WPSR125	OWPF	WPSR250	WPF F&D	LFCC-40	HFCC-40	HFCC-28	HFCC-23	MFCC-40	PLP-19
SBC	0.50	0.75	1.19	2.96	3.89	6.71	7.10	8.00	8.21	12.92
WPSR125		0.26	0.71	2.53	3.48	6.38	6.79	7.69	7.91	12.70
OWPF			0.45	2.31	3.26	6.20	6.61	7.51	7.74	12.57
WPSR250				1.88	2.82	5.80	6.22	7.10	7.35	12.20
WPF F&D					0.86	3.93	4.36	5.09	5.41	10.25
LFCC-40						3.18	3.64	4.34	4.69	9.64
HFCC-40							0.47	0.94	1.39	6.21
HFCC-28								0.44	0.90	5.69
HFCC-23									0.50	5.51
MFCC-40										4.91

As it can be seen from Table 3, the experimental results for some speech feature sets are not statistically different. In detail, the SBC of Sarikaya & Hansen and the WPSR125, OWPF and WPSR250 of Siafarikas et al. are not statistically different. Furthermore, the WPF of Farooq & Datta is statistically identical to the LFCC-FB40, as well as the HFCC features with the MFCCs.

Finally, summarizing the results presented in Table 2, we can see that the SBC speech features demonstrated relative reduction of the WER by more than 20% and 30%, when compared to the baseline MFCC and the PLP, respectively.

CONCLUSION

We would like to stress that the evaluation results presented here demonstrate that the widely-used Mel-frequency cepstral coefficients are not the most appropriate choice of parameters when maximization of the absolute speech recognition performance is desired. We deem that developers of speech recognizers will benefit considerably from the present work, since it could save duplication of efforts for implementing and comparing multiple speech features.

ACKNOWLEDGEMENTS

This work was supported by the MoveOn project (IST-2005-034753).

REFERENCES

1. Rabiner, L.R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *P. IEEE*, 77(2):257–286.
2. Forney, G.D. 1973. The Viterbi algorithm. *P. IEEE*, 61(3):268–278.
3. Gong, Y. 1995. Speech recognition in noisy environments: A survey. *Speech Commun.*, 16(3):261–291.
4. Welling, L., Kanthak, S., Ney, H. 1999. Improved methods for vocal tract normalization. *ICASSP'99*, vol. 2, pp.761–764.
5. Zhan, P., Westphal, M. 1997. Speaker normalization based on frequency warping. *ICASSP'97*, pp. 1039–1042.
6. Garau, G., Renals, S., Hain, T. 2005. Applying vocal tract length normalization to meeting recordings. *INTERSPEECH'05*, Lisbon, Portugal, pp. 265–268.
7. Deng, L., Huang, X. 2004. Challenges in adopting speech recognition. *Commun. ACM*, 47(1):69–75.

8. Davis, S.B., Mermelstein, P. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE T Acoust., Speech Signal P.*, 28(4):357–366.
9. Hermansky, H. 1990. Perceptual linear predictive (PLP) analysis for speech. *J. Acoust. Soc. Am.*, 87(4):1738–1752.
10. Assaleh, K.T., Mammone, R.J. 1994. Robust cepstral features for speaker identification. *ICASSP'94*, Adelaide, Australia. Vol.1, pp. 129–132.
11. Sarikaya, R., Hansen, J.H.L. 2000. High resolution speech feature parameterization for monophone-based stressed speech recognition. *IEEE Signal Proc. Let.*, 7(7):182–185.
12. Farooq, O., Datta, S. 2001. Mel filter-like admissible wavelet packet structure for speech recognition. *IEEE Signal Proc. Let.*, 8(7):196–198.
13. Siafarikas, M., Ganchev, T., Fakotakis, N. 2004. Wavelet packets based speaker verification. *Odyssey 2004*, Toledo, Spain. pp. 257–264.
14. Siafarikas, M., Ganchev, T., Fakotakis, N., Kokkinakis, G. 2005. Overlapping Wavelet Packet Features for Speaker Verification. *INTER-SPEECH'05*, Lisbon, Portugal, pp. 3121–3124.
15. Kim, D.S., Lee, S.Y., Kil, R.M. 1999. Auditory processing of speech signals for robust speech recognition in real-world noisy environments. *IEEE T Speech Audi P.*, 7(1):55–69.
16. Skowronski, M.D., Harris, J.G. 2004. Exploiting independent filter bandwidth of human factor cepstral coefficients in automatic speech recognition. *J. Acoust. Soc. Am.*, 116(3):1774–1780.
17. Lee, K.-F., Hon, H.-W., Reddy, R. 1990. An overview of the SPHINX speech recognition system. *IEEE T Acoust. Speech Signal P.*, 38(1):35–45.
18. Garofolo, J. 1998. Getting started with the DARPA-TIMIT CD-ROM: An acoustic phonetic continuous speech database. National Institute of Standards and Technology (NIST), Gaithersburgh, MD, USA.
19. Zheng, F., Zhang, G., Song, Z. 2001. Comparison of Different Implementations of MFCC, *J Comput. Sci. Technol.*, 16(6):582–589.
20. Slaney, M. 1998. Auditory toolbox. Version 2. Technical Report #1998-010, Interval Research Corporation.
21. Moore, B.C.J., Glasberg, B.R. 1983. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *J. Acoust. Soc. Am.*, 74(3):750–753.
22. Young, S.J., Odell, J., Ollason, D., Valtchev, V., Woodland, P. 1995. The HTK Book. Version 2.1, Department of Engineering, Cambridge University, UK.
23. Fletcher, H. 1940. Auditory patterns. *Rev. Mod. Phys.*, 12:47–65.
24. RobustGroup's Open Source Tutorial – Learning to use the CMU SPHINX Automatic Speech Recognition system. Available: http://www.speech.cs.cmu.edu/sphinx/tutorial.html
25. The CMU-Cambridge Statistical Language Modelling Toolkit, v2. Available: http://www.speech.cs.cmu.edu/SLM/toolkit_documentation.html