Frequency-Warped and Stabilized Time-Varying Cepstral Coefficients

Trond Skogstad, Torbjørn Svendsen

Department of Electronics and Telecommunications
Norwegian University of Science and Technology
Trondheim, Norway
tronds@iet.ntnu.no, torbjorn@iet.ntnu.no

Abstract

This paper presents a set of cepstral parameters based on time-varying linear prediction. The lattice filter structure is utilized to accommodate efficient stabilization of models and a Bark-like warped frequency scale. As the proposed cepstral features are based on non-stationary spectral analysis there is a potential for complementary information not captured in conventional features. In classification and recognition experiments, the proposed features are shown to improve performance when augmenting MFCCs.

Index Terms: cepstral coefficients, non-stationary analysis, time-varying linear prediction, speech features

1. Introduction

A vital first step in ASR is the parameterization of the speech signal. All prevailing front ends for ASR (e.g. MFCCs [1] and PLPs) are based on short-time Fourier analysis and thereby on an assumption of stationarity, the validity of which is debatable. While some phones (e.g. vowels) can be almost stationary for periods of over 50ms, others (e.g. stops) can have realizations with durations of less than 10ms and are probably at no time truly acoustically stable [2].

The feature extraction is usually done on a fixed scale, with the frame length parameter chosen as a certain tradeoff between validity and fidelity: The Fourier transform only makes sense as an estimate of the power spectral density if the analysis frame is short enough to be stationary. On the other hand, if the signal actually is stationary, a longer frame allows better frequency resolution and more reliable feature estimates.

The value of a description of the temporal dynamics of the speech signal has been demonstrated through the introduction of the ∆- and ∆∆-features in the MFCC parameterization. These are usually calculated by a simple line fitting around the current frame (usually some variant of \( ∆c_i(k) = \sum_{l=1}^{L} k(c_i(k + l) - c_i(k - l)) \)), and, dependent on the frame shift rate, contain little or no information from the current frame \( k \). A description of the temporal dynamics inside the current frame could therefore hold information complementary to the common MFCC parameterization, and would require either a frame subdivision and stationary analysis or a non-stationary analysis.

This paper proposes a new set of time-varying cepstral features, building on the framework of time-varying linear prediction (TVLP or TVAR) developed in [3] [4] [5] and extending our prior work in [6]. The TVLP models replace the usual assumption of stationarity with an assumption of an orderly evolution of the filter coefficients inside the analysis frame, expressible in a known set of basis functions. This non-stationary analysis provides a succinct description of the dynamics of the speech signal. In ASR, however, cepstral features representing the log-magnitude spectrum of the speech signal are usually preferred to the LP filter coefficients. In [6] we demonstrated that the basis function description of the TVLP coefficients could be brought through the recursion [7] to LP cepstral coefficients (LPCCs), while keeping a concise representation in terms of basis function weights. The cepstral basis function weights were also shown to be suitable as features in ASR. However, a satisfactory solution to the stability issues of TVLP models was not found.

In the current work, we focus on the related time-varying lattice filter models [4] and describe two new algorithms for stabilization of such models. Additionally, the lattice filter framework provides an opportunity to incorporate a Bark-like frequency warping, simulating the non-linear frequency sensitivity of human hearing.

2. Time-varying linear prediction

Linear predictive analysis has been a mainstay of speech technology for four decades, due to its role in the source-filter model of speech production and, by extension, its concise parametric representation of the spectral content of the speech signal. In the classical formulation of the problem, the signal \( x(n) \) is assumed to be stationary over the duration of the analysis window and modeled as a linear combination of the past \( P \) samples \( \hat{x}(n) = a_1x(n - 1) + a_2x(n - 2) + \cdots + a_Px(n - P) \).

Minimizing the prediction error power leads to the well-known Yule-Walker equations, which can be efficiently solved for the filter coefficients \( a_i \) by the Levinson-Durbin recursion. Most efforts of extending the model to the non-stationary case have been focused on approximating the time-evolution of the filter coefficients by a \( M \)-th order basis function expansion,

\[
a_i(n) = \sum_{j=1}^{M} a_{ij}g_j(n). \tag{1}
\]

This has the advantage of turning a linear non-stationary problem to a linear time-invariant problem by replacing a scalar process with a vector one [4]. This can be seen by forming the vectors \( X_n = [g_1(n)x(n) \ldots g_M(n)x(n)]^T \) and \( \theta = [a_{11} \ldots a_{1M} \ldots a_{21} \ldots a_{2M} \ldots a_P \ldots] \) and stating the predictor of the current sample \( x(n) \) as

\[
\hat{x}(n) = [X_{n-1}^T \ldots X_{n-P}^T] \theta. \tag{2}
\]

Similar to the stationary case, using the predictor (2) and minimizing the prediction error leads to a set of Yule-Walker-like equations. Several families of basis functions \( g_j(n) \) have been

Unlike stationary LP analysis, the filters resulting from TVLP analysis are not necessarily stable. This is especially problematic in the current setting, since the recursion from LP coefficients to LPCs (detailed in Section 3) assumes a stable all-pole model. In [8] a constrained minimization is performed, guaranteeing stability. However, this entails a considerable departure from the original statement as a linear estimation problem and comes at a high computational cost. This is a consequence of the minimization being done in the domain of the characteristic polynomial $A(z)$, while the stability constraints are stated in terms of the roots of the polynomial (no poles outside the unit circle), leading to an iterative algorithm with sequential linearization of the constraints. In [6] the problem was circumvented by having the back-end recognizer drop frames which were detected as unstable, a somewhat unsatisfactory solution which deteriorated the recognition rates.

2.1. Time-varying lattice filters

Linear prediction filters can be implemented in a lattice structure [9][10]. This lattice formulation also leads to an alternative algorithm for estimating the model parameters, originally suggested by Burg. The estimation is done in the domain of the reflection coefficients $k_i$, which appear in the Levinson-Durbin recursion. Burg’s method iterates over the model order $i \in \{1, \ldots, P\}$, at each step computing the reflection coefficients by minimizing the sum of the forward and backward prediction error,

$$J_f = \sum_{n=i}^{N} f_i^2 (n) + b_i^2 (n),$$  

leading to the maximum entropy estimate of the LP filter coefficients.

The lattice filter framework was extended to the non-stationary case in [4] and to the non-stationary frequency-warped case in [11]. Here, the reflection coefficients are expressed in a manner analogous to the filter coefficients of (1),

$$k_i (n) = \sum_{j=1}^{M} k_{ij} g_j (n).$$  

In the context of speech processing and specifically LP as a model of speech production, there is no obvious physical reason to prefer the model implied by (1) to (4).

Figure 1 illustrates the basic cell in the lattice filter structure, slightly modified to be suitable in the current setting. The transfer function $D(z)$ is a simple delay if a linear frequency scale is used and an allpass filter for a warped frequency scale.

![Figure 1: Time-varying lattice filter structure](image)

is a good approximation.

Assuming (5) and following the notation of Figure 1, the forward and backward prediction error signals can be stated as $f_i (n) = b_{i-1} (n) + f_{i-1} (n)$ and $b_i (n) = b_{i-1} (n) + k_i (n) f_{i-1} (n)$. Defining the vectors $K_i = [k_{i1} \ldots k_{iM}]^T$, $F_i (t) = [g_{i1} f_i (n) \ldots g_{iM} f_i (n)]^T$ and $B_i (n) = [g_{i1} b_i (n) \ldots g_{iM} b_i (n)]^T$, a set of correlation terms can be computed as:

$$\Phi_i = \sum_{n=1}^{N} B_{i-1} (n) B_i^T (n) + F_{i-1} (n) F_i^T (n)$$  

$$\Psi_i = \sum_{n=1}^{N} B_{i-1} (n) f_i (n) + F_{i-1} (n) b_i (n),$$  

where in the linear frequency case $B_i (n) = B_i (n - 1)$ (row filtering with a simple delay). In the frequency warped case, $B_i (n)$ is formed by row filtering $B_i (n)$ with the allpass filter $D_u (z)$ (defined in the Section 2.2).

The reflection coefficients minimizing (3) is found as:

$$K_i^* = \Phi_i^{-1} \Psi_i$$  

2.2. Frequency-warped time-varying lattice filters

Most prominently featured in the Mel-scale filter banks of MFCC, the use of a non-linear frequency scale mimicking properties of the human auditory system has been found to be beneficial for ASR. In linear prediction, frequency warping can be implemented by substituting the delay elements $D_u (z) = z^{-1}$ with an allpass filter,

$$D_u (z) = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}.$$  

With a sampling frequency of $f_s = 16$ kHz, $\lambda = 0.576$ will give a good fit to the Bark Scale [12].

While the magnitude response of $D_u (z)$ is constant by definition, the non-linear phase response determines a non-linear frequency mapping. For $\lambda > 0$, a greater relative emphasis will be put on modeling the signal spectrum in low frequency regions. This can be seen by considering the prediction error power of a warped filter [13],

$$\sigma^2 = \int_{-\pi}^{\pi} R(\omega) d\omega = \int_{-\pi}^{\pi} R(\omega) \frac{1 - \lambda^2}{1 - \lambda^2 - 2 \lambda \cos(\omega)} d\omega,$$  

where $R(\omega)$ is the residual spectrum and $\frac{1 - \lambda^2}{1 - \lambda^2 - 2 \lambda \cos(\omega)}$ a weighting factor indicating that the predictor is not minimizing the error signal power, but rather the power of the error signal.
ror signal filtered by the first order low pass filter \( W(z) = (1 - X^2)^{1/2}/(1 - \lambda z^{-1}) \).

A similar warping could be done by applying the bilinear transform [7] after calculating the cepstral coefficients. However, in this work the cepstral basis function weights are of interest, not the cepstral coefficients at each time instance, complicating matters. In addition, it is reasonable to expect that actually modeling the warped spectrum rather than modeling a linear spectrum and then transforming should utilize a given number of model parameters best.

2.3. Stabilization of lattice filters

Unlike the time-invariant case, the minimization of (3) will not necessarily lead to filters that are stable for all time-instances \( n \). Compared to a model based on a direct estimation of the time-varying LP parameters \( \theta \), the lattice filter framework offers a clear-cut advantage: The stability of the filters can readily be checked at each step \( i \) by the stability criterion

\[-1 \leq k_i(n) \leq 1, \quad n = 1, \ldots, N. \tag{11}\]

Further, if at any step an instability is detected, it is easier to stabilize the filters in this domain than in the domain of \( A(z) \).

In the following, two methods for stabilizing the filters will be proposed, one analytic, suitable for small \( M \) and most families of basis functions, and a second, more general, quadratic programming approach for larger \( M \) and all basis function families. Both methods will solve the minimization problem

\[ K^*_i = \arg \min_{K_i} ||\Phi_i K_i - \Psi_i||^2, \tag{12}\]

under two different sets of constraints. When both methods are applicable, they lead to the same results.

The stability criterion (11) is met if \(-1 < \min(k_i(n)) \) and \( \max(k_i(n)) < 1 \), and it is therefore sufficient to check the extremal values of \( k_i(n) \). It is easiest to work with time-continuous versions of the chosen basis and finding the extremal values through differentiation. As an example: For \( M = 2 \) and with an unsealed cosine basis \((g_1(t) = \cos(\pi t))\), it is sufficient to check the endpoints \( t = 0 \) and \( t = 1 \). This leads to four linear constraints,

\[-(1 - \epsilon) \leq k_{i1} + k_{i2} \leq 1 - \epsilon \tag{13} \]
\[-(1 - \epsilon) \leq k_{i1} - k_{i2} \leq 1 - \epsilon \tag{14}\]

which combined define a stable region in the space of basis function weights. A small number \( \epsilon \) should be subtracted to avoid marginally stable filters. The optimal (stable) \( K^*_i \) can then be found as the point on boundary of the stable region minimizing the correlation-weighted distance to the optimal (unstable) \( K^*_i \),

\[ ||\Phi_i K_i - \Psi_i||^2 = ||\Phi_i K_i - \Phi_i K^*_i||^2 = ||\Phi_i (K_i - K^*_i)||^2. \tag{15}\]

This can easily be solved analytically by parameterizing the violated constraint(s) in \( K_i \), and differentiating.

For \( M > 2 \), the stability constraints become non-linear in the basis function weights. A possible solution is to quantize the parametric stability surfaces and find the minimum of (15) through a simple search. The more general solution is to again consider the discrete versions of the basis functions and have two constraints for each time instance:

\[
\begin{bmatrix}
g_1 & \cdots & g_M \\
-g_1 & \cdots & -g_M
\end{bmatrix}
K_i \leq (1 - \epsilon)1, \tag{16}
\]

where \( g_j \) is the \( j \)-th basis function as a column vector \( g_j = [g_j(1), \ldots, g_j(N)]^T \), and \( 1 \) is the \( 2N \) column vector \( 1 = [1, \ldots, 1]^T \). By forming the matrix \( Q_i = 2\Phi_i^T \Psi_i \) and the vector \( c_i = -\Phi_i^T \Psi_i \) and rewriting (12) as

\[ K^*_i = \arg \min_{K_i} \frac{1}{2} K_i^T Q_i K_i + c_i^T K_i, \tag{17}\]

this can be recognized as a quadratic programming problem with inequality constraints, which can be efficiently computed by off-the-shell solvers.

3. Recursive computation of cepstral coefficients

Cepstral features are known to outperform LP filter coefficients and reflection coefficients in ASR [1]. In this section the transformation of the time-varying models of Section 2.1 to a time-varying cepstral representation

\[ c_i(n) = \sum_{j=1}^{M_i} c_{ij} g_j(n) \tag{18}\]

will be discussed. The cepstral basis function weights \( c_{ij} \) will be of special interest as possible features in ASR. Note that the number of basis functions \( M \) is dependent on the cepstral coefficient index \( i \). In the interest of brevity, the discussion in this section will be limited to cosine basis functions. Similar results will hold for e.g. polynomial bases.

In the time-invariant case, if the all-pole model

\[ H(z) = \frac{G}{1 - \sum_{i=1}^P a_i z^{-i}} \tag{19}\]

has all poles inside the unit circle, the first \( P \) LPCCs can be computed directly from the LP filter coefficients \( a_i \) by the recursion [7]:

\[ c_i = a_i + \sum_{k=1}^{i-1} \frac{k}{i} c_k a_{i-k}, \quad 0 < i \leq P. \tag{20}\]

The LP filter coefficients \( a_i \) are related to the reflection coefficients \( k_i \) by

\[ a_i = k_i, \quad i = 1, \ldots, P \tag{21}\]
\[ a_k = a_{i-k} - k_i a_{i-k}, \quad 1 \leq k < i \tag{22}\]

where \( a_i = a_i^T \).

In the time-varying case, these recursions hold true for each instance \( n \). However, by noticing that the key parts of each step in the recursions are the products \( c_k a_{i-k} \) and \( k_i a_{i-k}^{-1} \), and remembering the cosine product identity

\[ \cos(u) \cos(v) = \frac{1}{2} (\cos(u - v) + \cos(u + v)), \tag{23}\]

it is evident that at no point in the recursion is a non-cosine term introduced. In other words, reflection coefficients parameterized in a cosine basis (4) will lead to LP coefficients and cepstral coefficients parameterized in a cosine basis. Finding the basis weights \( c_{ij} \) of (18) is just a matter of, at each step in the recursions, grouping the terms according to frequency. The dimensionality of the parameterization will increase as a result of the introduction of new basis functions through (23), but as the variance of \( c_{ij} \) is rapidly decreasing with both increasing
and $j$, truncation can be done to reduce the parameter space without changing the time-trajectory of $c_i(n)$ significantly.

As the filter gain $G$ is not available in a parametric form, it is difficult to find an appropriate description of $c_i(n)$. A representation of the energy in the residual signal can be made by calculating $\log(j\pi(n))$ and projecting the resulting time series down to the cosine basis.

4. Experiments

A series of phone classification and continuous phone recognition experiments were performed to demonstrate the value of the proposed time-varying cepstral coefficients. All experiments were conducted on the phonetically labeled TIMIT database. The standard NIST 462 speaker training set, 24 speaker core test set and 50 speaker development set were used. Following common practice [14], the 61 phonetic labels were folded into 48 phones before training and further folded into 39 categories before evaluation. In the phone classification experiments the manual phonetic labeling was used to determine segment boundaries.

Large margin, single-state acoustic models (LM-GMM for classification, LM-GMM/HMM for recognition) were trained using the framework developed in [15]. In the interest of a fair comparison, Linear Discriminant Analysis was used to reduce all feature sets to 39 dimensions. Note that this was done even on feature sets of an initial dimensionality of 39 (including the baseline), as preliminary experiments indicated this gave slightly lower error rates, probably due to a decorrelation of the parameters.

The classification-only experiments were done mainly to illustrate the appropriateness of a warped frequency scale and of cepstral features $c_i$ (TVCC) over other LP-based features $k_{ij}$ (TVRC) and $a_{ij}$ (TVLPC). In all LP-based feature sets the prediction order was $P = 12$ and a cosine basis with $M = 3$ (TVLPCs and TVCCs truncated down) was used, with a frame length of 50ms and frame shift of 10ms. Three energy parameters were added, calculated as described at the end of Section 3. When warped (WTVCs), $\lambda = 0.576$. The features were stabilized using the quadratic programming approach.

We also performed recognition and classification experiments combining the WTVCs features with the 39 MFCCs. A common frame length of 50ms and frame shift of 10ms was chosen, hopefully getting the best of both methods: better estimates of MFCCs when in a stationary frame, combined with a good description of the dynamics of a non-stationary frame in the TVCCs. The space of system parameters was not exhausted and better configurations may exist. For comparison, the traditional feature set of 39 MFCCs (with the conventional choice of frame length, 25ms) was included as a baseline (BL).

Table 1 summarizes the error rates of the classification (C#) and recognition (R#) experiments, with $\#$ indicating the number of mixture components per phone. The tendency is clear: TVCCs is outperforming TVRCs which in turn is outperforming TVLPCs. Frequency warping is beneficial. There is a slight gap from the proposed features to the conventional MFCCs. The combination of WTVCs and MFCCs are consistently best over all experiments, demonstrating that the WTVCs contain information not captured in MFCCs.

5. Conclusions

A set of new time-varying cepstral features based on non-stationary LP analysis was proposed. The lattice filter frame

| Table 1: Results from phone segment classification and continuous phone recognition experiments. Error rates in percent. |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|            | C1          | C2          | C3          | R1          | R2          | R4          |
| TVRC        | 31.3        | 28.8        | 28.4        | -           | -           | -           |
| TVLPC       | 34.4        | 31.2        | 31.3        | -           | -           | -           |
| TVCC        | 30.1        | 28.7        | 27.7        | -           | -           | -           |
| WTVC        | 27.9        | 26.3        | 25.6        | -           | -           | -           |
| MFCC (BL)   | 25.4        | 24.6        | 23.9        | 32.2        | 31.2        | 30.7        |
| MFCC+WTVC   | 24.2        | 23.6        | 22.9        | 30.2        | 29.9        | 29.4        |

work was leveraged to accomplish efficient stabilization of models and a Bark-like frequency warping. In a series of classification and recognition experiments, frequency warped TVCCs outperformed other features based on time-varying linear prediction. Finally, frequency-warped TVCCs was shown to contain complementary information by successfully augmenting MFCCs.

6. References