CONTINUOUS FORMANT-TRACKING
APPLIED TO VISUAL REPRESENTATIONS OF THE SPEECH
AND SPEECH RECOGNITION

A. Álvarez, R. Martínez, V. Nieto, V. Rodellar and P. Gómez
Departamento de Arquitectura y Tecnología de Sistemas Informáticos
Universidad Politécnica de Madrid
Campus de Montegancedo, s/n, 28660 Boadilla del Monte, Madrid, Spain
Tel.: +34.1.336.73.84, Fax: +34.1.336.74.12, E-mail: pedro@pino.datsi.fi.upm.es

ABSTRACT
Through the present paper, a methodology to create Visual Representations of Speech for Speech Perception Enhancement Applications, based on the use of a Continuous Formant-Tracking Algorithm, is presented. The specific mathematical and computational issues introduced for such treatment are given, and a specific case for Computer-Aided Language Learning oriented to the Phonetic Specificities of English for Spanish Speakers is also presented. This specific technique may also be used in statistically normalizing Speech Data for Speech Recognition Systems. In this context, an example of a Robust to Noise Speech Recognizer, which uses Formant Dynamic Information is shown.

1. INTRODUCTION
Formant Dynamics is an interesting research field in Speech Perception, Speech Parametrization, Synthesis and Recognition. The paper contains a methodology to create Visual Representations of Speech for Speech Perception Enhancement Applications. Our approach uses the Gradient-Adaptive Lattice Algorithm [1][2] and a Continuous Formant-Tracker also presented in this paper.

The Gradient-Adaptive Lattice Algorithm is chosen, as it produces good peak spectra, which may be traced with relatively high accuracy. The algorithm models the vocal tract in which each filter stage represents one section of the tube and the forward and backward waves are also modeled [3].

This approach has several potential advantages over more conventional segmental systems. One example can be the noise. In a particular frequency band influences all cepstral or spectral coefficients. In this kind of situations, when the estimation of the formant is obscured, the position will be recovered by using consistency constraints with respect to the adjacent frames to estimate the formant location [4]. On the other hand, speaker normalization should be more feasible when formant frequencies are known explicitly [5].

Formants are the single most important source of evidence for the identifications of phonetic segments, as their relative positioning are primary features of different groups of sounds: vowel, liquids, glides and nasals. Formant transitions between vowels, provide also an useful information for the classification of fricatives and plosives [6].

2. GENERAL FRAMEWORK
To estimate the formant structure of a speech fragment, the method suggested in Figure 1 is used:

<table>
<thead>
<tr>
<th>Speech Trace u(t)</th>
<th>LPC parameters }</th>
<th>Spectrogram f[m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient-Adaptive Lattice</td>
<td>Levinson-Durbin Algorithm</td>
<td>Transfer Function Evaluation</td>
</tr>
<tr>
<td>Vector template</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Formant Tracking Algorithm</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. General Framework for the Formant Tracker.

The speech trace u(t) sampled at a given rate is LPC-extracted using a Gradient Adaptive Lattice to produce sets of PARCOR vectors h_n with a re-sampling index n each 5 msec. with a dimension of 16, 24 or 32, depending on the accuracy sought for Formant Extraction. The PARCOR vectors are transformed to LPC Parameters a_n using the Levinson-Durbin Algorithm, which establishes a relationship between the set of Inverse-Filter Parameters |a_n| and the associated set of PARCOR Parameters |h_n|:

\[ a_{in} = a_{in}^{-1} + h_{jn} a_{jn}^{-1} \quad 1 \leq i \leq j \quad 1 \leq j \leq k \] (1)

The components of the LPC vector |a_n| are the coefficients of an all-pole function, being K the dimensionality of the LPC vector:

\[ f_n(z) = \frac{1}{1 - \sum_{i=1}^{K} a_{in} z^{-i}} \] (2)

This function gives an approximation to the power spectrum of the speech trace in the LMS sense, and the LPC Spectrogram \( f(m) \) of the speech trace may be extracted by the evaluation of this function on the unity circle, this operation being done by the block Transfer Function Evaluation:

\[ f_n(m) = f_n(z = e^{in\Omega}) \quad 0 \leq m \leq M - 1 \] (3)

m being the frequency index, M the total number of frequency channels and \( \Omega \) the resolution in frequency. The process of Formant Extraction is related with the detection of the angular frequency associated with the poles of function (2).
1 - \sum_{i=1}^{K} a_i z_i^{-i} = 0; \quad z = z_{in}; \quad 1 \leq i \leq k

(4)
\varphi_i = \ln\{\ln z_{in}\}

(5)

In practice, the process of Formant Extraction is carried out by detecting the local maxima \( m \) for the instantaneous spectrum at a given \( n = n_k \).

2. FORMANT-TRACKER OPERATION

2.1. Introduction

The aim of the Continuous Formant Tracker is to obtain Formant Maps, which preserve their continuity and stability. To achieve this objective, a method for transforming Peak Representations of the Speech Trace (Fig. 7) into useful Formant Representations (Fig. 9) is used.

The starting hypothesis is to consider an extended concept of formant. Independently of the activity of the vocal chords, if we consider, following a tube model [2], that the vocal tract architecture existing in particular moment enhances several groups of frequencies from the spectrogram and lessens others, we will have different scenarios for every kind of sound produced. Taking into account that the movements of the vocal tract are quite slow compared with the air propagation speed, we can assure that maxima distribution in the spectrogram cannot change dramatically from one time slice to the next.

As a result of that, the Formant Tracker Algorithm must reward those representations in which formant positions for adjacent frames are quite near each other: Continuity Criterion. Also the differences between the formant positions calculated for 2 consecutive frames should be enclosed into a narrow band of frequencies: Stability Criterion.

2.2. Formant-Detection Steps

The Formant Detection Process comprises different stages, following the criteria described above as can be seen in Figure 2.

**Formant-Detecting Algorithm**

1. Initialize the search by taking those maxima \( M_x \) from the espectrogram in which for consecutive-in-time pairs of points, is achieved: \( M_{x_i} - M_{x_{i+1}} < \theta \), being \( \theta = 50Hz \).
2. Apply the transition rules to join 2 elementary tracks form the set calculated previously.
3. Apply the direct-union rules for 2 elementary tracks not matched in step 2.
4. Detect formant tracks shifted between 2 other ones.
5. Delete formant tracks of length less than 2 ms of time.

**Figure 2.** Formant-Detecting Algorithm.

Starting from the spectrogram produced for one speech trace as the one in Figure 6, the first step consists on picking up all the maxima points (peeks) for every time slot (Figure 7). After that, we choose all the frequency values belonging to 2 adjacent frames if they are near enough (Figure 8). At the end of this step we have a reduced set of points practically equal in terms of frequency. Each one of these sets constitutes an Elementary Track.

**Transition Rules**

1. Select 2 non-adjacent elementary tracks \( T_a, T_b \). The last element of \( T_a \) and the first element of \( T_b \) should accomplish that they are not separated in time more than \( t_b - t_a < 50 ms \).
2. Starting with the last element of \( T_a \) and finishing in the first element of \( T_b \), look for an elementary path with the following restrictions:
   - At least \( m > 2 \) intermediate points should be \( \maxima \) ones calculated in the first step of the Formant-Detecting Algorithm and they should not belong to a previous elementary track or path.
   - For 2 consecutive values in frequency of the set of points belonging to this elementary path, it is mandatory: \( f_{i+1} - f_i < \theta \), being \( \theta \) the time index and \( \theta = 175Hz \) if \( f_i \) and \( f_{i+1} < 2800Hz; \theta = 350Hz \) for higher frequencies.

**Figure 3.** Transition Rules for the Formant-Detecting Algorithm.

The step 2 has as a goal to detect the natural transitions between structures created in the step 1. In this case it is necessary to have at least \( n/2 \) points belonging to the set of peaks originally extracted. Also the separation in frequency among points of these sets should be not very large as can be seen in Figure 3.

**Direct-Union Rules**

1. Select 2 elementary tracks \( T_a, T_b \). The last element of \( T_a \) and the first element of \( T_b \) should accomplish that they are not separated in time more than \( t_b - t_a < 15 ms \).
2. Starting with the last element of \( T_a \) \( (t_i) \) and finishing in the first element of \( T_b \) \( (t_{i+1}) \), look for an elementary path if it is satisfied: \( f_{i+1} - f_i < \theta \), being \( \theta \) the threshold defined above. In this case:
   - Create all the necessary intermediate points if there are not enought \( \maxima \) ones calculated in the first step of the Formant-Detecting Algorithm.

**Figure 4.** Direct- Union Rules for the Formant-Detecting Algorithm.

The step 3 detects situations, which do not observe the rules defined in the previous step. If we have 2 elementary tracks quite near in time and frequency they will be joined (Figure 4). Once reached this point we have several Formant Tracks that accomplish the initial hypothesis.

**Shifted Formant Rules**

1. Select an elementary path \( T \) in such a way that it can be divided into 3 consecutive tracks \( T_a, T_c, T_b \), being \( T_a = T_1 - T_2 \), \( T_c = T_{i-1} - T_i \), \( T_b = T_i - T_{i+1} \). \( F \) indicates formant number \( i \). Also it is necessary that the last element of \( T_a \) \( (t_i) \) and the first element of \( T_b \) \( (t_{i+1}) \) not to be separated in frequency time more than \( \theta = 100 ms \).
2. In such a case, if there are 2 tracks \( T_a, T_b \) being \( T_a < \theta < T_b < \theta \), but it is not possible to find a track \( T_c \) with \( T_a < T_c < T_b \), we will have a formant candidate previously not detected.
3. Create the new elementary path, starting with the last element of \( T_a \) and finishing in the first element of \( T_b \) with the following restrictions:
   - For 2 consecutive values in frequency of the set of points belonging to this elementary path, it is mandatory: \( f_{i+1} - f_i < \theta \), being \( \theta \) the time index and \( \theta = 175Hz \) if \( f_i \) and \( f_{i+1} < 2800Hz; \theta = 350Hz \) for higher frequencies.

**Figure 5.** Shiffted Formant Rules for the Formant-Detecting Algorithm.
The step 4 (Fig 5), is useful because takes into account those situations in which a part of a formant has not been detected. The main idea of this procedure is to look for parts in the generated structures in which a formant line is cut in two pieces but it seems to be a single line.

The step 5 just deletes formant structures rather short, so that at the end of this stage we have the final formant estimation as shown in Figure 9.

The aim of the Continuous Formant Tracker is to obtain useful representations throughout Normalized Formant Maps. Considering $f_i$ as the Normalized Formant $i$, the expression that relates it with $\varphi$, Formant $i$, is:

$$f_i = \left( \frac{\varphi - \varphi_{i_{\text{max}}} + \varphi_{i_{\text{min}}}}{2} \right) \frac{2}{\varphi_{i_{\text{max}} - \varphi_{i_{\text{min}}}}}$$

(6)

where $\varphi_{i_{\text{max}}}$ corresponds to the maximum value of the first formant, which appears in /a/. The minimum value of such formant, $\varphi_{i_{\text{min}}}$ is associated to /il/. The minimum and maximum values of $\varphi_{i}$ correspond to /al/ and /il/, respectively.

The sounds of interest for our study comprise mainly vowels, glides and diphthongs, as these are difficult sounds to be perceived and produced by students of Foreign Languages, although in general, those sounds in which formant dynamics is determinant in their perception and discrimination [7] could also be studied using this technique.

The plots presented in Figure 10 have been produced using the technique being proposed, and represent a set of English diphthongs processed as an example.

The system may be used as a Microphonic Joystick, for Perception and Production Reinforcement in applications of Computer-Assisted Language Learning/Training (CALL-CALT) [9]. This is especially useful in Accent Reduction for non-native speakers studying English as a Foreign Language [10].

These ideas can also be applied to Speech Recognition. The aim of the IVORY ESPRIT project [11] is to develop a Robust-to-Noise Speech Recognizer. The inclusion of Normalized Formant Information is motivated as a way of increasing the robustness of the whole system against the noise.
One of the biggest problems with the noise, even when a Noise C cancellator is used, is that momentary spurious information may degrade dramatically the performances of the recognizer.

In our case, the Formant-Tracking Module calculate formants $F_1$-$F_4$, their first derivatives and also the associated energy for every formant frequency. The block diagram of the system is shown in Figure 12.

![Block diagram of the Isolated-Word Speech Recognition System designed for the IVORY Project.](image)

**Figure 11.** Block diagram of the Isolated-Word Speech Recognition System designed for the IVORY Project.

### 4. CONCLUSIONS

The Continuous Formant-Tracker requires neither a prior pre-segmenting of the speech into sonorant, obstruent and silence segments nor a division of the sonorant regions into subsegments. The algorithm works properly for the three situations and further criteria based on the Formant Tracker results may be used to classify the regions, if required.

![General outlook of the Speech Visualizing Interface. On the left hand screen a visual representation of the word /ship/ may be seen.](image)

**Figure 12.** General outlook of the Speech Visualizing Interface. On the left hand screen a visual representation of the word /ship/ may be seen.

The technique being proposed is highly efficient in producing meaningful representations of different Speech Traces. A MS WINDOWS Speech Visualizing Interface application, based on the General Framework described in Figure 1, has been produced. The application shown in Figure 12, is intended for Computer-Aided Language Learning, especially for supporting training in the specialities of English Phonetics.

On the other hand, the application of these ideas to Speech Recognition is also in progress in the frame of the IVORY ESPRIT project.

### 5. ACKNOWLEDGMENTS

This research is funded by ESPRIT Project IVORY (Integrated Voice Recognition System), no. 20277, PASO Project ALAS, nº 249, Grants TIC-96-1889-CE, TIC-95-1022. and project TER96-1938-C02-01.

### 6. REFERENCES