Deformable Spectrograms

Manuel Reyes-Gomez
LabROSA
Department of Electrical Engineering
Columbia University
mjr59@ee.columbia.edu

Nebojsa Jojic
Microsoft Research
One Microsoft Way
Redmond,WA.
jojic@microsoft.com

Daniel P.W. Ellis
LabROSA
Department of Electrical Engineering
Columbia University
dpwe@ee.columbia.edu

Abstract

Speech and other natural sounds show high temporal correlation and smooth spectral evolution punctuated by a few, irregular and abrupt changes. In a conventional Hidden Markov Model (HMM), such structure is represented weakly and indirectly through transitions between explicit states representing 'steps' along such smooth changes. It would be more efficient and informative to model successive spectra as *transformations* of their immediate predecessors, and we present a model which focuses on local deformations of adjacent bins in a time-frequency surface to explain an observed sound, using explicit representation only for those bins that cannot be predicted from their context. We further decompose the log-spectrum into two additive layers, which are able to separately explain and model the evolution of the harmonic excitation, and formant filtering of speech and similar sounds. Smooth deformations are modeled with hidden transformation variables in both layers, using Markov Random fields (MRFs) with overlapping subwindows as observations; inference is efficiently performed via loopy belief propagation. The model can fill-in deleted time-frequency cells without any signal model, and an entire signal can be compactly represented with a few specific states along with the deformation maps for both layers. We discuss several possible applications for this new model, including source separation.

1 Introduction

Hidden Markov Models (HMMs) work best when only a limited set of distinct states need to be modeled, as in the case of speech recognition where the models need only be able to discriminate between phone classes. When HMMs are used with the express purpose of accurately modeling the full detail of a rich signal such as speech, they require a large number of states. In [1](Roweis 2000), HMMs with 8,000 states were required to accurately represent one person's speech for a source separation task. The large state space is required because it attempts to capture every possible instance of the signal. If the state space is not large enough, the HMM will not be a good generative model since it will end up with a "blurry" set of states which represent an average of the features of different segments of the signal, and cannot be used in turn to "generate" the signal.

In many audio signals including speech and musical instruments, there is a high correlation between adjacent frames of their spectral representation. Our approach consists of exploiting this correlation so that explicit models are required only for those frames that cannot be accurately predicted from their context. In [2](Bilmes 1998), context is used to increase the modelling power of HMMs, while keeping a reasonable size parameters space, however the correlation between adjacent frames is not explicity modeled. Our model captures the general properties of such audio sources by modeling the evolution of their harmonic components. Using the common source-filter model for such signals, we devise a layered generative graphical model that describes these two components in separate layers: one for the excitation harmonics, and another for resonances such as vocal tract formants. This layered approach draws on successful applications in computer vision that use layers to account for different sources of variability [3, 4, 5](Jojic 2001,Levin 2002,Jojic 2003). Our approach explicitly models the self-similarity and dynamics of each layer by fitting the log-spectral representation of the signal in frame t with a set of transformations of the log-spectra in frame $t-1$. As a result, we do not require separate states for every possible spectral configuration, but only a limited set of "sharp" states that can still cover the full spectral variety of a source via such transformations. This approach is thus suitable for any time series data with high correlation between adjacent observations.

We will first introduce a model that captures the spectral de-

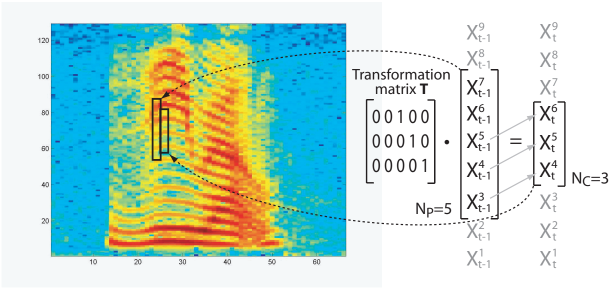


Figure 1: The $N_C = 3$ patch of time-frequency bins outlined in the spectrogram can be seen as an "upward" version of the marked $N_P = 5$ patch in the previous frame. This relationship can be described using the matrix shown.

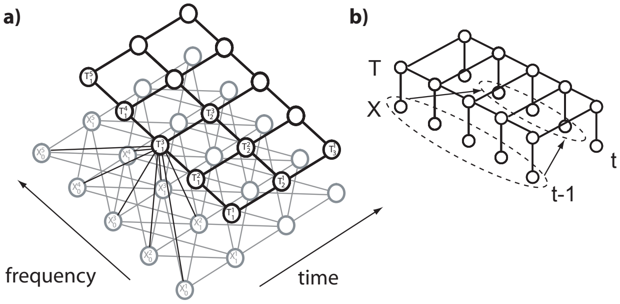


Figure 2: a) Graphical model b) Graphical simplification.

formation field of the speech harmonics, and show how this can be exploited to interpolate missing observations. Then, we introduce the two-layer model that separately models the deformation fields for harmonic and formant resonance components, and show that such a separation is necessary to accurately describe speech signals through examples of the missing data scenario with one and two layers. Then we will present the complete model including the two deformation fields and the "sharp" states. This model, with only a few states and both deformation fields, can accurately reconstruct the signal. This paper fully describes the operation and implementation of this complete model, which was only described as future work in [6](Reyes-Gomez 2004).

Finally, we briefly describe a range of existing applications including semi-supervised source separation, and discuss the model's possible application to unsupervised source separation.

2 Spectral Deformation Model

Figure 1 shows a narrow band spectrogram representation of a speech signal, where each column depicts the energy content across frequency in a short-time window, or time-frame. The value in each cell is actually the log-magnitude of the short-time Fourier transform; in decibels, $x_t^k = 20log(abs(\sum_{\tau=0}^{N_F-1} w[\tau]x[\tau - t \cdot H]e^{-j2\pi\tau k/N_F}))$, where t is the time-frame index, k indexes the frequency bands, N_F is the size of the discrete Fourier transform, H is the hop between successive time-frames, $w[\tau]$ is the N_F-point short-time window, and $x[\tau]$ is the original time-domain signal. We use 32 ms windows with 16 ms hops. Using the subscript C to designate current and P to indicate previous, the model predicts a patch of N_C time-frequency bins centered at the k^{th} frequency bin of frame t as a "transformation" of a patch of N_P bins around the k^{th} bin of frame $t-1$, i.e.

$$\vec{X}_t^{[k-n_C,k+n_C]} \approx \vec{T}_t^k \cdot \vec{X}_{t-1}^{[k-n_P,k+n_P]} \qquad (1)$$

where $n_C = (N_C - 1)/2$, $n_P = (N_P - 1)/2$, and T_t^k is the particular $N_C \times N_P$ transformation matrix employed at that point on the time-frequency plane. We use overlapping patches to enforce transformation consistency, [5](Jojic 2003).

Figure 1 shows an example with $N_C = 3$ and $N_P = 5$ to illustrate the intuition behind this approach. The selected patch in frame t can be seen as a close replica of an upward shift of part of the patch highlighted in frame $t-1$. This "upward" relationship can be captured by a transformation matrix such as the one shown in the figure. The patch in frame $t-1$ is larger than the patch in frame t to permit both upward and downward motions. The generative graphical model for a single layer is depicted in figure 2. Nodes $\mathcal{X} = \{X_1^1, X_1^2, ..., X_t^k, ..., X_T^K\}$ represent all the time-frequency bins in the spectrogram. For now, we consider the continuous nodes $\mathcal{X}$ as observed, although below we will allow some of them to be hidden when analyzing the missing data scenario. Discrete nodes $\mathcal{T} = \{T_1^1, T_1^2, ..., T_t^k, ..., T_T^K\}$ index the set of transformation matrices used to model the dynamics of the signal. Each $N_C \times N_P$ transformation matrix $\vec{T}$ is of the form:

$$\begin{pmatrix} \vec{w} & 0\,0 \\ 0 \;\; \vec{w} & 0 \\ 0\,0 & \vec{w} \end{pmatrix} \qquad (2)$$

i.e. each of the N_C cells at time t predicted by this matrix is based on the same transformation of cells from $t-1$, translated to retain the same relative relationship. Here, $N_C = 3$ and $\vec{w}$ is a row vector with length $N_W = N_P - 2$; using $\vec{w} = (0\ 0\ 1)$ yields the transformation matrix shown in figure 1. To ensure symmetry along the frequency axis, we constrain N_C, N_P and N_W to be odd. The complete set of $\vec{w}$ vectors include upward/downward shifts by whole bins as well as fractional shifts. An example set, containing each $\vec{w}$ vector as a row, is:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & .25 & .75 \\ 0 & 0 & 0 & .75 & .25 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & .25 & .75 & 0 \\ . & . & . & . & . \\ .75 & .25 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{3}$$

The length N_W of the transformation vectors defines the supporting coefficients from the previous frame $\vec{X}_{t-1}^{[k-n_W,k+n_W]}$ (where $n_W = (N_W - 1)/2$) that can "explain" X_t^k.

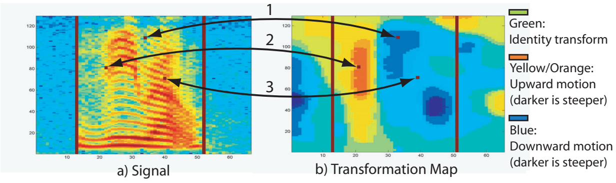


Figure 4: Example transformation map showing corresponding points on original signal.

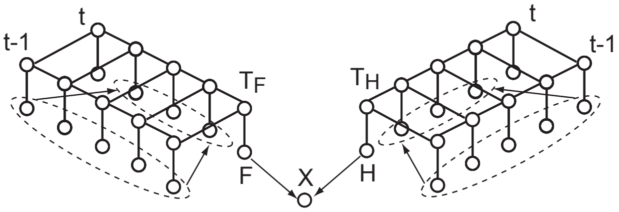


Figure 5: Graphical representation of the two-layer source-filter transformation model.

For harmonic signals in particular, we have found that a model using the above set of $\vec{w}$ vectors with parameters $N_W = 5$, $N_P = 9$ and $N_C = 5$ (which corresponds to a model with a transformation space of 13 different matrices T) is very successful at capturing the self-similarity and dynamics of the harmonic structure.

The transformations set could, of course, be learned, but in view of the results we have obtained with this predefined set, we defer the learning of the set to future work. The results presented in this paper are obtained using the **fixed** set of transformations described by matrix 3.

The clique "local-likelihood" potential between the time-frequency bin X_t^k, its relevant neighbors in frame t, its relevant neighbors in frame $t-1$, and its transformation node T_t^k has the following form:

$$\psi\left(\vec{X}_t^{[k-n_C,k+n_C]}, \vec{X}_{t-1}^{[k-n_P,k+n_P]}, T_t^k\right) = \mathcal{N}\left(\vec{X}_t^{[k-n_C,k+n_C]}; \vec{T}_t^k \vec{X}_{t-1}^{[k-n_P,k+n_P]}, \Sigma^{[k-n_C,k+n_C]}\right) \tag{4}$$

The diagonal matrix $\Sigma^{[k-n_C,k+n_C]}$, which is learned, has different values for each frequency band to account for the variability of noise across frequency bands. For the transformation cliques, the horizontal and vertical transition potentials $\psi_{hor}(T_t^k, T_{t-1}^k)$ and $\psi_{ver}(T_t^k, T_t^{k-1})$, are represented by transition matrices.

For observed nodes $\mathcal{X}$, inference consists in finding probabilities for each transformation index at each time-frequency bin. Exact inference is intractable and is approximated using Loopy Belief Propagation [7, 8] (Yedidia 2001,Weiss 2001) Appendix A gives a quick review of the loopy belief message passing rules, and Appendix B presents the specific update rules for this case.

The transformation map, a graphical representation of the *expected* transformation node across time-frequency, provides an appealing description of the harmonics' dynamics as can be observed in figure 4. In these panels, the links between three specific time-frequency bins and their corresponding transformations on the map are highlighted. Bin 1 is described by a steep downward transformation, while bin 3 also has a downward motion but is described by a less steep transformation, consistent with the dynamics visible in the spectrogram. Bin 2, on other hand, is described by a steep upwards transformation. These maps tend to be robust to noise (see fig 7), making them a valuable representation in their own right.

3 Inferring Missing Data

If a certain region of cells in the spectrogram are missing, like in the case of corrupted data, the corresponding nodes in the model become hidden. This is illustrated in figure 3, where a rectangular region in the center has been removed and tagged as missing. Inference of the missing values is performed again using belief propagation, the update equations are more complex since there is the need to deal with continuous messages, (Appendix C). The posteriors of the hidden continuous nodes are represented using Gaussian distributions, the missing sections on figure 3 part b), are filled in with the means of their inferred posteriors, figure 3 part c), and d). The transformation node posteriors for the missing region are also estimated, in the early stages on the "fill-in" procedure the transformation belief from the "missing" nodes are set to uniform so that the transformation posterior is driven only by the reliable observed neighbors, once the missing values have been filled in with some data, we enable the messages coming from those nodes.

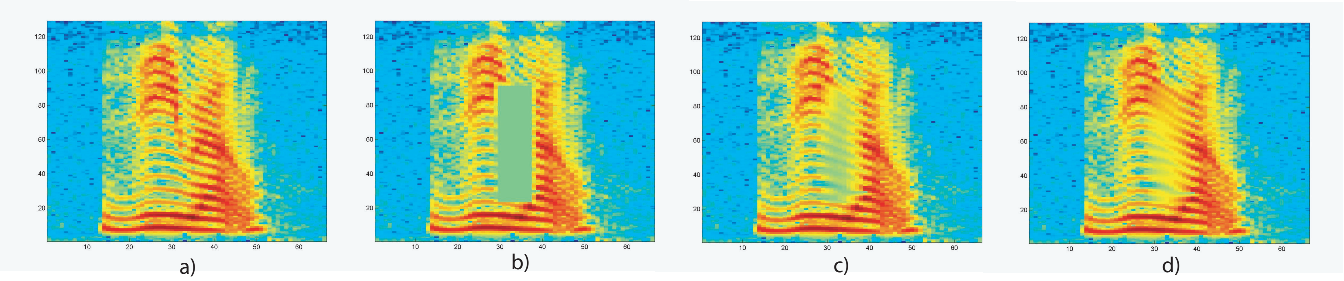
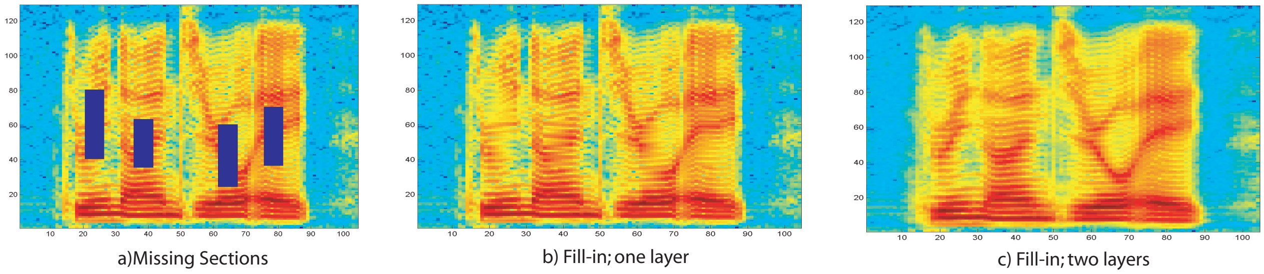


Figure 3: Missing data interpolation example a) Original, b) Incomplete, c) After 10 iterations, d) After 30.



a)Missing Sections

b) Fill-in; one layer

c) Fill-in; two layers

Figure 6: (a) Spectrogram with deleted (missing) regions. (b) Filling in using a single-layer transformation model. (c) Results from the two-layer model.

4 Two Layer Source-Filter Transformations

Many sound sources, including voiced speech, can be successfully regarded as the convolution of a broad-band *source excitation*, such as the pseudo-periodic glottal flow, and a time-varying resonant *filter*, such as the vocal tract, that 'colors' the excitation to produce speech sounds or other distinctions. When the excitation has a spectrum consisting of well-defined harmonics, the overall spectrum is in essence the resonant frequency response sampled at the frequencies of the harmonics, since convolution of the source with the filter in the time domain corresponds to multiplying their spectra in the Fourier domain, or adding in the log-spectral domain. Hence, we model the log-spectra X as the sum of variables F and H, which explicitly model the formants and the harmonics of the speech signal. The source-filter transformation model is based on two additive layers of the deformation model described above, as illustrated in figure 5. Variables F and H in the model are hidden, while, as before, X can be observed or hidden. The symmetry between the two layers is broken by using different parameters in each, chosen to suit the particular dynamics of each component. We use transformations with a larger support in the formant layer ($N_W = 9$) compared to the harmonics layer ($N_W = 5$). Since all harmonics tend to move in the same direction, we enforce smoother transformation maps on the harmonics layer by using potential transition matrices with higher self-loop probabilities. An example of the transformation map for the formant layer is shown in figure 7, which also illustrates how these maps can remain relatively invariant to high levels of signal corruption; belief propagation searches for a consistent dynamic structure within the signal, and since noise is less likely to have a well-organized structure, it is properties of the speech component that are extracted. Inference in this model is more complex, but the actual form of the continuous messages is essentially the same as in the one layer case (Appendix C), with the addition of the potential function relating the signal X_t^k with its transformation components H_t^k and F_t^k at each time-frequency bin:

$$\psi(X_t^k, H_t^k, F_t^k) = \mathcal{N}(X_t^k; H_t^k + F_t^k, \sigma^k) \qquad (5)$$

The first row of figure 10 shows the decomposition of a speech signal into harmonics and formants components, illustrated as the means of the posteriors of the continuous hidden variables in each layer. The decomposition is not perfect, since we separate the components in terms of differences in dynamics; this criteria becomes insufficient when both layers have similar motion. However, separation improves modeling precisely when each component has a different motion, and when the motions coincide, it is not really important in which layer the source is actually captured. Figure 6 a) shows the first spectrogram from figure 10 with deleted regions; notice that the two layers have distinctly different motions. In b) the regions have been filled via inference in a single-layer model; Notice that since the formant motion does not follow the harmonics, the formants are not captured in the reconstruction. In c) the two layers are first decomposed and then each layer is filled in; the figure shows the addition of the filled-in

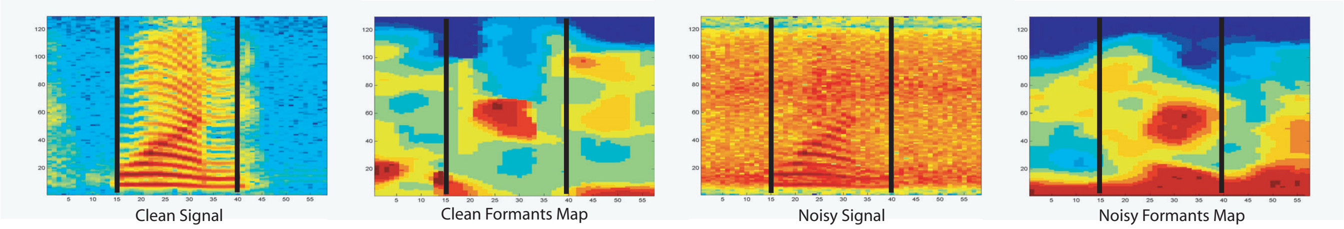


Figure 7: Formant tracking map for clean speech (left panels) and speech in noise (right panels).

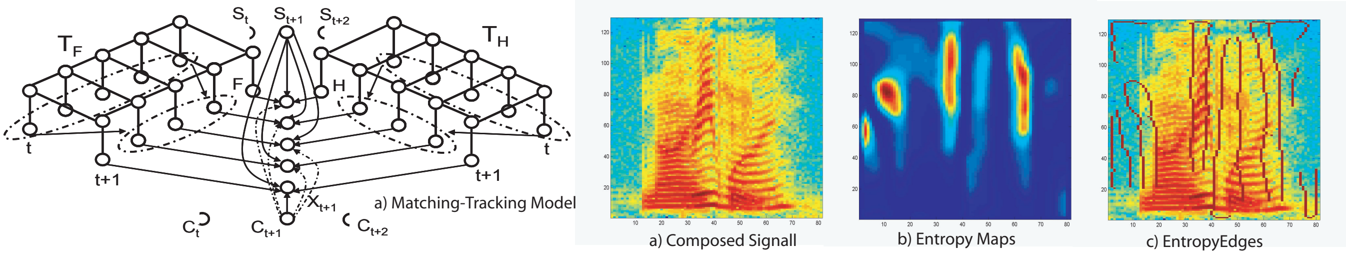


Figure 8: Left: Graphic model of the matching-tracking model; Right: Entropy Map and Entropy Edges

version in each layer.

5 Matching-Tracking Model

Prediction of frames from their context is not always possible such as when there are transitions between silence and speech or transitions between voiced and unvoiced speech, so we need a set of states to represent these unpredictable frames explicitly. We will also need a second "switch" variable that will decide when to "track" (transform) and when to "match" the observation with a state. The first row of figure 8 shows a graphical representation of this model. At each time frame, discrete variables S_t and C_t are connected to all frequency bins in that frame. S_t is a uniformly-weighted Gaussian Mixture Model containing the means and the variances of the states to model. Variable C_t takes two values: When it is equal to 0, the model is in "tracking mode"; a value of 1 designates "matching mode". The potentials between observations x_t^k, harmonics and formants hidden nodes h_t^k and fk_t respectively, and variables S_t and C_t is given by:

$$\psi\left(x_t^k, h_t^k, f_t^k, S_t, C_t = 0\right) = \mathcal{N}\left(x_t^k; h_t^k + f_t^k, \sigma^k\right) \quad (6)$$

$$\psi\left(x_t^k, h_t^k, f_t^k, S_t = j, C_t = 1\right) = \mathcal{N}\left(x_t^k; \mu_j^k, \phi_j^k\right) \quad (7)$$

Inference is done again using loopy belief propagation. Defining ϕ as a diagonal matrix, the M-Step is given by:

$$\begin{aligned} \mu_j &= \frac{\sum_t (Q(S_t = j) Q(C_t = 0) X_t)}{\sum_t (Q(S_t = j) Q(C_t = 0))} \\ \sigma_k &= \frac{\sum_t (Q(C_t = 1)(x_t^k - (f_t^k + h_t^k)))^2}{\sum_t (Q(C_t = 1))} \\ \phi_j &= \frac{\sum_t (Q(S_t = j) Q(C_t = 0)(X_t - \mu_j))^2}{\sum_t (Q(S_t = j) Q(C_t = 0))} \end{aligned} \quad (8)$$

$Q(S_t)$ and $Q(C_t)$ are obtained using the belief propagation rules. $Q(C_t = 0)$ is large if eqn. 6 is larger than eqn. 7. In early iterations when the means are still quite random, eqn. 6 is quite large, making $Q(C_t = 0)$ large with the result that the explicit states are never used.

To prevent this we start the model with large variances ϕ and σ, which will result in non-zero values for $Q(C_t = 1)$, and hence the explicit states will be learned.

As we progress, we start to learn the variances by annealing the thresholds i.e. reducing them at each iteration. We start with a relatively large number of means, but this becomes much smaller once the variances are reduced; the lower-thresholds then control the number of states used in the model. The resulting states typically consist of single frames at discontinuities as intended. Figure 9 a) shows the frames chosen for a short speech segment, (the spectrogram on figure 3.), the signal can be regenerated from the model using the states and both estimated motion fields. The reconstruction is simply another instance of inferring missing values, except the motion fields are not reestimated since we have the true ones. Figure 9 shows several stages of the reconstruction.

6 Applications

We have built an interactive model that implements formant and harmonics tracking, missing data interpolation, formant/harmonics decomposition, and semi-supervised source separation of two speakers. Videos illustrating the use of this demo are available at: `http://www.ee.columbia.edu/~mjr59/def_spec.html`.

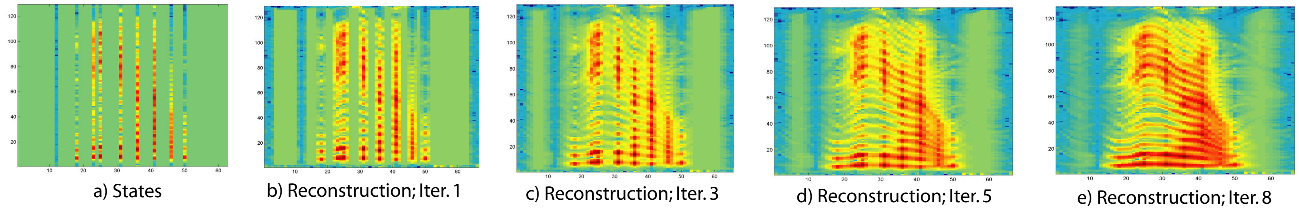

a) States b) Reconstruction; Iter. 1 c) Reconstruction; Iter. 3 d) Reconstruction; Iter. 5 e) Reconstruction; Iter. 8

Figure 9: Reconstruction from the matching-tracking representation, starting with just the explicitly-modeled states, then progressively filling in the transformed intermediate states.

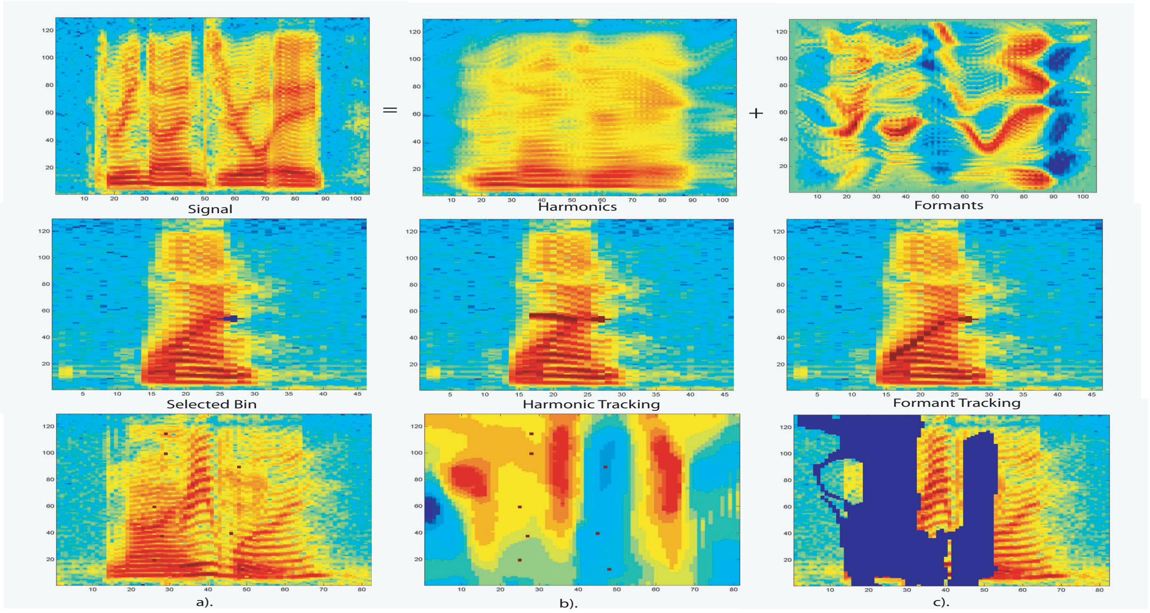


Figure 10: First row: Harmonics/Formants decomposition (posterior distribution means). Row 2: Harmonics/Formants tracking example. The transformation maps on both layers are used to track a given time-frequency bin. Row 3: Semi-supervised Two Speakers Separation. a) The user selects bins on the spectrogram that she believes correspond to one speaker. b) The system finds the corresponding bin on the transformation map. c) The system selects all bins whose transformations match the ones chosen; the remaining bins correspond to the other speaker.

Formants and Harmonics Tracking: Analyzing a signal with the two-layer model permits separate tracking of the harmonic and formant ‘ancestors’ of any given point. The user clicks on the spectrogram to select a bin, and the system reveals the harmonics and formant “history” of that bin, as illustrated in the second row of figure 10.

Semi-Supervised Source Separation: After modeling the input signal, the user clicks on time-frequency bins that appear to belong to a certain speaker. The demo then masks all neighboring bins with the same value in the transformation map; the remaining unmasked bins should belong to the other speaker. The third row of figure 10 depicts an example with the resultant mask and the “clicks” that generated it. Although far from perfect, the separation is good enough to perceive each speaker in relative isolation.

Missing Data Interpolation and Harmonics/Formants Separation: Examples of these have been shown above.

Features for Speech Recognition: The phonetic distinctions at the basis of speech recognition reflect vocal tract filtering of glottal excitation. In particular, the dynamics of formants (vocal tract resonances) are known to be powerful “information-bearing elements” in speech. We believe the formant transformation maps may be a robust discriminative feature to be use in conjunction with traditional features in speech recognition systems, particularly in noisy conditions; this is future work.

7 Potential Unsupervised Source Separation Applications

The right hand of figure 8 illustrates the *entropy* of the distributions inferred by the system for each transforma-

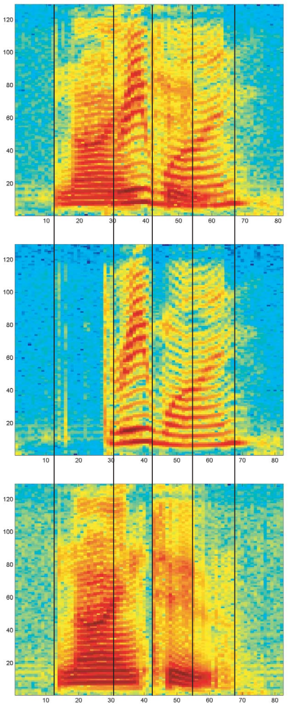


Figure 11: First pane shows the composed spectrogram, second and third spectrograms correspond to the individual sources, vertical lines correspond to the frames learned as states. Notice how the model captures the switches of dominant speaker.

tion variable on a composed signal. The third pane shows ‘entropy edges’, boundaries of high transformation uncertainty. With some exceptions, these boundaries correspond to transitions between silence and speech, or when occlusion between speakers starts or ends. Similar edges are also found at the transitions between voiced and unvoiced speech. High entropy at these points indicates that the model does not know what to track, and cannot find a good transformation to predict the following frames. These “transition” points are captured by the state variables when the Matching-Tracking model is applied to a composed signal, figure 11, the state nodes normally capture the first frame of the “new dominant” speaker. The source separation problem can be addressed as follows: When multiple speakers are present, each speaker will be modeled in its own layer, further divided into harmonics and formants layers. The idea is to reduce the transformation uncertainty at the onset of occlusions by continuing the tracking of the “old” speaker in one layer at the same time as estimating the initial state of the “new” speaker in another layer – a realization of the “old-plus-new” heuristic from psychoacoustics. This is part of our current research.

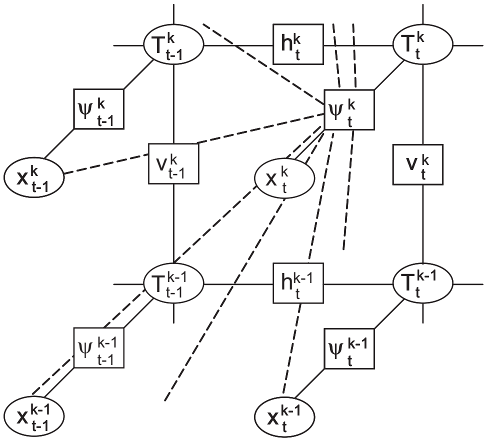


Figure 12: Factor Graph

8 Conclusions

We have presented a harmonic/formant separation and tracking model that effectively identifies the different factors underlying speech signals. We show that this model has a number of useful applications, several of which have already been implemented in a working real-time demo. The model we have proposed in this paper captures the details of a speech signal with only a few parameters, and is a promising candidate for sound separation systems that do not rely on extensive isolated-source training data.

9 Appendices

A: Loopy Belief Propagation

The sum-product algorithm [9](Kschischang 2001) can be used to approximate inference on graphical models with loops. The algorithm update rules applied to the factor graph representation of the model are:
Variable to local function:

$$m_{x\to f}(x) = \prod_{h\in n(x)\backslash f} m_{f\to x}(x) \tag{9}$$

Local function to variable:

$$m_{f\to x}(x) = \sum_{\sim x} f(X) \prod_{y\in n(f)\backslash x} m_{y\to f}(y) \tag{10}$$

where $X = n(f)$ is the set of arguments of the function f.

B: Update Rules for the Spectral Deformation Model

Figure 12 depicts a section of the factor graph representation of our model. Function nodes h^k_t and v^k_t represent respectively the potential cliques (transition matrices) $\psi_{hor}(T^k_t, T^k_{t-1})$ and $\psi_{ver}(T^k_t, T^{k-1}_t)$. Function node ψ^k_t, which represents the local likelihood potential defined in eq. 4, is connected to N_C “observation” variables in frame

t ($[x_t^{k-n_C}..x_t^{k+n_C}]$, n_C = (N_C - 1)/2) and to N_P “observation” variables in frame $t-1$.

When variables x_t^k are actually observed, only discrete messages between function nodes h_t^k, v_t^k and variable nodes T_t^k are required by the algorithm. Applying recursively the above update rules, we obtain the following forward recursion for the horizontal nodes on the grid:

$$m_{T_t^k \to h_t^k}(T_t^k) = (\sum_{T_{t-1}^k} h_t^k(T_t^k, T_{t-1}^k) m_{T_{t-1}^k \to h_{t-1}^k}(T_{t-1}^k)) \\ \psi(\vec{X}_t^{[k-n_C:k+n_C]}, \vec{X}_{t-1}^{[k-n_P:k+n_P]}, T_t^k) g(T_t^k) \quad (11)$$

where $g(T_t^k) = m_{v_t^k \to T_t^k}(T_t^k) m_{v_t^{k+1} \to T_t^k}(T_t^k)$. A similar backward recursion can also be found. The messages for the vertical chains can be updated through analogous upward/downward recursions.

C: Loopy Belief with Continuous-Valued Messages

The message from function node ψ_t^k to variable x_j^i has the form.

$$m_{\psi_t^k \to x_j^i}(x_j^i) = \\ \int_{\vec{y},\vec{z}} \frac{1}{C} exp^{\frac{1}{2}(\alpha x_j^i - \Gamma\vec{y} + \vec{z})' \Sigma^{-1}_{[r-n_C:r+n_C]} (\alpha x_j^i - \Gamma\vec{y} + \vec{z})} \\ \mathcal{N}(\vec{y}; \mu_y, \Sigma_y) \mathcal{N}(\vec{z}; \mu_z, \Sigma_z) d\vec{y} d\vec{z} \quad (12)$$

Where j is either $t-1$ or t and $i \in [k-n_P, k+n_P]$ if $j = t-1$ or $i \in [k-n_C, k+n_C]$ if $j = t$. Vector $\vec{y}$ is formed by the values on $X_{t-1}^{[r-n_P:r+n_P]}$ other than x_j^i if $j = t-1$ or the whole vector if $j = t$. Vectors $\vec{z}$ and $\vec{X}_t^{[r-N_C:r+N_C]}$ have an analogous relationship. Vector α and matrix Γ come from the most likely (or weighted mean) of the transformation matrix used at T_t^k.

Vectors $\vec{y}$ and $\vec{z}$ are obtained by concatenating individual variables x_r^s. Therefore $\mathcal{N}(\vec{y}; \mu_y, \Sigma_y)$ and $\mathcal{N}(\vec{z}; \mu_z, \Sigma_z)$ should be obtained by completing the square of the multiplication of the gaussian messages from the relevant individual variables x_r^s to the function node ψ_t^k. For simplicity and to speed up the process we approximate them instead by delta functions $\delta(\vec{y} - \mu_y)$ and $\delta(\vec{z} - \mu_z)$, where μ_y and μ_z are obtained as explain below. Then the messages reduce to: $m_{\psi_t^k \to x_j^i}(x_j^i) = \frac{1}{C} exp^{\frac{1}{2}(\alpha x_j^i - \Gamma\mu_y + \mu_z)' \Sigma^{-1} (\alpha x_j^i - \Gamma\mu_y + \mu_z)}$.

The posterior probability of node x_t^k, $q(x_t^k)$, is equal to the multiplication of all its incoming messages. We approximate this multiplication with a Gaussian distribution, $q'(x_t^k) = \mathcal{N}(x_t^k; \mu_{x_t^k}, \phi_{x_t^k})$. Minimizing their KL divergence we find:

$$\mu_{x_t^k} = \frac{\sum_{i=1}^{N_C+N_P} \alpha_i' \Sigma_i^{-1} (\Gamma_i \vec{y}_i - \vec{z}_i)}{\sum_{i=1}^{N_C+N_P} \alpha_i' \Sigma_i^{-1} \alpha_i^{-1}} \quad (13)$$

The values displayed by the missing data application are these mean values. The means of the variable to local function nodes messages, $m_{x_t^k \to \psi_j^i}(x_t^k)$, have the same form as in equation 13, just subtracting the numerator and denominator factor corresponding to the incoming message from the corresponding function. Since we use diagonal variances, parameters μ_y and μ_z in 12 are found by concatenating the means of the relevant messages $m_{x_t^k \to \psi_j^i}(x_t^k)$. When using the two layer model, an extra message comes from the other layer adding extra factors in the numerator and denominator of equation 13.

Acknowledgements

This work was supported by Microsoft Research and by the NSF under grant no. IIS-0238301. We want to thank Sumit Basu for insightful feedback on this research. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF.

References

[1] S. Roweis, “One-microphone source separation”, Advances in NIPS, MIT Press, 2000.

[2] J. Bilmes, “Data-driven extensions to HMM statistical dependencies”, Proc. ICSLP, 1998.

[3] N. Jojic and B. Frey, “Learning flexible sprites in video layers”, Proc. CVPR, 2001.

[4] A. Levin, A. Zomet, and Y. Weiss “Learning to perceive transparency from the statistics of natural scenes”, Proc. NIPS, 2002.

[5] N. Jojic, B. Frey, and A. Kannan, “Epitomic Analysis of Appearance and Shape”, Proc. ICCV, 2003.

[6] M. Reyes-Gomez, N. Jojic, and D. Ellis, “Towards single-channel unsupervised source separation of speech mixtures:The layered harmonics/formants separation-tracking model”, SAPA04. Korea 2004.

[7] J.S. Yedidia, W.T. Freeman, and Y. Weiss, “Understanding Belief Propagation and its Generalizations”, Exploring Artificial Intelligence in the New Millennium, Chapter 8.

[8] Y. Weiss and W.T. Freeman, “Correctness of Belief Propagation in Gaussian Graphical Models of Arbitrary Topology”, Neural Computation, V13, No 10, pp 2173-2200, 2001.

[9] F. Kschischang, B. Frey, and H.-A. Loeliger, “Factor Graphs and the Sum-Product Algorithm”, IEEE Transactions on information theory, Vol. 47 No. 2, 2001.