MULTI-INSTRUMENT MUSICAL TRANSCRIPTION USING A DYNAMIC GRAPHICAL MODEL

Brian K. Vogel[a,c]*, Michael I. Jordan*[a,b]*, David Wessel*[c]

[a] Dept. of Electrical Engineering and Computer Science, University of California, Berkeley, CA 94720
[b] Dept. of Statistics, University of California, Berkeley, CA 94720
[c] The Center for New Music and Audio Technologies, Berkeley, California 94709, USA

ABSTRACT

We present a dynamic graphical model (DGM) model for automated multi-instrument musical transcription. By multi-instrument transcription, we mean a system capable of listening to a recording in which two or more instruments are playing, and identifying both the notes that were played and the instruments that played them. Our transcription system models two musical instruments, each capable of playing at most one note at a time. We present results for two-instrument transcription on piano and violin sounds.

1. INTRODUCTION

In this paper, we present some results on the problem of multi-instrument musical transcription. By multi-instrument transcription, we mean a system capable of listening to a recording in which two or more instruments are playing, and identifying which instrument is playing which note. The general musical transcription problem involves listening to a musical piece and producing a reasonable musical score.

A robust automated musical transcription system would have applications to the music information retrieval (MIR) community, such as query by humming [1] for example. Good results on automated musical transcription may also carry over to the related problem of automated speech recognition.

The current state of the art in automated transcription systems is still far from a solution to the problem of robustly transcribing most interesting music. Even the seemingly simple task of transcribing a monophonic recording (a single instrument playing at most one note) becomes a hard problem once we consider rhythmically expressive performances. There has been some previous work on the audio-to-score problem (e.g., [2]). In this paper we focus on the simpler (but still hard) problem of going from an audio recording to a piano-roll like output (e.g., a MIDI file).

Several authors have recently reported progress on the difficult problem of transcribing music in which more than one note is sounding simultaneously (polyphonic transcription) [3] [4] [5]. There has been progress on the problem of instrument classification for monophonic recordings [6] [7]. Our work differs from these efforts in that it attempts to solve both problems simultaneously—we combine polyphonic transcription and instrument classification.

Our transcription system models a simple musically interesting example of multi-instrument polyphonic transcription consisting of two musical instruments, each capable of playing at most one note at a time. An example of a musical piece conforming to this model would be any of J.S. Bach's Two-part Inventions, where each part is performed on a different instrument.

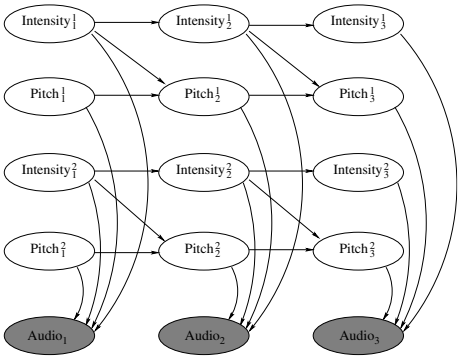

Fig. 1. A DGM for two-instrument transcription. Shaded nodes are observed data; unshaded nodes are "hidden" or "latent" variables.

We use a dynamic graphical model (DGM) (also known as a dynamic Bayesian network (DBN) [10]) to model note pitch and dynamic level envelopes for two monophonic musical instruments. A key feature of our model is the use of a note-event timbre model that includes both a spectral model and a dynamic intensity versus time model (i.e., a time envelope model). We present some results for two-instrument transcription of synthesized piano and violin sounds, using sampled acoustic instrument sounds for the synthesis.

2. MODEL

Figure 1 shows a DGM for a two-instrument transcription system. The hidden state variables $Intensity_t^m$ and $Pitch_t^m$ represent the instantaneous intensity (i.e., dynamic level) and pitch, respectively, of instrument m at time t. The observation variables $Audio_t$ represent the observed audio feature data at time t.

The top two hidden chains model the dynamic intensity and pitch evolution of one instrument (e.g., violin) while the lower two hidden chains model the dynamic intensity and pitch evolution of another instrument (e.g., piano). The hidden state variables correspond to the discrete set of allowable intensity and pitch values. We use the junction tree algorithm to compute the mode of the posterior distribution of intensity levels and notes (the Viterbi path) [9].

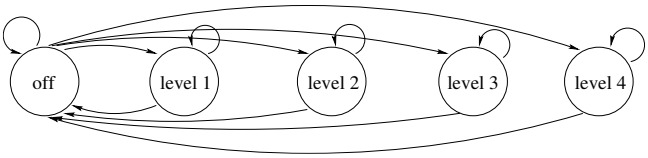

Fig. 2. A state transition diagram for an instrument characterized by a constant sound level after the note onset. Five discretized intensity levels are shown for clarity. Our actual implementation uses ten intensity levels. The "off" state denotes zero intensity.

3. INTENSITY ENVELOPE TRANSITION MODEL

The timbre of a musical instrument is influenced by both the spectral content, and the way in which the sound level changes over time. In the piano, the sound level immediately begins decaying after the initial hammer strike. This is also the case for plucked string instruments such as the guitar. However, in instruments where energy is continually supplied during the playing of a note, such as bowed string instruments, brass instruments, wind instruments and organs, the sound level fluctuates less during the playing of a note.

We propose two models of intensity envelopes for musical instruments. The "constant envelope model" is for instruments characterized by a steadier sound level after the note onset. The "decaying envelope model" is for instruments characterized by a gradual sound decay after the note onset.

The constant envelope model is shown in Figure 2. The state transition diagram in the figure comprises five discretized intensity levels, including the note-off state (zero intensity). Transitions from the note-off state to any nonzero intensity level are allowed. Self-loop transitions are allowed on all states. However, any outgoing transition from a nonzero intensity state must return to the note-off state. Thus, realizations of this state transition model will always result in note intensity envelopes consisting of a transition from the note-off state to some nonzero intensity level, followed by some number of self loops while the note sustains, followed by a transition back to the note-off state. This model defines a geometric distribution over note durations. Specifically, if the self-loop probability is p_{self}, then the probability that we remain at the same intensity for n time slices is $p_{self}(n) = (1 - p_{self})p_{self}^{n-1}$. Thus different expected note durations can be modeled by adjusting p_{self}.

The decaying envelope model is shown in Figure 3 for modeling the intensity envelope of instruments characterized by a decaying sound level after the note onset. In this model, transitions from the note-off state to any nonzero intensity level are allowed. Self-loop transitions are allowed on all states. However, any outgoing transition from a nonzero intensity state must lead to the next lower intensity state. Thus, realizations of this state transition model will always result in note intensity envelopes consisting of a transition from the note-off state to some nonzero intensity level, followed by some number of self loops, followed by a transition to the next lower intensity state, and so on, until the note-off state is reached.

4. PITCH TRANSITION MODEL

We place constraints on the times at which the pitch state can change. Specifically, pitch state change events within a single note envelope event should be disallowed. This is done by making the

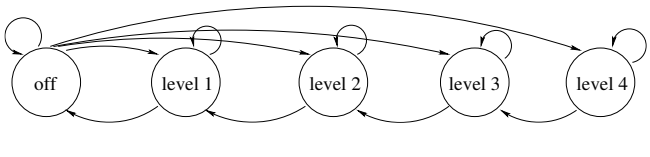

Fig. 3. A state transition diagram for an instrument characterized by a decaying sound level after the note onset. Five discretized intensity levels are shown for clarity. Our actual implementation uses ten intensity levels. The "off" state denotes zero intensity.

pitch state conditional probability distribution a function of both the previous pitch state and the previous intensity state. In our model, the pitch state is only allowed to change when the previous intensity state was the "note-off" state. In particular, the pitch transition model for instrument m is given by

$$P(Pitch_t^m = j | Intensity_{t-1}^m = k, Pitch_{t-1}^m = i) = \begin{cases} \delta(i,j) & \text{if } k > 0 \text{ (stay in the same state)} \\ T^m(i,j) & \text{if } k = 0 \text{ (pitch transition)} \end{cases}$$

where the $Intensity_{t-1}^m = 0$ state denotes the "note-off" state. $T^m(i,j)$ represents the pitch transition model for whole note events for instrument m. That is, $T^m(i,j) = P(Pitch_t^m = j | Pitch_{t-1}^m = i)$. We choose to make $T^m(i,j)$ instrument specific to reflect the fact that the set of allowable pitches can depend on the instrument. In our current implementation, we set $T^1(i,j) = T^2(i,j)$ so that instrument classification performance only depend on the intensity transition model and the observation model (and not on the pitch ranges of the instruments).

We use a method for specifying the state transition probabilities inspired by Shepard's notion of the pitch helix [8]. Perceived musical dissimilarity between two pitches is taken to be proportional to their Euclidean distance on the helix. We can easily construct a transition matrix based on the pitch helix as follows: Let $x(i)$ and $x(j)$ in $\mathbb{R}^3$ represent the locations of pitches i and j in the space in which the helix is embedded, where i and j can range over the K possible pitch values. For each i and j, we set the $(i,j)th$ element of a $K \times K$ matrix T equal to the following Gaussian kernel evaluated at $x(i)$ and $x(j)$:

$$T(i,j) = \exp(-\frac{1}{2}(x(i) - x(j))^T C^{-1}(x(i) - x(j)))$$

The rows of T are then normalized to make the matrix stochastic. C is diagonal, with two of the parameters tied to maintain rotational symmetry about the helix. So, the resulting transition matrix depends on two scalar parameters.

5. OBSERVATION MODEL

We model a time slice of the magnitude spectrogram as a series of narrow harmonically-spaced bump functions for each instrument, uniformly sampled in frequency at the values of the spectrogram frequency bins. Each harmonic bump is modeled as having a magnitude reflecting both its relative prominence with respect to the other harmonics and the dynamic level.

Our observation model is motivated by the following Gaussian process model. Consider one time slice $y_t(f)$ of the continuous spectrum. By continuous spectrum, we mean the spectrum obtained by replacing the DFT of the standard spectrogram with the

discrete time Fourier transform so that frequency is continuous-valued. We model the spectrum of a harmonic musical signal as a series of narrow bump functions that are harmonically spaced. That is, conditional on the fundamental frequency $Pitch_t$ of the musical signal, we model the spectrum as consisting of a series of bump functions located at integer multiples of $Pitch_t$. Each bump function is given a scale parameter $\alpha_n(Pitch_t)$ that can depend on $Pitch_t$. The motivation for this is that the relative spectral content of an instrument can depend on what pitch is being played. The intensity envelope at time t scales all of the harmonics. Our harmonic spectral model for a single instrument is the following:

$$y_t(f) = Instrument_t^1(f) + \xi(f)$$

$$Instrument_t^1(f) = Intensity_t^1 \sum_{n=1}^{H} \alpha_n(Pitch_t^1) b(f - nPitch_t^1)$$

where $\xi(f)$ is a zero mean Gaussian noise process and $b(f) = \exp(-f^2/\sigma)$. Our model for two instruments then becomes

$$y_t(f) = Instrument_t^1(f) + Instrument_t^2(f) + \xi(f).$$

A spectrogram time slice gives us $y_t(f)$ at the uniformly spaced frequencies $f_i, i = 1...N$ corresponding to the N spectrogram frequency bins. Conditional on the hidden state variables we have the following Gaussian observation model:

$$p(Audio_t | Intensity_t^1, Pitch_t^1, Intensity_t^2, Pitch_t^2)$$
$$= \mathcal{N}(y_t | \mu(Intensity_t^1, Pitch_t^1, Intensity_t^2, Pitch_t^2), \sigma_\xi^2 I),$$

where $Audio_t = [y_t(f_1), ..., y_t(f_N)]^T$, $\mu(Intensity_t^1, Pitch_t^1, Intensity_t^2, Pitch_t^2) = [\mu_1, ..., \mu_N]^T$ and

$$\mu_i = Instrument_t^1(f_i) + Instrument_t^2(f_i).$$

6. EXPERIMENTS

We present results for two audio clips of synthesized piano and violin sounds. We used a wavetable MIDI synthesizer so that the synthesized sounds actually consist of recorded piano and violin sounds. We take the truth score to be the MIDI file. Clip 1 consists of several piano and violin notes that have note durations greater than 0.5 seconds, while clip 2 (taken from Bach's two-part Invention #8) consists of note durations of about 0.13 seconds. The input sound clips can be heard at http://chaos.cnmat.berkeley.edu/transcription/sounds/. In this experiment, we use the constant envelope transition model from Figure 2 for instrument 1 (violin). We use the decaying envelope transition model from Figure 3 for instrument 2 (piano). Our expectation is that the decaying envelope model is closer to being a reasonable transition model for a piano than a violin. The intensity transition probabilities and the α_n harmonic magnitude parameters for each instrument are learned by an EM-based estimation procedure on a single-instrument version of the DGM in Figure 1. The harmonic width parameter σ^2 and the pitch transition parameters of C were chosen manually.

We used 10 intensity states for each instrument, discretized uniformly in log magnitude over a 60 dB dynamic range. The lowest intensity state corresponds to the note-off event. Any outgoing transition from the note-off event is interpreted as a note-on event. Performing inference on the DGM to compute the path of maximum posterior probability therefore gives us explicit note-on events. Note that thresholding or other post-processing would be needed to find the note onset events if we had used a continuous intensity state model. We used the same 12 allowable pitch states for each instrument, corresponding to the 12 semi-tones from A-flat below middle C to A-flat above middle C.

We defined a transcription error rate measure that is analogous to the word error rate used in automated speech recognition and other transcription systems. We measure the percentage *transcription error rate* as

$$100 \frac{\text{Insertions + Substitutions + Deletions}}{\text{Total Notes in Score}}$$

The insertions, substitutions, and deletions are computed separately for each instrument and then summed. Therefore, an instrument misclassification of a note onset event counts as one insertion and one deletion. Figure 4 shows the estimated note intensity envelopes for clip 1. Figure 5 shows the transcription results for clip 1. The transcription error rate was 12.5%. There were 8 notes total, with a single insertion error (on instrument 1 at around 3 seconds). The system was tested on several other short sound clips with generally similar results.

The use of an instrument-specific spectral model did not affect the results on clip 1. However, by using an instrument-specific spectral model and no time envelope model, the error rate was 50%. We have also found that placing constraints on the points in time at which pitch changes can occur significantly improves performance. Specifically consider the model that is obtained by removing the diagonal edges in Figure 1; this model is known as a factorial hidden Markov model (FHMM) [9]. In such a model, the pitch can change at any time. However, pitch changes within the time envelope of a single note should be disallowed since a time envelope represents the variation in sound level during the duration of a *single* note event. Indeed, the error rate for the simpler FHMM was 25%.

Clip 2 represents an example in which our results are the poorest. With an instrument-specific spectral model the transcription error rate for this clip was 93.75 %. There were 16 notes total, with 1 substitution, 14 deletions, and no insertions. The transcription error rate for clip 2 with a tied spectral model was 100 %.

Clip 2 differs from the other clips that we tested in having very short note durations. We believe that the poor performance on clip 2 reflects a poor choice of intensity envelope transition parameters for the short time scale, as the training data consists of notes of longer duration. The actual polyphony of clip 2 is also greater than the two voices modeled because of the fast tempo. We are currently experimenting with other intensity transition models that may better suited to modeling short note durations.

7. CONCLUSIONS

We have presented a DGM for automated multi-instrument musical transcription and presented results for synthesized piano and violin sounds. Multi-instrument polyphonic transcription is a challenging problem since it requires a suitable timbre model. A key feature of our model is a timbre model that includes both a spectral model and a time envelope model, yielding a combined approach to polyphonic transcription and instrument classification. Our model also has the feature that computing posterior modes for the DGM in Figure 1 yields explicit note-on events, as well as dynamic level versus time. If we had modeled the intensity state as being continuous-valued, some kind of post-processing would be required to estimate the note-on events.

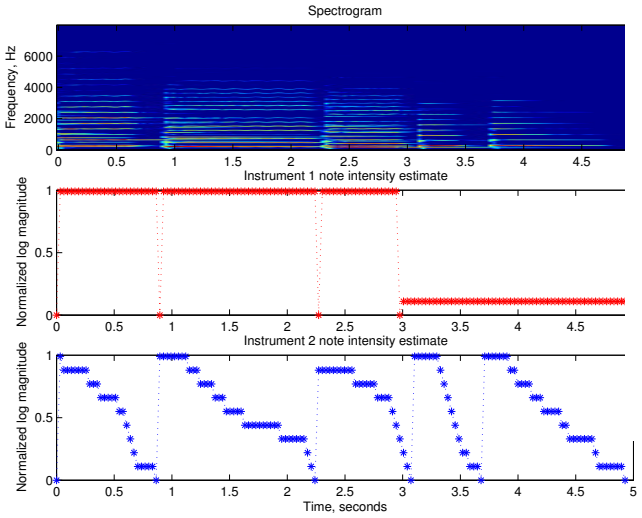

Fig. 4. Estimated note intensity envelopes for synthesized piano and violin sounds. Instrument 1 corresponds to the violin, and Instrument 2 corresponds to the piano. The spectrogram of the input audio file is shown at the top.

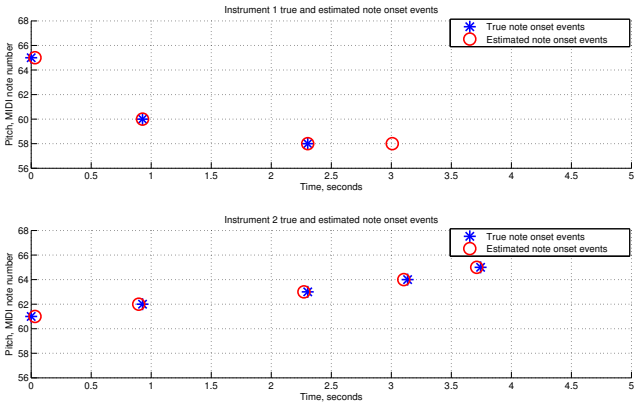

Fig. 5. Transcription results for synthesized piano and violin sounds. Instrument 1 corresponds to the violin, and Instrument 2 corresponds to the piano.

Our model extends immediately to the case of more than two instruments. However, the complexity for exact inference in a DGM is exponential in the number of hidden nodes in a time slice for our class of models. Specifically, for our DGM the time complexity is $O(TMK^{M+1})$ where T is the number of time slices, M is the number of hidden nodes per time slice, and K is the number of states for a hidden node. Thus, if we restrict ourselves to exact inference, the model is limited in practice to a small number of instruments. However, there is a large literature on algorithms for approximate posterior inference in large-scale dynamic graphical models [10]; these algorithms are directly applicable to our model.

8. ACKNOWLEDGMENTS

We wish to acknowledge the Jerry and Evelyn Hemmings Chambers Chair in Music for support of this research.

9. REFERENCES

[1] Nishimura, T., Hashiguchi, H., Takita, J., Zhang, J.X., Goto, M., and Oka, R., "Music signal spotting retrieval by a humming query using start frame feature dependent continuous dynamic programming," *Proc. ISMIR 2001*, pp.211-218, October 2001

[2] Cemgil, A.T., Desain, P., and Kappen, H.J., "Rhythm quantization for transcription," *Computer Music Journal* 24:2:60-76, 2000

[3] Raphael, C., "Automatic transcription of piano music," *Proc. ISMIR 2002*

[4] Goto, M., "A predominant-F0 estimation method for CD recordings: MAP estimation using EM algorithm for adaptive tone models," *Proc. ICASSP'01*, pp.V-3365-3368, May 2001

[5] Klapuri, A.P., "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness", *IEEE Transactions on Speech and Audio Processing, Vol. 11, No.6*,pp.804-816,2003

[6] Brown, J.C., Houix, O. & McAdams, S., "Feature dependance in the automatic identification of muisical woodwind instruments," *J. Acoust. Soc. Am.* 109, pp. 1064-1072, 2001

[7] Martin, K., *Sound-source recognition: A theory and computational model*. PhD Thesis, MIT, 1999

[8] Shepard, R., "Structural representations of musical pitch." In D. Deutsch, editor, *The Psychology of Music*, pages 344-385. Academic Press, New York, 1982

[9] Ghahramani, Z., Jordan, M.I., "Factorial hidden Markov models," *Machine Learning* 29, 1994

[10] Murphy, K., *Dynamic Bayesian networks: Representation, inference and learning.* PhD Thesis, Dept. Computer Science, UC Berkeley, 2002