PITCH DETECTION IN POLYPHONIC MUSIC USING INSTRUMENT TONE MODELS

Yipeng Li and DeLiang Wang

Department of Computer Science and Engineering
& Center for Cognitive Science
The Ohio State University
{liyip, dwang}@cse.ohio-state.edu

ABSTRACT

We propose a hidden Markov model (HMM) based system to detect the pitch of an instrument in polyphonic music using an instrument tone model. Our system calculates at every time frame the salience of a pitch hypothesis based on the magnitudes of harmonics associated with the hypothesis. A hypothesis selection method is introduced to choose pitch hypotheses with sufficiently high salience as pitch candidates. Then the system applies an instrument model to evaluate the likelihood of each candidate. The transition probability between successive pitch points is constructed using the prior knowledge of the musical key of the input. Finally an HMM integrates the instrument likelihood and the pitch transition probability. Quantitative evaluation shows the proposed system performs well for different instruments. We also compare a Gaussian mixture model and kernel density estimation for instrument modeling, and find that kernel density estimation gives better overall performance while the Gaussian mixture model is more robust.

Index Terms— Pitch detection, instrument modeling, hidden Markov model.

1. INTRODUCTION

A system capable of detecting the pitch of a particular instrument in polyphonic music is useful in many areas. For example, it can serve as a pre-processor for pitch-based music sound separation. Such a system may also help to automatically analyze an instrumentalist's performance when sounds from other instruments are also present. Another area where such a system is useful is automatic music transcription. If the identities of instruments used in polyphonic music are known, the system can detect each musical line played by a different instrument and transcribe the music automatically.

It should be pointed out that the pitch detection discussed in this paper is different from melody detection. Melody detection attempts to find pitches that form a melody line while we aim to detect pitches from the instrument we are concerned with. Almost all existing melody detection algorithms [1, 2, 3] do not detect pitches based on their source, therefore detected pitches may not be from the same instrument. In this paper we explore the possibility of using instrument tone models for pitch detection in polyphonic music.

To our knowledge, few systems were proposed to detect the pitch of a particular instrument in polyphonic music. Eggink and Brown [4] developed a system to detect the pitch of a solo instrument in accompanied sonatas and concertos. This system bears some similarity to ours in that it also detects the pitch of one instrument using an instrument tone model. Their system generates several pitch candidates and selects the most likely succession of pitches based on local knowledge such as instrument likelihood and temporal knowledge such as pitch transition probability. Unlike theirs, our system integrates the instrument likelihood and pitch transition probability in a more principle way by using a Hidden Markov Model (HMM). In the current study we do not consider polyphonic instruments, such as piano, which can play more than one note simultaneously under normal playing conditions. Instead we only consider instruments that can play only one note at a time, such as a clarinet and a flute. We call the sounds of instruments other than the concerned one as accompaniment.

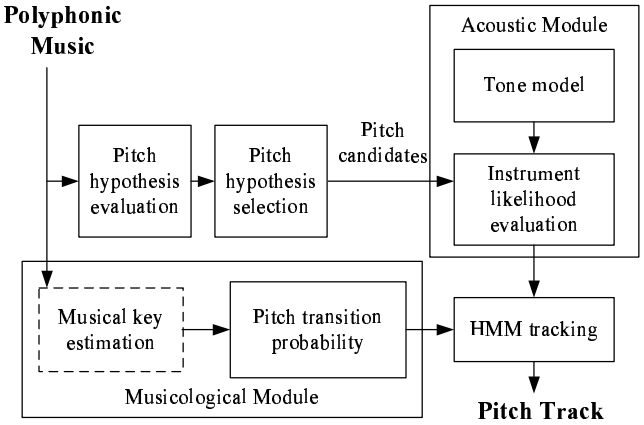

Fig. 1. Schematic diagram of the proposed system.

This paper is organized as follows. Section 2 describes the proposed system. Section 3 presents quantitative evaluation results of the proposed system. A conclusion is presented in Section 4.

2. SYSTEM DESCRIPTION

Our proposed system is illustrated in Figure 1. The input to the system is monaural polyphonic music. The system, in the pitch hypothesis evaluation step, performs time-frequency analysis and evaluates the salience of each pitch hypothesis based on the relative magnitudes of harmonics associated with the hypothesis. Then the pitch hypothesis selection step chooses pitch hypotheses with sufficiently high salience as pitch candidates. The instrument likelihoods of pitch candidates are evaluated in the acoustic module using a tone model of the instrument. The transition probability of successive pitches is constructed in the musicological module using the musical key of the input. In this study the musical key information is used as prior knowledge. Finally the instrument likelihood and the transition probability of pitch candidates are integrated in an HMM and the Viterbi algorithm is used to find the most likely pitch sequence, which is considered as the pitch track of the instrument. Our system is similar to Ryynänen and Klapuri's [3] but differs in a significant way: in the acoustic model their system evaluates pitch candidates primarily based on the instrument-independent salience while our system evaluates pitch candidates based on a tone model that characterizes the spectral shape of the instrument.

2.1. Pitch Hypothesis Evaluation

The goal of this processing step is to evaluate the salience of each pitch hypothesis. The input sampled at 44.1 kHz is first divided into frames of 40 ms in length with 50% overlap. Each frame is Hamming windowed and zero-padded before the Fourier Transform. After the time-frequency transformation, the harmonics associated with a pitch hypothesis are extracted. A pitch hypothesis is the center frequency of a frequency bin in the range between 80 Hz and 2000 Hz. For each pitch hypothesis, its first J harmonics are generated assuming perfect harmonicity. Then frequency bins neighboring a predicted harmonic frequency are searched to find the strongest spectral peak. The salience of a pitch hypothesis F, $S(F)$, is obtained as the sum of the magnitude of its harmonics.

2.2. Pitch Hypothesis Selection

$S(F)$ measures the salience of the harmonic structure associated with hypothesis F. This measure is usually biased towards pitch hypotheses that are octave-lower than the true one. That is, if F is the true pitch, $S(F/2)$, $S(F/3)$, etc, also have high values comparable to, if not higher than, $S(F)$. This is because low-number harmonics are usually stronger than high-number harmonics. Another reason is that in the low-frequency range spectral components are more densely populated—polyphonic music generally has more energy in the low-frequency range. However, by using a tone model, this bias to octave-lower pitch hypotheses does not pose a problem: the harmonic structure of such a pitch hypothesis usually has several missing harmonics and therefore does not fit the generally smooth tone model well. Octave-higher pitch hypotheses have lower salience since high-number harmonics are usually weaker. But for reasons to be discussed in Section 2.3 the tone model used tends to favor octave-higher hypotheses. To address this problem, we choose only pitch hypotheses with sufficiently high $S(F)$ as candidates. More specifically, we choose a pitch hypothesis F as a candidate if $S(F)$ is a local maximum and

$$S(F) > \theta_1 \cdot \max_f S(f) \tag{1}$$

where $S(f)$ is the salience function. Ideally, θ_1 should be a function of the relative strength between the specified instrument and the accompaniment. But in this study, we use a fixed value of 0.5, which gives good performance.

2.3. Acoustic Module

The acoustic module evaluates the likelihood of each pitch candidate using a tone model of the instrument concerned. Musical instrument identification in monophonic music is relatively easy compared to that in polyphonic music. Only recently the latter has begun to receive some attention [5, 6] but these systems' applicability to facilitating pitch detection remains to be seen. Eggink and Brown [7] developed a system to recognize solo instruments in accompanied sonatas and concertos. Their system uses the relative magnitudes of harmonics (normalized to the overall magnitude of the harmonics) as the feature for classification and shows that the feature is robust against background accompaniment. Therefore we adopt this feature for instrument likelihood evaluation. To reduce intra-instrument variation, the relative magnitudes are log-compressed. Since the relative magnitudes of harmonics are dependent on the note played, we model the tones of an instrument as a set of models, each of which is conditioned on a note within the pitch range of the instrument.

Using relative magnitudes discards the energy of the harmonic structure associated with a pitch hypothesis. We have found that this makes the system prone to pitch octave-higher errors. For example, if F is the true pitch and $2F$, $3F$, etc, are the harmonics, then a pitch hypothesis with frequency $2F$ will have harmonics from every other harmonic of F. Since high-number harmonics are usually weaker, the octave-higher pitch hypothesis has a harmonic structure

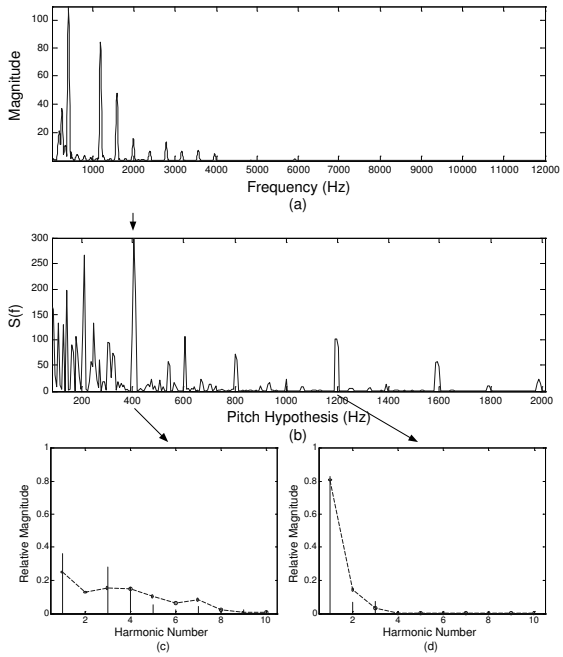

Fig. 2. (a) magnitude spectrum of a frame when a clarinet is playing note C4 accompanied by a piano. (b) salience of pitch hypotheses. The arrow indicates the true pitch of the clarinet. (c) relative magnitudes of harmonics associated with the true pitch. The dashed circle curve is the average relative magnitude of harmonics associated with note G4. (d) relative magnitudes of harmonic associated with a pitch hypothesis that is 3 times higher than the true one. The dashed circle curve is the average relative magnitude of harmonic associated with note D6, which is 3 times of G4.

with the first few harmonics dominant, which is similar to the harmonic structure of an actual high-pitched sound. Figure 2 illustrates the problem clearly. Figure 2(a) shows the spectrum of one frame when a clarinet is playing note G4 accompanied by a piano which is weak in this case. Figure 2(b) plots $S(f)$, the salience of all pitch hypotheses. Note that octave-lower pitch hypotheses also have high salience. The arrow above the box indicates the true pitch. Figure 2(c) shows the observed relative magnitudes of harmonics associated with the true pitch. The dashed circle curve is the average relative magnitudes of harmonics associated with note G4 played by a clarinet. The observed relative magnitudes follow the envelope of the average although they do not match exactly. Figure 2(d) shows the relative magnitudes of harmonics associated with a pitch hypothesis that is 3 times of the true one. In this case, the first harmonic is dominant while other harmonics are considerably weaker, especially for high-number harmonics. The dashed circle curve is the average relative magnitude of harmonics associated with note D6 played by a clarinet (2 times higher than G4). As can be seen these two harmonic structures match better. As a result, the likelihood for the octave higher pitch hypothesis is higher than that of the true pitch. Note this problem results from the feature used, i.e., the relative magnitudes of harmonics. The ways used to evaluate likelihood are irrelevant. This is why we introduce the pitch hypothesis selection in Section 2.2 to reduce the octave-higher errors.

The instrument can be modeled as a Gaussian mixture as in [7] using the relative magnitudes as the feature. However this modeling may not be a good choice for several reasons. First, the true number of mixing components is unknown. Second, when the amount of training data is limited, the parameters of the estimated Gaussian mixture model (GMM) may have large variance; in other words, the

parameters may not be well estimated. In order to better characterize the distribution of the training data, we explore kernel density estimation to model an instrument. Similar to [8], the observation probability for a given candidate F and a given instrument I is formulated as:

$$p_I(\tilde{\boldsymbol{X}}(F)) = \sum_{i=1}^{N} \frac{1}{N h_1 ... h_J} \prod_{j=1}^{J} K(\frac{\tilde{X}(jF) - \tilde{X}_i(jF)}{h_j}) \quad (2)$$

where $\tilde{\boldsymbol{X}}(F) = (\tilde{X}(F), \tilde{X}(2F), ..., \tilde{X}(JF))$ is a vector of the log of the relative magnitudes of harmonics associated with pitch hypothesis F and $\tilde{X}(jF)$ is the log of the relative magnitude of the j^{th} harmonic. $\tilde{X}_i(jF)$ is the relative magnitude of the j^{th} harmonic in the i^{th} training sample. The summation is over all the N training samples and multiplication is over all the J feature dimensions. A popular choice of $K(\cdot)$ is a one-dimension Gaussian. h_j's are parameters called bandwidths that define the amount of smoothing for the empirical distribution. Optimal values of h_j's may be determined by the least-square cross-validation method [9]. For simplicity we use the standard deviation of training samples in j^{th} dimension as h_j, which appears to work well in testing.

During instrument likelihood evaluation, for a given pitch candidate, the note with a pitch that is closest to the candidate is identified and the corresponding tone model is used. To reduce the computational complexity we choose the kernel that yields the highest probability for likelihood evaluation. Note this is equivalent to the Nearest Neighborhood estimation if h_j's are chosen to be the same for all dimensions.

To incorporate the pitch hypothesis selection into a probabilistic framework, we define the likelihood of a pitch hypothesis F as:

$$p(\boldsymbol{X}|F) = \begin{array}{ll} p_I(\tilde{\boldsymbol{X}}(F)) & \text{if } F \text{ is a candidate} \\ \theta_2 & \text{otherwise} \end{array} \quad (3)$$

where $\boldsymbol{X}$ is the magnitude spectrum. Here only part of the spectrum is used for likelihood evaluation. Strictly speaking, a likelihood measure that accounts for the spectral components other than the harmonics of F should also be used. But since the selected harmonics provide an adequate description of the hypothesis and other spectral components are irrelevant, the formulation is reasonable. θ_2 is set to some small value indicating no reliable likelihood may be drawn from the observation about the hypothesis F.

2.4. Musicological Module

The musicological module constructs the pitch transition probability based on the musical key of the input. One way to characterize the pitch transition probability is to use the distribution of pitch intervals. Interval distribution depends on the key of the music since music composed with different keys usually uses different sets of notes. In [10], interval distributions are collected with respect to Major and Minor keys. The key of a music piece may be detected automatically [11] but we use it here as prior knowledge. Note that the interval distribution specifies the transition at the note level while the proposed system evaluate pitch hypothesis at the frame level. To bridge the gap, we make a simplifying assumption that the frame level pitch transition follows the same distribution as that at the note level. More specifically, we assume:

$$p(F_i - F_{i-1}) = p_{note}(MIDI(F_i) - MIDI(F_{i-1})) \quad (4)$$

where F_i is the pitch hypothesis at time frame i. $p(F_i - F_{i-1})$ is the frame level pitch transition probability and is assumed to be the same for different instruments. p_{note} is the distribution of intervals. $MIDI(F)$ maps F in Hz to the corresponding MIDI number according to:

$$MIDI(F) = 69 + [12 \times \log_2(\frac{F}{440})] \quad (5)$$

2.5. HMM Tracking

The pitch generation process can be modeled as a continuous HMM [12]. In each frame, a hidden node represents the pitch state space, which consists of all the pitch hypothesis, and an observation node represents the observed magnitude spectrum. The transition probability between consecutive frames is specified in the musicological module. Finally the Viterbi algorithm is used to find the most likely sequence of pitch generation and transition, which is considered to be the pitch track of the instrument we are interested in.

3. EVALUATION

To evaluate the proposed system, we construct a database consisting of 3 pieces of western classic music composed by J. S. Bach (*Invention No. 1, Sinfonia No. 1*, and *Chorale Harmonizations*) with the degree of polyphony varying from 2 to 4 and with different tempi. The MIDI data of these pieces are taken from [13] without further editing. A musical line played by a single instrument is randomly selected for evaluation. Note that the selected line may not be the melody of a piece. Audio files of these pieces can be generated from MIDI data using MIDI synthesizers. But MIDI synthesizers generate sound by looping a sample of the instrument to be synthesized. As a result, the synthesized sound has rather stable spectral contents, which is very different from real music recordings. Since it is difficult to get multi-track recordings where different instruments are recorded in different tracks, we generate audio files from MIDI data by substituting recorded note samples from the RWC music instrument database [14] for notes specified in MIDI files. Specifically, during the synthesis the note specified in a MIDI file is identified and the note sound sample with the closest average pitch is used for that note. If the length of the recorded note sample is longer than the note duration specified in MIDI, then the note sample is truncated to meet the specified note duration. If it is the other way, i.e., the length of the recorded note sample is shorter than the note duration specified in MIDI, the whole recorded note sample is used. The accompaniment is still synthesized using MIDI synthesizers. The synthesis is done for each piece and for each of the four instruments from different instrument families: clarinet, flute, violin, and trumpet. Since the chosen pieces may not be composed for the instrument to be synthesized, some notes may not be within the pitch range of an instrument. In this case, the out-of-range note is simply skipped. The audio files generated in this way do not approximate real performances well, but they show realistic spectral and temporal variations. The first 30 seconds of each piece are used for testing.

Each synthesized line of an instrument is mixed with its accompaniment in 4 different line-to-accompaniment ratios: 10, 5, 0, and –5 dB; we call such ratio as signal-to-noise ratio (SNR). The higher the SNR, the stronger the line played by the instrument compared to the accompaniment and the easier the task of detecting the corresponding pitch track of the instrument concerned. The sound sample of a note usually has pitches different from the nominal pitch value of the note because of instrument tuning or other factors. To get more accurate reference pitch tracks we apply Praat [15] to the synthesized signal of an instrument. The detected pitch tracks are further inspected to correct octave errors.

The RWC database provides sound samples of many different instruments. For the same instrument, samples are also recorded for a note with respect to different manufacturers and different playing techniques. We use sound samples from different manufacturers of each instrument to construct tone models for training. The pitches of a note are detected using Praat and used to extract the relative magnitudes of harmonics. We use $J = 10$ harmonics as in [4].

Table 1 shows the average detection accuracies at the frame level for different instruments under different SNRs using kernel density estimation. A detected pitch is considered correct if it is within 3% of the reference pitch. Detecting the presence of the instrument concerned is not implemented in this paper and it is currently under investigation. Therefore the pitch detection accuracies are evaluated

Table 1. Average detection accuracies using kernel density estimation

SNR	Instrument (%)			
	Clarinet	Flute	Violin	Trumpet
10dB	88.8	92.0	79.2	86.1
5dB	87.1	87.4	67.3	86.0
0dB	75.5	80.1	53.4	84.9
–5dB	41.0	51.2	38.7	68.5

Table 2. Average detection accuracies using GMM

SNR	Instrument (%)			
	Clarinet	Flute	Violin	Trumpet
10dB	74.3	92.2	66.2	84.6
5dB	70.4	82.9	59.1	80.2
0dB	62.4	70.1	50.9	73.4
–5dB	49.9	54.7	42.6	60.2

at the frames when the instrument is present. Except for violin, the proposed system has good detection accuracies even at 0 dB. Violin sounds tend to be less stable compared to other instruments tested, which contributes to the low detection accuracy. One reason for the relatively poor performance in the –5 dB case is that pitch hypothesis selection removes pitch hypotheses that correspond to the true pitch because its salience is low compared to that of the strongest sound from the accompaniment. Another reason may be that as the accompaniment becomes stronger, the relative magnitudes of the instrument concerned are corrupted. Eggink and Brown [4] reported the average frame-level pitch detection accuracy of 63% at 0 dB for a similar task. Their test signals are synthesized using a MIDI synthesizer from MIDI data. Our system appears to obtain significantly better results. However, caution should be exercised since the test database used in their study is different from ours.

We also evaluate the system using a GMM as the tone model. The relative magnitudes of harmonics of each note are modeled as a GMM with 3 mixing components to model the three different stages of a note: attack, sustain, and decay. To reduce the number of parameters to be estimated because of the limited training data, diagonal covariance matrices are used. The parameters of GMMs are estimated using the toolbox in [16]. The average detection accuracies are listed in Table 2. Compared to kernel density estimation, the GMM model tends to have lower detection accuracies when SNR is high. However for very low SNR, such as –5 dB, the GMM model performs better, suggesting that GMM is more robust in low SNRs.

HMM tracking helps to improve the pitch detection accuracies. Compared to only selecting the pitch candidates with the highest likelihood as the true pitch, the HMM tracking contributes 5% accuracy in average for SNRs of 10 and 5 dB, while for 0 and –5 dB, the contribution is smaller. This shows that the HMM tracking is more helpful when the likelihood evaluation gives reasonable results. Hypothesis selection significantly improves the performance of the proposed system. Without the hypothesis selection, the average detection accuracies are 10.6%, 9.8%, 5.2%, and 3.0% lower in absolute terms for SNRs of 10, 5, 0, and –5 dB, respectively.

4. CONCLUDING REMARKS

In this paper we have proposed a system to detect the pitch of a particular instrument in polyphonic music. An HMM is used to incorporate the tone model and pitch transition probability to perform robust pitch estimation. Quantitative evaluation shows that the proposed system performs well for different instruments, different degrees of polyphony, and different SNR situations.

We have also compared GMM and kernel density estimation for instrument modeling and found that kernel estimation yields better detection accuracies. However non-parametric models are vulnerable to outliers and they may not handle the intra-instrument variation well. To further evaluate tone modeling, more data need be collected. A common phenomenon in music audio processing is that harmonics from different instruments may overlap, which affects instrument likelihood evaluation. For high SNR situations, the overlapping of harmonics does not seem to pose a problem since the resulting change of relative magnitudes of harmonics is small. However, in low SNR situations, this effect can be significant and may cause the poor performance. One way to deal with this problem is to apply spectral smoothing [17] before harmonic structure is used for instrument likelihood evaluation. With an instrument model, it is possible to detect when the instrument concerned is not playing. This detection mechanism is currently under investigation.

Acknowledgments. This research was supported in part by an AFOSR grant (F49620-04-1-0027) and an NSF grant (IIS-0534707).

5. REFERENCES

[1] M. Goto, "A real-time music-scene-description system: predominant-F0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Communication*, vol. 43, pp. 311–329, 2004.

[2] Y. Li and D. L. Wang, "Separation of singing voice from music accompaniment for monaural recordings," *IEEE Transactions on Audio, Speech, and Language Processing*, in press.

[3] M. P. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *WASPAA*, 2005.

[4] J. Eggink and G. Brown, "Extracting melody lines from complex audio," in *ISMIR*, 2004.

[5] E. Vincent and X. Rodet, "Instrument identification in solo and ensemble music using independent subspace analysis," in *ISMIR*, 2004.

[6] S. Essid, G. Richard, and B. David, "Instrument recognition in polyphonic music," in *ICASSP*, 2005, pp. III.245–248.

[7] J. Eggink and G. Brown, "Instrument recognition in accompanied sonatas and concertos," in *ICASSP*, 2004, pp. IV.217–220.

[8] N. Roman, D. L. Wang, and G. Brown, "Speech segregation based on sound localization," *Journal of the Acoustical Society of America*, vol. 114, no. 4, pp. 2236–2252, 2003.

[9] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. New York: Chapman and Hall, 1986.

[10] D. Huron, *Sweet Anticipation: Music and the Psychology of Expectation*. Cambridge, Massachusetts: MIT Press, 2006.

[11] A. Shenoy, R. Mohapatra, and Y. Wang, "Key determination of acoustic musical signals," in *IEEE International Conference on Multimedia and Expo*, 2004, pp. 1771–1774.

[12] M. Wu, D. L. Wang, and G. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 3, pp. 229–241, 2003.

[13] D. Huron, *The Humdrum Toolkit: Reference Manual*. Stanford, California: Center for Computer Assisted Research in the Humanities, 1995.

[14] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Music genre database and musical instrument sound database," in *ISMIR*, 2003.

[15] P. Boersma and D. Weenink. (2002) Praat: Doing phonetics by computer, version 4.0.26. [Online]. Available: http://www.fon.hum.uva.nl/praat

[16] K. Murphy. (2005) HMM toolbox for MATLAB. [Online]. Available: http://www.cs.ubc.ca/˜murphyk/Software/HMM/hmm.html

[17] A. P. Klapuri, "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.