Robust Temporal and Spectral Modeling for Query By Melody

Shai Shalev-Shwartz
Hebrew University
Jerusalem, Israel
shais@cs.huji.ac.il

Shlomo Dubnov
Ben-Gurion University
Be'er-Sheva,Israel
dubnov@bgumail.bgu.ac.il

Nir Friedman
Hebrew University
Jerusalem, Israel
nir@cs.huji.ac.il

Yoram Singer
The Hebrew University
Jerusalem, Israel
singer@cs.huji.ac.il

ABSTRACT

Query by melody is the problem of retrieving musical performances from melodies. Retrieval of real performances is complicated due to the large number of variations in performing a melody and the presence of colored accompaniment noise. We describe a simple yet effective probabilistic model for this task. We describe a generative model that is rich enough to capture the spectral and temporal variations of musical performances and allows for tractable melody retrieval. While most of previous studies on music retrieval from melodies were performed with either symbolic (e.g. MIDI) data or with *monophonic* (single instrument) performances, we performed experiments in retrieving live and studio recordings of operas that contain a leading vocalist and rich instrumental accompaniment. Our results show that the probabilistic approach we propose is effective and can be scaled to massive datasets.

Categories and Subject Descriptors

H.2.4 [**Database Management**]: Systems—*Multimedia databases, query processing*; H.3.3 [**Information storage and retrieval**]: Information Search and Retrieval—*Query formulation, Retrieval models*; I.5.4 [**Pattern Recognition**]: Applications—*Signal processing.*

General Terms

Algorithms, Experimentation.

Keywords

Music Information Retrieval, Query by Melody, Graphical Models, Spectral Modeling.

1. INTRODUCTION

A natural way for searching a musical audio database for a song is to look for a short audio segment containing a melody from the song. Most of the existing systems are based on textual information, such as the title of the song and the name of the composer. However, people often do not remember the name of the composer and the song's title but can easily recall fragments from the soloist's melody.

The task of *query by melody* attempts to automate the music retrieval task. It was first discussed in the context of query by humming [11, 13, 14]. These works focus on converting hummed melodies into symbolic MIDI format (MIDI is an acronym for Musical Instrument Digital Interface. It is a symbolic format for representing music). Once the query is converted into a symbolic format the challenge is to search for musical performances that approximately match the query. Most of the research so far has been conducted with music stored in MIDI format [12] or in *monophonic* (i.e. single vocal or instrument) recordings (see for instance [9, 7] and the references therein). In this paper, we suggest a method for query by melody where the query is posed in symbolic form as a monophonic melody and the database consists of real polyphonic recordings.

When dealing with real polyphonic recordings we need to address several complicating factors. Ideally melodies can be represented as sequences of notes, each is a pair of frequency and temporal duration. In real recordings two major sources of difficulty arise. The first is the high variability of the actual durations of notes. A melody can be performed faster or slower than the one dictated by the musical score. This type of variation is often referred to as tempo variability. Furthermore, the tempo can vary within a single performance. For instance, a performance can start with a slow tempo which gradually increases. The second complicating factor is the high variability of the spectrum due to many factors such as differences in tone colors (timbre) of different singers/instruments, the intentional variation by the leading vocalists (e.g. vibrato and dynamics) and by "spectral masking" of the leading vocal by the accompanying vocals and orchestra.

We propose to tackle these difficulties by using a generative probabilistic approach that models the temporal and spectral variations. We associate each note with a hidden tempo variable. The tempo variables capture the temporal variations in the durations of notes. To enable efficient computation, the hidden tempo sequence is modeled as a first order Markov process. In addition, we also describe a simple probabilistic spectral distribution model that is robust to the masking noise of the accompanying instruments and

SIGIR'02, August 11-15, 2002, Tampere, Finland.

singers. This spectral distribution model is a variant of the harmonic likelihood model for pitch detection [16]. Combining the temporal and spectral probabilistic components, we obtain a joint model which can be thought of as a dynamic Bayesian network [8]. This representation enables efficient alignment and retrieval using dynamic programming.

This probabilistic approach is related to several recent works that employ Hidden Markov Models (HMM) for music processing. Raphael [15] uses melody information (pitches and durations of notes) in building an HMM for a score following application. A similar approach is taken by Durey and Clements [9] who use the pitch information of notes for building HMMs for melody retrieval. However, both approaches were designed for and evaluated on monophonic music databases. Most work on polyphonic music processing addressed tasks such as music segmentation into textures [6], polyphonic pitch tracking [18], and genre classification [17, 10]. We believe that the approach we describe in this paper is a step toward an effective retrieval procedure for massive musical datasets.

2. PROBLEM SETTING

In our setting, we are given a melody and our task is to retrieve musical performances containing the requested melody and to find its location within the retrieved performances. A melody is a sequence of notes where each note is a pair of a pitch value and a duration value. Our goal is to retrieve melodies from audio signals representing real performances.

Formally, let $\mathcal{R}_+$ denote the positive real numbers. Let $f_l, f_h \in \mathcal{R}_+$ be frequency values (in Hz) and let $[f_l, f_h]$ be a *diapason*. A *diapason* of a singer (or an instrument) is the range of pitch frequencies that are in use by the singer (or by the instrument). For instance, a tenor singer typically employs a *diapason* of $[110Hz, 530Hz]$. Let Λ denote the set of all possible frequencies of notes. In the well-tempered Western music tuning system, $\Lambda = \{f_{ref} \cdot 2^{s/12} | s \in \mathcal{Z}\}$, where $f_{ref} = 440Hz$. Let $\Gamma = [f_{low}, f_{high}] \cap \Lambda$ denote all the possible pitches of notes in the *diapason*. A melody is described formally by a sequence of pitches, $\boldsymbol{p} \in \Gamma^k$, and a sequence of durations, $\boldsymbol{d} \in {\mathcal{R}_+}^k$, in a predefined time units (e.g. seconds or samples).

A performance of a melody is a discrete time sampled audio signal, $\boldsymbol{o} = o_1, ..., o_T$. A performance is formally entirely defined given the melody: play or sing using pitch p_1 for the first d_1 seconds, then play or sing pitch p_2 for the next d_2 seconds, and so on and so forth. In reality, a melody does not impose a rigid framework. The actual frequency content of a given note varies with the type of instrument that is played and by the performer. Examples for such variations are the vibrato and timbre effects. The accompaniment also greatly influences the spectral distribution. While playing a note using pitch p, we are likely to see a local concentration of energy close to multiples of the frequency p in the power spectrum of the signal. However, there may be other spectral regions with high levels of energy. We will address this problem later on in this section. Another source of variation is local scaling of the durations of notes as instructed by the melody. The performer typically uses a tempo that scales the duration and moves from one tempo to another, thus using a different time scale to play the notes. Therefore, we also need to model the variation in the tempo which we describe now.

Rallentando	1.2 1.2 1.25 1.3 1.3
Accelerando	0.7 0.65 0.6 0.5 0.5

Table 1: Examples of scaling factor sequences: In the first sequence the scaling factors are gradually increasing and thus the tempo is decreasing ("Rallentando"). In the second example the scaling factors are decreasing and the tempo is increasing ("Accelerando").

A tempo sequence is a sequence of scaling factors, $\boldsymbol{m} \in {\mathcal{R}_+}^k$. The actual duration of note i, denoted $\widetilde{d}_i$ is d_i scaled by m_i, $\widetilde{d}_i = d_i m_i$. Seemingly, allowing different scaling factors for the different notes adds a degree of freedom that makes the melody duration values redundant. However, a typical tempo sequence does not change rapidly and thus reflects most of the information of the original durations (up to a scaling factor). Table 1 shows two examples of tempo sequences. A pitch–duration–tempo triplet $(\boldsymbol{p}, \boldsymbol{d}, \boldsymbol{m})$ generates an actual pitch–duration pair $(\boldsymbol{p}, \widetilde{\boldsymbol{d}})$.

In order to describe the generation of the actual performance audio signal $\boldsymbol{o}$ from $(\boldsymbol{p}, \widetilde{\boldsymbol{d}})$ we introduce one more variable, $\boldsymbol{s} \in {\mathcal{R}_+}^k$ where s_i is the starting time (sample number) of note i in the performance. We define $s_i = 1 + \sum_{j=1}^{i-1} \widetilde{d}_j$ for $i = 1, \ldots, k+1$. Notes generate consecutive blocks of signal samples. Let $\overline{o}_i = (o_{s_i}, ..., o_{s_{i+1}-1})$ be the block of samples generated by note i.

The power spectrum of $\overline{o}_i$ varies significantly from performance to performance, according to various factors such as the spectral envelope of the soloist and pitches of accompaniment instruments. Since our goal is to locate and retrieve a melody from a dataset that may contain thousands of performances, we resort to a very simple spectral model and do not explicitly model these variables. We use an approximation to the likelihood of a block spectrum given its pitch.

3. FROM MELODY TO SIGNAL: A GENERATIVE MODEL

To pose the problem in a probabilistic framework, we need to describe the likelihood of a performance given the melody, $P(\boldsymbol{o}|\boldsymbol{p}, \boldsymbol{d})$. We cast the tempo sequence $\boldsymbol{m}$ as a hidden random variable, thus the likelihood can be written as,

$$P(\boldsymbol{o}|\boldsymbol{p}, \boldsymbol{d}) = \sum_{\boldsymbol{m}} P(\boldsymbol{o}, \boldsymbol{m}|\boldsymbol{p}, \boldsymbol{d}) \quad . \tag{1}$$

For simplicity, we assume that the tempo sequence does not depend on the melody. While this assumption, naturally, does not always hold, we found empirically that these types of correlations can be ingnored in short pieces of performances. With this assumption and the identity $\widetilde{\boldsymbol{d}} = \boldsymbol{dm}$,

Equ. (1) can be rewritten as,

$$P(\boldsymbol{o}|\boldsymbol{p},\boldsymbol{d}) = \sum_{\boldsymbol{m}} P(\boldsymbol{m}|\boldsymbol{p},\boldsymbol{d})P(\boldsymbol{o}|\boldsymbol{p},\boldsymbol{d},\boldsymbol{m})$$
$$= \sum_{\boldsymbol{m}} P(\boldsymbol{m})P(\boldsymbol{o}|\boldsymbol{p},\boldsymbol{d},\boldsymbol{m})$$
$$= \sum_{\boldsymbol{m}} P(\boldsymbol{m})P(\boldsymbol{o}|\boldsymbol{p},\widetilde{\boldsymbol{d}}) \quad .$$

We now need to specify the prior distribution over the tempo, $P(\boldsymbol{m})$, and the posterior distribution of the signal given the pitches and the actual durations of the notes $P(\boldsymbol{o}|\boldsymbol{p},\widetilde{\boldsymbol{d}})$.

3.1 Tempo modeling

We chose to model the tempo sequence as a first order Markov process. As we see in the sequel this choice on one hand allows an efficient alignment and retrieval, and on the other hand, was found empirically to be rich enough. Therefore, the likelihood of $\boldsymbol{m}$ is given by,

$$P(\boldsymbol{m}) = P(m_1)\prod_{i=2}^{k} P(m_i|m_{i-1}) \quad .$$

We use the log-normal distribution to model the conditional probability $P(m_i|m_{i-1})$, that is $\log_2(m_i) \sim \mathcal{N}(\log_2(m_{i-1}),\rho)$, where ρ is a scaling parameter of the variance. The prior distribution of the first scaling factor $P(m_1)$ is also assumed to be log-normal around zero with variance ρ, $\log_2(m_i) \sim \mathcal{N}(0,\rho)$. In our experiments, the parameter ρ was determined manually according to musical knowledge. This parameter can also be learned from MIDI files.

3.2 Spectral Distribution Model

In this section we describe our spectral distribution model. There exist quite a few models for the spectral distribution of singing voices and harmonic instruments. However, most of these models are rather general. These models typically assume that the musical signal is contaminated with white noise whose energy is statistically independent of the signal. See for instance [16] and the references therein. In contrast, we assume that there is a leading instrument, or soloist, that is accompanied by an orchestra or a chorus. The energy of the accompaniment is typically highly correlated with the energy of the soloist. Put another way, the dynamics of the accompaniment matches the dynamics of the soloist. For instance, when the soloist sings pianissimo the chorus follows her with pianissimo voices. We therefore developed a simple model whose parameters can be efficiently estimated that copes with the correlation in energy between the leading soloist and the accompaniment. In Fig. 1 we show the spectrum of one frame of a performance signal from our database. The harmonics are designated by dashed lines. It is clear from the figure that there is a large concentration of energy at the designated harmonics. The residual energy, outside the harmonics, is certainly non-negligible but is clearly lower than the energy of the harmonics. Thus, our assumptions, although simplistic, seem to capture to a large extent the characteristics of the spectrum of singing with accompaniment.

Using the definition of a block $\overline{o}_i$ from Sec. 2, the likelihood of the signal given the sequences of pitches and durations can be decomposed into a product of likelihood values

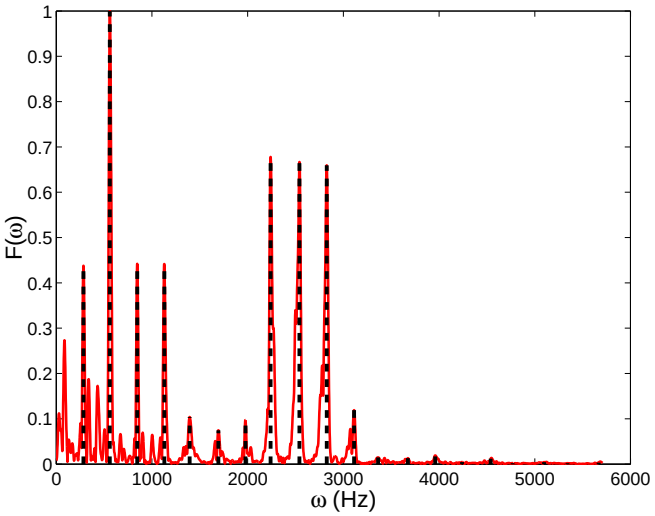


Figure 1: The spectrum of a single frame along with an impulse train designating the harmonics of the soloist.

of the individual blocks,

$$P(\boldsymbol{o}|\widetilde{\boldsymbol{d}},\boldsymbol{p}) = \prod_{i=1}^{k} P(\overline{o}_i|p_i) \quad .$$

Therefore, the core of our modeling approach is a probabilistic model for the spectral distribution of a whole block given the underlying pitch frequency of the soloist. Our starting point is similar to the model presented in [16]. We assume that a note with pitch p_i attains high energy at frequencies which are multiples of p_i, namely at $p_i h$ for integer h. These frequencies are often referred to as harmonics. Since our signal is band limited, we only need to consider a finite set of harmonics h, $h \in \{1,2,...,H\}$. For practical purposes we set H to be 20 which enables a fast parameter estimation procedure. Let $F(\omega)$ denote the observed energy of the block $\overline{o}_i$ at frequency ω. Let $S(\omega)$ denote the energy of the soloist at frequency ω. The harmonic model assumes that $S(\cdot)$ is bursts of energy centered at the harmonics of the pitch frequency, $p_i h$, and we model it as a weighted sum of delta functions,

$$S(\omega) = \sum_{h=1}^{H} A(h)\delta(p_i h - \omega) \quad , \qquad (2)$$

where $A(h)$ is the volume gain for the harmonic whose index is h. The residual of the spectrum at frequency ω is denoted $N(\omega)$ and is equal to $N(\omega) = F(\omega) - S(\omega)$. We now describe a probabilistic model that leads to the following log-likelihood score,

$$\log P(\overline{o}_i|p_i) \propto \log \frac{\|S\|^2}{\|N\|^2} \quad , \qquad (3)$$

where $\|\cdot\|$ denotes the ℓ_2-norm.

To derive the above equation we assume that the spectrum of the ith block, F, is comprised of two components. The first component is the energy of the soloist, $S(\omega)$ as defined in Equ. (2). The second component is a general masking

noise that encompasses the signal's energy due to the accompaniment and affects the entire spectrum. We denote the noise energy at frequency ω as $\eta(\omega)$. The energy of the spectrum at frequency ω is therefore modeled as,

$$F(\omega) = \sum_{h=1}^{H} A(h)\left(\eta(\omega) + \delta(p_i h - \omega)\right) \quad . \tag{4}$$

We now impose another simplifying assumption by setting the noise η to be a multivariate normal random variable and further assuming that the noise values at each frequency ω are statistically independent with equal variance. Thus, the noise density function is

$$f(\eta|v) = \frac{1}{(2\pi v)^{L/2}} e^{-\frac{\|\eta\|^2}{2v}} \tag{5}$$

where v is the variance and L is the number of spectral points computed by the discrete Fourier transform. (We chose $L = 2^{15}$ to get a good spectral resolution.) Taking the log of the above density function we get,

$$\log f(\eta|v) = -\frac{L}{2}\log(2\pi v) - \frac{\|\eta\|^2}{2v} \quad . \tag{6}$$

The gain values $A(h)$ are free parameters which we need to estimate from the spectrum. Assuming that the noise level is relatively small compared to the bursts of energy at the harmonics of the pitch frequency, we set the value of $A(h)$ to be $F(p_i h)$. We also do not know the noise variance v. For this free parameter we use the simple maximum likelihood (ML) estimate which can be easily found as follows. The maximum likelihood estimate of v is found by taking the derivative of $\log f(\eta|v)$ with respect to v,

$$\frac{\partial \log f(\eta|v)}{\partial v} = -\frac{L}{2}\frac{2\pi}{2\pi v} + \frac{\|\eta\|^2}{2v^2} = 0 \quad \Rightarrow \; v^* = \frac{\|\eta\|^2}{L} \quad .$$

Rearranging Equ. (4), the noise value at frequency ω, $\eta(\omega)$, can be written as,

$$\eta(\omega) = \frac{F(\omega) - \sum_{h=1}^{H} A(h)\delta(p_i h - \omega)}{\sum_{h=1}^{H} A(h)} \quad .$$

By using above equation for $\eta(\omega)$ along with values set for $A(h)$ and the maximum likelihood estimate v^* in Equ. (6) we get,

$$\begin{aligned} \log f(\eta|v*) &= -\frac{L}{2}(\log(2\pi) + \log(\|\eta\|^2) - \log(L) - 1) \\ &= c + \frac{L}{2}\log\left(\frac{\|S\|^2}{\|N\|^2}\right) \quad . \end{aligned} \tag{8}$$

Since the stochastic ingredient of our spectral model is the accompanying noise, the noise likelihood above also constitute the likelihood of the spectrum.

To summarize, we now overview our approach for retrieval. We are given a melody $(\boldsymbol{p}, \boldsymbol{d})$ and we want to find an audio signal $\boldsymbol{o}$ which represents a performance of this melody. Using our probabilistic framework, we cast the problem as the problem of finding a signal portion $\boldsymbol{o}$ whose likelihood given the melody, $P(\boldsymbol{o}|\boldsymbol{p}, \boldsymbol{d})$, is high. Our search strategy is as follows. We find the best alignment of the signal to the melody as we describe in the next section. The score of the alignment procedure we devise is also our means for retrieval. We then rank the segments of signals in accordance with their likelihood scores and return the segments achieving high likelihoods scores.

1. **Initialization**

$$\forall_{1\le t\le T}, \gamma(0,t,1) = 1$$

2. **Recursion**

$$\gamma(i,t,\xi) = \max_{\xi' \in M} \gamma(i-1, t', \xi') P(\xi|\xi') P(o_{t'+1},\ldots,o_t|p_i)$$

where $t' = t - d_i\xi$.

3. **Termination**

$$P^* = \max_{1\le t\le T,\ \xi\in M} \gamma(k,t,\xi)$$

Figure 2: The alignment algorithm.

4. ALIGNMENT AND RETRIEVAL

Alignment of a melody to a signal is performed by finding the best assignment of a tempo sequence. Formally, we are looking for the scaling factors $\boldsymbol{m}^*$ that attain the highest likelihood score, $\boldsymbol{m}^* = \arg\max_{\boldsymbol{m}} P(\boldsymbol{o}, \boldsymbol{m}|\boldsymbol{d}, \boldsymbol{p})$. Although the number of possible sequences of scaling factors $\boldsymbol{m}$ grows exponentially with the sequence length, the problem of finding $\boldsymbol{m}^*$ can be efficiently solved using dynamic programming, as we now describe.

Let $\boldsymbol{m}^i = (m_1, ..., m_i)$ denote the scaling factors of the first i notes of a melody. Let $\boldsymbol{o}^t = (o_1, ..., o_t)$ denote the first t samples of a signal. Let M be a discrete set of possible scaling factor values. For $\xi \in M$, let $M_{i,t,\xi}$ be a set of all possible sequences of i scaling factors, $\boldsymbol{m}^i$, such that $m_i = \xi$ is the scaling factor of note i and $t = \sum_{j=1}^{i} m_j d_j$ is the actual ending time of note i. Let $\gamma(i,t,\xi)$ be the joint likelihood of $\boldsymbol{o}^t$ and $\boldsymbol{m}^i \in M_{i,t,\xi}$

$$\gamma(i,t,\xi) = \max_{\boldsymbol{m}^i \in M_{i,t,\xi}} P(\boldsymbol{o}^t, \boldsymbol{m}^i|\boldsymbol{p}, \boldsymbol{d})$$

The pseudo code for computing $\gamma(i,t,\xi)$ recursively is shown in Fig. 2.

The most likely sequence of scaling factors $\boldsymbol{m}^*$ is obtained from the algorithm by saving the intermediate values that maximize each expression in the recursion step. The complexity of the algorithm is $O(kT|M|^2D)$, where k is the number of notes, T is the number of samples in the digital signal, $|M|$ is the number of all possible tempo values and D is the maximal duration of a note. Using a pre-computation of the likelihood values we can reduce the time complexity by a factor of D and thus the run time of the algorithm reduces to $O(kT|M|^2)$. It is important to clarify that the pre-computation does not completely determine a single pitch value for a frame. It calculates the probability of the frame given each possible pitch in the diapason.

As mentioned above, our primary goal is to retrieve the segments of signals representing the melody given by the

query. Theoretically, we need to assign a segment $\boldsymbol{o}$ its likelihood score, $P(\boldsymbol{o}|\boldsymbol{p},\boldsymbol{d}) = \sum_{\boldsymbol{m}} P(\boldsymbol{m}|\boldsymbol{p},\boldsymbol{d})$. However, this marginal probability is rather expensive to compute. We thus approximate this probability with the joint probability of the signal and most likely sequence of scaling factors, $P(\boldsymbol{o},\boldsymbol{m^*}|\boldsymbol{p},\boldsymbol{d})$. That is, we use the likelihood score of the alignment procedure as a retrieval score.

5. EXPERIMENTAL RESULTS

To evaluate our algorithm we collected 50 different melodies from famous opera arias, and queried these melodies in a database of real recordings. The recordings consist of 832 performances of opera arias performed by more then 40 different tenor singers with full orchestral accompaniment. Each performance is one minute. The data was extracted from seven audio CDs [2, 3, 5, 1, 4], and saved in *wav* format. Most of the performances (about 90 percent) are digital recordings (DDD/ADD). Yet, some performances are digital remastering of old analog recordings (AAD). This introduced additional complexity to the retrieval task due to varying level of noise.

The melodies for the experiments were extracted from MIDI files. About half of the MIDI files were downloaded from the Internet [1] and the rest of the MIDI files were performed on a MIDI keyboard and saved as MIDI files.

We compared three different tempo-based approaches for retrieval. The first method simply uses the original durations given in the query without any scaling. We refer to this simplistic approach as the *Fixed Tempo* (FT) model. The second approach uses a single scaling factor for all the durations of a given melody. However, this scaling factor is determined independently for each signal so as to maximize the signals likelihood. We refer to this model as the *Locally Fixed Tempo* (LFT) model. The third retrieval method is our variable tempo model that we introduced in this paper. We therefore refer to this method as the *Variable Tempo* (VT) model. By taking a prefix subset of each melody used in a query we evaluated three different lengths of melodies: 5 seconds, 15 seconds, and 25 seconds.

To assess the quality of the spectral distribution model described in Sec. 3.2, we implemented the spectral distribution model described in [16]. This model assumes that the harmonics of the signal are contaminated with noise whose mean energy is independent of the energy of the harmonics. We refer to our model as the *Harmonics with Scaled Noise* (HSN) model and to the model from [16] as the *Harmonics with Independent Noise* (HIN) model.

To evaluate the performances of the methods we used three evaluation measures: *one-error*, *coverage* and *average precision*. To explain these measures we introduce the following notation. Let N be the number of performances in our database and let M be the number of melodies that we search for. (As mentioned above, in our experiments $N = 832$ and $M = 50$.) For a melody index i we denote by Y_i the set of the performances containing melody i. The probabilistic modeling we discussed in this paper induces a natural ordering over the performances for each melody. Let $R_i(j)$ denote the ranking of the performance indexed j with respect to melody i. Based on the above definitions we now give the formal definitions of the performance measures we used for evaluation.

One-Error. The one-error measures how many times the top-ranked performance did *not* contain the melody posed in the query. Thus, if the goal of our system is to return a single performance that contains the melody, the one-error measures how many times the retrieved performance did not contain the melody. Formally, the definition of the one-error is,

$$\text{OErr} = \frac{1}{M}\sum_{i=1}^{M}[\![\arg\min_j R_i(j) \notin Y_i]\!] \ .$$

where $[\![\pi]\!] = 1$ if predicate π holds and 0 otherwise.

Coverage. While the one-error evaluates the performance of a system with respect to the top-ranked performance, the goal of the coverage measure is to assess the performance of the system for all of the possible performances of a melody. Informally, Coverage measures the number of excess (non-relevant) performances we need to scan until we retrieve all the relevant performances. Formally, Coverage is defined as,

$$\text{Cov} = \frac{1}{M}\sum_{i=1}^{M}(\max_{j\in Y_i} R_i(j) - |Y_i|) \ .$$

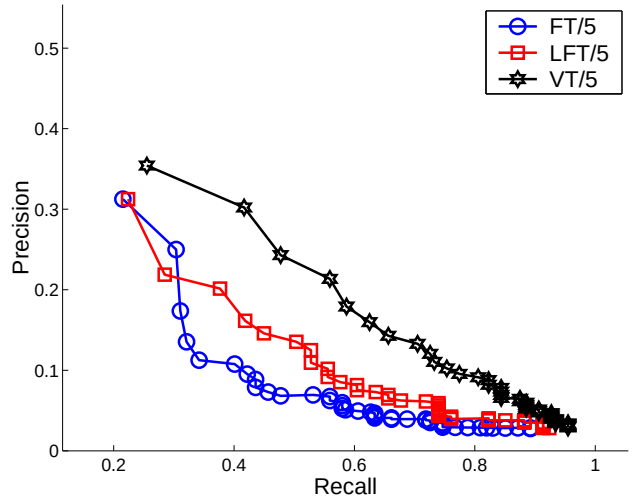
Average Precision. The above measures do not suffice in evaluating the performances of retrieval systems as one can achieve good (low) coverage but suffer high one-error rates, and vice versa. In order to assess the ranking performance as a whole we use the frequently used average precision measure. Formally, the average precision is defined as,

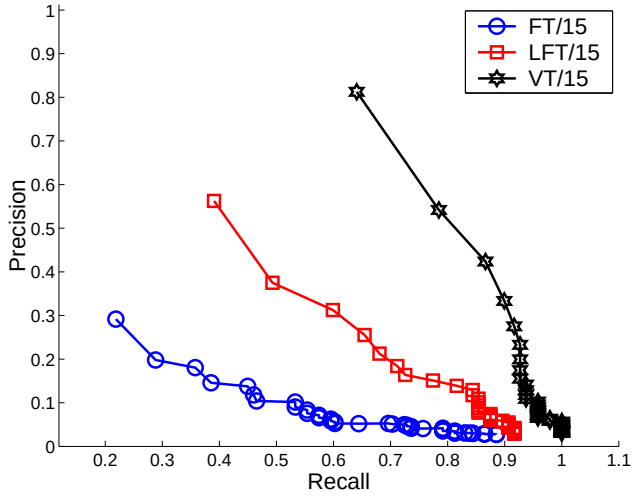
$$\text{AvgP} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{|Y_i|}\sum_{j\in Y_i}\frac{|\{j' \in Y_i | R_i(j') \le R_i(j)\}|}{R_i(j)} \ .$$

In addition we also use precision versus recall graphs to illustrate the overall performances of the different approaches discussed in the paper. A precision-recall graph shows the level of precision for different recall values. The graphs presented in this paper are non-interpolated, that is, they were calculated based on the precision and recall values achieved at integer positions of the ranked lists.

In Table 2 we report results with respect to the performance measures described for the FT, LFT, and VT models. For each tempo model we conducted the experiments with the two spectral distribution models HIN and HSN. It is clear from the table that the *Variable Tempo* model with the *Harmonics with Scaled Noise* spectral distribution outperforms the rest of the models and achieves superior results. Moreover, the performance of the *Variable Tempo* model consistently improves as the duration of the queries increases. In contrast, the *Fixed Tempo* does not exhibit any improvement as the duration of the queries increases and the *Locally Fixed Tempo* shows only a moderate improvement when using fifteen second long queries instead of five second long queries and it does not improve as the duration grows to twenty five seconds. A reasonable explanation for these phenomena is that the amount of variability in a very short query is naturally limited and thus the leverage

[1] http://www.aria-database.com,
http://www.musicscore.freeserve.co.uk,
http://www.classicalmidi.gothere.uk.com





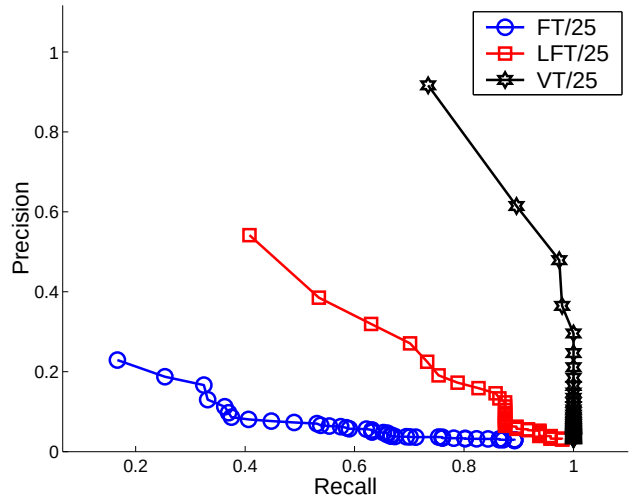
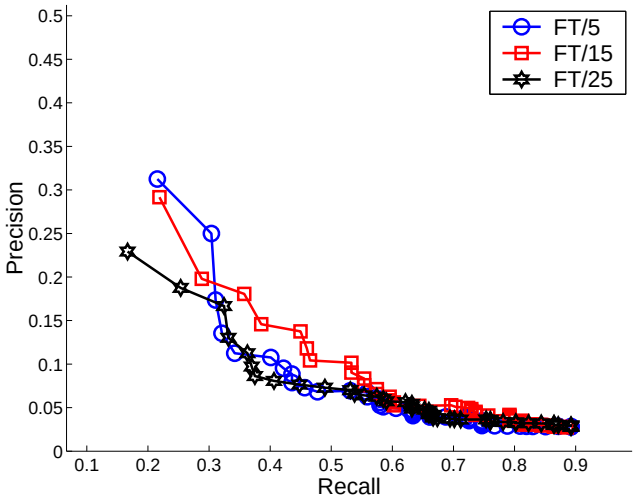
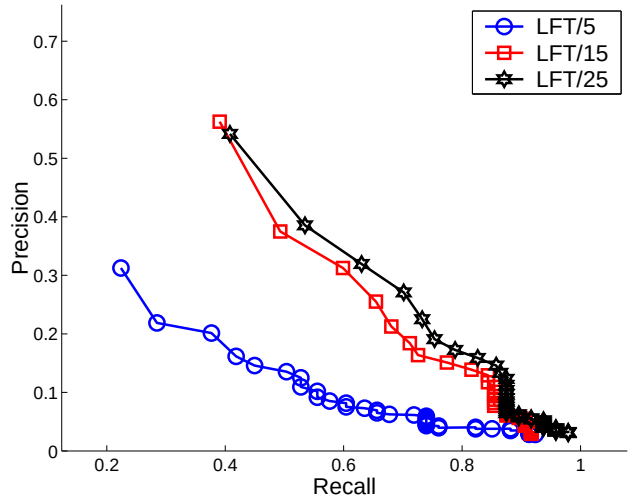


Figure 3: Precision-recall curves comparing the performance of three tempo models for queries consisting of five seconds (top), fifteen second (middle), and twenty five seconds (bottom).





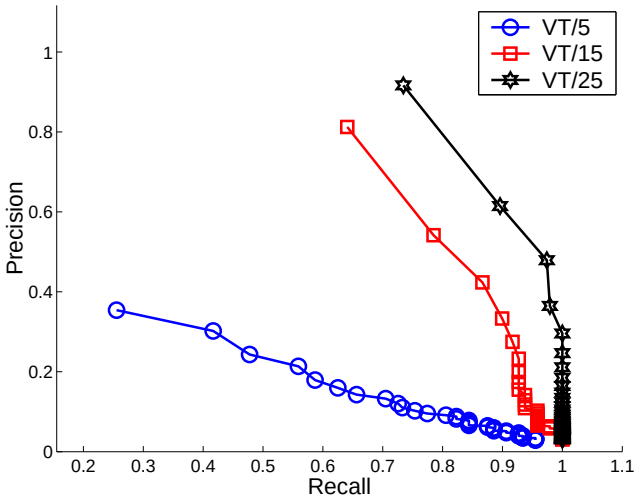


Figure 4: Precision-recall curves comparing the performance of each of the tempo models for three different query lengths.

			Spectral Distribution Model					
			HSN			HIN		
			AvgP	Cov	Oerr	AvgP	Cov	Oerr
Melody length (sec.)	25	VT	**0.95**	**0.21**	**0.08**	0.92	0.40	0.10
		LFT	0.66	5.90	0.46	0.63	5.98	0.48
		FT	0.34	20.69	0.77	0.33	22.46	0.79
	15	VT	**0.86**	**1.75**	**0.19**	0.83	3.02	0.19
		LFT	0.66	8.10	0.44	0.66	8.15	0.42
		FT	0.38	19.83	0.71	0.36	19.08	0.73
	5	VT	**0.51**	**10.67**	**0.65**	0.46	11.83	0.69
		LFT	0.43	17.33	0.69	0.37	17.94	0.75
		FT	0.38	22.96	0.69	0.35	21.67	0.75

Table 2: Retrieval results

gained by accurate tempo modeling which takes into account the variability in tempo is rather small. Thus, as the query duration grows the power of the variable tempo model is better exploited. The *Locally Fixed Tempo* can capture the average tempo of a performance but clearly fails to capture changes in the tempo. Since the chance of a tempo change grows with the duration of the query the average tempo stops from being a good approximation and we do not see further improvement in the retrieval quality.

In Fig 3 we give precision-recall graphs that compare the three tempo models. Each graph compares FT, LFT and VT for different query durations. The VT model clearly outperforms both the FT and LFT models. The longer the query the wider the gap in performance. In Fig 4 we compare the precision-recall graphs for each model as a function of the query duration. Each graph shows the precision-recall curves for 5, 15, and 25 seconds queries. We again see that only the VT model consistently improves with the increase in the query duration. Using a globally fixed tempo (FT) is clearly inadequate as it results in very poor performance – precision is never higher than 0.35 even for low level of recall. The performance of the LFT model is more reasonable. A precision of about 0.5 can be achieved for a recall value of 0.5. However, the full power of our approach is utilized only when we use the VT model. We achieve an average precision of 0.92 with a recall of 0.75. It seems that with the VT model we reach an overall performance that can serve as the basis for large scale music retrieval systems.

Lastly, as a final sanity check of the conjecture of the robustness of the VT model we used the VT and LFT model with three long melody queries (one minute) and applied the retrieval and alignment process. We then let a professional musician listen to the segmentation and browse the segmented spectogram. An example of a spectogram with a segmentation of the VT model is given in Fig 5. The example is of a performance where the energy of accompaniment is higher than the energy of the leading tenor. Nonetheless, a listening experiment verified that our system was able to properly segment and align the melody posed by the query. Although these perceptual listening tests are subjective, the experiments indicated that the VT model also provides an accurate alignment and segmentation.

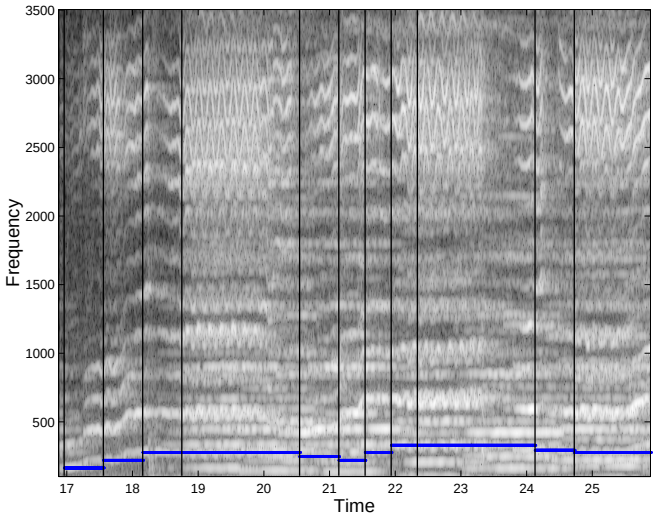


Figure 5: An illustration of the alignment and segmentation of the VT model. The pitches of the notes in the melody are overlayed in solid lines.

6. DISCUSSION

In this paper we presented a robust probabilistic model for query by melody. The proposed approach is simple to implement and was found to work well on polyphony-rich recordings with various types of accompaniments. The probabilistic model that we developed focuses on two main sources of variability. The first is variations in the actual durations of notes in real recordings (tempo variability) and the second is the variability of the spectrum mainly due to the "spectral masking" of the leading vocal by the accompanying vocals and orchestra. In this work we assumed that the pitch information in a query is accurate and only the duration can be altered in the performance. This assumption is reasonable if the queries are posed using a symbolic input mechanism such as a MIDI keyboard. However, an easier and more convenient mechanism is to hum or whistle a melody. This task is often called "query by humming". In addition to the tempo variability and spectral masking, a query by humming system also needs to take into account imperfections in the pitch of the hummed melody. Indeed, much of the work on query by humming have been devoted to music retrieval using noisy pitch information. The majority of the work on query by humming though have focused on search of noisy queries in symbolic databases. Since the main thrust of this research is searches in real polyphonic recordings, it complements the research on query by humming and can supplement numerous systems that search in symbolic databases. We plan to extend our algorithm so it can be combined with a front end for hummed queries. In addition, we have started conducting research on supervised methods for musical genre classification. We believe that by combining highly accurate genre classification with a robust retrieval and alignment we will be able to provide an effective tool for searching and browsing for both professionals and amateurs.

Acknowledgments

We would like to thank Moria Koffman for her help in creating the queries used in the experiments and Leo Kontorovitch for useful comments on the manuscript.

7. REFERENCES

[1] Arien. famous tenor arias. Decca 450 005-2.

[2] Best of opera. Disky 701812.

[3] Les 40 tenors. EMI 5720072 (Two CDs).

[4] Luciano pavarotti. nessun dorma, arias and duets. Decca 467 462-2.

[5] The young domingo. BMG 63527 (Two CDs).

[6] J. Aucouturier and S. M. Segmentation of musical signals using hidden markov models. *Audio Engineering Society*, May 2001.

[7] R. Dannenberg. An on-line algorithm for real-time accompaniment. *Proc. Int'l Computer Music Conference*, 1984.

[8] T. Dean and K. Kanazawa. A model for reasoning about persistent and causation. *Computational Intelligence*, 5(3):142–150, 1989.

[9] A. S. Durey and M. A. Clements. Melody spotting using hidden Markov models. In *Proceedings of the International Symposium on Music Information Retrieval*, pages 109–117, Bloomington, IN, October 2001.

[10] J. Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999.

[11] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. Query by humming: Musical information retrieval in an audio database. In *ACM Multimedia*, pages 231–236, 1995.

[12] K. Lemstrom and J. Tarhio. Searching monophonic patterns within polyphonic sources. In *Proc. Content-Based Multimedia Information Access (RIAO'2000)*, pages 1261–1279, April 2000.

[13] R. McNab, L. Smith, D. Bainbridge, and I. Witten. The new zealand digital library melody index, 1997.

[14] M. Mongeau and D. Sankoff. Comparison of musical sequences. *Computers and the Humanities, 24:161– 175*, 1990.

[15] C. Raphael. Automatic segmentation of acoustic musical signals using hidden markov models. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 21(4), Apr. 1999.

[16] J. Tabrikian, S. Dubnov, and Y. Dickalov. Speech enhancement by harmonic modeling via map pitch tracking. *ICASSP2002, Orlando, Florida*, 2002.

[17] G. Tzanetakis and P. Cook. Audio information retrieval (air) tools. In *Proc. International Symposium on Music Information Retrieval*, 2000.

[18] P. Walmsley, S. Godsill, and P. Rayner. Polyphonic pitch tracking using joint bayesian estimation of multiple frame parameters. In *Proc. 1999 Ieee Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 1999.