An End-to-End Machine Learning System for Harmonic Analysis of Music

Yizhao Ni, Matt McVicar, Raúl Santos-Rodríguez, and Tijl De Bie

***Abstract*—We present a new system for the harmonic analysis of popular musical audio. It is focused on chord estimation, although the proposed system additionally estimates the key sequence and bass notes. It is distinct from competing approaches in two main ways. First, it makes use of a new improved chromagram representation of audio that takes the human perception of loudness into account. Furthermore, it is the first system for joint estimation of chords, keys, and bass notes that is fully based on machine learning, requiring no expert knowledge to tune the parameters. This means that it will benefit from future increases in available annotated audio files, broadening its applicability to a wider range of genres. In all of three evaluation scenarios, including a new one that allows evaluation on audio for which no complete ground truth annotation is available, the proposed system is shown to be faster, more memory efficient, and more accurate than the state-of-the-art.**

***Index Terms*—Audio chord estimation, harmony progression analyzer (HPA), loudness-based chromagram, machine learning, meta-song evaluation.**

I. Introduction

BESIDES by the melody, the quality of Western tonal music is strongly determined by its chords and chord progressions, key, and bassline [1]. For example, a minor chord often evokes a more melancholic mood, while a major chord may sound more positive. Furthermore, various genres will make use of different types of chord progressions of different complexities (see, e.g., Section IV-C in [2]), with jazz music typically being the most complex in this respect. The interplay and dynamics of chords are therefore essential in the way humans perceive and experience music [1], [3]. Unfortunately, transcribing chords in music audio is nontrivial, even for trained musicians, and hard to automate.

Manuscript received September 14, 2011; revised December 02, 2011; accepted February 06, 2012. Date of publication February 20, 2012; date of current version April 04, 2012. This work is supported in part by the EPSRC under Grant EP/G056447/1 and in part the PASCAL2 EU Network of Excellence. The work of M. McVicar was supported by the Bristol Centre of Complexity Sciences (BCCS), University of Bristol. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Daniel P. W. Ellis.

Y. Ni, M. McVicar, and T. De Bie are with the Intelligent Systems Lab, Department of Engineering Mathematics, University of Bristol, Bristol BS8 1UB, U.K. (e-mail: Yizhao.NI@gmail.com, matt.mcvicar@bris.ac.uk, tijl.debie@gmail.com).

R. Santos-Rodríguez is with the Signal Theory and Communications Department, Universidad Carlos III de Madrid, 28903 Getafe, Spain (e-mail: rsrodriguez@tsc.uc3m.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2012.2188516

Chord estimation as we define it in this paper is the task of automatically detecting chord labels and boundaries from the audio of a musical piece. Loosely speaking, the structure of most approaches to this task consists of dividing a song into a high time resolution sequence of windows, known as *time frames*. Subsequently, an algorithm is used to assign a chord label to each frame, based on the features extracted and the local context. Key estimation from music audio involves a similar procedure in order to determine the key and key changes during the song. In recent years, audio chord estimation and tonal key estimation have been very active fields [2], [4]–[20] due to the increasing popularity of using such mid-level tonal features in music information retrieval (MIR). Of course, the chords, keys, and basslines of a song are not independent. Thus, addressing them simultaneously seems a natural strategy, which we adopt in this paper.

The annual Music Information Retrieval Evaluation eXchange (MIREX[1]) evaluation has a task dedicated to chord estimation, where participants attempt to predict chord labels and boundaries for a collection of songs. The chords in this evaluation have been simplified to an alphabet of 25 classes: 12 major chords, 12 minor chords, and a no-chord symbol for periods of silence, speaking or other times when no chord label can be assigned. In the MIREX *Audio Chord Estimation* (ACE) evaluation 2010, an expert knowledge based system [11] achieved the best performance, with 80.22% chord estimation accuracy on a collection of The Beatles, Queen, and Zweieck songs. Just one year later, three machine learning (ML)-based systems: [18], [20] together with the system proposed in this paper, have broken this record and achieved state-of-the-art performances[2]. Even though the results seem to be very promising, a number of problems are impeding further progress. For instance, most metrics do not consider chords other than major and minor, restricting the development of more complicated ACE systems. Furthermore, due to the limited amount of the data, it is becoming increasingly probable that the existing ACE systems (and "pre-trained" expert-systems in particular) are overfitting MIREX evaluation data.

Notation and Definitions

Let $\mathbf{x} = [x_1, \ldots, x_s, \ldots, x_S]$ be a mono audio signal with x_s indicating the value of the sth sample. In chord estimation research, this signal is usually converted into a 12-dimensional representation of the harmonic content, one such vector for each time frame (indexed by t with $1 \leq t \leq T$). This vector is known as a *chroma* [4] vector, and it reflects the distribution of salience

[1]http://www.music-ir.org/mirex/wiki/MIREX_HOME

[2]http://nema.lis.illinois.edu/nema_out/mirex2011/results/ace/

1558-7916/$31.00 © 2012 IEEE

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

over the 12 pitch classes. In this paper the sequence of chromagrams for the audio signal $\mathbf{x}$ are gathered as the columns of a matrix $\bar{\mathbf{X}} \in \mathbb{R}^{12\times T}$, with T indicating the number of frames. The key, chord and bass note annotations are denoted by $\mathbf{k} \in \mathcal{A}_k^{1\times T}$, $\mathbf{c} \in \mathcal{A}_c^{1\times T}$, and $\mathbf{b} \in \mathcal{A}_b^{1\times T}$, with $\mathcal{A}_k$, $\mathcal{A}_c$ and $\mathcal{A}_b$ representing the alphabets of keys, chords, and bass notes, respectively. Suppose we have a collection of N pieces of audio. We will then denote the collection of chromagrams and annotations for keys, chords and bass notes as $\mathcal{X} = \{\bar{\mathbf{X}}^n \in \mathbb{R}^{12\times T_n}\}_{n=1}^N$, $\mathcal{K} = \{\mathbf{k}^n \in \mathcal{A}_k^{1\times T_n}\}_{n=1}^N$, $\mathcal{C} = \{\mathbf{c}^n \in \mathcal{A}_c^{1\times T_n}\}_{n=1}^N$, and $\mathcal{B} = \{\mathbf{b}^n \in \mathcal{A}_b^{1\times T_n}\}_{n=1}^N$.

II. Background

In the prevailing scheme of key/chord/bass note estimation, two stages can be distinguished: the extraction of the chromagram and the estimation of the keys/chords/bass notes based on these chromagrams. We will here briefly summarize the state-of-the-art on these two aspects.

A. Chromagram Extraction

The following building blocks and strategies have been used in the literature for the computation of chroma features:

- *Short-Time Fourier Transform (STFT)*. The computation of chromagrams based on the STFT was proposed in [4]. It has been applied to various key/chord estimation tasks in the early 2000s (e.g., [5]) and is implemented in the MIR toolbox [21]. To compute chromagrams, for each musical frequency the signal power in nearby frequency bins is summed, after which these energies are wrapped onto one octave to yield 12 pitch class salience.
- *Constant Q Transform*. Proposed by [22], the constant Q transform forms the basis of many chromagrams, such as the NNLS chromagram [23]. This transform is inspired by the fact that musical pitches are equidistant in log-frequency, and thus it is more reasonable to calculate pitch class energy in the log-frequency domain. This is equivalent to using filter bandwidths that increase linearly with the frequency of the pitch that is filtered out.
- *Harmonic/Percussive Sound Separation (HPSS)*. The authors of [24] exploited the fact that harmony typically has a well-defined frequency content, whilst percussive information is typically localized in time to separate the harmonic from the percussive components in the audio. They showed that chromagrams computed from the reconstructed harmonic content only yields better chord estimation performance.
- *Reducing Local Variations of Chords*. The majority of ACE systems require the selection of a frame-wise hop rate, which is normally faster than chord changes to capture chord boundaries. However, this may result in chromagrams that are sensitive to local noise signal and chord variations [25]. To reduce local variation one can apply *beat synchronization*, a process of summarizing (usually taking the mean/median of) frame-wise features that occur between two beats, yielding fewer but longer beat-synchronous frames [7], [26]. The rationale for doing this is that chord labels between two consecutive beats tend to be the same; hence, the use of a beat-synchronous frame eliminates the effect of local variation. A recent approach related to this is to consider the boundary information provided by repeated segments of a song, then smoothing chroma vectors corresponding to the same repetition to reduce variation [20].
- *Tonal Centroid Re-Weighting*. Based on the observation that close harmonic relations such as fifths and major/minor thirds have relatively large Euclidean distances in the pitch class space, in [27] it was suggested to instead use a six-dimensional projection of the chroma vector, known as a tonal centroid, resulting in a musically more sensible metric embedding. Tonal centroid features were experimentally shown to be superior to constant Q chromagrams in [15], but they were shown to be inferior to beat-synchronous chromagrams in the same paper.
- *Timbre Invariant Audio Re-Weighting*. Exploiting the fact that lower Mel-frequency cepstral coefficients (MFCCs) are closely related to timbre, the authors of [28] presented a chromagram that is more robust to changes in timbre. The feature regards the logarithm of the spectrum as pitch class salience and discards the lower MFCCs of the pitches before wrapping them onto one octave. The proposed CRP (chroma discrete cosine transform-reduced log pitch) feature is the first chromagram that departs from using energy as pitch class salience, and the ACE system based on this feature [20] achieved state-of-the-art performance in MIREX ACE evaluation 2011.
- *Human Auditory Perception Re-Weighting*. An attempt to extract pitch class salience based on human auditory perception was proposed in [12]. The idea is to transform different frequency spectra by arc-tangent functions so as to weigh human auditory sensitivity. As we will discuss in Section III, this scheme is similar to the technique known as A-weighting [29], but its motivation is more heuristic.

B. Estimation Algorithm

The most common method for predicting the chord sequence given chroma vectors makes use of hidden Markov models (HMMs, [30]). An HMM is a probabilistic model in which the sequence being modeled is assumed to be a Markov process of hidden variables with a parallel chain of observed variables that are dependent on these hidden variables. In chord estimation, the chromagrams (or other spectral features, see [31] and [32]) form the *observed variables* while the chords are the *hidden variables* [see the dotted box in Fig. 1(a)]. The parameters of the HMM can then be tuned by an expert, or estimated from data. For convenience, we will refer to the former type of systems as *expert systems* and to the latter as *ML based systems*.

The ML approach, pioneered for chord estimation in [5], most often estimates the parameters by expectation–maximization [33], or using maximum likelihood if a fully annotated training set is available (possibly with Laplace correction). More recently, also a discriminative parameter estimation approach [34] was adopted in [9], which directly attempts to optimize estimation performance instead of the likelihood function.

Later, it was noted that one can exploit chord transition characteristics under different tonal keys [3], such that estimating

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

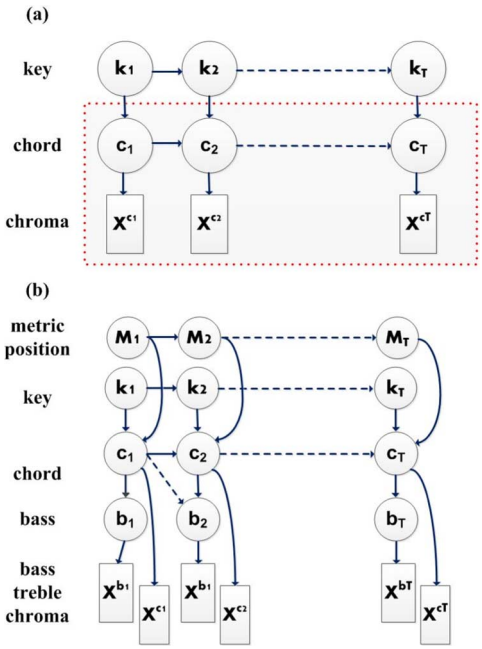

Fig. 1. Review of the development of HMM topologies for key/chord estimation systems. (b) is adapted from [11].

chords and keys simultaneously came naturally. This was achieved by deploying more complex HMM topologies, sometimes referred to as dynamic Bayesian networks [2], [13]–[15], [17], [19] These methods make use of interacting key/chord chains to propagate key-to-chord information [i.e., Fig. 1(a)]. Mathematically, this HMM topology formalizes a probability distribution $P(\mathbf{k}, \mathbf{c}, \bar{\mathbf{X}}|\Theta)$ jointly for the chroma vectors $\bar{\mathbf{X}}$ and the annotations, with Θ representing the parameters of this distribution. Given the optimal parameters Θ^*, the key/chord estimation task is equivalent to finding $\{\mathbf{k}^*, \mathbf{c}^*\}$ that maximize the joint probability: $\{\mathbf{k}^*, \mathbf{c}^*\} = \arg\max_{\mathbf{k},\mathbf{c}} P(\mathbf{k}, \mathbf{c}, \bar{\mathbf{X}}|\Theta^*)$.

For more complicated models such as those, only a handful of systems learn the parameters Θ entirely from a training set of songs and annotations [2], [15]. Most approaches are based at least partially on expert knowledge, where parameters are set on the basis of music theoretic knowledge of the developers [13], [14], [17], [19]. For example, the key and chord transition parameters can be set by an expert, often informed by perceptual key-to-key and chord-to-key relationships [3].

The superiority of this dual-chain HMM topology over the standard one has now been established, but still it is not perfect for modeling harmony progressions. For instance, different chords may share the same bass note, resulting in similar frequency content in the bass range. This information is useful for distinguishing complex chords (e.g., inversions), but it is difficult to capture by dual-chain key/chord HMMs. Although estimating the bass note of a chord by including the bassline as an additional sequence has been investigated in parallel with research on including the key [6], [8], [16], these lines of research did not converge until the publication of a novel expert system, namely the *Musical Probabilistic model* (denoted by *MP*) [11].

The structure of the MP model is shown in Fig. 1(b). It was hailed as the first system to combine most musical qualities into a single model, allowing the simultaneous inference of key, chord, and bass pitch classes. This represented a leap forward in harmonic analysis research, for the first time allowing the prediction of complex chords. However the complexity of the structure has also increased the search space and has resulted in severe memory consumption and processing time issues, restricting its practical use. Also tuning the parameters of such a complex model becomes a daunting task.

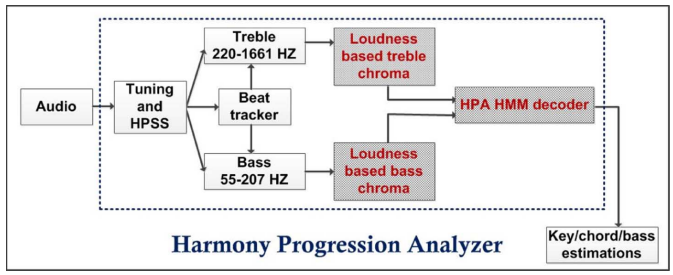

Fig. 2. Components of the proposed harmony progression analyzer (HPA). Shaded blocks show the novelties as compared to the state-of-the-art.

C. Contributions in This Paper

In the current paper, we propose a new system, *Harmony Progression Analyzer* (HPA), for harmonic analysis of audio. The HPA system integrates some important contributions made recently in the literature (e.g., using [26] to beat-synchronize chromagrams), and adds a number of novel elements. These novelties pertain to the chromagram computation and the HMM topology and parameter estimation. For the former, relying on empirical studies, we propose a simple alternative chromagram that reflects perceived loudness of pitch classes. The latter enables HPA to simultaneously estimate keys, chords as well as bass notes, allowing us to improve on [5], [7], [10], [13]–[15], [17]–[20] which can only predict a library of 25 major/minor chords. Regarding the HMM topology, HPA is similar to MP [11]. However, the HPA system differs in that its parameters are learnt from data, whereas in MP they were set by an expert. This ML foundation also enables us to utilize search space reduction techniques, significantly reducing memory consumption and computation times—both of which were significant limitations of MP [11]. More importantly, ML-based parameter tuning expands HPA's scope of applicability to a wider range of genres, as long as training data is available.

Fig. 2 illustrates the key components of the HPA, and the rest of the paper is organized based on this flow chart. HPA first tunes the set of frequencies on the equal-tempered scale [35] and extracts the harmonic part of the audio via *HPSS* [24]. It then calculates the proposed beat-synchronous *loudness based chromagram* (denoted by LBC), motivated and detailed in Section III. The chromagrams are eventually passed through the HPA HMM decoder (described in Section IV) to estimate keys, chords and bass notes. In Section V, we discuss our evaluation methods, including a new strategy that relies on possibly noisy untimed chord sequences ubiquitous on guitar tab websites [36], [37], instead of requiring accurate annotations for the test songs. The experimental results and discussions are given in Section VI. Finally, we draw the conclusions and propose directions of future work in Section VII.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

The software to train and use the HPA system is made in MATLAB. It is freely available for research purposes, including comprehensive documentation.[3]

III. Loudness-Based Chromagram

In this paper, we note that perception of loudness is not linearly proportional to the power or amplitude spectrum, and hence existing chromagrams typically do not accurately reflect human perception of the audio's spectral content. Indeed, the empirical study in [38] showed that loudness is approximately linearly proportional to the so-called *sound pressure level* (SPL), defined as the $\log_{10}$ of the power spectrum.

Let F be the set of frequencies on the equal-tempered scale (possibly tuned to a particular song) over a reasonable range. Then a typical chromagram extraction approach first computes the energy (or amplitude) $\mathbf{X} \in \mathbb{R}^{|F|\times T}$ for all frequencies $f \in F$ at all time frame indices $t \in \{1,\ldots,T\}$. Then $X_{f,t}$ reflects the salience at frequency f and frame t. Here, we suggest to use the $\log_{10}$ of the power spectrum instead, which leads to the SPL matrix denoted and computed as follows:

$$\mathcal{L}_{f,t} = 10\log_{10}\left(\frac{\|X_{f,t}\|^2}{\|p_{ref}\|^2}\right), f \in F, t = 1,\ldots,T \quad (1)$$

where p_{ref} indicates a reference pressure level and, with s_t the sample time corresponding to frame index t

$$X_{f,t} = \frac{1}{L_f}\sum_{n=0}^{L_f-1} x_{\lceil s_t - L_f/2\rceil + n} w_{n,f} \exp\left\{\frac{-j2\pi Q n}{L_f}\right\} \quad (2)$$

is a constant Q transform [22] and $w_{n,f}$ is the hamming window. The frequency dependent bandwidth L_f is defined as $L_f = Q(SR/f)$, where Q represents the constant resolution factor and SR is the sampling rate of $\mathbf{x}$.

Furthermore, low and high frequencies require higher sound pressure levels for the same perceived loudness as mid-frequencies [38]. To compensate for this, we propose to use *A-weighting* [29] to transform the SPL matrix into a representation of the perceived loudness of each of the pitches as follows:

$$\mathcal{L}'_{f,t} = \mathcal{L}_{f,t} + A(f), f \in F, t = 1,\ldots,T \quad (3)$$

where

$$R_A(f) = \frac{12200^2 \cdot f^4}{(f^2+20.6^2)\cdot\sqrt{(f^2+107.7^2)(f^2+737.9^2)}\cdot(f^2+12200^2)}$$
$$A(f) = 2.0 + 20\log_{10}(R_A(f)). \quad (4)$$

It is known that loudnesses are additive if they are not close in frequency [39]. This allows us to sum up loudness of sounds in the same pitch class, yielding

$$X'_{p,t} = \sum_{f\in F}\delta(M(f)+1,p)\mathcal{L}'_{f,t}, p = 1,\ldots,12, t = 1,\ldots,T. \quad (5)$$

Here δ denotes an indicator function and

$$M(f) = \left(\left\lfloor 12\log_2\left(\frac{f}{f_A}\right) + 0.5\right\rfloor + 69\right) \bmod 12 \quad (6)$$

with $f_A = 440$ Hz a reference frequency. Finally, to account for the fact that overall sound level should be irrelevant in estimating harmonic content, our loudness-based chromagram $\bar{\mathbf{X}} \in \mathbb{R}^{12\times T}$, is obtained by normalizing $\mathbf{X}$:

$$\bar{X}_{p,t} = \frac{X'_{p,t} - \min_{p'} X'_{p',t}}{\max_{p'} X'_{p',t} - \min_{p'} X'_{p',t}} \forall\, p,t. \quad (7)$$

Note that this normalization is invariant with respect to the reference level and hence a specific p_{ref} is not required. Note also that the normalization is a nonlinear operation, such that the effect of A-weighting becomes nonlinear too, deemphasizing low-frequency SPLs.

The human auditory perception re-weighting proposed in [12] aims at modeling loudness perception as well, but it uses arc-tangent functions (instead of the established A-weighting) to convert the pitch class energy (rather than SPLs) to a perceptual salience.

Fig. 3 illustrates an example of LBC compared to two recent chromagrams: NNLS and CRP.[4] Since NNLS regards the spectral energy as pitch class salience, the loudness of low-energy contents might be underestimated. The advantage is that the background spectrum and high-energy pitches are well-separated, but it also risks filtering out relevant low-energy pitches (e.g., the G note in the second C:maj chord in Fig. 3) and yields inaccurate chroma vectors in these cases.

In contrast, chromagrams using a logarithmic transformation of the spectrum (i.e., CRP and LBC) emphasize variations among low-energy pitches, reflecting the fact that these are perceptually stronger than their energy suggests. A possible risk is that this also amplifies the noise. Compared with CRP, the fundamental difference of LBC is that it focuses on modeling the loudness of the pitches, instead of eliminating timbre influence. For CRP this seems to result in a larger background spectrum, such that it requires postprocessing such as in [20] or a special emission probability model used in [40]. LBC on the other hand shows a relatively good balance between noise reduction and low-energy pitch preservation, and our experiments confirm this (see Section VI-A).

In order to extract LBC features, we first convert audio signals to mono 11 025 Hz, and separate the harmonic and percussive elements with our implementation of the *HPSS* algorithm. After tuning [35] we compute a bass and treble LBC for each song (denoted $\bar{\mathbf{X}}^{\mathbf{b}}$ and $\bar{\mathbf{X}}^{\mathbf{c}}$, respectively). The bass frequency range used is A1 (55 Hz) to $G\sharp 3$ ($\sim$ 207.65 Hz), and the treble frequency range is A3 (220 Hz) to $G\sharp 6$ ($\sim$ 1661.22 Hz), jointly covering five octaves. The beat tracker then estimates beat positions using our implementation of [26] and takes the median chroma feature between consecutive beats. To train the HPA HMM decoder, it also beat synchronizes the key/chord/bass annotations by taking the most prevalent labels between beats. A median feature vector (with the corresponding beat-synchronous annotations when training) is then treated as one time frame, which is passed to the HPA HMM decoder for training or decoding chord sequences.

[3]https://patterns.enm.bris.ac.uk/hpa-software-package

[4]The parameters for generating the chromagrams are specified in Section VI-A.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

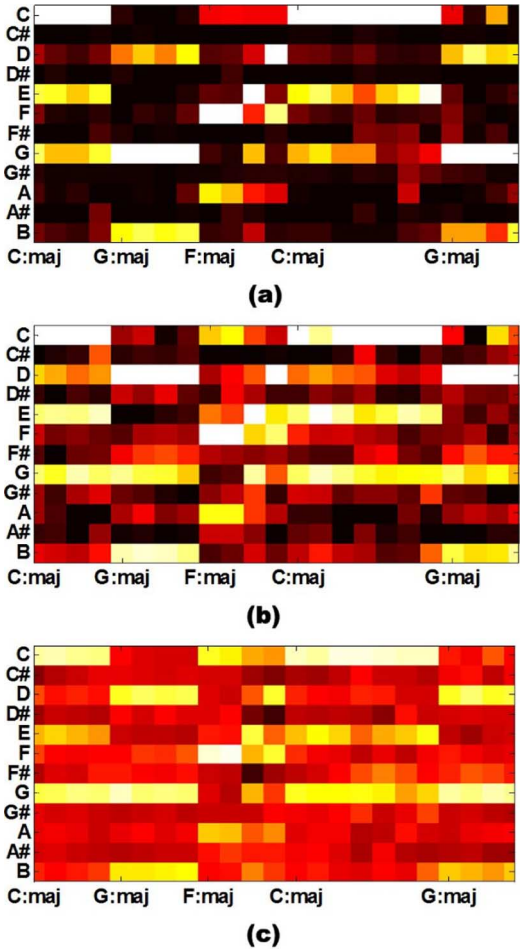

Fig. 3. Comparison of three chromagrams. (a) NNLS, (b) LBC, (c) CRP, on the track "Let It Be" (Lennon/McCartney) (segment 6.5 s to 16.5 s). The x-axis labels the chord progression and the y-axis represents the pitch classes.

IV. Harmony Progression Analyzer

In this section, we describe the *HMM topology* adopted for the HPA HMM decoder, the *parameter estimation* approach, and the *search space reduction* techniques.

A. HMM Topology

As depicted in Fig. 4, HPA's adopted HMM topology consists of three hidden and two observed variable layers. The hidden variables correspond to the key $\mathcal{K}$, the chord label $\mathcal{C}$ and the bass $\mathcal{B}$ annotations. Under this representation, a chord is decomposed into two aspects: chord label and bass note. For instance, take the chord A:maj/3: the chord state is $c = \mathrm{A{:}maj}$ and the bass state is $b = \mathrm{C\#}$. Correspondingly, we use the two LBC chromagrams discussed above: $\bar{\mathbf{X}}^{\mathbf{c}}$ for the treble frequency range, which is emitted by the hidden chord sequence $\mathbf{c}$, and $\bar{\mathbf{X}}^{\mathbf{b}}$ for the bass range, emitted by the hidden bass sequence $\mathbf{b}$. This decomposition will allow HPA to distinguish between complex chords and inversions.

Under this framework, the set Θ of a HPA HMM topology has the following parameters:

$$\Theta = \{p_i(k), p_i(c), p_i(b), p_t(k|\bar{k}), p_t(c|\bar{c}, k), p_t(b|c), p_t(b|\bar{b}), p_e(\bar{\mathbf{X}}^{\mathbf{c}}|c), p_e(\bar{\mathbf{X}}^{\mathbf{b}}|b)\} \quad (8)$$

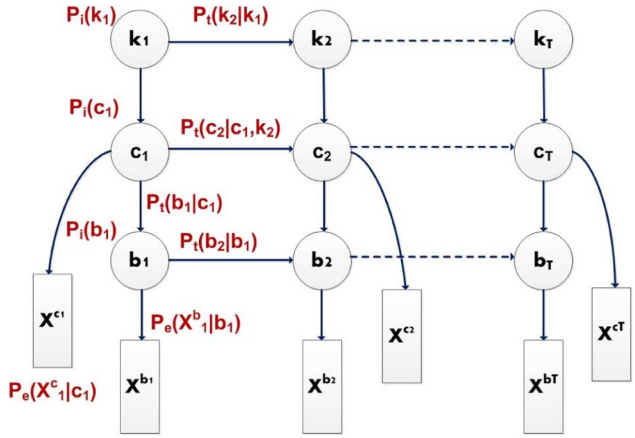

Fig. 4. HMM topology of the HPA system. The probability functions are parameters of the system, which are learnt via maximum-likelihood estimation.

where p_i, p_t, and p_e denote the initial, transition and emission probabilities respectively. The joint probability of the feature vectors $\{\bar{\mathbf{X}}^{\mathbf{c}}, \bar{\mathbf{X}}^{\mathbf{b}}\}$ and the corresponding annotation sequences $\{\mathbf{k}, \mathbf{c}, \mathbf{b}\}$ of a song is then given by the formula[5]

$$P(\bar{\mathbf{X}}^{\mathbf{c}}, \bar{\mathbf{X}}^{\mathbf{b}}, \mathbf{k}, \mathbf{c}, \mathbf{b}|\Theta) = p_i(k_1)p_i(c_1)p_i(b_1)\prod_{t=2}^{T} p_t(k_t|k_{t-1})$$
$$p_t(c_t|c_{t-1}, k_t)p_t(b_t|c_t)p_t(b_t|b_{t-1})\prod_{t=1}^{T} p_e(\bar{\mathbf{X}}_t^{\mathbf{c}}|c_t)p_e(\bar{\mathbf{X}}_t^{\mathbf{b}}|b_t). \quad (9)$$

B. Parameter Estimation

The parameters $p_i(k)$, $p_i(c)$, $p_i(b)$ correspond to the initial probabilities of key, chord and bass respectively. They can be learnt via *maximum likelihood estimation* (MLE), e.g., $p_i(k) = \#(k_1 = k)/\#k_1, \forall\, k \in \mathcal{A}_k$, where $\#$ indicates number of.

For the transitions, $p_t(c|\bar{c}, k)$ represents the probability of a chord change under a certain key. Since the chord transition is strongly influenced by the underlying key [13], this probability is modeled as key dependent. Under the assumption that relative chord transitions are key independent, we transposed all sequences to a common key k and learned $p_t(c|\bar{c}, k)$ from the transposed sequences. This multiplied the effective amount of data by 12 in computing the MLE solution

$$p_t(c|\bar{c}, k) = \frac{\#(c_t = c\ \&\ c_{t-1} = \bar{c}\ \&\ k_t = k)}{\sum_{c'} \#(c_t = c'\ \&\ c_{t-1} = \bar{c}\ \&\ k_t = k)},\ \forall\, c, \bar{c}, k. \quad (10)$$

Note that this transposition trick has also been employed by other researchers, including [10] and [15]. Since it allows us to get 12 times as much information from the data, the method is also applied to the estimations of $p_t(k|\bar{k})$, $p_t(b|c)$, and $p_t(b|\bar{b})$.

Apart from modeling chord transitions, $p_t(k|\bar{k})$ is applied to model key changes during a song, of which the MLE solution is calculated as follows:

$$p_t(k|\bar{k}) = \frac{\#(k_t = k, k_{t-1} = \bar{k})}{\sum_{k'} \#(k_t = k', k_{t-1} = \bar{k})}, \forall\, k, \bar{k} \in \mathcal{A}_k. \quad (11)$$

[5]Note that we use $p_t(b_t|b_{t-1}, c_t) = p_t(b_t|c_t)p_t(b_t|b_{t-1})$, which from a purely probabilistic perspective is not correct. However, this simplification reduces computational cost and results in better performance in practice.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

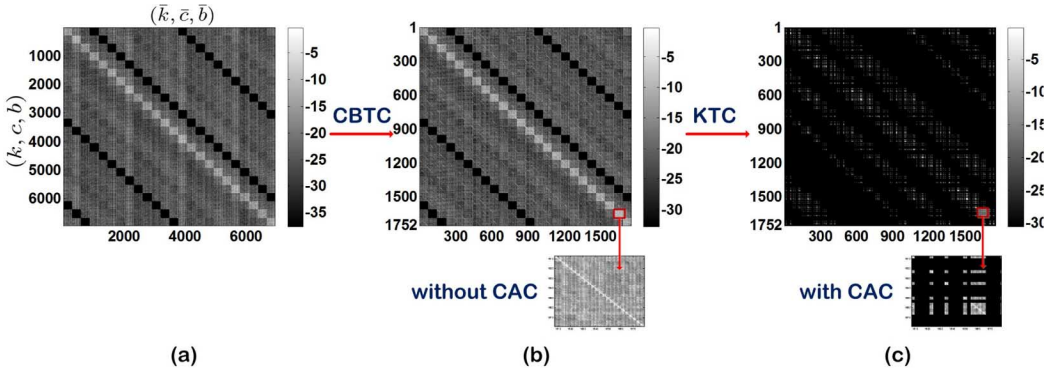

Fig. 5. Example of applying search space reduction techniques to the transition probability matrix $p'_t((k,c,b)|(\bar{k},\bar{c},\bar{b}))$ (the probabilities are in $\log$ form) of a song. The song used in this example is "Ticket to Ride" (Lennon/McCartney). Using chord to bass transition constraint (CBTC) to constrain available tuples, the dimension of the transition probability matrix can be reduced [Fig. (a) to Fig. (b)]. From Fig. (b) to Fig. (c) the matrix becomes sparser via constraining possible key transitions (KTC), indicating a further reduction of search space. A similar effect can be observed by comparing details of Fig. (b) and Fig. (c), when the chord alphabet constraint (CAC) is applied.

The parameter $p_t(b|c)$ models the probability of a bass note under a chord label so as to capture chord inversions:

$$p_t(b|c) = \frac{\#(b_t = b, c_t = c)}{\sum_{b'} \#(b_t = b', c_t = c)}, \forall\, b \in \mathcal{A}_b, \forall\, c \in \mathcal{A}_c. \quad (12)$$

A transition link $p_t(b|\bar{b})$ is also added, with the purpose of modeling the continuity of bass notes and capturing ascending and descending bassline progressions, which is given by

$$p_t(b|\bar{b}) = \frac{\#(b_t = b, b_{t-1} = \bar{b})}{\sum_{b'} \#(b_t = b', b_{t-1} = \bar{b})}, \forall\, b, \bar{b} \in \mathcal{A}_b. \quad (13)$$

Finally, the emission probabilities $p_e(\bar{\mathbf{X}}^{\mathbf{c}}|c)$ and $p_e(\bar{\mathbf{X}}^{\mathbf{b}}|b)$ are modeled as 12-dimensional single Gaussians, of which the mean vectors and full covariance matrices are also learnt via MLE.

The MLE estimations of Θ reveal the fundamental qualitative difference between HPA and MP [11], of which the parameters are tuned manually. Compared with MP, HPA can be easily retrained as new data becomes available. In this sense, it is potentially more applicable and flexible to a wider range of harmonic analysis tasks.

C. Search Space Reduction

Given the optimal parameters Θ^* via MLE, the decoding task can be formalized as the computation of the key, chord and bass sequences $\{\mathbf{k}^*, \mathbf{c}^*, \mathbf{b}^*\}$ that maximize the joint probability

$$\{\mathbf{k}^*, \mathbf{c}^*, \mathbf{b}^*\} = \arg\max_{\mathbf{k},\mathbf{c},\mathbf{b}} P(\bar{\mathbf{X}}^{\mathbf{c}}, \bar{\mathbf{X}}^{\mathbf{b}}, \mathbf{k}, \mathbf{c}, \mathbf{b}|\Theta^*). \quad (14)$$

Mathematically, this is equivalent to a standard HMM with a tuple of observed and hidden variables

$$P(\bar{\mathbf{X}}^{\mathbf{c}}, \bar{\mathbf{X}}^{\mathbf{b}}, \mathbf{k}, \mathbf{c}, \mathbf{b}|\Theta^*) = \prod_{t=1}^{T} p'_e\big((\bar{\mathbf{X}}^{\mathbf{c}}_t, \bar{\mathbf{X}}^{\mathbf{b}}_t)|(c_t, b_t)\big)$$
$$p'_i\big((k_1, c_1, b_1)\big) \prod_{t=2}^{T} p'_t\big((k_t, c_t, b_t)|(k_{t-1}, c_{t-1}, b_{t-1})\big) \quad (15)$$

with the following emission and transition probabilities $\forall\ (k,c,b)$:

$$\begin{cases} p'_i\big((k,c,b)\big) = p_i(k)p_i(c)p_i(b) \\ p'_t\big((k,c,b)|(\bar{k},\bar{c},\bar{b})\big) = p_t(k|\bar{k})p_t(c|\bar{c},k)p_t(b|c)p_t(b|\bar{b}) \\ p'_e\big((\bar{\mathbf{X}}^{\mathbf{c}}, \bar{\mathbf{X}}^{\mathbf{b}})|(c,b)\big) = p_e(\bar{\mathbf{X}}^{\mathbf{c}}|c)p_e(\bar{\mathbf{X}}^{\mathbf{b}}|b). \end{cases} \quad (16)$$

The computational complexity of solving this task with the *Viterbi* algorithm [30] is $O\big(|\mathcal{A}_k|^2|\mathcal{A}_c|^2|\mathcal{A}_b|^2|T|\big)$. This complexity is high due to the large number of possible chord/key/bass note combinations (see Fig. 5(a) for an example), especially when one would like to use a large chord vocabulary [11]. In order to reduce the decoding time, we introduce three ML-based search space reduction techniques, of which the effectiveness is also shown in Fig. 5. Although the necessity of search space reduction has been explored before [11], as far as we aware we are the first to *learn* the thresholds from data.

1) Chord Alphabet Constraint (CAC): It is unlikely that all chords will be used in a single song. Therefore, if it is possible to find out which chords are used in a song, we will be able to constrain the chord alphabet without loss of performance. One ML-based method is to utilize two-stage estimation. In particular, using a simple HMM with only chords as the hidden chain, we first apply a max-Gamma decoder [30] to a chromagram $\bar{\mathbf{X}}$ and obtain a first estimation. From this prediction we derive a unique chord alphabet $\mathcal{A}'_c$ for this song. Then, we force the HPA HMM chord transition probability to be zero for chords which are absent in this alphabet:

$$p'_t(c|\bar{c},k) = \begin{cases} p_t(c|\bar{c},k) & \text{if } c, \bar{c} \in \mathcal{A}'_c \\ 0 & \text{otherwise.} \end{cases} \quad (17)$$

2) Key Transition Constraint (KTC): Music theory dictates that not all key changes are equally likely. If a song does change key, the modulation is most likely to move to a related key [3]. Thus, we suggest to rule out a priori the key transitions that are seen the least often in the training set, which is equivalent to applying a threshold pruning in dynamic programming [41]. Formally, this can be done by constraining the key transition probability as

$$p'_t(k|\bar{k}) = \begin{cases} p_t(k|\bar{k}), & \text{if } \#(k_t = k\ \&\ k_{t-1} = \bar{k}) > \gamma \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

where γ is a positive integer indicating the threshold.

3) Chord to Bass Transition Constraint (CBTC): Similar to the key transition constraint, we can also constrain the chord to bass transitions. A constraint is imposed on $p_t(b|c)$ such that the bass notes can only be one of τ ($\tau \leq 12$) candidates for a given chord, which is equivalent to histogram pruning in dynamic programming. Mathematically, the frequencies of each chord-to-bass emission are ranked and only the most common τ are permissible:

$$p'_t(b|c) = \begin{cases} p_t(b|c), & \text{if } b \text{ is one of the top } \tau \text{ bass notes for } c \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

Reassuringly, when $\tau = 3$, we discovered the constraint is equivalent to using root position, first and second inversions of a chord.

V. Evaluation Strategies

As mentioned in Section I, standard metrics for evaluating ACE systems are designed for major and minor chords only. Furthermore, they require fully annotated test data (i.e., the MIREX ACE data), and overfitting this data is becoming a significant risk. To tackle these issues, we complement established metrics with a new one that allows evaluation on songs for which no accurate ground truth annotation is available. This section summarizes the three evaluation strategies used.

A. Major/Minor Chord Evaluation

In this evaluation, carried out on the MIREX ACE dataset, we restrict ourselves to a chord alphabet of 25 chords, which is consistent with the MIREX ACE evaluation. We use this evaluation to verify the effectiveness of the proposed chromagram, compare the HPA system with other ML-based systems that can not predict complex chords, as well as to analyze our search space reduction techniques.

B. Complex Chord Evaluation

This evaluation (also on the MIREX data) is designed to evaluate complex chord estimation performance [11] with 11 chord types,[6] resulting in 121 unique chords. To the best of our knowledge the only prior systems that can handle this vocabulary are MP [11] (the top-performing MIREX ACE evaluation 2010) and Chordino [23], both expert systems. In this evaluation, we first investigate the performance of different chromagrams used in the first evaluation. We then compare our HPA system with the standard HMM as well as with the two expert systems, analyzing their performances as well as memory and time efficiencies.

C. Meta-Song Evaluation

Finally, we go beyond the MIREX dataset and propose a new evaluation strategy, to which we refer as *meta-song evaluation*. This strategy can be applied to test songs for which only so-called untimed chord sequences (UCS) are available (see [36] for an example). These UCSs are noisy and do not contain chord onset times, but they can be found in abundance on guitar tab websites, making this evaluation strategy applicable to unprecedented amounts of songs of a wide variety of genres.

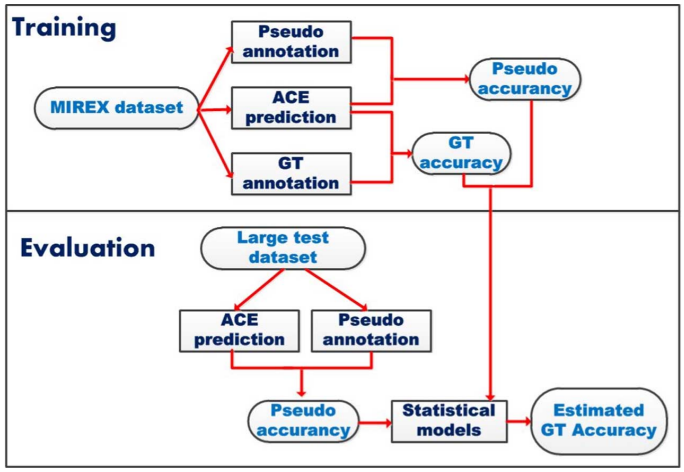

Fig. 6. Diagram of the proposed meta-song evaluation.

The meta-song evaluation works by aligning the UCS of a song with its audio (using the jump alignment method from [36]), thus constructing an approximate ground truth annotation, called a *pseudo annotation*. An ACE system can then be scored in the usual way by comparing its chord estimations against the pseudo annotations, leading to a *pseudo accuracy*. To take into account that this pseudo accuracy is bound to be somewhat biased and noisy, we trained a regression model on the MIREX data to calibrate the relation between the pseudo accuracy and the true ground truth (GT) accuracy, as well as the uncertainty on this relation. This model can then be applied to compute a confidence interval for the GT accuracy from a pseudo accuracy for a test set song. A diagrammatic representation of this approach is given in Fig. 6.

Using meta-song evaluation, we can evaluate HPA and three well-known systems (MP [11], Chordino [23], and labROSA [10]) on a much larger number of songs from different genres, allowing us to carry out an extensive analysis of the existing ACE systems, including their generalizations to different genres.

Although the present paper is the first to apply this new evaluation technique, due to space constraints the mathematical details are provided in a technical report [42]. It was also presented as a late-breaking poster at ISMIR 2011 [43].

VI. Experiments

The audio dataset used in Evaluation V-A and V-B is the same as the MIREX ACE evaluation 2011[7], which contains 217 songs by The Beatles, Queen, and Zweieck. The GT key and chord annotations were obtained from http://isophonics.net, while the bass notes were extracted directly from the GT chord annotations. Since these audio signals were extracted from remastered versions of the original CDs, they might have been shifted in the time domain (e.g., adding small periods of silence to the beginning) and also stretched. To counteract this, we followed the suggestion of the labROSA group [10] and realigned the GT

[6]maj, min, maj/3, maj/5, maj6, maj7, min7, 7, dim, aug and 'N'.

[7]http://www.music-ir.org/mirex/wiki/2011:Audio_Chord Estimation

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

annotations to the audio before the experiments. We discovered that the maximum shift was 0.4 s and the maximum required linear stretch was 0.5%.

For Evaluation V-C, an additional audio dataset consisting of 1840 songs from a variety of genres is used for evaluation, and the UCSs to create the pseudo annotations are extracted from the online chord database http://www.e-chords.com/.

A. Major/Minor Chord Evaluation

In this evaluation, we used a full key alphabet (12 major and 12 minor keys), but restricted ourselves to a chord alphabet of 25 chords. There were 13 bass states corresponding to the 12 pitch classes as well as "no bass." In accordance with the MIREX ACE train-test setup, we randomly split 2/3 of songs from each album to form the training set, while the remaining 1/3 were used for testing. Two chord evaluation metrics analogous[8] to those proposed in the MIREX ACE evaluation are used: the *chord overlap ratio* (denoted by OR), defined as $1/N\sum_n p_n$ with p_n indicating the chord estimation accuracy of the nth prediction; and the *chord weighted average overlap ratio* (denoted by WAOR), defined as $\sum_n CF_n/\sum_n TF_n$, with CF_n and TF_n denoting the number of correct and total samples of the nth prediction, respectively. Both evaluations were performed with a 1-ms sampling rate, as used in the MIREX ACE evaluation. In total 102 train-test runs were done to assess variance.[9]

1) Comparison of Chromagrams: We first compared the proposed loudness based chromagram (LBC) with the following popular chromagram implementations:

- *STFT Chromagram* (denoted by STFTC), which is implemented by the MIR toolbox [21]. The bandwidth was optimized and set as 0.19 seconds.
- *NNLS Chromagram* [23] (denoted by NNLS), which is based on a constant Q transform followed by the removal of harmonics using non-negative least squares (see [11, Ch. 5]). NNLS was acknowledged as the state-of-the-art in MIREX ACE 2010. In our experiments, we used the NNLS Vamp plugin[10] to generate this chromagram.
- *Tonal Centroid Re-Weighting* [27] applied to NNLS chromagram (denoted by TCRC).
- *CRP Chromagram* [28] (denoted by CRP), which is the latest log-pitch chromagram implemented by [44]. We optimized the parameters using cross-validation: the STFT bandwidth was set as 0.19 second, the MFCCs parameters $\gamma = 1000$, $\xi = 25$ and the smoothing window $w = 1$.
- *Human Auditory Perception* chromagram as described in [12] (denoted by HAPC).

The frequency range of all chromagrams was fixed as A1 (55 Hz, MIDI pitch 33) to $G\sharp 6$ ($\sim$ 1661.22 Hz, MIDI pitch 92), and a standard HMM system was used to generate chord estimations. For each chromagram, tuning and HPSS were applied *a priori*[11] and then two strategies were used for constructing time frames: one with a frame-wise constant hop length of 0.096s; and one with beat synchronization, where the beats are estimated by our implementation of [26].

TABLE I
PERFORMANCE OF CHROMAGRAMS AND SYSTEMS ON THE MAJOR/MINOR CHORD ESTIMATION TASK. BOLD NUMBERS REFER TO THE BEST RESULTS. THE IMPROVEMENT OF BEAT-SYNCED LBC IS SIGNIFICANT AT A LEVEL $<10^{-70}$ OVER THE PERFORMANCES OF OTHER CHROMAGRAMS UNDER A PAIRED T-TEST. THE IMPROVEMENT OF HPA IS SIGNIFICANT AT A LEVEL $<10^{-45}$ OVER THE PERFORMANCES OF OTHER SYSTEMS

CHROMA	CONSTANT HOP (OR [%], WAOR [%])	BEAT SYNCHRONIZATION (OR [%], WAOR [%])
LBC	$(71.36 \pm 1.28, 70.55 \pm 1.45)$	$(\mathbf{77.82 \pm 1.31}, \mathbf{77.22 \pm 1.24})$
STFTC	$(68.98 \pm 1.21, 67.78 \pm 1.40)$	$(73.10 \pm 1.24, 72.02 \pm 1.40)$
NNLS	$(68.05 \pm 1.50, 67.4 \pm 1.87)$	$(73.67 \pm 1.36, 72.88 \pm 1.46)$
TCRC	$(\mathbf{72.05 \pm 1.31}, \mathbf{71.71 \pm 1.58})$	$(72.75 \pm 1.41, 72.6 \pm 1.29)$
CRP	$(63.61 \pm 1.96, 63.00 \pm 1.89)$	$(67.00 \pm 2.03, 66.39 \pm 1.94)$
HAPC	$(60.60 \pm 1.16, 60.40 \pm 1.06)$	$(68.38 \pm 1.18, 68.07 \pm 1.16)$

SYSTEM	ORIGINAL FEATURE (OR [%], WAOR [%])	LBC (OR [%], WAOR [%])
labROSA	$(73.71 \pm 1.68, 72.72 \pm 1.50)$	$(75.23 \pm 1.67, 74.48 \pm 1.57)$
K-HMM	$(74.08 \pm 1.50, 73.98 \pm 1.36)$	$(78.22 \pm 1.21, 77.61 \pm 1.30)$
HPA	$(\mathbf{79.37 \pm 1.28}, \mathbf{78.82 \pm 1.23})$	$(\mathbf{79.37 \pm 1.28}, \mathbf{78.82 \pm 1.23})$

The results are shown in Table I (upper table), from which we observe a consistent improvement of beat-synchronous features over constant hop representations. This observation is consistent with results in [7], [11], [26]. In the constant hop scheme LBC performed better than the other chromagrams except TCRC. This is probably because tonal centroid re-weighting already uses harmonic information to reduce chromagram variability. Therefore, with beat synchronization the performance of LBC can be greatly improved, while the improvement of TCRC is limited. Compared with the latest NNLS and CRP chromagrams, LBC also achieves a large improvement in the beat synchronization scheme. To summarize, LBC is significantly better than all other chromagrams when beat synchronization is applied, and comparable to the best one without beat synchronization, verifying the effectiveness of the proposed chromagram.

2) Comparison of Systems: Apart from the standard HMM system [i.e., row LBC presented in Table I (upper table)], we also compared HPA with two pure ML-based systems:

- The *labROSA* ACE system [10] (denoted by labROSA), which is the implementation of [9]. It utilizes an advanced discriminative HMM [45] and does chord estimation on the basis of beat-synchronous chromagrams. In our experiments, the system is used with the transposition trick mentioned in Section IV-C to enhance performance, which is equivalent to their *EW1* submission in MIREX ACE 2010.
- The *key-specific HMM* system [2], [15] (denoted by K-HMM). This system is analogous to HPA, the main difference being the absence of the bass chain, and the use of tonal centroid re-weighting of the constant Q chromagram.

For both systems, we reported two results: one with their original chromagrams; and one with the LBC as a chromagram. Table I (lower table) shows the results. Again, we observed consistent improvements by using LBC over the original features.

[8]Although we tried to make evaluations as close to MIREX as possible, they still return slightly different results, e.g. the MP system achieved 80.22% (chord overlap ratio) and 79.45% (chord weighted average overlap ratio) in MIREX ACE 2010, but the same estimations (with the same GT annotations) only scored 78.99% (OR) and 77.74% (WAOR) in our evaluations.

[9]That is, each song in the dataset would be tested 34 times.

[10]http://isophonics.net/nnls-chroma

[11]Since NNLS and CRP have their own tuning methods, only HPSS was applied to these two chromagrams.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

When all systems use the LBC, both HPA and K-HMM outperform the standard HMM, showing the benefit obtained from the key information used. Also labROSA benefits from the LBC features, but its performance falls short even of the standard HMM, despite its use of a more sophisticated learning method. This reveals a disadvantage of discriminative HMMs: because it has a set of parameters which is many times more than that of a standard HMM, the model risks to overfit when there is insufficient training data.

We also compared HPA with K-HMM (both using LBC features) by their chord confusion matrices (Fig. 7). Clearly Fig. 7(b) is sparser than Fig. 7(a), indicating that HPA confuses fewer chord types. This may be caused by the simplification of the chord alphabet, which confuses simpler models by the resulting chord variations. For example, K-HMM has difficulty in distinguishing $\mathrm{A{:}maj} = (A, C\sharp, E)$ from $F\sharp{:}\min = (F\sharp, A, C\sharp)$ (dotted box in Fig. 7(a)). This is because simplifying $F\sharp{:}\min 7 = (F\sharp, A, C\sharp, E)$ as $F\sharp{:}\min$ increases the variation of $F\sharp{:}\min$, making it closer to A:maj. However, since $F\sharp{:}\min 7$ and A:maj do not share the same bass note, HPA can use its bass chain to distinguish them. This indicates that increasing the complexity of models, such as separating treble and bass contents, could help harmonic estimation. Nevertheless, using the bass chain also has adverse effects: if the two chords (e.g., $D{:}\mathrm{maj}/5 = (A, D, F\sharp)$ and A:maj) share the same bass note, then the estimation of HPA might be influenced by its bass chain, making it difficult to classify these chords correctly [dotted box in Fig. 7(b)]. This problem is more serious when the model is too complex for the available training data. Results corroborating this effect were recently presented in [37]. We expect this problem can be solved by better balancing the influence of chord/bass chains via discriminative methods, which may lead to a more powerful discriminative HPA system in our future work.

3) Analysis of Search Space Reduction Techniques: Finally, we investigated our search space reduction techniques. Fig. 8(a) shows that using a reasonable cutoff γ to impose a key transition constraint (KTC) can reduce the decoding time dramatically while retaining a high performance. The same trend is also observed when applying a reasonable τ to the chord to bass transition constraint [CBTC, dotted curves in Fig. 8(b)]. Furthermore, using a chord alphabet constraint [CAC, solid curves in Fig. 8(b)] did not decrease the performance, although the decoding time is also reduced. In fact applying CAC could lead to a slight improvement, possibly due to fewer "false" chord changes in chord estimations.

Combining these techniques, we can speed up decoding without decreasing the performance, allowing us to apply HPA to more complex chord representations in the next evaluation.

B. Complex Chord Evaluation

In this evaluation, the ACE systems are evaluated on a chord estimation task using the complex chord vocabulary. This is a more realistic case in audio chord estimation, but currently only two systems, the musical probabilistic model (MP) [11] and Chordino [23], are dedicated to this large vocabulary. Compared with Section VI-A, the chord vocabulary increases dramatically in this scenario. Hence, we use *leave-one-out* cross-validation (rather than three-fold cross-validation) so as to make maximal use of the existing dataset.

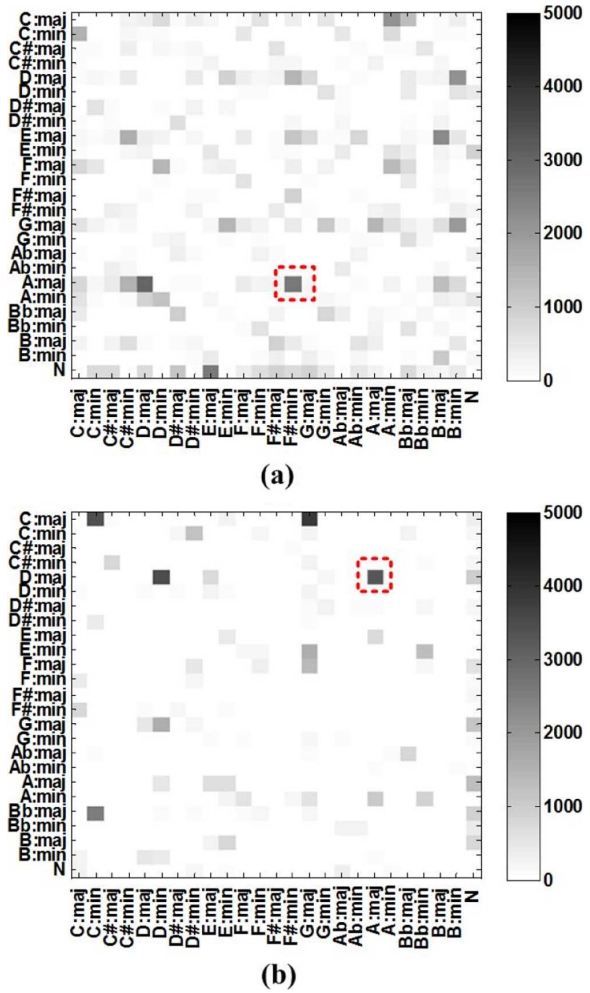

Fig. 7. Difference between the chord confusion matrices of HPA and K-HMM. (a) shows the number of times that K-HMM misclassified chord i as chord j minus that of HPA, thresholded at 0, while (b) shows the opposite. (a) K-HMM-HPA. (b) HPA-K-HMM.

Four metrics were used to evaluate the performance. *Chord precision* (CP) scores 1 if the GT and estimated chords are identical at a given frame and 0 otherwise (e.g., the score between A:maj/3 and A:maj is 0). *Note-based chord precision* (NCP), which scores 1 if all notes are identical between GT and estimated chords and 0 otherwise (e.g., the score between A:maj/3 and A:maj is 1 but that between A:maj and A:maj7 is 0). MIREX OR and WAOR, which only compare the first two intervals in the GT and the estimated chords (e.g., the score between A:maj and A:maj7 is 1, while that between A:maj and A:min is 0). Note that the additional CP and NCP metrics are stricter than OR and WAOR, hence more suitable for evaluating estimations of complex chords such as dominant 7th, augmented, and diminished chords.

1) Generalization of Chromagrams: We first investigate the generalization of all chromagrams used in Section VI-A using a standard HMM. All chromagrams were beat-synchronized in this scenario. Table II (upper table) shows the results, from which we observed a similar trend of improvements of LBC on MIREX evaluations. However, the improvements on stricter

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

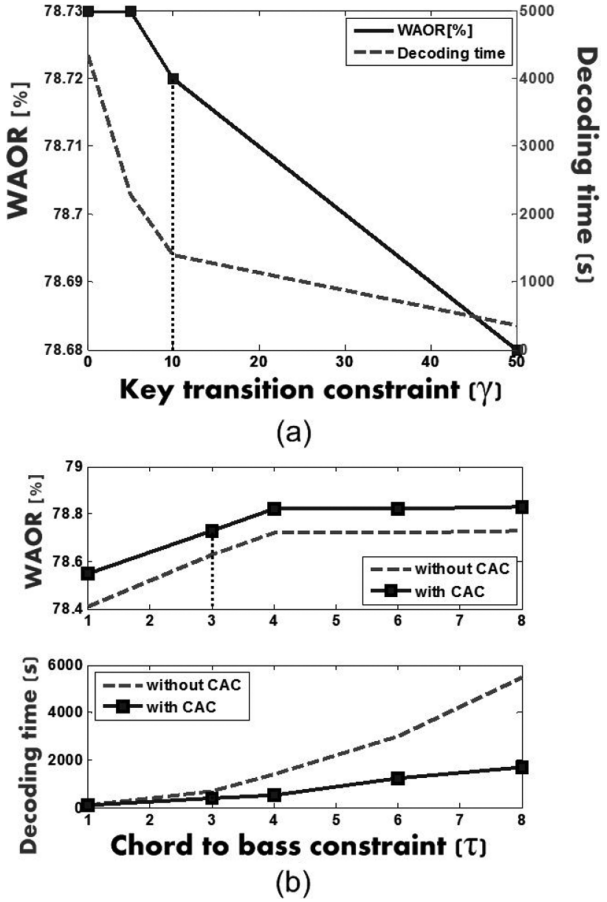

Fig. 8. Performances and decoding times of HPA using different search space reductions. The experiments in (a) were done without chord alphabet constraint and τ is fixed at 3. In (b), "CAC" refers to chord alphabet constraint and the experiments were carried out with γ fixed at 10. From (a) to (b), the decoding time is reduced dramatically (from ~ 5000 s to ~ 500 s), while the system still retains a high performance.

TABLE II
PERFORMANCE OF DIFFERENT CHROMAGRAMS (UPPER TABLE) AND DIFFERENT SYSTEMS (LOWER TABLE) ON THE COMPLEX CHORD ESTIMATION TASK. BOLD NUMBERS REFER TO THE BEST RESULTS

CHROMA	CP [%]	NCP [%]	OR [%]	WAOR [%]
LBC	60.25	**63.36**	**79.11**	**78.34**
STFTC	58.95	60.09	74.58	73.30
NNLS	45.9	47.47	69.35	68.78
TCRC	**60.57**	61.70	77.13	76.65
CRP	42.53	47.43	67.26	66.55
HAPC	42.24	45.63	69.35	68.78

SYSTEM	CP [%]	NCP [%]	OR [%]	WAOR [%]
HPA-P	**70.26**	**71.96**	**82.98**	**82.96**
HPA-L	64.46	66.12	81.28	81.05
HPA-N	55.61	58.54	78.55	77.69
MP	53.00	57.83	78.43	77.84
CH	50.31	52.35	77.53	76.86

metrics (e.g., CP) decrease. This is because all chromagrams are affected by the same problem: the fact that there is insufficient data to train a different emission probability model for each chord/bass combination when complex chords are used. It is no surprise that TCRC is least affected by this, as it is the only chromagram using some form of expert knowledge to compensate for this.

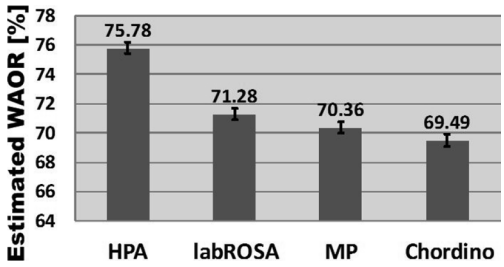

Fig. 9. Estimated average accuracies of the ACE systems on 1840 songs. An error bar represents the confidence interval of performance with a 95% confidence level.

To tackle this problem, one can obtain more fully annotated data, which is costly and not scalable. Alternatively, one can use more sophisticated models that model the dependency between musical elements (e.g., chord and bass), such as HPA and MP.

2) Comparison of Systems: Here we use three variants of HPA: the first one is a pre-trained HPA system[12] (denoted by HPA-P), which provides an upper bound of performance HPA could achieve should more data become available. The second is HPA using leave-one-out cross-validation (denoted by HPA-L), to assess its generalization ability and to compare it with that of the standard HMM. The final one is leave-one-out HPA using an NNLS chromagram (denoted by HPA-N), to compare HPA's ML-based HMM topology with the topology of MP when using the same chromagram.

We first compared HPA-L with the standard HMM (i.e., row "LBC" in the upper table) in Table II. HPA-L achieves an improvement amounting to a relative reduction in error rate by 12.2% on WAOR, which is greater than the 7.0% improvement achieved on the major/minor evaluation task. This result indicates that when the difficulty of the task increases but the training data is limited, one can benefit from a richer model that respects the structure of music.

We then compared the HPA variants with the state-of-the-art MP system in the same table. Note that the results of MP are a bit different from what have been presented in MIREX ACE 2010, due to slight differences between evaluation metrics as mentioned in Section VI-A. Encouragingly, even the leave-one-out system HPA-N achieves a slight improvement over MP, especially on stricter evaluations. We found that one cause of the low performance of MP is that it missed out many short-lived chords, possibly due to its strong chord self-transition probability. Another cause observed is that it predicted many complex chords (notably 6ths and 7ths). Both observations reveal the weakness of expert systems where parameters are tuned manually, possibly in ways that are suboptimal.

We also compared HPA's and MP's processing time and memory consumption for decoding the MIREX dataset [Table III (left table)]. Results on two specific songs, one with an average length and one being the longest song in the dataset, are also presented. On average HPA consumes only 1/10 of the memory and is 1.7 times faster than MP, verifying the effectiveness of the search space reduction techniques. By

[12]The system is equivalent to our HP_huo submission to the MIREX ACE evaluation 2011. For this system, the parameters τ and γ are fixed at 3 and 10. All other parameters are trained using the whole MIREX dataset.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

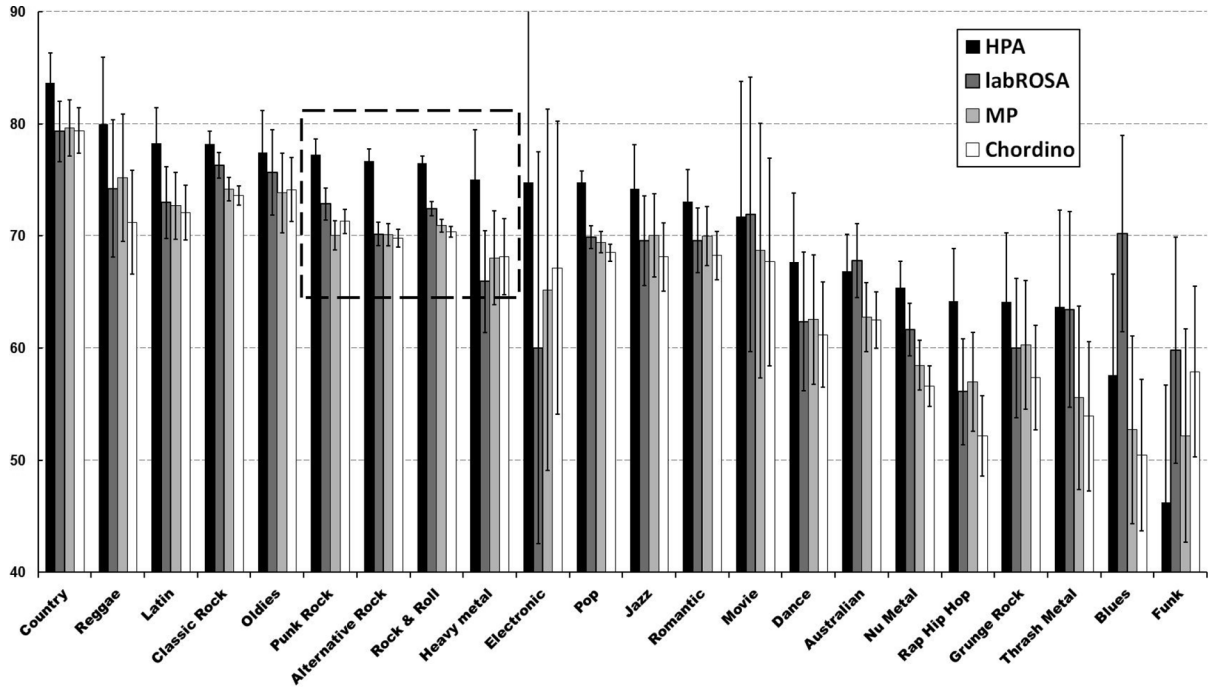

Fig. 10. Estimated average WAORs of the ACE systems on each genre. Error bars represent a confidence interval of performance with 95% confidence.

TABLE III
COMPARISON OF PROCESSING TIME AND MEMORY CONSUMPTION BETWEEN THE HPA, MP, AND CH SYSTEMS ON THE COMPLEX CHORD ESTIMATION TASK. THE TWO TRACKS USED FOR COMPARISON: "TICKET TO RIDE" (LENNON/MCCARTNEY, 190 SECONDS),"I WANT YOU (SHE'S SO HEAVY)" (LENNON/MCCARTNEY, 467 SECONDS). LEFT TABLE: COMPUTER RUNNING CENTOS 5.6 WITH 12 INTEL (R) X5650 CORES AT 2.67 GHZ, 24G RAM; RIGHT TABLE: MACHINE RUNNING OSX LEOPARD WITH AN INTEL DUO CORE 2.4G CPU, 4 GB RAM

	Proc. time (s)		Mem. (GB)	
	HPA-P	MP	HPA-P	MP
MIREX	**14796**	25114	**1.4**	15
Ticket to Ride	**66**	105	**0.4**	5.9
I Want You	**218**	313	**1.4**	15

System	Proc. time (s)	
	Feature Extraction	Decoding
CH	**9511**	
HPA-P	14756	818

comparison, running MP is not practical for regular users due to its large memory consumption.

Finally, we compared HPA to Chordino [23] (denoted by CH), which uses the same NNLS chroma features as MP but a simpler model. This comparison is informative since CH's computation/memory cost is more in line with that of HPA. To imitate the behavior of regular users, this test was done on an Apple Macintosh computer. Table III (right table) shows that HPA is a factor 1.6 slower than CH due to its more costly chromagram calculation, although the decoding process is very fast. As shown in Table II (lower table), HPA's accuracy improvement over CH is considerable though, making HPA a valuable alternative to CH.

C. Meta-Song Evaluation

In this final evaluation, we estimated and compared the performance on the 25-class chord alphabet of HPA (trained on the entire MIREX dataset) with three ACE systems: the ML-based labROSA system [10] (also trained on the whole MIREX dataset), and the expert systems MP [11] and Chordino [23]. The evaluation set consists of 1840 songs from different genres, for which the GT annotations do not exist but where we can derive the pseudo annotations from `www.e − chords.com`. To estimate performance, WAOR is used as the evaluation metric, and the linear regression model presented in [42] is applied to model the relationship between pseudo accuracies and true WAORs.

Average results on the whole dataset are presented in Fig. 9. The estimated WAOR of labROSA is slightly better than expert systems MP and CH. This implies a better generalization of ML-based systems over expert ones. HPA however outperforms all other systems.

We also categorized the songs by their genres and estimated the WAOR performances of the systems on each genre. The results are illustrated in Fig. 10. HPA performs better on most of the genres, especially on those related to Rock (dashed box in Fig. 10). This is probably to the fact that HPA is trained on songs mainly from the Rock genre. Only in a few genres such as Funk and Blues, do other methods perform better than HPA. This seems to suggest that further improvements can be gained using a meta-system, combining different systems' estimations. This will be investigated in our future work.

VII. CONCLUSION AND FUTURE WORK

In this paper, we introduced a novel key, chord and bass simultaneous estimation system—the harmony progression analyzer (HPA)—that purely relies on ML techniques. The experimental results verify that the HPA system achieves state-of-the-art performance on chord estimation, and it can be sped up significantly using the search space reduction techniques without severely decreasing the performance.

HPA uses a novel chromagram extraction method, which is inspired by loudness perception studies and achieves better estimation performance. Second, HPA purely relies on ML techniques, which promises further improvements if more data become available. Finally, HPA achieves an excellent tradeoff be-

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

tween performance and memory and time complexities, making it applicable to real world harmonic analysis tasks.

For future work, we aim to improve the processing time for chromagram extraction. This can be done by moving to faster programming languages such as C/C++. We will also move towards discriminative approaches using the same HMM topology, which may potentially address the problem of HPA discussed in Section VI-A2 where we noticed the bass chain was "overpowering" the chord chain and leading to incorrect estimations. Finally, an investigation of combining systems' estimations using ML techniques is also a direction of our future work.

ACKNOWLEDGMENT

The authors would like to thank M. Mauch for kindly making the code of his Musical Probabilistic (MP) model available to them, as well as several helpful discussions on the evaluation of chord estimation systems.

REFERENCES

[1] W. Piston, *Harmony*, 4th ed. New York: Norton, 1978.
[2] K. Lee, "A system for acoustic chord transcription and key extraction from audio using hidden Markov models trained on synthesized audio," Ph.D. dissertation, Stanford Univ., Stanford, CA, 2008.
[3] C. L. Krumhansl, *Cognitive Foundations of Musical Pitch*. Oxford, U.K.: Oxford Univ. Press, 1990.
[4] T. Fujishima, "Real time chord recognition of musical sound: A system using common lisp music," in *Proc. ICMC*, 1999, pp. 464–467.
[5] A. Sheh and D. Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," in *Proc. ISMIR*, 2003.
[6] T. Yoshioka, T. Kitahara, K. Komatani, T. Ogata, and H. Okuno, "Automatic chord transcription with concurrent recognition of chord symbols and boundaries," in *Proc. ISMIR*, 2004.
[7] J. P. Bello and J. Pickens, "A robust mid-level representation for harmonic content in music signals," in *Proc. ISMIR*, 2005.
[8] K. Sumi, K. Itoyama, K. Yoshii, K. Komatani, T. Ogata, and H. Okuno, "Automatic chord recognition based on probabilistic integration of chord transition and bass pitch estimation," in *Proc. ISMIR*, 2008.
[9] A. Weller, D. Ellis, and T. Jebara, "Structured prediction models for chord transcription of music audio," in *Proc. ICMLA*, 2009.
[10] D. Ellis and A. Weller, "The 2010 LABROSA chord recognition system," in *Proc. ISMIR (MIREX)*, 2010.
[11] M. Mauch, "Automatic chord transcription from audio using computational models of musical context," Ph.D. dissertation, Queen Mary Univ. of London, London, U.K., 2010.
[12] S. Pauws, "Musical key extraction from audio," in *Proc. ISMIR*, Barcelona, Spain, 2004, pp. 96–99.
[13] K. Noland and M. Sandler, "Key estimation using a hidden Markov model," in *Proc. ISMIR*, 2006.
[14] K. Lee and M. Slaney, "Acoustic chord transcription and key extraction from audio using key-dependent HMMs trained on synthesized audio," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 291–301, Feb. 2008.
[15] K. Lee and M. Slaney, "A unified system for chord transcription and key extraction using hidden Markov models," in *Proc. ISMIR*, 2007.
[16] M. P. Ryynäen and A. P. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Comput. Music J.* , no. 3, pp. 72–86, 2008.
[17] H. Papadopoulous and G. Peeters, "Local key estimation based on harmonic and metric structures," in *Proc. DAFX*, 2009.
[18] M. Khadkevich and M. Omologo, "Use of hidden Markov models and factored language models for automatic chord recognition," in *Proc. ISMIR*, 2009.
[19] T. Rocher, M. Robine, P. Hanna, L. Oudre, Y. Grenier, and C. Févotte, "Concurrent estimation of chords and keys from audio," in *Proc. ISMIR*, 2010.
[20] T. Cho and J. P. Bello, "A feature smoothing method for chord recognition using recurrence plots," in *Proc. ISMIR*, 2011.
[21] O. Lartillot and P. Toiviainen, "A Matlab toolbox for musical feature extraction from audio," in *Proc. Int. Conf. Digital Audio Effects*, 2007.
[22] J. Brown, "Calculation of a constant q spectral transform," *J. Acoust. Soc. Amer.*, vol. 89, no. 1, pp. 425–434, 1991.
[23] M. Mauch and S. Dixon, "Approximate note transcription for the improved identification of difficult chords," in *Proc. ISMIR*, 2010.
[24] N. Ono, K. Miyamoto, J. Roux, H. Kameeoka, and S. Sagayama, "Separation of a monaural audio signal into harmonic/percussive components by complimentary diffusion on spectrogram," in *Proc. EUSIPCO*, 2008.
[25] T. Cho, R. J. Weiss, and J. P. Bello, "Exploring common variations in state of the art chord recognition systems," in *Proc. Sound Music Comput. Conf.*, 2010, pp. 1–8.
[26] D. Ellis and G. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," in *Proc. ICASSP*, 2007, pp. 1429–1433.
[27] C. Harte, M. Sandler, and M. Gasser, "Detecting harmonic change in musical audio," in *Proc. Audio Music Comput. for Multimedia Workshop*, Santa Barbara, CA, 2006.
[28] M. Müller and S. Ewert, "Towards timbre-invariant audio features for harmony-based music," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 3, pp. 649–662, Mar. 2010.
[29] M. T. Smith, *Audio Engineer's Reference Book*. Waltham, MA: Focal Press, 1999.
[30] L. R. Rabiner, "A tutorial on hidden Markov models and selected application in speech recognition," *Proc. IEEE*, vol. 77, no. 8, pp. 257–286, Feb. 1989.
[31] B. Su and S. Jeng, "Multi-timbre chord classification using wavelet transform and self-organized map neural networks," in *Proc. ICASSP*, 2001, pp. 3377–3380.
[32] G. Cabral, F. Pachet, J. Briot, and S. Paris, "Automatic x traditional descriptor extraction: The case of chord recognition," in *Proc. ISMIR*, 2005.
[33] M. R. Gupta and Y. Chen, "Theory and use of the EM algorithm," *Found. Trends Signal Process.*, vol. 4, pp. 223–296, Mar. 2011.
[34] Y. Altun, I. Tsochantaridis, and T. Hofmann, "Hidden Markov support vector machines," *Proc. ICML*, 2003.
[35] C. Harte and M. Sandler, "Automatic chord identification using a quantised chromagram," in *Proc. AES Conf.*, 2005.
[36] M. McVicar, Y. Ni, R. Santos-Rodriguez, and T. De Bie, "Using online chord databases to enhance chord recognition," *J. New Music Res.*, vol. 40, no. 2, pp. 139–152, 2011.
[37] M. McVicar, Y. Ni, R. Santos-Rodriguez, and T. De Bie, "Leveraging noisy online databases for use in chord recognition," in *Proc. ISMIR*, 2011.
[38] H. Fletcher, "Loudness, its definition, measurement and calculation," *J. Acoust. Soc. Amer.*, vol. 5, no. 2, pp. 82–82, 1933.
[39] T. D. Rossing, *The Science of Sound (Second Edition)*. Boston, MA: Addison-Wesley, 1990.
[40] N. Jiang, P. Grosche, V. Konz, and M. Müller, "Analyzing chroma feature types for automated chord recognition," in *Proc. 42nd AES Conf.*, 2011.
[41] R. Bisiani, "Beam search," *Encycl. Artif. Intell.*, pp. 56–58, 1987.
[42] Y. Ni, M. Mcvicar, R. Santos-Rodriguez, and T. De Bie, "Meta-song evaluation for chord recognition,' 2011, arXiv:1109.0420v1, Tech. Rep.
[43] Y. Ni, M. McVicar, R. Santos-Rodriguez, and T. De Bie, "Meta-song evaluation for chord recognition," in *Proc. ISMIR (Late Breaking Demo)*, 2011.
[44] M. Müller and S. Ewert, "Chroma toolbox: Pitch, chroma, CENS, CRP," in *Proc. ISMIR*, 2011.
[45] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proc. ICML*, 2004.

Yizhao Ni received the M.Sc. degree from the University College London, London, U.K., and the Ph.D. degree in machine learning for machine translation from the University of Southampton, Southampton, U.K.

He is a Post-Doc Research Assistant at the Intelligent Systems Lab, University of Bristol, Bristol, U.K. His current research interests lie in the development and application of machine learning methods to music information retrieval, machine translation, and bioinformatics.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.

Matt McVicar is currently pursuing the Ph.D. degree at the Intelligent Systems Lab, University of Bristol, Bristol, U.K., supervised by T. De Bie.

His current research interests are the automatic transcription of chords from musical audio and their applications to higher level tasks such as mood prediction.

Raúl Santos-Rodríguez received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Universidad Carlos III de Madrid, Getafe, Spain, in 2007 and 2011, respectively.

He is a Research Assistant at the Signal Theory and Communications Department, Universidad Carlos III de Madrid, and a Visiting Fellow at the Department of Engineering Mathematics, University of Bristol, Bristol, U.K. His current research interest lies in machine learning and its application to signal processing.

Tijl De Bie received the Ph.D. degree in machine learning and advanced optimization techniques from the University of Leuven, Leuven, Belgium, in 2005.

He is a Senior Lecturer in artificial intelligence at the University of Bristol, Bristol, U.K., where he was first appointed as a Lecturer in January 2007. Before that, he was a Research Assistant at the University of Leuven and the University of Southampton, and during the Ph.D. degree he spent research visits at UC Berkeley and UC Davis. His current research interests include statistical pattern analysis algorithms, the use of optimization theory to design such algorithms, and their application to music information retrieval, bioinformatics, and web and text mining.

Authorized licensed use limited to: Robert Gordon University. Downloaded on October 06,2022 at 10:13:47 UTC from IEEE Xplore. Restrictions apply.