BASS BOOST: CHORD ESTIMATION WITH INVERSIONS IN BEATLES SONGS

Elena Georgieva[1]
etg259@nyu.edu

Michal Goldstein[1]
mng320@nyu.edu

Juan Sierra[1]
jds833@nyu.edu

[1]Music and Audio Research Laboratory, New York University

ABSTRACT

Previous approaches for automatic chord recognition focus on major and minor thirds, overlooking other significant features of the chords, such as chord inversions. A chord is inverted when a note other than the root appears in the bass, and inverted chords can alter a chord's perceptual function in a song. In this paper, we added bass information to major/minor chord estimates generated by an existing machine-learning-based method. This was accomplished by estimating the bass f_0 track using the CREPE pitch tracker [1], and adding the bass line to the chord information. In order to evaluate our method, we sampled 10 songs from the Beatles dataset [2] and compared the estimated chord labels, combined with the bass information, to the reference annotation. [1]

1. INTRODUCTION

Estimating chord progressions using automatic chord estimation (ACE) is a fundamental task in the field of Music Information Retrieval (MIR). One of the most common approaches to fulfill this task is to compute the chroma features of the musical excerpt, which are represented as the energy of each pitch class on the equal-tempered scale over time. The chroma matrix is subsequently used to predict the chord label of each time frame through a pattern-matching process.

While this process can be accomplished using hand-crafted templates, better results can be achieved using data-driven approaches such as deep-neural networks [3]. Existing models such as this one include only major and minor triad information, yet other information, including extended chords such as seventh chords, diminished and augmented chords, suspensions, and inversions, are omitted.

In functional harmony, humans perceive chords based on the chord's function in a song [4]. Existing work on automatic chord transcription classifies chords based on pitch class as opposed to by function. In this project ,we propose including inversion information in chord analysis as a step towards classifying chords based on function.

Inversions are a crucial chord feature, as the note in the bass strongly influences the sound and function of a chord. For example, if a tonic chord appears in second inversion, the fifth scale degree appears in the bass and the chord functions as a dominant rather than a tonic chord.

2. RELATED WORK

The goal of this project is to estimate chords with inversions from audio files. While there is not a lot of work in past literature on extracting inversions specifically, previous researchers have attempted to include more chords features in chord annotations, for example by extracting 7th chords from audio files. For instance, Deng and Kwok used Gaussian Mixture Modelling and Hidden Markov Model networks to automatically estimate chords, including 7th chords [5]. Similarly, Nadar, Abeßer, and Grollmisch used deep convolutional neural networks for automatic chord recognition, extending the commonly used major/minor vocabulary include seventh chords [6].

While not a lot of work have been conducted in estimating inverted chords from audio files, several studies have focused on representing inverted chord from annotated labels. In particular, Chen and Su used a recurrent neural network to investigate various interrelated chord functions such as key modulation, chord inversion, and secondary chords [7]. In addition, Two Python packages, music21 [8] and mingus [9] offer functionality including returning the quality of a chord, the musical interval between two notes, and converting from chords to roman numerals while taking into account the inversion of the chord.

One group of researchers did approached a problem very similar to the one we focus on in the paper. Yizhao and their team used machine learning to perform harmonic estimation, including estimation of key sequence and bass notes from audio files [10]. Their chromagram, which is used to estimate chords, takes into account human perception of loudness and are calculated separately for the treble range and the bass range. Subsequently, they combine the estimated chord label, based on the treble chromagram, and the bass note, based on the bass chromagram, to a single chord label. Their model provides fast, memory ef-

[1] All code used in this paper is publicly available at https://github.com/jdsierral/BassBoost

ficient, and accurate harmonic estimates. However, since the publication of this paper more efficient, data-driven, chord-extraction algorithms have been developed.

3. METHODS

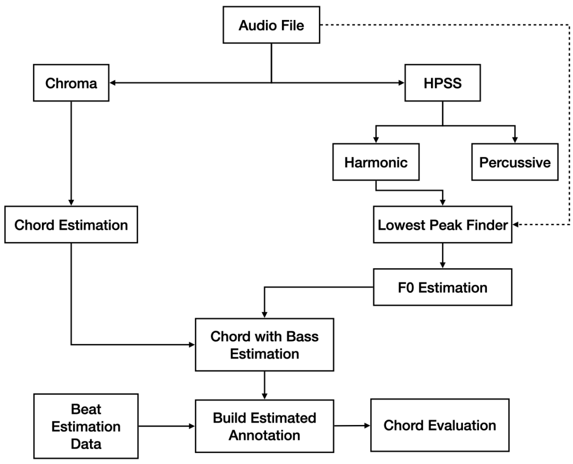

Figure 1. Flow Chart of the general estimation process

To estimate chords with inversions, we followed the steps illustrated in *Figure. 1*

First, we chose a set of 10 Beatles songs, spanning most of their albums. From each audio file, we obtained a set of chroma features using Madmom's Deep Chroma Processor. Chroma represents how much energy of each pitch class {C, C, D, ...} is present in each frame of the signal. Next, this chroma information is represented as a chromagram and passed to the Deep Chroma Chord Recognition Processor, which calculates the most likely chord at each time frame.

Next, we used Harmonic Percussive Source Separation (HPSS) to separate the harmonic and percussive content of the audio signal. HPSS was implemented through simple vertical and horizontal median filtering on the Short Time Fourier Transform (STFT) of our source signal. We chose to take only the harmonic content generated by HPSS. Although the percussive elements can have significant low frequency energy, they do not represent relevant harmonic information [11]. This harmonic signal was used to estimate the lowest frequency F0, which is the line usually played by the bass. The harmonic signal was also low-pass filtered with a 4^{th} order Butterworth filter at 250 Hz to constrain the f_0 estimation to lower frequencies.

From the f_0 value in Hertz, we obtain the musical pitch class of the bass. At first, we attempted multiple methods for the f_0 estimation including estimation through the STFT directly and through the YIN algorithm [12]. Ultimately, we chose the CREPE method [1], as it provided the most robust and precise pitch information. After we obtained the pitch tracks, the f_0 values were quantized and mapped to exact pitch classes. For this task, we are not interested in the the precise f_0 value in Hertz, nor are we interested in the octave.

Next, we used a beat tracking algorithm [13] from the audio signal processing library Madmom to estimate the beats in our source material. We treat the beats as frames, and obtained multiple F0 values for each frame. We selected the statistical mode of each beat frame to be the final f_0 estimate for that beat frame. Ultimately, we ended up with one bass note value per estimated beat.

The next step was to combine the estimated bass f_0 values with the chord estimations for each song. This produced an estimated chord and bass note value for each beat frame. For example, at one beat there could be a C Major chord with a G bass note value.

This approach produces a very musically oriented estimation of said inversions since it takes into account both the general chord estimation through the chroma information, and the harmonic rhythm through beat estimation. Ultimately, we ended up with many sections of repeated chords. Whenever there is a harmonically relevant inversion in any of the beats, our method reveals that, even though the initial chord estimation could have identified the chord as unchanged for said beat.

4. EVALUATION

4.1 Bass Line and Chord Evaluation

The Beatles dataset includes a set of chord annotations including beat, chord, and key information. We extracted bass line "ground truth" from the chord annotations. We assumed the name of each chord to be the root except when there was inversion information in the chord annotations. In that case, we calculated the pitch class of the bass of the chord from the provided information. We only used this information to evaluate the bass F0 information we estimated from the audio. We plot an example comparison of the estimated and reference F0 information for the song "Penny Lane" 2.

We evaluated the chords again using the Beatles dataset chord evaluations.

For the evaluation, we calculated three scores. The first score represents the number of estimations that match those provided in the dataset annotations, compared at the level of major and minor thirds. We calculated this using `mir_eval.thirds_inv`. The second score is the number of correct triads that come directly from madmom's deep nearual network chord estimations. We used `mir_eval.thirds`. This compares major/minor thirds and ignores extended notes. The third score is the fraction of correct madmom estimations that were correctly expanded with our estimated inversion information. This third score was created to avoid conflating madmom's success metric with that of our inversion estimates.

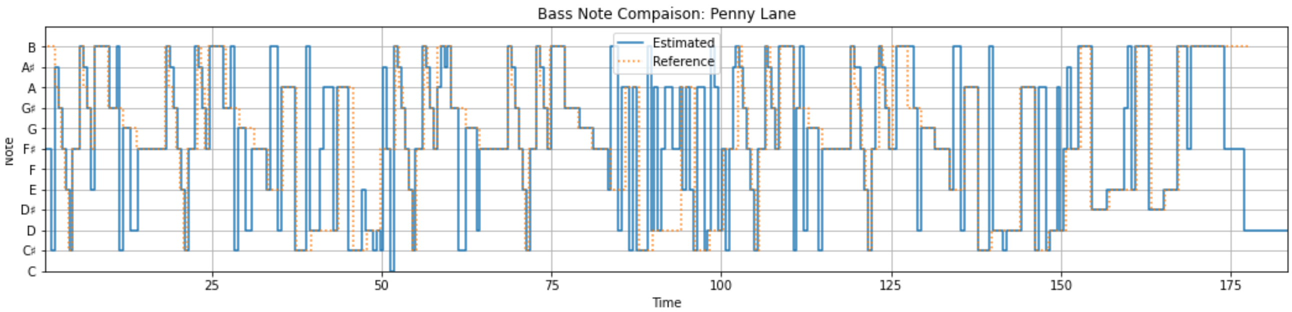

Figure 2. Estimated and Reference F0 information for the Song "Penny Lane." We calculated the Estimated F0 information through HPSS, filtering, and the Crepe pitch tracker. The Reference F0 information comes from the dataset.

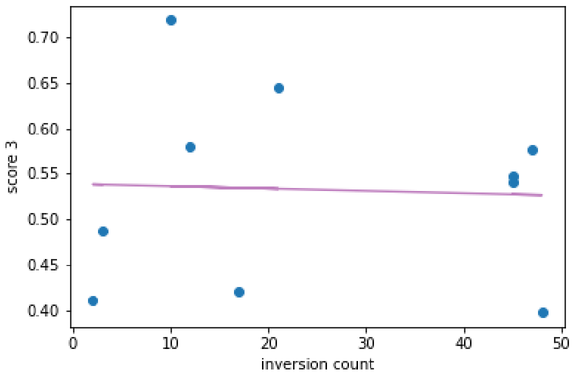

Figure 3. The linear relationship between the number of inverted chords in each song, and the chord evaluation score for that song

4.2 HPSS Evaluation

The HPSS process was intended to improve the effectiveness of the chord estimation. We calculated the chord evaluation scores and for the 10 songs with and without the HPSS processing. We found the chord evaluations perform better without HPSS than with HPSS processing, thoug the difference was not significant ($p = .8$)????.

5. FUTURE WORK

In the future, we are interested in exploring functional harmony recognition. We would like to convert the chord information into roman numerals with inversions (i.e. I^6), representing the chord's function in the song.

We are interested in applying this work to Bach Chorales. Bach Chorales are a well-studied body of work, but automatic harmonic analyses of the chorales do not include inversion information.

Beyond this, we believe this chord estimation with inversions tool can be used for the automatic creation of sheet music and karaoke tracks.

Song	Score 1	Score 2	Score 3
Please Please Me	0.33	0.69	0.49
All My Loving	0.33	0.80	0.41
Yesterday	0.40	0.62	0.65
Michelle	0.26	0.62	0.42
Here, There And Everywhere	0.38	0.53	0.72
For No One	0.40	0.69	0.58
Getting Better	0.33	0.58	0.58
The Fool On The Hill	0.25	0.62	0.40
Penny Lane	0.41	0.74	0.54
Mother Nature's Son	0.40	0.73	0.55

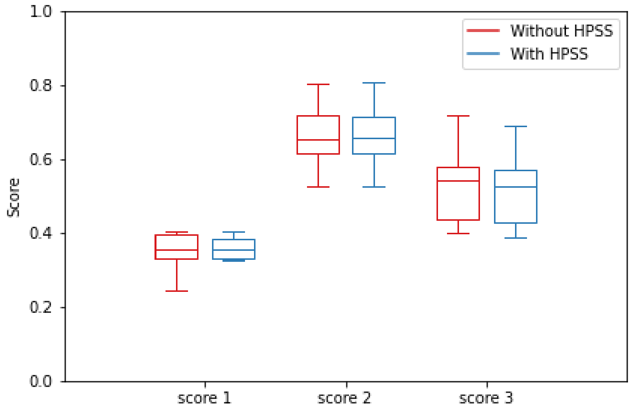

Figure 4. Chord evaluation scores for the 10 selected songs. Scores 1 and 2 represents the number of estimates that match the datset annotations including inversions an disregarding inversion, respectively. Score 3 is the fraction of correct madmom estimations that were correctly expanded with our estimated inversion information. The scores for each song were determined with and without HPSS. The table includes the without-HPSS values only

6. ACKNOWLEDGMENTS

Thank you to Dirk Vander Wilt for his guidance on this project.

7. REFERENCES

[1] J. W. Kim, J. Salamon, P. Li, and J. P. Bello, "Crepe: A convolutional representation for pitch estimation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 161–165.

[2] C. Harte, "Towards automatic extraction of harmony information from music signals," Ph.D. dissertation, 2010.

[3] F. Korzeniowski and G. Widmer, "A fully convolutional deep auditory model for musical chord recognition," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2016, pp. 1–6.

[4] A. Patel, "Language, music, syntax and the brain," in *Nature Neuroscience*, vol. 6, no. 7, 2003.

[5] J. Deng and Y.-K. Kwok, "Automatic chord estimation on seventhsbass chord vocabulary using deep neural network," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 261–265.

[6] C.-R. Nadar, J. Abeßer, and S. Grollmisch, "Towards cnn-based acoustic modeling of seventh chords for automatic chord recognition," in *International Conference on Sound and Music Computing. Málaga, Spain*, 2019.

[7] T.-P. Chen, L. Su *et al.*, "Functional harmony recognition of symbolic music data with multi-task recurrent neural networks." in *ISMIR*, 2018, pp. 90–97.

[8] M. Cuthbert and C. Ariza, "Music21: A toolkit for computer-aided musicology and symbolic music data." in *International Society for Music Information Retrieval (ISMIR)*, 2010, pp. page 637–642.

[9] B. Spaans, "mingus," https://github.com/bspaans/python-mingus, 2015.

[10] Y. Ni, M. McVicar, R. Santos-Rodriguez, and T. De Bie, "An end-to-end machine learning system for harmonic analysis of music," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1771–1783, 2012.

[11] D. Fitzgerald, "Harmonic/percussive separation using median filtering," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, vol. 13, 2010.

[12] A. De Cheveigné and H. Kawahara, "Yin, a fundamental frequency estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002.

[13] S. Böck and M. Schedl, "Enhanced beat tracking with context-aware neural networks," in *Proc. Int. Conf. Digital Audio Effects*, 2011, pp. 135–139.