Effective Music Feature NCP: Enhancing Cover Song Recognition with Music Transcription

Conference Paper · August 2017

DOI: 10.1145/3077136.3080680

4 authors:

Yao Cheng
Anhui University
114 PUBLICATIONS **3,548** CITATIONS
SEE PROFILE

Chen Xiaoou
Peking University
43 PUBLICATIONS **345** CITATIONS
SEE PROFILE

Deshun Yang
Peking University
32 PUBLICATIONS **227** CITATIONS
SEE PROFILE

Xiaoshuo Xu
Peking University
7 PUBLICATIONS **30** CITATIONS
SEE PROFILE

Effective Music Feature NCP: Enhancing Cover Song Recognition with Music Transcription

Yao Cheng
Institute of Computer Science and Technology
Peking University, Beijing, China
chengyao@pku.edu.cn

Xiaoou Chen
Institute of Computer Science and Technology
Peking University, Beijing, China
chenxiaoou@pku.edu.cn

Deshun Yang
Institute of Computer Science and Technology
Peking University, Beijing, China
yangdeshun@pku.edu.cn

Xiaoshuo Xu
Institute of Computer Science and Technology
Peking University, Beijing, China
xuxiaoshuo@pku.edu.cn

ABSTRACT

Chroma is a widespread feature for cover song recognition, as it is robust against non-tonal components and independent of timbre and specific instruments. However, Chroma is derived from spectrogram, thus it provides a coarse approximation representation of musical score. In this paper, we proposed a similar but more effective feature Note Class Profile (NCP) derived with music transcription techniques. NCP is a multi-dimensional time serie, each column of which denotes the energy distribution of 12 note classes. Experimental results on benchmark datasets demonstrated its superior performance over existing music features. In addition, NCP feature can be enhanced further with the development of music transcription techniques. The source code can be found in github[1].

CCS CONCEPTS

•**Information systems** →*Similarity measures; Sentiment analysis;*

KEYWORDS

Cover Song Recognition;Dynamic Programming

1 INTRODUCTION

Cover song recognition, which is also called music version identification, was extensively studied in recent years. Partly because its potential commercial values such as music copyright protection and management. Another reason is that finding the transformation of music piece that retains its essential identity helps us develop intelligent audio algorithms that recognize common patterns among musical excerpts.

Music versions are usually performed with their own characteristics in reality, to adapt to different singer or live atmosphere or specific instruments, sometimes just for musical aesthetics. The variations among music versions are the big obstacles for cover song recognition research. What we need to do is to find an effective feature or sequence matching method that satisfy key invariance, tempo invariance and structure invariance.

Up to now, Chroma has been widely used in cover song recognition, as it is robust against non-tonal components and independent of timbre and specific instruments. Afterwards, various variants of Chroma have been come up with in succession. In the past years, Ellis[4] enhanced Chroma to make it synchronized with beats detected by a dynamic programming method. The enhanced chroma, also called Beat-Chroma, is insensitive to tempo variance. Harmonic Pitch Class (HPCP) [5] was firstly proposed by Emilia, and exhibited a better performance in Serra's work[13]. Chroma Energy distribution Normalized Statistics (CENS) [11] was another widely used variant of Chroma. An overview of these features can be found in [7].

Moreover, part of researchers held that Chord Sequence can be regarded as a good representation in cover song recognition. Attempts can be found in [1, 8, 10]. Lee[10] extracted Chord Sequence from Chroma using Hidden Markov Model, and measured similarity between songs by Dynamic Time Warping[13]. Bello[1] extended Lee's method[10] with a BLAST string alignment method widely used in bioinformatics. Furthermore, Maksim[8] derived Chord Profile by folding Chord Sequence into a 24-dimensional feature, so that Chord Profile can be directly applied in large scale cover song recognition owing to its fixed low dimension. However, none of these proposed features exhibited a promising future.

When the size of database is not very large, for instance hundreds of songs, performing Q^*_{max}[13] seems like a state of the art method. However, when the number of songs reaches up to thousands or even tens of thousands, the sequence matching methods didn't work at all because of its high computational cost. To adapt to large scale cover song recognition, most existing methods took measures to speed up the computational efficiency at the expense of accuracy. Some approaches attempted to design a fixed low dimensional feature, obtained by dimension-reduction algorithms or just derived from Chroma. After the low dimensional features extracted, a lot of information retrieval algorithms can be exploited to solve the original problem. Typical recent research works included cognition-inspired descriptors[15], 2DFM[2, 6], Chrod Profile[8], MPLPLC[3], combining features[12], music shapelets[14] and so on. Unfortunately, even the state of the art methods for large scale

[1]https://github.com/gmcather/NCP-exp

SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan.
 ISBN 978-1-4503-5022-8/17/08…$15.00.
DOI: http://dx.doi.org/10.1145/3077136.3080680

cover song recognition are still far away from business application. Considering the facts mentioned above, we mainly concentrated on small scale cover song recognition.

MIDI, short for Musical Instrument Digital Interface, is a technical standard that allows a wide variety of electronic musical instruments to connect and communicate with one another. It provided a symbolical representation form for music. Assuming that we had the MIDI representations of each song, the difficulty of cover song recognition research would be greatly decreased, as symbolic features such as melody or chord can be more easily extracted from MIDI. However, MIDI files of songs are not easy to obtain in reality, as composers are not willing to publish them for copyright protection.

Inspired by the above facts, we attempted to substitute Converted-MIDI (converted from audio by music transcription techniques) for Edited-MIDI (the original MIDI written by composer), and explored whether experimental results can be improved further over previous methods. In this work, we derived NCP feature from Converted-MIDI, and demonstrated its superior performance over Beat-Chroma and CENS. Besides we also explored why NCP showed a better result.

2 NCP FEATURE EXTRACTION

In the following subsections, we elaborated the procedures of NCP feature extraction.

2.1 Wav2MIDI

It's easier to extract extract more precise pitch information from original MIDI files than audio files, however, original MIDI files are usually hard to being accessible for the copyright reason. Alternatively, we collected MIDI files converted from audio pieces with music transcription tools. Henrique [9] summarized the widespread music transcription benchmark tools, to name a few, WIDI[2], Akoff Composer[3], Waon[4] and Sound2MIDI[5]. In our work, WIDI was used as the main MIDI conversion tool, as it is one of great commercial music transcription tools, and Waon was used for contrast experiment.

2.2 MIDI2Events

MIDI, short for Musical Instrument Digital Interface, is a technical standard that describes how to perform a song, and widely used for electronic music. To parse a MIDI file, we resorted to an open source tool called Pretty-MIDI[6]. Before devling into the parsing process, let's clarify some basic relevant concepts.

- **Track**: A MIDI file consists of multiple tracks, each track stores the performance by a specific instrument.
- **Event**: Each track contains a series of events, each event consists of four attributes: start time, end time, pitch and velocity, and usually represents a short action on a specific instrument.
- **Pitch**: Each pitch ranges from 0 to 127.
- **Velocity**: Velocity represents how strong the note or key is pressed.
- **Start Time and End Time**: Start Time and End Time describes when the event started and ended.

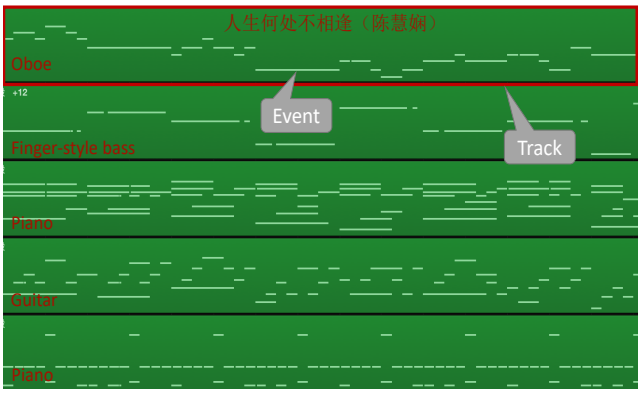

Figure 1: The Edited-MIDI shown by GarageBand

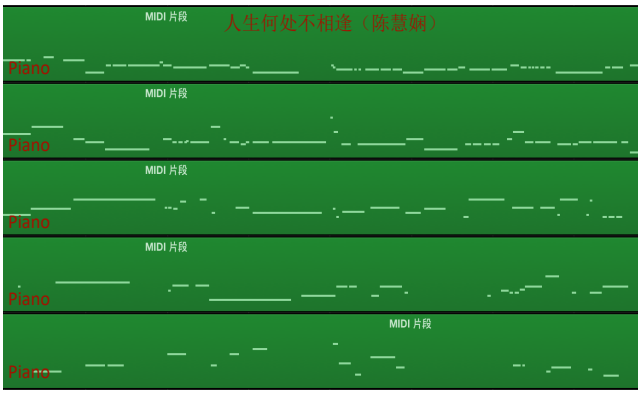

Figure 2: The Converted-MIDI shown by GarageBand

With Pretty-MIDI, we can easily separate different tracks from MIDI, extract events from tracks. We made a distinction between Edited-MIDI and Converted-MIDI. Edited-MIDI, also refers to original MIDI, is composed by musicians. However, Converted-MIDI is converted from audio file with music transcription tool, therefore some useful information might lose during conversion procedure. In reality, Even though Converted-MIDI sounds not as euphonious as Edited-MIDI, its main melody can be clearly perceived. The samples of Converted-MIDI and Edited-MIDI on github[7] were presented for listening. Both of them refer to the same Chinese Pop Music. Two synchronized music pieces were cut down from samples mentioned above, and presented in Figure 1 and Figure 2 respectively. By observing, Edited-MIDI distinguished the instruments in each track well, yet Converted-MIDI was not capable of recognizing instruments in tracks, and marked them by Piano. Moreover, different music transcription tools varied in the quality of conversion as well.

[2]http://www.widisoft.com/english/products.html
[3]http://akoff.com/download.html
[4]https://sourceforge.net/projects/waon/
[5]http://sound2midi.software.informer.com/
[6]https://github.com/craffel/pretty-midi
[7]https://github.com/gmcather/NCP-exp

2.3 Events2NCP

Inspired by the extraction process of Chroma, we devised the following computation procedures to extract NCP feature, shown in Algorithm 1. Note that all the events should be taken as the input of Algorithm, regardless of which of track they come from.

Algorithm 1 Algorithm for computing NCP feature

Input:
An array of start time of Events, $EventStart$
An array of end time of Events, $EventEnd$
An array of pitch of Events, $EventPitch$
An array of velocity of Events, $EventVelocity$
The number of Events, N
Output:
An matrix represents NCP feature, F
The number of rows of NCP feature, T

// compute earliest start time and latest end time
$EarliestTime$, $LatestTime$ = 0, $+\infty$
for each $i \in [0, N)$ **do**
  update $EarliestTime$ with $EventStart[i]$
  update $LatestTime$ with $EventEnd[i]$
end for

// initialize matrix F, T
$T = \lfloor (LatestTime - EarliestTime)/0.1 \rfloor$
$F_{T*12} = \mathbf{0}_{T*12}$

// accumulate events into F
for each $i \in [0, N)$ **do**
  $p = EventPitch[i]\%12$
  $v = EventVelocity[i]$
  for each $j \in [0, T)$ **do**
    $t = EarliestTime + j * 0.1$
    if $t \geq EventStart[i]$ **and** $t < EventEnd[i]$ **then**
      $F_{j,p}$ += v
    end if
  end for
end for

// normalize F by row
for each $i \in [0, T)$ **do**
  find the maximum value in i-th row, r
  if $r \neq 0$ **then**
    for each $j \in [0, 12)$ **do**
      $F_{i,j}$ /= r
    end for
  end if
end for

Algorithm 1 described how to derive NCP feature F from Events. To begin with, the earliest start time and the latest end time are precomputed for later use. Next, each event was split into pieces by 0.1s time unit. Then, each split event was accumulated to NCP feature F. At last, each column of F was normalized to accommodate to the variance of loudness. Note that we took 0.1s as the least time unit for the reason that a musical note usually lasts for 0.1 second when performing music.

3 EXPERIMENT AND ANALYSIS

3.1 Datasets and Metric

Two datasets were used in our experiments as following:

- ***Covers80***[8]: A public widespread cover song dataset built by LabROSA[9], a collection of 80 songs, each performed by 2 artists.
- ***Covers38***[10]: A private dataset built by ourselves, a collection of 38 Chinese pop songs, each performed by 3∼4 artists, 132 songs in all. Each of 38 songs has its corresponding Edited-MIDI for the following verification experiment.

Metric: We adopted Mean Average Precision (MAP)[11] to measure the relevance of retrieved songs.

3.2 The performance of NCP

Two types of NCP features, which are NCP-WIDI from WIDI tool and NCP-Waon from Waon tool respectively, were presented in experiments, in comparison to widespread features Beat-Chroma[4] and CENS[11]. Q^*_{max}[13] was conducted to measure the similarity between songs. In experiments, we carefully adjusted significant parameters, for example the length of embedding window m in Q^*_{max}, to make sure each features achieved their best performances. The experimental results on *Covers80* and *Covers38* datasets were presented in Table 1.

Table 1: Comparison among Beat-Chroma, CENS and NCP

MAP	Beat-Chroma	CENS	NCP-Waon	NCP-WIDI
Covers80	0.554	0.596	0.613	**0.645**
Covers38	0.656	0.781	0.799	**0.828**

Both NCP-WIDI and NCP-Waon outperformed Beat-Chroma and CENS on *Covers80* and *Covers38*, this evidence indicated NCP feature is more suitable for cover song recognition at least for Q^*_{max} algorithm. Moreover, The fact that NCP-Waon performed not as well as NCP-WIDI, not only presented the differences of music transcription tools in cover song recognition, but also reminded us that the performance can be improved further with the development of music transcription techniques.

3.3 Explore the evidence

Chroma is computed by a DFT or constant-Q transform method, where different frequency bands are divided into their corresponding Chroma bins. While NCP was directly derived from MIDI, taking music transcription techniques into account, so it provided a more accurate music symbolical representation over Chroma. Therefore we infered that it is the differences of features mentioned above that caused their preformance differences, and verified our guess in the following experiments.

For simplicity, we clarified some concepts in advance.

- $\mathbf{F_{edited}}$**:** NCP feature derived from an Edited-MIDI.

[8] http://labrosa.ee.columbia.edu/projects/coversongs/covers80/
[9] http://labrosa.ee.columbia.edu
[10] https://github.com/gmcather/NCP-exp/cover38
[11] http://en.wikipedia.org/wiki/Information_retrieval#Mean_average_precision

- **$\mathbf{F_{widi}}$:** NCP feature derived from a song of *mp3* or *wav* form by WIDI.
- **$\mathbf{F_{chroma}}$:** Chroma feature derived from a song of *mp3* or *wav* form.

The more the similarity between music feature and F_{edited}, the closer the distance between music feature and authentic music symbolical feature representation. To verify F_{widi} is closer to the authentic music symbolical representation than F_{chroma}, we collected N(=132) songs from *Covers38*, each contains its corresponding Edited-MIDI and audio file. Here we adopted Q^*_{max} to measure the similarity between different features. The similarity Sim_w between F_{edited} and F_{widi}, or the similarity Sim_c between F_{edited} and F_{chroma} can be computed with the following formulations.

$$Sim_w = Q^*_{max}(F_{edited}, F_{widi}) \tag{1}$$

$$Sim_c = Q^*_{max}(F_{edited}, F_{chroma}) \tag{2}$$

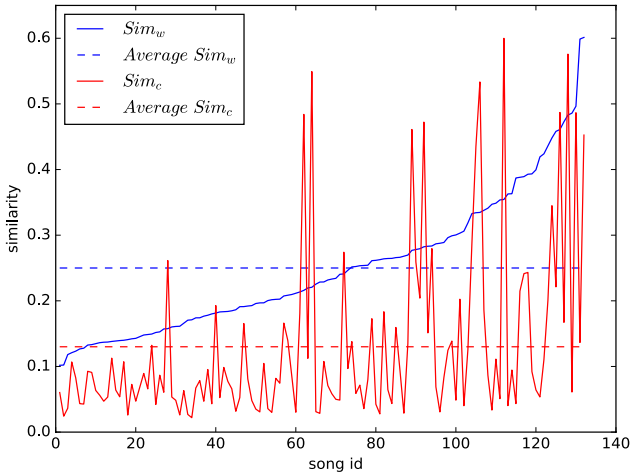

Figure 3: Sim_w and Sim_c for 132 songs

We computed both Sim_w and Sim_c for all 132 songs, sorted all pairs of (Sim_w, Sim_c) in an ascending order by Sim_w, and showed the outcome in Figure 3. Sim_w is larger than Sim_c among almost all songs, and only a few songs exhibited an opposite result. In addition, Sim_w is approximately twice as large as Sim_c on average. From all these evidences, we found NCP is more suitable for cover song recognition over Chroma.

4 CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel music feature NCP by exploiting existing music transcription techniques, and demonstrated its superior performance over current popular features on small scale datasets. Moreover, we also explored and verified the important factors that contribute to the great performance. We conjectured the performance of NCP can be improved further with the development of music transcription techniques.

However, NCP cannot be directly applied to large scale cover song recognition due to its varied and long sequence. In the future work, to acclimate NCP to large scale cover song recognition, we will attempt to derive a fixed and low dimensional feature that still preserved distinctive musical information from NCP. For instance, 2DFM might be a great attempt to perform the transformation.

ACKNOWLEDGMENTS

This work was supported by the Natural Science Foundation of China under Grant No.61370116.

REFERENCES

[1] Juan Pablo Bello. 2007. Audio-Based Cover Song Retrieval Using Approximate Chord Sequences: Testing Shifts, Gaps, Swaps and Beats. In *International Society for Music Information Retrieval Conference.*

[2] Thierry Bertin-Mahieux and Daniel PW Ellis. 2012. Large-Scale Cover Song Recognition Using the 2D Fourier Transform Magnitude. In *International Society for Music Information Retrieval Conference.*

[3] Ning Chen, J Stephen Downie, Haidong Xiao, Yu Zhu, and Jie Zhu. 2015. Modified Perceptual Linear Prediction Liftered Cepstrum (MPLPLC) Model for Pop Cover Song Recognition. In *International Society for Music Information Retrieval Conference.*

[4] Daniel PW Ellis and Graham E Poliner. 2007. Identifying Cover Songs with Chroma Features and Dynamic Programming Beat Tracking. In *IEEE International Conference on Acoustics, Speech and Signal Processing.*

[5] Emilia Gómez. 2006. Tonal Description of Polyphonic Audio for Music Content Processing. In *INFORMS Journal on Computing.*

[6] Eric J Humphrey, Oriol Nieto, and Juan Pablo Bello. 2013. Data Driven and Discriminative Projections for Large-Scale Cover Song Identification. In *International Society for Music Information Retrieval Conference.*

[7] Nanzhu Jiang, Peter Grosche, Verena Konz, and Meinard Müller. 2011. Analyzing Chroma Feature Types for Automated Chord Recognition. In *Audio Engineering Society Conference: 42nd International Conference: Semantic Audio.* Audio Engineering Society.

[8] Maksim Khadkevich and Maurizio Omologo. 2013. Large-Scale Cover Song Identification Using Chord Profiles.. In *International Society for Music Information Retrieval Conference.*

[9] Henrique BS Leão, Germano F Guimarães, Geber L Ramalho, Sérgio V Cavalcante, and others. 2003. Benchmarking Wave-to-MIDI Transcription Tools. In *University of São Paulo.*

[10] Kyogu Lee. 2006. Identifying Cover Songs from Audio Using Harmonic Representation. In *MIREX task on Audio Cover Song Identification.*

[11] Meinard Müller, Frank Kurth, and Michael Clausen. 2005. Audio Matching via Chroma-Based Statistical Features. In *International Society for Music Information Retrieval Conference.*

[12] Julien Osmalsky, Jean-Jacques Embrechts, Peter Foster, and Simon Dixon. 2015. Combining Features for Cover Song Identification. In *International Society for Music Information Retrieval Conference.*

[13] Joan Serra. 2011. Identification of Versions of the Same Musical Composition by Processing Audio Descriptions. In *Department of Information and Communication Technologies.*

[14] Diego Furtado Silva, Vinícius Mourão Alves de Souza, Gustavo Enrique de Almeida Prado Alves Batista, and others. 2015. Music Shapelets for Fast Cover Song Regognition. In *International Society for Music Information Retrieval Conference.*

[15] JMH van Balen, Dimitrios Bountouridis, Frans Wiering, Remco C Veltkamp, and others. 2014. Cognition-inspired Descriptors for Scalable Cover Song Retrieval. In *International Society for Music Information Retrieval Conference.*