



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

# 노래 신호의 자동 전사

Automatic Transcription of Singing Voice Signals

2017 년 8 월

서울대학교 융합과학기술대학원

융합과학부 디지털정보융합전공

허 훈

# Abstract

Automatic music transcription refers to an automatic extraction of musical attributes such as notes from an audio signal to a symbolic level. The symbolized music data are applicable for various purposes such as music education and production by providing higher-level information to both consumers and creators. Although the singing voice is the easiest one to listen and play among various music signals, traditional transcription methods for musical instruments are not suitable due to the acoustic complexity in the human voice. The main goal of this thesis is to develop a fully-automatic singing transcription system that exceeds existing methods. We first take a look at some typical approaches for pitch tracking and onset detection, which are two fundamental tasks of music transcription, and then propose several methods for each task. In terms of pitch tracking, we examine the effect of data sampling on the performance of periodicity analysis of music signals. For onset detection, the local homogeneity in the harmonic structure is exploited through the cepstral analysis and unsupervised classification. The final transcription system includes feature extraction and probabilistic model of the harmonic structure, and note transition based on the hidden Markov model. It achieved the best performance (an F-measure of 82%) in the note-level evaluation including the state-of-the-art systems.

**Keywords:** Automatic music transcription, music information retrieval, onset detection, pitch estimation, singing voice, harmonic structure

**Student Number:** 2011-31243

# Contents

<b>Abstract</b>	<b>i</b>
<b>Contents</b>	<b>ii</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>ix</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Definitions . . . . .	5
1.2.1 Musical keywords . . . . .	5
1.2.2 Scientific keywords . . . . .	7
1.2.3 Representations . . . . .	7
1.3 Problems in singing transcription . . . . .	9
1.4 Topics of interest . . . . .	10
1.5 Outline of the thesis . . . . .	13
<b>Chapter 2 Background</b>	<b>16</b>

2.1	Pitch estimation . . . . .	17
2.1.1	Time-domain methods . . . . .	17
2.1.2	Frequency-domain methods . . . . .	18
2.2	Note segmentation . . . . .	20
2.2.1	Onset detection . . . . .	20
2.2.2	Offset detection . . . . .	23
2.3	Singing transcription . . . . .	24
2.4	Evaluation methodology . . . . .	26
2.4.1	Pitch estimation . . . . .	26
2.4.2	Note segmentation . . . . .	27
2.4.3	Dataset . . . . .	28
2.5	Summary . . . . .	31

**Chapter 3 Periodicity Analysis by Sampling in the  
Time/Frequency Domain for Pitch Tracking 32**

3.1	Introduction . . . . .	32
3.2	Data sampling . . . . .	34
3.3	Sampled ACF/DF in the time domain . . . . .	37
3.4	Sampled ACF/DF in the frequency domain . . . . .	38
3.5	Iterative F0 estimation . . . . .	40
3.6	Experimental setup . . . . .	42
3.7	Result . . . . .	46
3.8	Summary . . . . .	49

**Chapter 4 Note Onset Detection based on Harmonic Cepstrum  
Regularity 50**

4.1	Introduction . . . . .	50
4.2	Cepstral analysis . . . . .	52
4.3	Harmonic cepstrum regularity . . . . .	56
4.3.1	Harmonic quefrency selection . . . . .	57
4.3.2	Sub-harmonic regularity function . . . . .	58
4.3.3	Adaptive thresholding . . . . .	59
4.3.4	Picking onsets . . . . .	59
4.4	Experiments . . . . .	61
4.4.1	Dataset description . . . . .	61
4.4.2	Evaluation results . . . . .	62
4.5	Summary . . . . .	64

**Chapter 5 Robust Singing Transcription System using Local Homogeneity in the Harmonic Structure 66**

5.1	Introduction . . . . .	66
5.2	F0 tracking . . . . .	71
5.3	Feature extraction . . . . .	72
5.4	Mixture model . . . . .	76
5.5	Note detection . . . . .	80
5.5.1	Transition boundary detection . . . . .	81
5.5.2	Note boundary selection . . . . .	83
5.5.3	Note pitch decision . . . . .	84
5.6	Evaluation . . . . .	86
5.6.1	Dataset . . . . .	86
5.6.2	Criteria and measures . . . . .	87
5.6.3	Experimental setup . . . . .	89

5.7	Results and discussions . . . . .	90
5.7.1	Failure analysis . . . . .	95
5.8	Summary . . . . .	97
<b>Chapter 6 Conclusion and Future Work</b>		<b>99</b>
6.1	Contributions . . . . .	99
6.2	Future work . . . . .	103
6.2.1	Precise partial tracking using instantaneous frequency . .	103
6.2.2	Linguistic model for note segmentation . . . . .	105
<b>Appendix</b>		<b>108</b>
	Derivation of the instantaneous frequency . . . . .	108
<b>Bibliography</b>		<b>110</b>
<b>초 록</b>		<b>124</b>

# List of Figures

Figure 1.1	Music services and the accessibility for the listener, performer, and developer/provider. . . . .	3
Figure 1.2	Four representations of the first four measures of Ludwig van Beethoven’s Piano Sonata No. 8 in C minor, Op. 13. . . . .	8
Figure 1.3	Best performance results in the MIREX audio onset detection task during 2005-2016. . . . .	9
Figure 2.1	General workflow of onset detection. . . . .	21
Figure 2.2	General workflow of singing transcription. . . . .	25
Figure 3.1	An example of amplitude and prominence. . . . .	36
Figure 3.2	An illustrative comparison between the original detection function and the sampled function by using data amplitude and prominence. (Top) an input signal. (Middle) autocorrelation functions. (Bottom) difference functions. The original difference function is reversely normalized for comparison. . . . .	39
Figure 3.3	Gross error rates per data sampling ratio. . . . .	48



Figure 4.1	Architecture of the proposed onset detection system. . .	53
Figure 4.2	Waveform and spectrogram of a clarinet, a violin, and a singing voice signal. . . . .	56
Figure 4.3	Comparison between a sustain and a transient. Each vertical line represents its relative amplitude of cepstral coefficient. . . . .	57
Figure 4.4	(a) Waveform of a violin signal. (b) Detection function and adaptive threshold. (c) Five harmonic quefrequencies. (d) Five sub-harmonic cepstral coefficients. . . . .	60
Figure 4.5	F-measure comparison for different classes of onset. . . .	64
Figure 5.1	Schematic flow underlying the proposed transcription system. . . . .	70
Figure 5.2	A two-dimensional example of the vector rotation. (a) A scatter plot of the original and the rotated data. Eigenvectors and eigenvalues are depicted by the direction and the length of arrows. (b) A density plot of angles for both data when normalized onto the unit circle. . . . .	75
Figure 5.3	Note counts by different stream lengths and the heuristic regression of the maximum number of clusters. Each dot in the scatter plot represents a stream. Variances in the box plot are shown with stream groups divided in a step of 0.2 s. The regression function $g(T) = \min(\lceil 5T \rceil, 5)$ is depicted by the red line. . . . .	79
Figure 5.4	Flowchart on the cluster optimization. . . . .	80
Figure 5.5	Transitions in the hidden Markov model. . . . .	83

Figure 5.6	Transcription result from an excerpt of <code>afemale10.wav</code> in the dataset. . . . .	85
Figure 5.7	Average F-measures in three criteria by different number of harmonic partials. . . . .	92
Figure 5.8	Precision-recall curves in COn criterion for two transition detection methods. . . . .	92
Figure 5.9	Evaluation comparison of the proposed system (marked by asterisk) and other algorithms. Labels on the y-axis indicate the criteria and their numerical measure. Items marked by crosses are not publicly announced. . . . .	93
Figure 5.10	Failure analysis for a case of incorrect transcription caused by long-tail release. The ending part of the note formed a different cluster (from 8.5 to 9 seconds). . . . .	96
Figure 6.1	A screenshot of the program implementation of the proposed system. . . . .	102
Figure 6.2	Instantaneous frequency of harmonic components in a singing voice signal. . . . .	104
Figure 6.3	IPA vowel chart. . . . .	106

# List of Tables

Table 2.1	Confusion matrix in binary classification. . . . .	27
Table 2.2	Datasets for music transcription. . . . .	30
Table 3.1	Pitch tracking errors comparison over the time-domain methods. . . . .	46
Table 3.2	Pitch tracking errors comparison over the frequency- domain methods. . . . .	47
Table 4.1	Dataset details. . . . .	62
Table 4.2	Performance of the proposed algorithm. . . . .	63
Table 5.1	Parameter configuration. . . . .	90
Table 5.2	Computational time of the proposed system. . . . .	90

# Chapter 1

## Introduction

### 1.1 Motivation

Music is one of the most popularly produced and consumed content today. In the last decade and a half, it is reported that over 100,000 albums are released a year worldwide [1]. Perceiving or not, people routinely consume music all the time in their life. Background music is always being played during commute, work, exercise, even watching TV programs at home after work. According to a survey of 3,000 people by an American survey organization, the average American listens to four hours of music each day [2]. They listen to music via a variety of channels including radio, internet streaming, and their owned music. According to another survey, the average person will spend 13 years of their lives listening to music [3]. As such, it is evident that music is a very deeply ingrained content in our lives.

Nevertheless, the use of music content still remains unidirectional because

listeners receive only limited information in the music. In a typical listening environment such as online streaming, only the raw audio and a few metadata (e.g. artist, song title, album title) are available. Musically highly-trained people can immediately play the melody that they just listened, but most ordinary people need a musical score (i.e. sheet) to practice their playing or singing. People may purchase the scores from online websites to get more information about the melody they listened, but most of these scores are not guaranteed to be correctly transcribed because those are manually processed by individuals. The most authoritative source is the one that the creator directly publishes, but it is very rare for creators to distribute it publicly. Also, procedures for music creation are sometimes done even without the score. Considering the fact that the band music has become popular in the contemporary music, it is now very common that the inspiration of the artist is realized directly in the form of audio signal.

For decades, listeners have been provided only with the raw audio and basic metadata, which caused the unidirectional consumption of music. As the streaming has become the primary channel of listening to music, fortunately, latest services that provide high-level information are growing. Beatport [4] established a platform for buying and selling multi-track audio sources for music creators and DJs. Some musical attributes such as *tempo* and *key* are also provided with the downloaded audio, but these are manually tagged by creators and therefore not applicable to existing songs. Pandora [5] and Last.fm [6] are the ones that analyze many users' listening patterns and statistics to provide personalized music recommendation services, but the use of the information inherent in the content itself is very limited. Echonest [7] performs the content-









Provider	 Spotify  MUSIC  TIDAL	PANDORA last.fm	 beatport	 echonest
Information/Service	Audio Metadata (song title, artist, album title) Cover image	Song/playlist recommendation Listening statistics	Multitrack audio source Musical properties (key, tempo)	Content analysis data (key, tempo, beat)
Accessibility	 Listener		 Artist/Performer	
	 Developer/Provider			

Fig. 1.1 Music services and the accessibility for the listener, performer, and developer/provider.

based analysis of *tempo*, *key*, and *beat* in music and distributes them through a web API, but it is more useful for service developers and researchers rather than most general listeners.

Automatic music transcription (AMT) allows bi-directional and interactive consumption, and the potential for the content to be expanded and reproduced. It refers to a task which extracts a musical notation in the form of symbolic data from audio recordings. AMT not only generates musical scores for songs that the score does not exist but also can help consumers to fully understand and enjoy music. The symbolized music does not necessarily have to be in a form of traditional Western music score, but it often consists of numerical data of the musical attributes of each note at a more fundamental level. In some aspects, this is likened to inferring the recipe after taste of a dish or reverse-engineering the source code of a computer program [8].

Music contents can be utilized in various fields when they are provided as symbolized data together with audio signals and metadata. First, it can be used for educational purposes for those who want to learn to sing or play musical instruments. Songs2See [9], a project started at the Fraunhofer Institute in Germany, is a game application to learn by playing real instruments. Users can practice the traditional instruments such as guitar, piano, saxophone, flute, bass, clarinet to play popular classical music, and receive the points by their performance. Time-scale modification and pitch-shifting technologies are applied for adjusting tempo and key of accompaniment music. AMT is also used for more creative purposes such as automatic accompaniment generation. Microsoft’s Songsmith [10, 11] is an application that automatically creates accompaniments that match singer’s voices. This service is not only for beginners who are not musically trained at all but also for musicians who want to get musical inspiration easily. More recently, mobile learning market is expanding these days with the advances in mobile technologies. Smartphones or tablet devices are becoming an important platform for learning and training. Music education on mobile platforms is one of the most popular areas, and mobile apps such as Yousician [12] provide a real-time tutoring service for beginners who like to play guitar, piano, or violin. In such applications, users’ performance is recorded through a built-in microphone, and transcribed into note-level data in order to guide users to play a given music score correctly.

AMT is also useful as an fundamental study for higher-level music information retrieval (MIR) tasks. It can be used as a front-end in tasks such as query-by-humming (QBH) or melodic similarity analysis. Also, this will give useful information in musicology-based MIR studies such as tonality analysis

and harmonics theory.

In the rest of this chapter, we define some important keywords in both musical and scientific field. We also introduce several ways to represent music and music signals, and describe the scope of our research. Next, the problem statement in the singing transcription, which is the destination that this thesis is heading for, is followed. Finally, we summarize our major contributions and the outline of the thesis, before the description of about the detailed algorithms and system.

## 1.2 Definitions

As this thesis deals with aspects of music and singing voice, this section describes some important definitions for a clear understanding of relevant terms and how they will be used throughout this dissertation.

### 1.2.1 Musical keywords

**Pitch** is the ordering of sounds on a frequency-related scale extending from low to high. Pitch can be expressed in two major units: Hz, the unit of frequency, and MIDI<sup>1</sup> note number, which converts the frequency in semitone scale. In the early days, the conversion formula was into a range of [1, 88] corresponding to the frequencies of a typical 88-key piano keyboard, but later 106-key range is used as a wider pitch range is required. The MIDI note number is converted from the frequency into the logarithmic scale as  $12 \log_2(\text{frequency}/440) + 69$ . The voiced speech of a typical adult male will have a fundamental frequency

---

<sup>1</sup>MIDI (Musical Instrument Digital Interface) is an industry standard specification for transmitting and sharing performance data of electronic musical instruments.



(F0) from 85 to 180 Hz, and that of a typical adult female from 165 to 255 Hz [13]. In the case of singing, the highest F0 increases to about 1 kHz.

**Onset** is the exact time when a note starts. An onset can be hard or soft depending on the attack time, which is “the time taken for initial run-up” of the amplitude envelope [14]. Typical examples of hard onset include pitched percussive instruments such as vibraphone, xylophone, and keyboard instruments such as piano. These instruments are relatively easy to detect because all notes have a clear attack in their amplitude envelope. On the other hand, bowed string instruments such as violin or cello can be an example of the soft onset. Similarly, singing voice can also be classified as soft onset.

**Duration** is the time during which a note is playing. The time of which an onset plus its duration is called the *offset*, indicating the end of the note. Since it is often ambiguous to locate clearly where the notes end on most instruments, it is generally considered more difficult to detect offset than onset.

**Articulations** refers to performance techniques which affect the transition or continuity on a single note or between multiple notes or sounds. This includes *staccato* which infers playing strongly, and *slur* playing smoothly on several notes. As a particular type of articulation, *legato* refers to an articulation in which the melody is played from note to note with no intervening silence. This playing technique is closely related to soft onset appeared in such as string instruments and singing voices.

In this dissertation, a transcription system interprets a melody from a single source at the symbolic level, on the basis of three attributes of musical notes:

onset, duration, and pitch.

### 1.2.2 Scientific keywords

**Harmonic** (also known as partial) is a wave which is added to the fundamental wave. An acoustic signal with a certain pitch can be expressed by the sum of  $k$  sinusoids as follows:

$$x(t) = \sum_k A_k(t) \cos(2\pi f_k(t) + \phi_k) \quad (1.1)$$

where  $A_k(t)$  is the amplitude,  $f_k(t)$  is the frequency, and  $\phi_k(t)$  is the phase at a time instance  $t$ . The *fundamental frequency* (F0) is closely related to the pitch of the complex tone. The relative energy of the harmonics determines the timbre of a tone.

**Harmonic structure** refers to the energy relationship between harmonic partials in an acoustic context. It is distinguished from the definition used in music theory, which is the combination of musical sounds in consonances and progressions in the musical context of mode and tonality.

### 1.2.3 Representations

We here introduce four ways to represent music and music signals. The first is *waveform*, the most physical form. From an acoustic perspective, the sound has the physical property of the wave. As a function of time, its amplitude is related to the sound pressure caused by waves propagating into the air.

The second is a *time-frequency representation* through the short-time Fourier transform (STFT) or the constant-Q transform (CQT). Unlike the waveform (of one-dimensional time series data), time-frequency representations

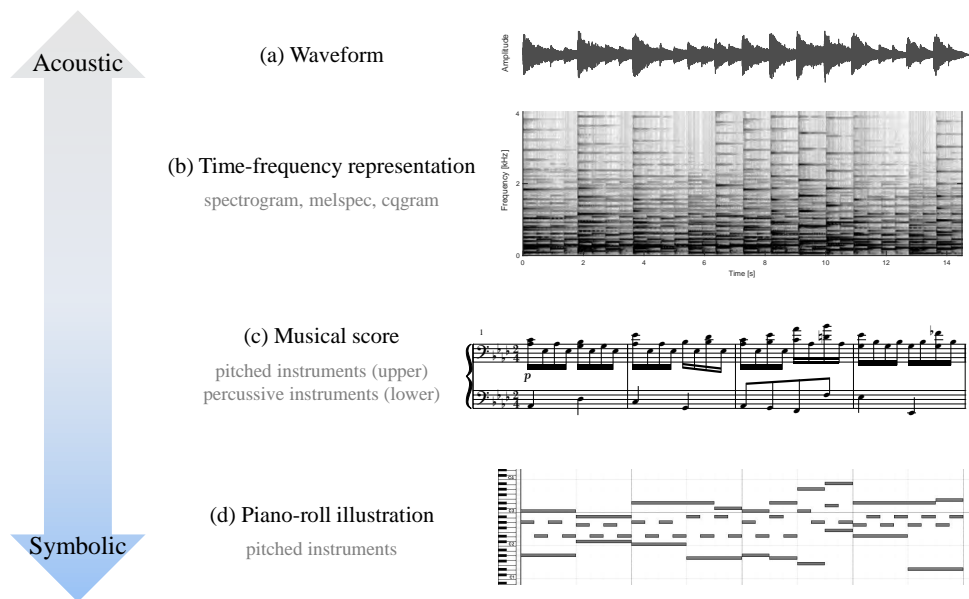


Fig. 1.2 Four representations of the first four measures of Ludwig van Beethoven’s Piano Sonata No. 8 in C minor, Op. 13.

consist of two axes: time and frequency. The frequency axis can be along with the mel scale that approximates the frequency perception in the human auditory system, or a logarithmic scale that is directly proportional to the chromatic scale in western music.

The third is *musical score*, the most traditional representation. It can express not only the basic attributes of note but also various musical information such as key, tempo, bar, and playing style.

The last representation is *piano-roll illustration*, which represents only three attributes of onset, duration, and pitch. It is the most symbolic representation in that it only expresses the basic attributes of notes more abstractly than the traditional musical score.

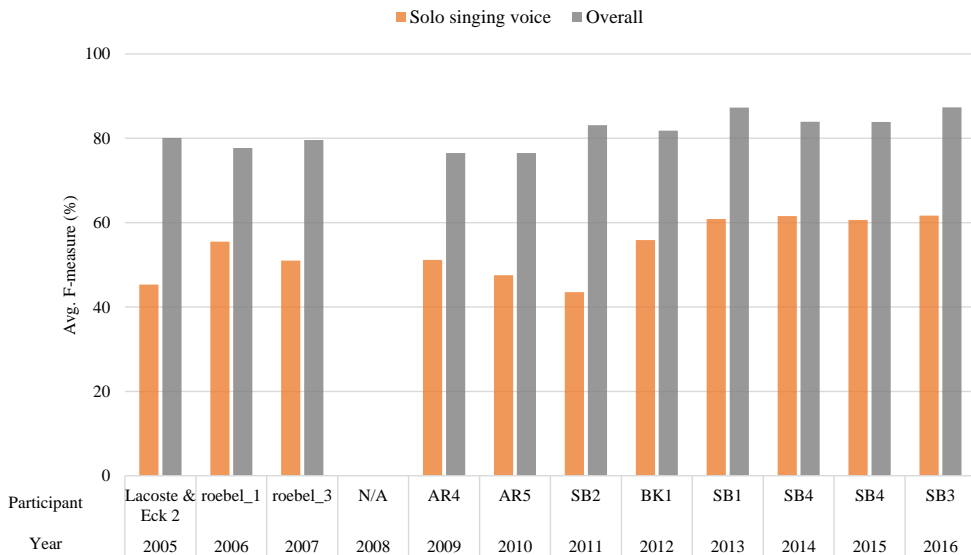


Fig. 1.3 Best performance results in the MIREX audio onset detection task during 2005-2016.

### 1.3 Problems in singing transcription

Most of MIR tasks are evaluated annually in the Music Information Retrieval Evaluation eXchange (MIREX)<sup>2</sup>. Audio onset detection is one of the most traditional tasks since 2005, and it aims to detect note onset in various musical signals such as mixed music and various types of instruments. As shown in Fig. 1.3, the performance gap between the overall score and singing voice has not been narrowed until 2016.

As such, singing voice is more difficult to transcribe than typical instruments due to its complex characteristics. We classified them into three factors as follows:

<sup>2</sup><http://www.music-ir.org/mirex/>

1. **Diverse tone:** Different instruments with the same kind have a common sound tone, even if they are played by different performers. However, the tone of the singing voice varies greatly depending on the gender and age of a singer. The singer's unique tone differ from another even with the same gender and age. In addition, the voices of the same singer have different harmonic structures by the pronunciation and singing style. Accordingly, it is not beneficial to use an intrinsic harmonic structure of the singing voice, which is used for the analysis of musical instruments.
2. **Erratic loudness:** Singing voice allows a richer musical expression than any other instrument, and this becomes one of the obstacles for automatic transcription. Crossing notes and notes, the loudness of singing voices is very unpredictable and immediately changing, and this is often observed even in the same note. This characteristic is not found in instruments such as piano and guitar, and makes it difficult to apply the traditional onset detection methods based on the energy change of the signal.
3. **Abundant articulation:** The delicate musical expression of a singing voice is found not only in terms of loudness but also in pitch. *Vibrato* produces a regular, pulsating change of pitch, and *ornament* decorates melody with a very short and fast note before or after a reference note, and *glissando* makes a continuous change of pitch from one to another.

## 1.4 Topics of interest

Automatic music transcription from audio has long been one of the most intriguing problems and a challenge in the field of music information retrieval,

because it requires a series of low-level tasks such as onset/offset detection and pitch estimation, followed by high-level post-processing for symbolic representation. Onset detection and pitch estimation, which are two fundamental topics in music transcription, have been considered separate tasks in early days. In the former case, there have been various attempts for note onset detection of a music signal since the mid-90s. To detect the onset of a single instrument signal, the first attempt was a simple approach using the derivative of the amplitude envelope [15]. However, because of the different characteristics depending on the type of instrument, researchers have attempted to find suitable approaches to the characteristics of instruments classified as pitched non-percussive (PNP) and non-pitched percussive (NPP).

For the latter, the term ‘pitch tracking’ has become more preferred for recent years, as it is more useful to utilize the pitch contour instead of a single pitch value from a quasi-periodic signal. The most traditional and effective method is to choose a frequency at the global maximum value of a detection function generated from such as the autocorrelation. This approach was very intuitive and somewhat effective, but the key was to minimize the ‘octave error,’ which means that estimates are sometimes doubled or half frequencies.

Among the three main topics that make up this thesis, the first two topics deal with pitch tracking and onset detection of general monophonic music signals, respectively. Regarding the pitch tracking, we exploit a data sampling method to reduce the octave error, which is one of the major cause of incorrect estimation, examining on several existing methods. The term ‘data sampling’ is used in this method to refer to the selective acquisition of extrema (i.e. maxima and minima) related to the predominant period instead of using all the given

samples, while generating a detection function such as autocorrelation from the discretized digital signal.

In terms of onset detection, we pay attention to the point that existing algorithms are not capable of detecting soft onsets, and present an alternative approach using the harmonic structure. It is noticed that the harmonic structure remains stable within a single note, even in the case of complex music signals where the amplitude envelope is unpredictable and the tone varies. We examine that this hypothesis can be applicable to onset detection of various kinds of musical instruments.

As the transcription of complex music signals has been more attempted, onset detection and pitch tracking is not treated as separate tasks anymore, and many approaches aim to solve them in a single framework. A typical example is an algorithm that transcribes polyphonic music based on non-negative matrix factorization (NMF) [16, 17] and recurrent neural networks (RNNs) [18]. More recently, RNNs have been used to detect the presence of singing voice from mixed music signals [19]. In general, transcription of polyphonic music is considered a more challenging problem than of monophonic music. The main reason for this is that multiple notes are often played simultaneously, and thus individual notes interfere by virtue of their harmonic relations [20]. When only a single instrument is targeted, polyphonic music has notable achievements in recent studies due to the regularized harmonic structure. However, the transcription of singing voices with complex characteristics still remains room for improvement, for the reasons explained in Section 1.3.

The goal of this thesis is ultimately extended to a fully-automatic transcription of singing voice signals. This includes seeking and enhancing the suitable

approach to the complex characteristics of singing transcription. We expand the hypothesis that was first attempted in our onset detection study, and develop a system that can effectively use the local homogeneity of the harmonic structure. Compared with other singing transcription systems, the proposed system aims to achieve high performance for various types of singing voices.

## 1.5 Outline of the thesis

**Chapter 2** provides an overview of the two main topics of music transcription: pitch estimation and note segmentation. Focusing on existing methods that are widely used, we investigate which approaches have been historically taken for each topic. In addition, we review the literature on singing transcription, which is the final objective of this thesis, and highlighted the differences between singing transcription and traditional transcription systems. Finally, the public datasets used in many studies related to music transcription are listed to aid in future research.

**Chapter 3** examines the effect of data sampling on periodicity analysis of a quasi-periodic signal. In dealing with the conventional pitch detection functions, such as the autocorrelation function or the average magnitude difference function, we select only a few samples that are regarded to be relevant to the periodicity, rather than using all the data. Periodicity analysis is performed in the time and the frequency domains, with an iterative linear estimation technique for precise refinement of the fundamental period. To evaluate the method, it is utilized as a monophonic pitch tracker, and we measure the performance using the traditional pitch error metrics. The experimental result shows the



improvement in the gross pitch error and the octave pitch error, compared to the original method without sampling.

**Chapter 4** presents a novel onset detection algorithm based on cepstral analysis. Instead of considering unnecessary mel-scale or any interests of non-harmonic components, we selectively focus on the changes in particular cepstral coefficients that represent the harmonic structure of an input signal. In comparison with a conventional time-frequency analysis, the advantage of using cepstral coefficients is that it shows the harmonic structure more clearly, and gives a robust detection function even when the envelope of waveform fluctuates or slowly increases. As a detection function, harmonic cepstrum regularity (HCR) is derived by the summation of several harmonic cepstral coefficients, but their quefrency indices are defined from the previous frame so as to reflect the temporal changes in the harmonic structure. Experiments show that the proposed algorithm achieves significant improvement in performance over other algorithms, particularly for pitched instruments with soft onsets, such as violin and singing voice.

**Chapter 5** discusses a comprehensive transcription system for monophonic singing voice based on harmonic structure analysis. Given a precise tracking of the fundamental frequency, a novel acoustic feature is derived to signify the harmonic structure in singing voice signals, regardless of the loudness and pitch. It is then used to generate a parametric mixture model based on the von Mises–Fisher distribution, so that the model represents the intrinsic harmonic structures within a region of smoothly connected notes. To identify the note boundaries, the local homogeneity in the harmonic structure is exploited by two

different methods: the self-similarity analysis and hidden Markov model. The proposed system identifies the note attributes including the onset time, duration and note pitch. Evaluations are conducted from various aspects to verify the performance improvement of the proposed system and its robustness, using the latest evaluation methodology for singing transcription. The results show that the proposed system significantly outperforms other systems including the state-of-the-art systems.

# Chapter 2

## Background

In this chapter, we present an overview of various methods for automatic music transcription. As aforementioned in the introduction, music transcription systems include two main topics of pitch estimation and note segmentation. Although the two topics are not independent when a transcription system is configured, we provide a review of each topic to help to understand the whole transcription system.

This chapter is organized as follows. Section 2.1 presents methods for pitch estimation in the time domain and the frequency domain. Section 2.2 introduces onset detection algorithms for music signals, and shortly addresses offset detection. Then, we provide a historical review of singing transcription in Section 2.3, and describe the evaluation methodology that is currently used in most papers related to music transcription in Section 2.4. Finally, Section 2.5 concludes the chapter.

## 2.1 Pitch estimation

Most pitch estimation algorithms are done in the time domain or the frequency domain. There are a few methods based on the cepstrum analysis, which was mainly attempted in speech processing, to use the quasi-periodic magnitude of the spectrum. However, it is not commonly used today because the estimation accuracy is not superior to the time-domain methods, despite the high complexity due to the double transform. Therefore, this section covers the time-domain and frequency-domain methods that are widely used.

### 2.1.1 Time-domain methods

Autocorrelation-based F0 estimation is the most straightforward method for pitch estimation. Basically, it calculates the correlation between a time-domain signal and its time-shifted version. Then, it finds the global maximum in the autocorrelation function (ACF), and converts the time lag at the global maximum into the frequency scale in Hz.

Given a time-domain signal  $x(t)$  and a frame length  $T$ , the autocorrelation function  $r_{t_0}(\tau)$  at time  $t_0$  is defined by

$$r_{t_0}(\tau) = \sum_{t=t_0}^{t_0+T-1} x(t)x(t+\tau) \quad (2.1)$$

where  $\tau$  is the time lag.

In practical cases, it is often observed that the actual F0 corresponds to one of the local maxima rather than the global maximum. This is called the ‘octave error,’ which is caused by higher harmonics that have stronger energy than the fundamental frequency does. To solve this problem, many autocorrelation-based methods refine the autocorrelation function so that the global maximum

corresponds to the F0. A pre-processing such as spectral whitening can be a possible solution.

Another popular time-domain method is YIN algorithm [21]. The YIN algorithm is the most widely used pitch estimator owing to its high performance despite its simple and clear structure. It has a similar structure to the auto-correlation method, but it uses the difference instead of the multiplication of signal amplitude.

The first step of the YIN algorithm is calculation of the absolute value of the amplitude difference

$$d_{t_0}(\tau) = \sum_{t=t_0}^{t_0+T-1} |s(t) - s(t + \tau)|. \quad (2.2)$$

Then, the cumulative mean-normalized difference function is derived from  $d_{t_0}(\tau)$  as

$$d'_{t_0}(\tau) = \begin{cases} 1, & \tau = 0 \\ \frac{d_{t_0}(\tau)}{(1/\tau) \sum_{j=1}^{\tau} d_{t_0}(j)}, & \text{otherwise.} \end{cases} \quad (2.3)$$

The F0 estimate is determined by the smallest  $\tau$  for which a local minimum of  $d'_{t_0}(\tau)$  is smaller than a threshold. A second-order polynomial interpolation is followed to adjust the time lag  $\tau$  precisely.

### 2.1.2 Frequency-domain methods

The advantage of using frequency-domain methods is that they are easy to extend to multiple-F0 estimation. The time-domain methods contain a post-processing step to emphasize the peak corresponding to a single F0, therefore the harmonic energy of the signal may be distorted. On the other hand, algorithms based on time-frequency representations such as the short-time Fourier

transform (STFT) are free to use the pure harmonic energy. Nonetheless, it is beneficial to use a time-domain method that yields a lower estimation error for purposes of singing transcription.

In early days, Noll proposed a pitch estimator called the harmonic product spectrum (HPS) for human speech signals [22]. The HPS method measures the maximum coincidence for harmonics for each spectral frame. The periodic correlation  $Y(k)$  of the spectrum  $X(k)$  is calculated by

$$Y(k) = \prod_{r=1}^R |X(rk)| \quad (2.4)$$

where  $R$  is the number of harmonics to consider. The F0 estimate is obtained at the global maximum of  $Y(k)$ .

The HPS algorithm is simple to implement, thus can be run in the real-time environment. However, the octave errors are commonly observed because the second harmonics are often stronger than the F0 in most real-world music signals. Moreover, the downsampling technique (expressed as  $X(rk)$  term) cannot help but lose the discrete data when the spectral resolution is insufficient.

Another frequency-domain method called subharmonic summation (SHS) [23] can give an alternative way to solve this resolution issue. This method applies a similar concept to the HPS algorithm to add up all the spectral components that have a harmonic relation with the F0, but the original spectral resolution is preserved as the SHS method uses the shifted spectrum instead of downsampling the spectrum.

## 2.2 Note segmentation

In the introduction chapter, we defined an onset as the time at which a note begins. From a signal processing perspective, three terms can be defined considering the acoustic characteristics at the beginning of a note [14]:

- Attack: The Time interval during which the amplitude envelope increases.
- Transient: Short intervals during which the signal evolves quickly in some nontrivial or relatively unpredictable way.
- Onset: A single instant chosen to mark the temporally extended transient.

The definition for offset in terms of signal processing are not found in the literature. This seems to be due to the difficulty in generalizing its signal characteristics, as the offset has very different aspects depending on instruments. For a monophonic signal, it is possible to locate the offset at which the frame begins to be unvoiced.

### 2.2.1 Onset detection

Most onset detection methods consist of three main steps as follows. First, preprocessing aims to emphasize the most important characteristics of onset detection in input signals. Techniques such as spectral whitening and transient/sustain separation are consistently mentioned in literature. Second, the reduction step converts the audio signal into a highly-processed detection function that represents the occurrence of the transient in the original input signal. This step determines the detection function, which is a key part of the algorithm. The last step is to select identifiable local maxima (i.e. peaks) in the detection function.

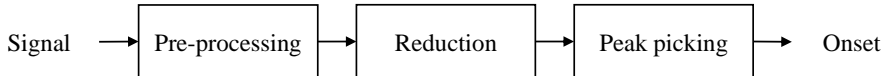


Fig. 2.1 General workflow of onset detection.

## Amplitude Envelope

The very first method for onset detection is reported to have been proposed by Schloss in the mid-80s [15]. The detection function in this method is originated from the amplitude envelope of the input signal  $x(n)$ :

$$E(n) = \sum_{m=-N/2}^{N/2-1} (x(n+m))^2 w(m) \quad (2.5)$$

where  $w(m)$  is a window function. The above formula does not imply the onset strength, and preprocessing such as low-pass filtering of the input signal  $x(n)$  must be accompanied.

## Time-Frequency Analysis

A method proposed by Goto et al. [24] is an example of applying the analysis in the time-frequency domain to onset detection. A component  $p(t, f)$  in the STFT magnitude that fulfills the below condition is regarded as an onset component.

$$\begin{cases} p(t, f) > pp, & pp = \max(p(t-1, f), p(t-1, f \pm 1), p(t-2, f)) \\ np > pp, & np = \min(p(t+1, f), p(t+1, f \pm 1)) \end{cases} \quad (2.6)$$

This approach allows to extract the detection function intuitively from the magnitude spectrum. However, it is vulnerable to noise, and has limits to be able to detect only a very strong onset.



## High Frequency Content

Since the late 90s, algorithms that are not only simple but decently performed have been studied. Considering the dramatic changes in the signal at the moment of the new note event, Masri proposed a detection function by weighting along with the frequency axis as below [25]:

$$E_k(n) = \sum_{k=2}^{N/2+1} W_k |X_k(n)|^2 \quad (2.7)$$

where  $W_k$  is the weight on the frequency bins. With  $W_k = k$ , the detection function increases when the stability of the signal rapidly collapses, because the higher frequency components have a greater weight. In particular, this method is known to be effective for the detection of non-pitched percussive (NPP) onsets.

## Psychoacoustic Knowledge

With the achievements in the psychoacoustics field, multilateral approaches have been applied to MIR systems. Based on the perceptual evidence that the loudness is perceived logarithmically by the human hear, Klapuri proposed the following detection function [26]:

$$W(t) = \frac{d}{dt} (\log E(t)) = \frac{\frac{dE(t)}{dt}}{E(t)} \quad (2.8)$$

where  $E(t)$  is the amplitude envelope. The local maxima of the detection function became closer to the actual perceived point by using the logarithmic scaling, which is the biggest difference from the previous methods.

## Spectral Flux/Difference

Among the various methods using the derivative of successive STFT magnitudes, the spectral flux (or difference) method is most widely used one. Like

the HFC algorithm, it is effective for NPP onsets but is often reported to perform better. The detection function is derived as below [27]:

$$SD(n) = \sum_{k=-N/2}^{N/2-1} \{H(|X_k(n)| - |X_k(n-1)|)\}^2 \quad (2.9)$$

where  $H(x) = \frac{x+|x|}{2}$ .

Given the magnitude spectrum at two successive frames, the distance between two vectors determines the detection function. When  $\ell_1$ -norm is used as a distance metric, then the detection function is called spectral flux. If  $\ell_2$ -norm or Kullback-Leibler divergence is used, it is called spectral difference.

### 2.2.2 Offset detection

Offset detection has been considered a minor task compared to onset detection. There are only a few studies on offset detection, because it does not give any usefulness by itself; Acoustic features that do not require offsets, such as inter-onset intervals, are already exploited in many papers. Another reason why it has been hardly attempted is that offsets are not as salient as onsets. In many instruments, decaying of a note is slow and gradual, so it is very difficult to measure the exact time of an offset. This ambiguity makes it hard to annotate the ground truth, and thus hard to evaluate.

For the reasons described above, most note segmentation methods do not prepare a specific process for offset detection. An exceptional example is an approach which models the activity of a note using a two-state hidden Markov model (HMM) [28]. In this approach, the offset is detected at the frame when an active pitch between two consecutive onsets changes from an active to an inactive state for the first time, so that the moment when offset detection occurs

can be specified.

## 2.3 Singing transcription

Since singing voice is much harder to transcribe due to its complex signal characteristics compared to typical instruments, singing transcription has begun to be attempted relatively late. The first algorithm for transcription of singing voice was proposed in McNab's work [29], which devised a note segmentation scheme using the RMS energy and the F0 of the signal. If the signal energy exceeds a given threshold, this method interprets it as the beginning of the note, and similarly, the note offset is inferred if the signal energy falls below a threshold. Additionally, a note boundary was detected when an F0 estimate was more than half a semitone from the average of the previous estimates during the segment. As shown in Fig. 2.2, McNab contributed to establishing a framework for general singing transcription. In this framework, note pitch identification is included in consideration of the difficulty lying in singing transcription in which the F0s are unstable within a note, which is the most different aspect from the transcription of other musical instruments.

Due to the vulnerability in legato note detection, however, the method based solely on the signal energy cannot be applied to most singing voice inputs. Therefore, many later researchers have tried to use the discontinuity of the F0. Pollastri used both the energy and F0 changes for note segmentation, as well as the zero crossing rate (ZCR) used to make voiced/unvoiced decisions [30]. Clarisse et al. performed note segmentation based on the adaptive adjustment of thresholds for signal energy. Later, their system was improved by adding several decision rules for richer expressions with *legato*, *vibrato*, and *tremolo*

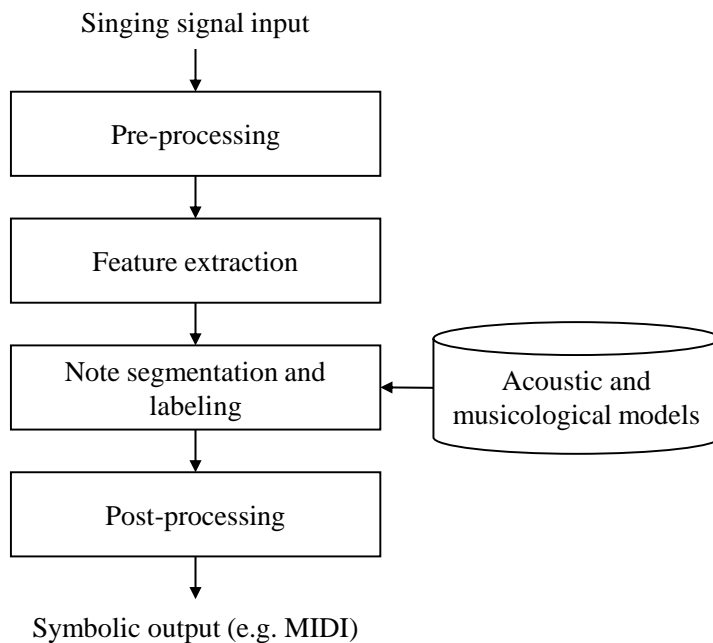


Fig. 2.2 General workflow of singing transcription.

[31].

Voicing can be another feature to detect not only a silence between notes but also some legato notes with a strong transient noise at the beginning of the note (i.e. attack). In particular, fricative consonants ([s], [f], [v]) and stop/plosive consonants ([p], [b], [t]) are the syllables come with very strong noise. The voicing detection is commonly embedded within the F0 tracking. The autocorrelation-based method determined an aperiodic signal if it cannot detect any clear peak in the ACF, or the peak amplitude is less than a threshold. In the YIN algorithm, the frame is determined as unvoiced when any of the difference (YIN) value is not less than a threshold.

## 2.4 Evaluation methodology

As music transcription has been studied for over a decade, methodology and framework to evaluate the performance have been well-established. This section provides a brief overview of the standard evaluation methods on pitch estimation and note segmentation. For more information on how algorithms are practically evaluated using real-world data, see Chapter 3 (for pitch estimation), and Chapter 4-5 (for note segmentation). Also, this section introduces public datasets used in various papers related to music transcription.

### 2.4.1 Pitch estimation

For monophonic speech/singing signals, a pitch estimator can be utilized for voice activity detection (VAD), which identifies whether the voice is active or not. Most pitch estimators include the voicing detection along with the F0 estimation accuracy in their performance evaluation.

To measure the estimation accuracy, two types of error metrics are commonly used. The first one is gross pitch error (GPE), for which the relative error to ground truth is greater than 20%. The measure of the GPE is defined by the ratio of the number of frames corresponding to the GPE and the total number of frames. The find pitch error (FPE), on the other hand, refers to an error where the relative error is less than 20%. In this case, statistics such as mean or standard deviation of the FPE are preferred rather than the ratio of frames.

In respect to voicing detection, *precision* (false alarm) and *recall* are used as measures. A ‘false alarm’ means an unvoiced frame that is estimated as voiced, and the precision will be low if it occurs frequently. Conversely, a high chance of

estimating a voiced frame as unvoiced implies that important information may be lost. In general, over 90% of recall is highly demanded for system stability. Precision and recall are the evaluation metrics for binary classification, and are described in detail in the following subsection.

## 2.4.2 Note segmentation

Precision and Recall are the measures that manifest the detection accuracy in pattern recognition and information retrieval related to binary classification. Precision is the ratio of the results that are predicted as relevant among the retrieved results, and recall is the ratio of the actually retrieved items among the items predicted as relevant. *F-measure* (also known as  $F_1$ -score) that combines the precision and the recall is widely used as the most representative measure for binary classification.

		True condition	
		Positive	Negative
Prediction	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Table 2.1 Confusion matrix in binary classification.

Those measures are calculated taking into account both the predicted documents and the ground truth. More specifically, the predicted results of two types (positive or negative) are combined with the true condition, thus all four outcomes are formulated consequently as shown in Table 2.1. The formulation of each measure is as follows:

$$\text{Precision} = \frac{n(\text{True Positives})}{n(\text{True Positives}) + n(\text{False Positives})} \quad (2.10)$$

$$\text{Recall} = \frac{n(\text{True Positives})}{n(\text{True Positives}) + n(\text{False Negatives})} \quad (2.11)$$

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.12)$$

For onset detection, the onset detected near the ground-truth onset time is treated as a true positive. The longest time difference determined as true positive is called a *tolerance window*, and it is usually given by 50 milliseconds. In other words, detected onsets that do not have any ground-truth onset within 50 ms are determined to be false positives. In the case of offset detection, a more loose criterion applies to the tolerance window, since accurate annotation is more difficult. It is generally given by 50 ms or 20% of note duration, whichever is larger.

### 2.4.3 Dataset

For a fair evaluation, it is important to evaluate the algorithms under the same experiment condition, including the dataset and measures. Because the dataset is particularly the factors that most directly affect the performance, it is necessary to choose carefully. Although the datasets used in the MIREX are considered to be a de facto standard, it is not publicly available in order to prevent participants from being strongly biased towards specific data. Instead, a variety of datasets for MIR tasks is available on the internet. We present here some widespread datasets for music transcription purposes.

In respect to pitch estimation, the dataset should provide audio recordings

and their corresponding ground-truth including the correct F0 sequence. The ground-truth F0 sequence is usually generated using an existing pitch estimator and then manually corrected. Since a pitch estimator is also treated as a voicing detector, the annotation may include data related to voiced/unvoiced frames. On the other hand, the dataset for note segmentation mainly provides the onset time and duration time of each note as the ground-truth. Some datasets such as Bach10 also provide the note pitch in the MIDI note number scale. Popular datasets for music transcription are listed in Table 2.2.



Table 2.2 Datasets for music transcription.

Dataset	Target task	Ground-truth	Audio type	Files	Reference
SVO1	Onset/offset detection, Pitch estimation	Onset time, offset time, note pitch	Solo singing voice (amateur singers)	22	Heo [32]
SVO2	Onset/offset detection	Onset time, offset time (partial)	Solo singing voice (singing experts)	14	Heo [32]
MIR-QBSH- corpus	Note detection, pitch estimation	Reference song (MIDI)	Solo singing voice	4,431	MIREX [33]
MIR-1k	Pitch estimation	F0 contour	Accompaniment and separable solo singing voice	1,000	MIREX [34]
TONAS	Onset/offset detection, pitch estimation	Onset time, du- ration time, note pitch	Solo singing voice (flamenco)	72	COFLA team [35]
Bach10	Onset detection, pitch estimation	Onset time, note pitch, F0 contour	Violin, clar- inet, saxophone, bassoon	40	Duan et al. [36]
Onset_Leveau	Onset detection	Onset time	Multi-type	20	Leveau et al. [37]
ODB	Onset detection	Onset time	Multi-type	20	University of Ali- cante [38]
RWC database (partially used)	Onset detection	Onset time	Solo instruments	Huge	Goto et al. [39, 40]
Jamendo Cor- pus	Vocal activity detec- tion	Vocal presence time	Mixed song	93	Ramona [41]

## 2.5 Summary

In this chapter, we have taken a look at a variety of methods related to three main topics: pitch estimation, note segmentation, and singing transcription. Some methods such as the YIN algorithm are still in use today, but many of the methods are no longer used due to their lower performance than recent achievements in the MIR field. However, by reviewing the past methods even if obsoleted, we hope that our readers have been helpful in understanding how to define problems and strategies for the purpose of music transcription. For example, focusing on how acoustic features are related to musical pitch and note onset can be a good starting point for seeking a novel approach.

The three main topics will be discussed in detail in the following chapters. First, we will present a pitch tracking method using data sampling to improve existing pitch estimation methods in Chapter 3. Second, a method for detecting soft onsets that have not been easily realized by existing methods will be discussed in Chapter 4. Lastly, in Chapter 5, we will describe our integrated system to transcribe note onset, duration, and pitch from singing voice signals.

## Chapter 3

# Periodicity Analysis by Sampling in the Time/Frequency Domain for Pitch Tracking

### 3.1 Introduction

In this chapter, based on the original work by Heo et al. [42], we describe the first stage of the singing transcription system proposed in this thesis. The estimation and tracking of the fundamental frequency in music and audio signals is one of the elementary techniques in the music information retrieval field. The analysis of a melody provides valuable data for many retrieval tasks. The pitch contour, which is an essential attribute in music transcription alongside note onset and offset, is utilized to analyze the harmonic structure of an input signal. Besides, frame-wise pitch values are utilized as a feature in recent applications, such as query-by-humming and audio fingerprinting. Since many pitch tracking algorithms have shown reliable performances for monophonic inputs, the range

of interest has expanded to multiple pitch estimation in recent years.

During several decades, there have been numerous attempts for pitch estimation and tracking. Most approaches can be classified into two groups: the time-domain approach and the frequency-domain approach. The time-domain approach estimates the period of a quasi-periodic signal. To specify the fundamental period, the autocorrelation function and the average magnitude difference function are widely used in many previous works. Methods using these functions can be computed and implemented easily. However, it is difficult for this approach to deal with polyphony detection, which can cause the octave errors. For music signals with unclear and complicated periodicity, sometimes doubled (or half) pitch is estimated compared to the reference pitch.

One of the most popular algorithms called YIN [21] prevents octave errors by utilizing several additional refinement steps, and its performance is still competitive today. To exploit higher-order statistics, correntropy is employed to analyze the temporal structure, showing advantages over traditional autocorrelation-based methods [43]. However, methods based on the autocorrelation or difference function can be prone to music signals with irregular harmonic structures. Although the fundamental frequency is the most dominant harmonic partial for most cases, the harmonic structure can sometimes be irregular depending on the instrument and the playing style. Probabilistic modeling is one solution, which measures the likelihood of matching up the harmonic structure with a pre-defined frequency map [44].

In the frequency-domain approach, the time-series data are transformed into the frequency domain using the short-time Fourier transform, and the most dominant part of the spectrum is detected. The harmonic product spectrum

(HPS) algorithm calculates the greatest common divisor of harmonic frequencies by the product of downsampled signals by degrees [22]. Pitch detection is also attempted by cepstral analysis [32], using the fact that the spectrum of a monophonic music signal contains equally-spaced harmonic partials. These approaches have the merit of expansibility to multiple pitch estimation.

Whichever approach is used, it is important to make a clear detection function, since most approaches determine the fundamental period by selecting the most predominant point of the detection function. For this purpose, we propose a method of analyzing periodicity using only a few relevant data. Data sampling is used here for effective computation as well as effective analysis of periodicity.

The rest of this chapter is organized as follows. We present the proposed method based on the sampling in both time and frequency domains from Section 3.2 to Section 3.4. The F0 refinement based on the least square method is explained in Section 3.5. Next, Section 3.6 describes the experimental setup to evaluate the pitch tracking performance. In Section 3.7, the results of the evaluation are reported along with the discussions. Finally, we draw our conclusion in the last section.

## 3.2 Data sampling

Given a quasi-periodic signal  $x(n)$ , two functions that are generally employed to represent the periodicity are defined by

$$D_{\text{ACF}}(m) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) x(n+m) \quad (3.1)$$

$$D_{\text{AMDF}}(m) = \frac{1}{N} \sum_{n=0}^{N-1} |x(n) - x(n+m)| \quad (3.2)$$

where  $D_{\text{ACF}}$  is the autocorrelation function (ACF) and  $D_{\text{AMDF}}$  is the average magnitude difference function (AMDF), respectively, and  $N$  is the length of the input signal. In general, these functions have a few local maxima at the smallest period and its integer-multiple values. For detection, it is desired that the maximum value is located at the period index which is corresponding to its fundamental value. Due to the cumulative property underlying these functions, however, the most dominant peak may not be distinguished clearly from its adjacent or harmonic values. In other words, it can be viewed that most data that are less relevant to the periodicity cause a negative influence in the period detection.

To reduce the influence of those data and emphasize the period-related data, we select a few samples, which are considered meaningful data for periodicity detection, rather than using all the data in calculating the detection functions. One simple way to distinguish meaningful samples is to choose parts that characterize the macroscopic shape of the data - the points where sharp changes occur, such as peaks and valleys. It is natural and empirical to focus on these characteristic points and find a similar shape to estimate the period of a quasi-periodic signal.

In addition to the effectiveness in the fundamental frequency detection, it is also efficient for the computation. Because the complexity of the ACF and the AMDF is proportional to  $N^2$ , using a few samples can dramatically reduce the computation time, particularly for huge input data.

We utilize peaks and valleys of the input data as the distinctive parts that imply periodicity for the two following reasons. Firstly, it is easy to distinguish inflection points from continuous data. Secondly, given that an audio signal

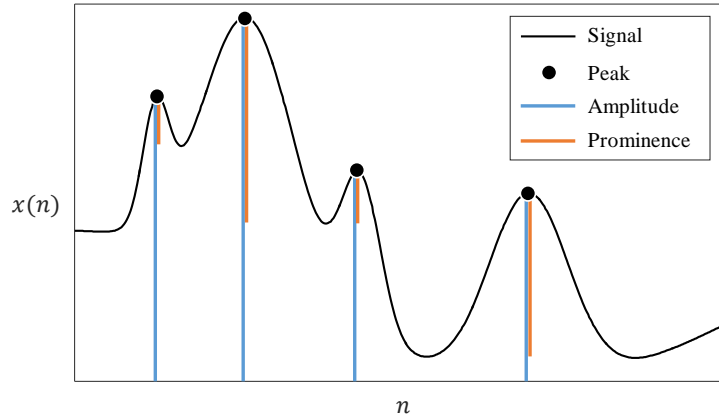


Fig. 3.1 An example of amplitude and prominence.

comprises of multiple sinusoid elements with different frequencies, each peak (or valley) represents the inflection point of more than one sinusoidal curve.

It is not necessary to sample as many peaks or valleys as possible. Rather, taking a few prominent samples into account would help to reduce the aperiodic influence. The prominence of a peak indicates the degree of how much the peak stands out relative to other peaks, which is defined by the height difference between the peak and its highest saddle point connecting it to its adjacent peaks. Figure 3.1 shows that an isolated peak with a low height could be more prominent than one that is higher.

The procedure for sampling data is as follows: Firstly, we normalize the given data to a range between 0 and 1. It is necessary to calculate the sampled ACF/DF through the basis function, which is explained in the next subsection. Secondly, peaks are selected in order of prominence. If valleys are used instead of peaks, the same procedures are conducted for the reversed-amplitude data.

By adjusting the minimum prominence condition, a trend in the performance of a pitch tracker can vary depending on the proportion of sampled data. The details are explained in Section 3.7.

### 3.3 Sampled ACF/DF in the time domain

Let  $n^+ = (n_1^+, n_2^+, \dots)$  denote a set of time for sampled peaks in a single frame of  $x(n)$ . The sampled auto-correlation function (ACF) and sampled difference function (DF) using only peaks are defined by summation of a basis function  $B$ ,

$$D^+(m) = \sum_{i,j} B(A_{ij}^+, \mu_{ij}, m) \quad (3.3)$$

for all  $i > j$ , where

$$A_{ij}^+ = \begin{cases} x(n_i^+)x(n_j^+) & \text{(sampled ACF)} \\ 1 - |x(n_i^+) - x(n_j^+)| & \text{(sampled DF)} \end{cases} \quad (3.4)$$

and  $\mu_{ij} = |n_i^* - n_j^*|$ . Similarly, we also define  $A_{ij}^-$  and  $D^-(m)$  by substituting  $x(n^+)$  with sampled valleys  $x(n^-)$ . Then, the sampled ACF/DF is derived with consideration of the periodicity by sampled peaks and sampled valleys concurrently as follows:

$$D(m) = D^+(m) + D^-(m) \quad (3.5)$$

The function  $B(A, \mu, m)$  is a basis function given in the form of the normal distribution  $Ae^{-b(m-\mu)^2}$ . It models a single peak with magnitude  $A$  and center point  $\mu$ .  $b$  is a positive constant which is related with the peak width. The summation form of the basis function enables the scattered sampled data to



be interpreted as a successive trend. It can be understood as a similar notion of expanding from a simple histogram to a weighted density function. Note that the equation means the histogram of the distances between two different time indices if the basis function is given by a delta function  $\delta(\mu_{ij})$ . Provided that all data are sampled and the delta basis function has a weight of  $A_{ij}$ , the equation will be identical to the original form of the autocorrelation and difference functions. Thus, sampled ACF/DF can be viewed as a restricted version of the original ACF/DF.

The weight of the basis function can be replaced with the product (or difference) of peak prominences at two different time indices rather than using the amplitude. In this case, the prominence is considered as the intrinsic amplitude. Figure 3.2 illustrates the original ACF/DF and the sampled form by using the amplitude and the prominence. Peaks become more salient in the sampled functions, and the first peak, which corresponds to the fundamental period is relatively significant among the other peaks.

Similar to the convention, the fundamental period is determined by choosing the global maximum in the weighted detection function, which is the sampled ACF/DF multiplied by an octave cost function. The octave cost function is a weight function that attenuates high-frequency candidates. For the direct comparison to the original ACF/DF, any additional refinements such as the normalization and the smoothing are not included.

### **3.4 Sampled ACF/DF in the frequency domain**

Utilizing the sampled ACF/DF in the frequency domain can be more useful than in the time domain because there are only a few spectral peaks in the spectrum

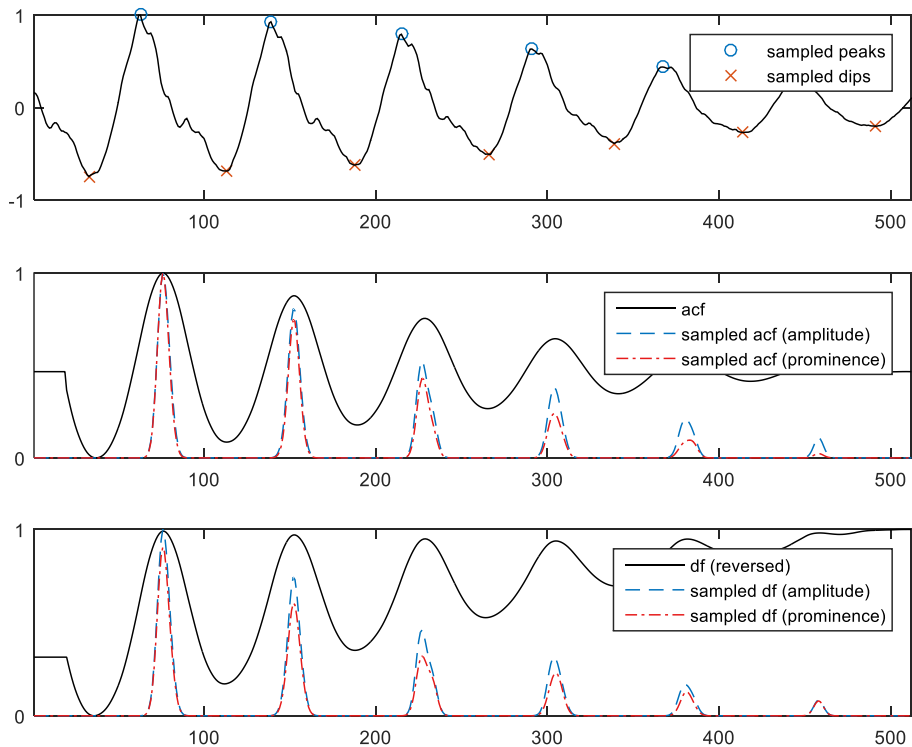


Fig. 3.2 An illustrative comparison between the original detection function and the sampled function by using data amplitude and prominence. (Top) an input signal. (Middle) autocorrelation functions. (Bottom) difference functions. The original difference function is reversely normalized for comparison.

magnitude and it can be simply done by sampling those peaks to recognize the periodicity. Since the magnitude of the spectrum of a monophonic music signal is periodic, the period of the frequency is estimated in the same manner as the time-domain case. This process is similar to the cepstral analysis in utilizing the period of the spectrum.

Unlike the time-domain approach, however, the length of the input data depends on the size of the FFT; not on the frame length. In the case of a time

series data, the greater the period, the lower the frequency. On the contrary, the fundamental frequency is directly proportional to the period of the spectrum magnitude, which could limit the frequency resolution, considering the fundamental frequency is concentrated in the low-frequency range in most cases. To solve this problem, we also propose an iterative F0 refinement algorithm based on the least square method. The details are described in the following subsection.

### 3.5 Iterative F0 estimation

With some given frequency bins of each spectral peak underlying the spectrum magnitude, this estimation technique enables the optimized interval between the spectral peaks to be reached. Given the locations of the selected peaks, F0 estimation can be substituted into the approximated greatest common divisor problem. Assuming that all harmonic partials are integer multiples of a particular integer, it is apparent that the greatest common integer divisor equals the fundamental frequency. However, it is not possible that all peaks are equally-spaced on the frequency bin axis, owing to the limited frequency resolution or inharmonic partials. Fortunately, a fundamental frequency candidate is roughly calculated through sampled ACF/DF. Therefore, the optimized interval between peak locations can be estimated by through the following steps.

Let  $x_n$  denote the frequency bin of the  $n$ -th spectral peak. By the assumption that all harmonic partials are integer multiples of the fundamental,  $x_n$  can be expressed by the linear sum of a harmonic partial and error term as

$$x_n = a_n x_0 + e_n \tag{3.6}$$

Note that both the harmonic order  $a_n$  and the fundamental frequency bin  $x_0$  are unknown. These are desired to be estimated subject to minimize the error. To this end, the cost function  $J$  is derived as a sum of squares of errors for each spectral peak by

$$J = \sum_{n=1}^N e_n^2 \quad (3.7)$$

$$= \sum_{n=1}^N (x_n - a_n x_0)^2. \quad (3.8)$$

To minimize this, we take its derivative and find  $x_0$  so that the derivative equals zero. By solving this, an equation for the fundamental frequency bin  $x_0$  is obtained as below:

$$x_0 = \frac{\sum_{n=1}^N a_n x_n}{\sum_{n=1}^N a_n^2} \quad (3.9)$$

Because we have only one observation of  $x_n$ , and both  $a_n$  and  $x_0$  are unknown, a numerical method is employed to estimate both unknown variables. As described in Algorithm 1,  $x_0$  is iteratively converged in sequence during the update and assignment steps.

Setting a proper initial value is essential to make  $x_0$  converge successfully within a few iterations. Using an initial value which is estimated through sampled ACF/DF prevents the iteration from converging incorrectly.

This refinement technique can be applied to general frequency-domain pitch detection algorithms. It would help to estimate the pitch more precisely with a very low computational complexity, especially when an insufficient frequency

---

**Algorithm 1** The iterative F0 refinement based on the least squares.

---

1: **Input:**

Partial frequencies  $f_1, \dots, f_N$  and their harmonic order  
 $a_1, \dots, a_N$ .

2: **Output:**

Refined fundamental frequency  $f_0$ .

3: **initialize**  $f_0$

4: **for**  $i = 1$  to  $N$  **do**

5:   **assignment step:** assign harmonic order,  $a_n = \left\lceil \frac{f_n}{f_0} \right\rceil$

6:   **update step:** calculate  $f_0$  using  $a_{1\dots i}, f_{1\dots i}$ ,  $f_0 = \frac{\sum_{j=1}^i a_j f_j}{\sum_{j=1}^i a_j^2}$

7: **end for**

8: **return**  $f_0$

---

resolution is given. Although this technique is also valid for time series data, it is not as effective compared to frequency-related data, because the reciprocal relationship in the time domain between the period and the frequency gives a sufficient resolution in the frequency range of interest.

### 3.6 Experimental setup

To evaluate the performance in pitch tracking by our sampling approach and many other methods, the MIR-1K dataset [34] is used. The dataset is widely used for various tasks in the music information retrieval field, including source separation and pitch tracking. It contains 1,000 audio recordings of the singing voice and the music accompaniment, and the total length of the dataset is 133 minutes. Each signal is recorded in separate channels, and only the singing voice signal in the right channel was used for the experiment with the original sample rate of 16 kHz. The pitch values in MIDI numbers, which are provided every 20

milliseconds, were used as the ground truth after converting into frequencies.<sup>1</sup>

A fixed frame length of 512 samples was equally used in the proposed method and the comparison group. In data sampling, we set the minimum peak prominence condition so that only peaks with prominence greater than 0.1 are sampled. The peak width constant  $b$  was set to 32. In the case of the frequency-domain sampled ACF/DF, a whitening process was employed that is described in [45].

Reference methods for pitch estimation and tracking include the seven following techniques which are publically available. Since the pitch values have been independently generated from different tools, the details of parameters need to be tuned separately as below:

- ac: This method performs the autocorrelation function of Boersma [46] in the Praat system. It is called with the command “To Pitch (ac)...0.02 93.75 15 no 0.0 0.0 0.01 0.0 0.0 800.” refers to applying 20ms of hop size and 512 samples of window size with no V/UV (voiced/unvoiced) cost.
- cc: This method implements a cross-correlation analysis [47]. It is also available in the Praat system, and called with the command “To Pitch (cc)... 0.02 93.75 15 no 0.0 0.0 0.01 0.0 0.0 0.0 800.” which performs the cross-correlation function using the same parameters as ac.
- shs: Also available in the Praat system, this method is based on spectral subharmonic summation [23]. It is called with the command: “To Pitch (shs)...0.02 93.75 4 1700 15 0.84 800 48.” as used in [21].
- SRH: The Summation of Residual Harmonics (SRH) method [48] is a

---

<sup>1</sup>The MIDI number is defined by  $12 \log_2 (\text{frequency}/440) + 69$ .

pitch tracker, focusing on the harmonicity of the residual signal. We used the SRH\_PitchTracking in the GLOAT Matlab package [49] and set the parameters as  $F0_{min}=94$ ,  $F0_{max}=800$ , and  $shift=20ms$ .

- HPS: The Harmonic Product Spectrum (HPS) method [22] is a pitch estimate algorithm which multiplies the original magnitude spectrum and its decimated spectra by an integer number. We used the implementation found on the website of “Audio Contents Analysis” [50] and used the same hop/window size as the SRH method.
- YIN: YIN [21] is one of the most well-known algorithms for pitch estimation. It is based on the autocorrelation method, considering the difference function instead, and it was further improved to reduce possible errors. We used the Matlab code freely available in [51] using the same  $minF0/maxF0$ , hop size as above, and 640 samples of window size.
- pYIN: pYIN (probabilistic YIN) [52] is a state-of-the-art method which is a modification of the YIN algorithm. It contains threshold parameters in a distributed form, in comparison with the single parameter in YIN. Also, a hidden Markov model (HMM) is Viterbi-decoded to find a smooth path through the fundamental frequency candidates. We used the plugin found in [53] with default parameter values except for the hop/window sizes, which were given the same values as other methods. The pitch values were annotated from the Sonic Annotator [54].

The assessment for all the above methods was performed using two traditional metrics: gross pitch error and fine pitch error [55]. Voicing activity decision, which is a common metric for the pitch tracking assessment, was not

used in our experiment, because the performance of voicing decision tends to vary depending on the parameter setting, and it has been revealed by a previous work [56] that it does not differ so much from different methods. We opted to focus on the pitch estimation accuracy since the aperiodic power, which is closely related to voicing decision, can be easily calculated by using F0. Instead, we define an additional error metric called “octave pitch error” to evaluate how robust the algorithm is, especially for the octave pitch error. Each of the three metrics is defined as follows:

- Gross Pitch Error (GPE): The proportion of frames that are labeled as voiced by ground truth, for which the relative pitch error is greater than 20%.
- Fine Pitch Error (FPE): The standard deviation of the relative pitch errors that are less than 20%, for frames that are labeled as voiced by ground truth.
- Octave Pitch Error (OPE): The proportion of frames with pitch errors greater than a semitone and pitch chroma errors less than a semitone.

For the last evaluation, the pitch tracking error rate is examined depending on the proportion of sampled data. A frame-level factor called the data sampling ratio is primarily defined by the ratio of the number of sampled data points and the number of all data points in a single frame. Then, for each data sampling ratio, we calculated the gross error rate, which is the proportion of frames that are decided as gross errors. Note that this metric is a function of data sampling ratio, which implies that the optimal proportion of sampling for the highest accuracy can be derived.



### 3.7 Result

The pitch tracking performance for various methods in the time domain is displayed in Table 3.1. The proposed methods are placed in the first four rows, denoting the sampled autocorrelation function as sACF and the sampled difference function as sDF. Through the comparison of the three error metrics between the sACF and the ac, it is observed that the sACF with prominence achieved the significant improvement compared to the original autocorrelation, in the aspect of the overall performance and the octave pitch error. This result supports the assumption that using a few meaningful data is more effective for the periodicity analysis rather than using the entire data. We also noticed that the peak prominence, which is considered the intrinsic amplitude, is more suitable than the peak amplitude for the fundamental period detection. The difference function, which is commonly regarded as a good alternative to the autocorrelation function, showed a slightly better performance compared to the autocorrelation function, when the peak amplitude is sampled. However, it is observed that the improvement is not as much as the sACF with promi-

Table 3.1 Pitch tracking errors comparison over the time-domain methods.

Method	GPE (%)	FPE (cents)	OPE (%)
sACF, prominence	1.8	23.3	1.1
sACF, amplitude	2.8	21.1	1.6
sDF, prominence	2.2	21.3	1.3
sDF, amplitude	2.4	21.2	1.4
ac	3.0	26.3	1.8
cc	4.0	32.7	2.0
YIN	1.1	22.9	0.5
pYIN	1.0	49.0	0.6

Table 3.2 Pitch tracking errors comparison over the frequency-domain methods.

Method	GPE (%)	FPE (cents)	OPE (%)
sACF	5.7	48.0	1.6
sACF (F0 refined)	4.8	27.3	1.8
sDF	11.6	45.2	5.2
sDF (F0 refined)	10.7	26.3	4.6
shs	4.5	27.9	0.7
SRH	2.8	34.6	1.3
HPS	24.3	24.4	5.8

nence, inferring that the prominence is more effective for autocorrelation-based approaches.

The YIN and the pYIN are employed as the reference method using the difference function. As a consequence, both of the YIN methods outperform among all methods including the frequency domain approaches. Those methods contain several refinement steps because those are regarded as the essential parts of the algorithm. Although it has been revealed in many related works that the refinement steps improve the overall performance, as we mentioned in Section 3.2, we did not utilize any refinements for the sACF and the sDF, except an octave cost function.

Table 3.2 describes the pitch tracking errors over the frequency-domain methods. The relatively poor result compared to the time-domain sACF and sDF shows that using the difference function upon the magnitude spectrum is not so much effective as the time-series data. That is because the peak level of each harmonic partial tends to decrease as the frequency goes toward higher harmonic orders, thus it is less periodic than the time-domain amplitude. The fine pitch error in the sACF and the sDF decreased significantly as the iterative F0 refinement method is used. It turns out that the proposed method enables

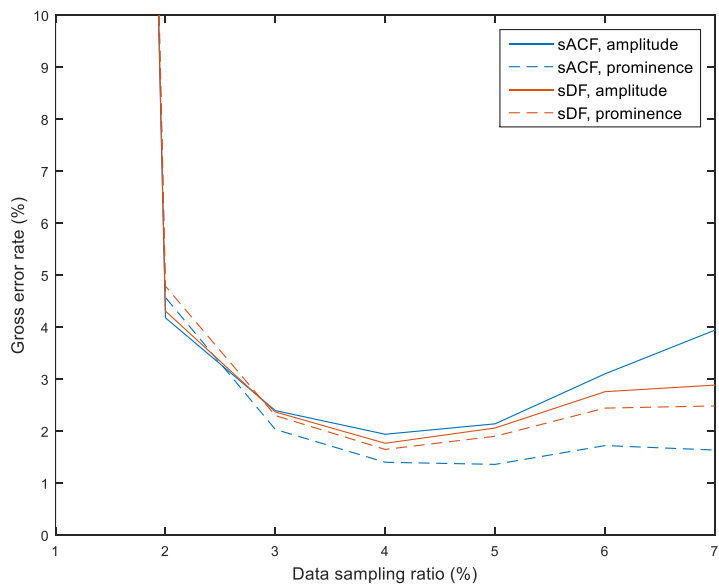


Fig. 3.3 Gross error rates per data sampling ratio.

to estimate the pitch more precisely. Meanwhile, HPS was not operated very stable in our experiment. It is assumed that this relatively poor result is caused by the limited audio quality of the input data, which does not give the sufficient number of higher-order harmonic partials.

The actual computation time for each method is not stated in this chapter, because we conducted the experiment for some methods in the different environment. Given the 1,000 audio files in the MIR-1k dataset, we measured the computation time of the Matlab based implementation. The sACF and the sDF took about an hour respectively, while the original autocorrelation and the difference function took over 6 hours per method.

The trend in the gross error rate per data sampling ratio is illustrated in Fig. 3.3, for the time-domain sACF and the sDF. When the sampling ratio is

less than 4%, data is sampled too sparsely to analyze the periodicity. When more than 5% of the entire data are sampled, interestingly, it appears that the impact on the gross error is growing due to the oversampling. The reliability of the gross error decreases as the data sampling ratio increases, because the proportion of sampling is limited as long as only peaks and valleys are sampled. Despite that, this result indicates that using the data as many as possible does not lead to the best performance.

### **3.8 Summary**

In this chapter, a data sampling method was presented for the estimation of the fundamental period of quasi-periodic signals. As dealing with the input data directly, this method is able to be utilized in the time and the frequency domains, and can be adapted in many conventional algorithms. The data sampling method is not only efficient for computation but also effective to analyze the periodicity. When evaluated over a public dataset which contains plenty of the singing voice recordings, the sampling method improved the pitch tracking performance compared to the original detection functions without sampling. An iterative F0 refinement technique was also presented, and the improvement in precise F0 tracking was proved in the experiment.

## Chapter 4

# Note Onset Detection based on Harmonic Cepstrum Regularity

### 4.1 Introduction

This chapter presents an onset detection algorithm which is an important part of the transcription system proposed in this thesis. The main goal of this study is focused on the detection of soft onsets that have not been clearly detected in existing onset detection algorithms. This study is the first attempt based on the idea on the harmonic structure, a key hypothesis of this thesis. This chapter is based on the research published in the proceedings of the IEEE International Conference on Multimedia and Expo [32].

For a couple of decades, onset detection of musical notes has been a major issue in the music information retrieval community. Being a fundamental low-level task in this field, accurate note onset detection can lead to solving many problems for advanced music analyses, including pitch estimation, tempo esti-

mation, automatic transcription, and many commercial applications of music and audio processing.

Masri proposed a well-known algorithm for pitched non-percussive (PNP) onset detection applying linear weight on high frequency content (HFC) [25]. Klapuri's sub-band energy change method used a filter bank model to approximate the human cochlea [26]. A more general approach is introduced by Duxbury, where the changes in the spectrum called spectral difference or spectral flux, are used to indicate musical onsets [27].

With all the contributions made so far, however, according to the Music Information Retrieval Evaluation eXchange (MIREX) 2012, onset detection still remains a challenging problem, particularly for soft onsets. The best result for onset detection of solo singing voice is an F-measure of 55.9%. In the case of solo sustained strings, averaged F-measure for all participants is only 52.8%. Since most of the traditional approaches use spectral energy and its difference via time-frequency analysis, detection function must be solely affected by the original source. A soft onset generally does not rapidly occur because it has a long attack interval or indistinguishable envelope shape, and becomes the main reason of difficulties in the peak-picking procedure.

Supervised machine learning-based approaches are also proposed to handle this problem. Toh et al. viewed onset detection as a classification problem, and used two Gaussian mixture models (GMMs) to classify audio features into onset and non-onset frames [57]. Also, they derived a fusion of four different types of acoustic features, including mel-frequency cepstral coefficients (MFCCs), linear predictive cepstral coefficients (LPCCs), and equal loudness phon values along critical bands. However, as in other machine learning algorithms, this approach

is supposed to rely heavily on the training data. Furthermore, it is extremely time-consuming and laborious to annotate a large amount of audio data at the frame level. Finally, MFCCs, which is the main feature of learning-based approaches, are hard to represent all kinds of characteristics of various musical instruments.

To resolve these issues on detecting soft onsets, we propose a novel approach based on harmonic cepstrum regularity (HCR), by focusing on the changes in the harmonic structure. HCR is expected to yield better results for sustained string instruments and singing voice which usually contains soft onsets and unexpected energy flow. Especially, it is robust to irregular changes in the formant structure caused by different singing styles. In addition, unlike spectral difference or spectral flux, HCR gives a steady detection function regardless of note strengths since only harmonic components in the cepstral domain are considered. Overview of the proposed system is illustrated in Fig. 4.1.

The rest of this chapter is organized as follows. In Section 2 we explain the reason why we applied cepstral analysis to this task. Section 3 describes the four main procedures to locate positive onsets, including harmonic quefrency selection, sub-harmonic regularity function, adaptive thresholding, and peak-picking. In Section 4, the experiments we performed to evaluate the improvement of the proposed algorithm are presented, and we finally draw our conclusions and present directions for future works in Section 5.

## 4.2 Cepstral analysis

Cepstrum is originally defined as the inverse Fourier transform of the log-magnitude Fourier spectrum. When a magnitude spectrum has a strong

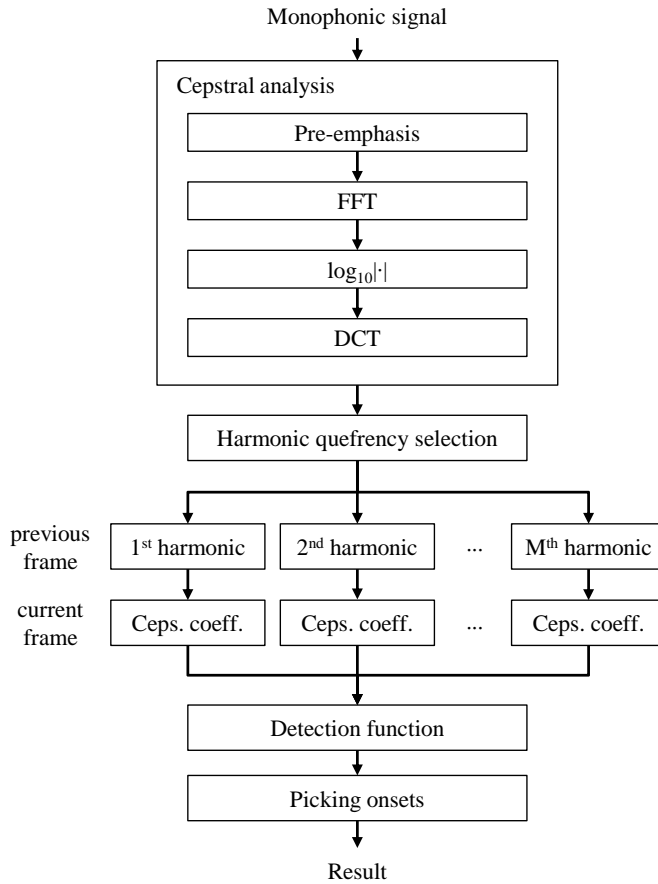


Fig. 4.1 Architecture of the proposed onset detection system.

periodicity—i.e. if a regular frequency interval is found between harmonic partials—cepstrum appears strong. On the other hand, weak cepstrum is generally obtained for noisy audio signals. The conversion to the logarithmic scale aims to adjust the dynamic range of the audio energy, so that the cepstrum simulates the way in which the human auditory system perceive the loudness.

In the field of image and audio processing, discrete cosine transform (DCT) is more preferred than inverse Fourier transform, because its “energy com-



action” property enables not to lose much information of the signal, while most of information are concentrated in the low-frequency range of the DCT. Cepstral coefficients are determined as the results of the DCT which is the last step of cepstral analysis. These are widely used as a feature to represent timbral characteristics in many audio processing tasks, including speech recognition, speaker identification, and genre classification.

It is well known that pitch is proportional to frequency. More specifically, the relation between pitch and frequency is logarithmic, not linear. When pitch increases by an octave, frequency would be doubled. Mel scale reflects this characteristic of the human auditory system. MFCC is derived by first taking the Fourier transform of a windowed signal, mapping the log-amplitudes of the resulting spectrum into the mel scale, then performing DCT on the mel log-amplitudes. The amplitude of the resultant cepstrum become the MFCC [58]. Similarly, LPCC can be derived as linear predictive coding coefficients transformed into cepstra.

Almost every algorithm using the MFCC generally truncates higher coefficients. In many cases, only the first 13 coefficients are said to be enough to store the signal characteristic. However, we use a sufficient number of cepstral coefficients which are the same size as the frame length, because we would need higher quefrequency resolution to get the coefficients corresponding to harmonic components more precisely. In the following experiments we conducted, we use a Hamming window of 2,048 length, and therefore the cepstral coefficients with the same length are obtained for every analysis frame. Using a 44.1 kHz sampling rate and an 87.5% overlap, we get a frame rate of five milliseconds which is short enough to detect the shortest possible musical note.

Another difference between the classic MFCC and our cepstral analysis is that we use a linear scale in frequency rather than the mel scale. Mel-scaling compensates differences between frequency and subjective pitch. This psychoacoustic knowledge definitely makes some advantages when human perception (i.e. timbre, masking, etc.) is an important issue, but in this situation we do not need to consider this because we are only interested in whether the harmonic structure is noticeable or not. It is known that the  $n$ -th order harmonic frequency for three types of musical instruments is derived as

$$f_n = \begin{cases} nf_1, & \text{open tube \& string instruments} \\ (2n - 1) f_1, & \text{closed tube instruments} \end{cases} \quad (4.1)$$

where  $f_1$  means the fundamental frequency [59]. For singing voice, a very complex calculation is required to derive the exact harmonic frequency but we can simplify vocal tract as a closed-tube type instrument. While harmonic frequencies are linearly related to the order, any scaling along frequency bands is not necessary.

We also take a pre-emphasis step by sending the input signal through a highpass filter. This process is to compensate the high-frequency part and emphasize the high-order cepstral coefficients that we want to concentrate on. Pre-emphasized signal  $s'_n$  is derived as

$$s'_n = s_n - \alpha s_{n-1} \quad (4.2)$$

where  $s_n$  is the original signal and the value of  $\alpha$  is usually defined between 0.9 and unity, and we set this value to 0.97.

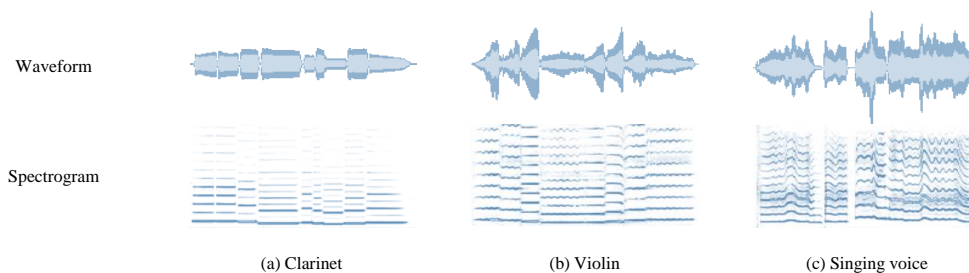


Fig. 4.2 Waveform and spectrogram of a clarinet, a violin, and a singing voice signal.

### 4.3 Harmonic cepstrum regularity

Figure 4.2 shows the waveform and spectrogram of three musical signals: a clarinet, a violin, and a singing voice signal. It is not difficult to recognize the onsets from the spectrogram simply by our intuition, because we tend to pay visual attention to several imaginary vertical lines. These are easily distinguishable due to the discontinuity of many horizontal lines, which indicate the energy of the harmonic components. In order to determine how regularly this harmonic structure of the input signals is maintained, we need to examine the amount of temporal change of the harmonic components.

To this end, we first extract the harmonic quefrequencies from their cepstral coefficients, and then check how much the energy changes in the selected quefrequencies. An important point is that the harmonic structure of the previous frame is applied to the current frame. In other words, cepstral coefficients of the previous harmonic quefrequencies are selected to build the detection function of the current frame. The cepstral coefficients of the current frame would not be different much from those of the previous frame if the harmonic structure is stable, and thus the peak locations would remain the same. On the other

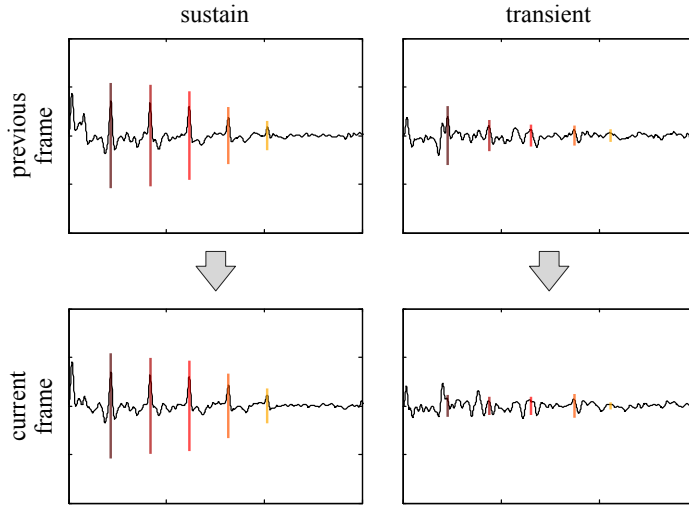


Fig. 4.3 Comparison between a sustain and a transient. Each vertical line represents its relative amplitude of cepstral coefficient.

hand, if the harmonic structure is changing rapidly, these peak locations will also change, resulting in smaller amplitudes for the coefficients found in the previous frame. As shown in Fig. 4.3, this causes the pronounced difference between a sustain and a transient.

### 4.3.1 Harmonic quefrency selection

Harmonic quefrency selection is an essential procedure to make the results reliable. Pitch estimation can be regarded as an advanced concept of this process, in a sense, so we can also extend this algorithm not only to note onset detection but also to monophonic music transcription while as long as the harmonic quefrencies are selected correctly. Therefore, conventional pitch estimators can be used to find the exact harmonic quefrencies. For example, YIN is a well-known pitch estimator [21] and the correntropy method is known to give the best result for singing voice [43]. The relationship between the fundamental frequency  $f_1$

and its corresponding quefrequency  $q_1$  follows as below:

$$q_1 = \frac{2f_s}{f_1} \quad (4.3)$$

In this chapter, however, we simply take an autocorrelation method to pick the harmonic peaks because pitch estimation is not our main goal and the autocorrelation function (ACF) is also enough to yield the reasonable result. One-dimensional Gaussian kernel function is applied to make peaks more salient, and the kernel size is set to 64 via experiments. Then we compute the ACF of the cepstral coefficient and derive the fundamental quefrequency  $q_1$  from the index of the maximum ACF value. For high order harmonics, we approximated their possible ranges based on the integer harmonics assumption, and the local maxima are chosen within the possible ranges.

### 4.3.2 Sub-harmonic regularity function

As a detection function, the harmonic cepstrum regularity function (HCR) is derived by the summation of all harmonic cepstral coefficients, where harmonic quefrequencies represent the harmonic structure of the previous frame. That is,

$$d_n = \sum_{k=1}^M C_{q_{k,n}, n-1} \quad (4.4)$$

where  $C$  is the cepstral coefficient matrix whose rows and columns represent quefrequency and the frame index, respectively.  $k$  is the harmonic order up to  $M$  (normally  $M = 5$ ), which depends on the instrument type and the degree of pre-emphasis. In general, the more harmonic components were used, the better result we would get as the detection function would fully describe the harmonic structure.

### 4.3.3 Adaptive thresholding

We now describe the adaptive thresholding procedure, which picks local minima of the detection function  $d_n$ . Since our HCR function represents how regular the harmonic structure is and this regularity is disrupted when onsets occur, local minima are supposed to be picked instead of maxima. There are several adaptive thresholding methods such as low-pass FIR filtering, median filtering, and low-pass filtering of the square of the detection function. We choose a method based on the local median because it is known to be robust by minimizing the effects caused by the outliers [14]. In this chapter, adaptive threshold  $\delta_n$  is determined not to miss local minima as

$$\delta_n = \delta + \text{median}\{d_{n-T}, \dots, d_{n+T}\} \quad (4.5)$$

where  $T$ , the size of the median filter, is set to 40 in our experiments.

A fixed thresholding is also used to detect silences of the input signal.  $\delta_c$  is a constant value separating silence from non-silence frames, and it is relevant to the signal-to-noise ratio of the signal. For general recordings, 20 percent of the maximum of  $d_n$  will be proper.

### 4.3.4 Picking onsets

Before picking onsets, a post-processing is performed to discard multiple false positive onsets adjacent to a true positive onset. Since the shortest note in our experiment data is longer than 15 ms, we dismiss multiple onsets whose duration are shorter than this interval.

Unlike many other approaches where the peaks in the detection function directly indicate onsets, we first compute the ‘transient sections,’ which means

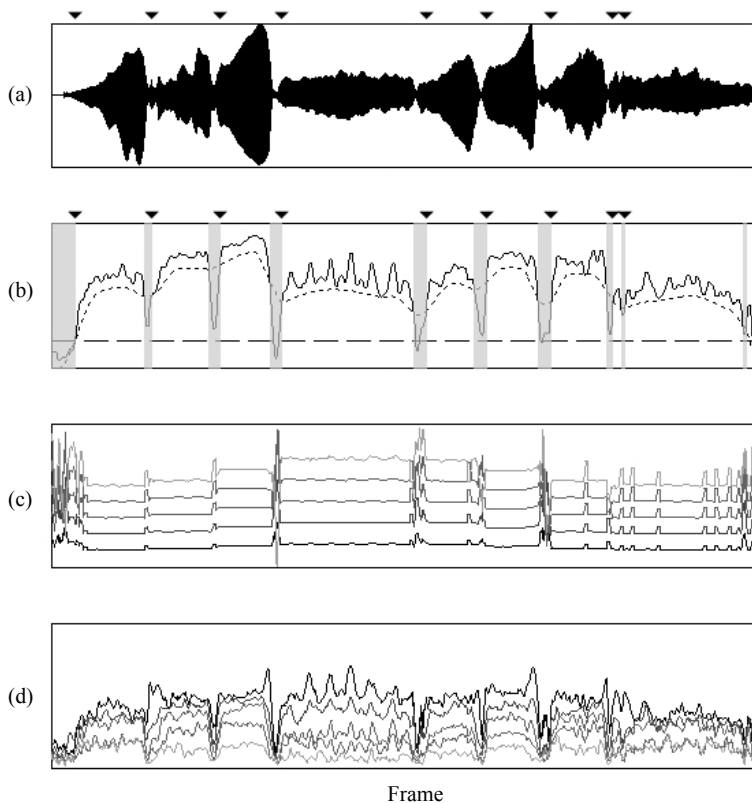


Fig. 4.4 (a) Waveform of a violin signal. (b) Detection function and adaptive threshold. (c) Five harmonic quefrequencies. (d) Five sub-harmonic cepstral coefficients.

the interval between different notes. Frames where the detection function  $d_n$  is greater than the adaptive threshold  $\delta_n$  or less than the fixed threshold  $\delta_c$  are classified as the transient section. Then, positive onsets are defined at the end of each transient section. Offsets can also be simply found (if needed) at the beginning of each transient section, as long as transient sections are well-defined.

Figure 4.4 shows the graphical summary of procedures described in this

section. An excerpt from a solo violin performance of “Ach Gott und Herr” from Bach10 database<sup>1</sup> [36] was used in this figure. Triangle markers in the first two plots indicate the ground-truth onsets and detected onsets, respectively. In the plot (b), detected onsets are located at the end of each transient section which is depicted as a gray-shaded area. The solid line indicates a detection function  $d_n$ , dotted line an adaptive threshold, and dashed line a fixed threshold, respectively. The detection function in (b) is obtained by summing across five sub-harmonic cepstral coefficients which are shown in the plot (d). We can observe in the plot (b) that the detection function is mostly stable within a note regardless of the waveform amplitude, the only exception being the fluctuation in the middle of the input signal due to the vibrato of the violin. In the plot (c) are illustrated five harmonic quefrequencies that correspond to five cepstral coefficients shown in (d).

## 4.4 Experiments

### 4.4.1 Dataset description

The experiments were performed on Bach10 database [36] and both commercial and non-commercial singing voice recordings, which contain 3,474 onsets in total. The Bach10 database is accompanied by the ground-truth onsets. For singing voice, 13 male and two female recordings are used, which contain more than 1,500 onsets. Onset labeling for singing voice recordings was cross-validated by three professional musicians. Ambiguous musical articulations such as glissando and non-pitched notes were excluded in the experiments. All data were preprocessed to be monaural signals sampled at 44.1 kHz. The detailed

---

<sup>1</sup>Details of the database can be found online at <http://music.cs.northwestern.edu/data/Bach10.html>.



information of the data set is reported in Table 4.1.

Table 4.1 Dataset details.

Instrument	Reference	Duration	Onsets
<i>Sustained string</i>			
Violin <sup>†</sup>	Bach10 dataset	5m 34s	425
<i>Woodwind</i>			
Saxophone <sup>*</sup>	Bach10 dataset	5m 34s	500
Clarinet <sup>*</sup>	Bach10 dataset	5m 34s	475
Bassoon <sup>*</sup>	Bach10 dataset	5m 34s	507
<i>Singing voice</i>			
Male <sup>†</sup>	Professional singers	11m 46s	1,533
Female <sup>†</sup>	Amateur singers	24s	34

<sup>†</sup> soft onset class; <sup>\*</sup> hard onset class

#### 4.4.2 Evaluation results

Like many other methods, we regarded an onset to be correctly detected (*CD*) if the ground-truth and the detected onset are within a 50-ms interval. Because of inaccuracy found on the annotation of the Bach10 database, a 70-ms tolerance window was used instead for some clips. We applied the same tolerance to all the comparison groups. If a detected onset is not within this interval, it is regarded as a false positive (*FP*). If a ground-truth onset is not detected (i.e. missing onset), there is a false negative (*FN*). Precision (*P*), Recall (*R*), and the F-measure (*F*) are used to evaluate the performance. These measures are defined as follows:

Table 4.2 Performance of the proposed algorithm.

Instrument	Onsets	Precision	Recall	F-measure
Violin	425	.874	.949	.910
Saxophone	500	.934	.960	.947
Clarinet	475	.878	.933	.904
Bassoon	507	.809	.829	.819
Singing voice (male)	1,533	.713	.737	.725
Singing voice (female)	34	.784	.829	.806
Total	3,474	.802	.836	.819

$$P = \frac{CD}{CD + FP} \quad (4.6)$$

$$R = \frac{CD}{CD + FN} \quad (4.7)$$

$$F = \frac{2PR}{P + R} \quad (4.8)$$

The overall results of our algorithm are summarized in Table 4.2. We can see that there is no large difference in performances between soft and hard onsets. Particularly for singing voice, although not directly comparable, F-measure was significantly improved by about 30% over the best performing algorithm of the MIREX 2012.

For comparison with other approaches, we implemented several algorithms aforementioned in Section 4.1 [25, 27] plus energy-based method, which was first introduced by Schloss [15]. Klapuri’s psychoacoustic knowledge-based approach [26] was implemented based on MIR Toolbox 1.4 [60]. Parameters for adaptive thresholding and peak-picking were fixed to the same values in every experiment.

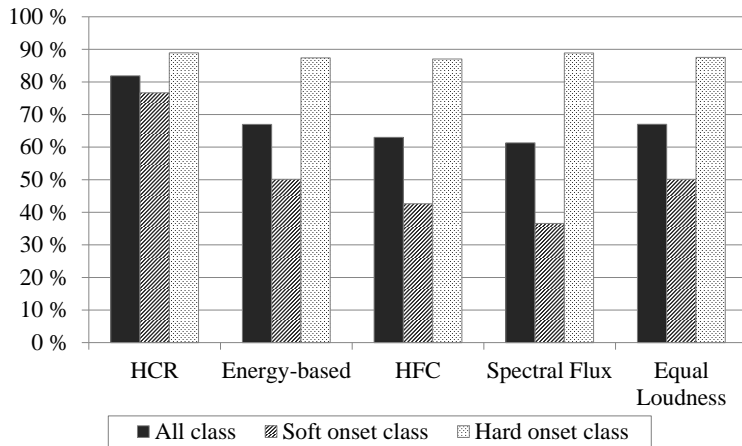


Fig. 4.5 F-measure comparison for different classes of onset.

We classified all recordings into the soft onset class and the hard onset class by the instrument type. Violin and singing voice are categorized into the typical soft onset class. All other instruments were classified as the hard onset class. As depicted in Fig. 4.5, HCR shows the remarkable improvement for the soft onset class. While an F-measure of other algorithms is below 50%, HCR achieves an F-measure of 76.7%. Considering that every algorithm yields a good performance for the hard onset class, it is obvious that the performance for the soft onset class makes the overall improvement.

## 4.5 Summary

In this chapter, we have proposed an automatic note onset detection algorithm for pitched instruments including singing voice signals. The presented algorithm is simple and yet achieves a significant improvement, especially for soft

onsets. Using the cepstral analysis, sub-harmonic regularity functions were derived from the changes in harmonic cepstral coefficients. The experiments were performed on over about 3,500 onsets on multi-instrument recordings from the Bach10 database and 15 singing voice recordings. The results showed that the proposed algorithm not only achieved performance comparable to other conventional algorithms for hard onsets but also outperformed significantly for soft onsets.

Since the proposed algorithm is able to locate the transient sections whose beginning and end position indicate note offset/onset, respectively, and also to find the fundamental quefrequency which is related to pitch, it has a potential to apply to an integrated automatic music transcription system. Future research will cover offset detection at the beginning of the transient section we already obtained. We also plan to extend it for the polyphonic music transcription.

## Chapter 5

# Robust Singing Transcription System using Local Homogeneity in the Harmonic Structure

### 5.1 Introduction

In this chapter, we present a robust singing transcription system based on the idea using the harmonic structure, which was roughly attempted in Chapter 4. While our previous approach indirectly measured the degree in which the harmonic structure was temporally maintained through cepstral analysis, this approach differs in that it accurately tracks each partial, and measures the local homogeneity in the harmonic structure. Based on our experience with pitch tracking and onset detection studies, this study covers a fully automatic singing transcription system from audio input to symbolic output. The latest F0 tracking method is used for harmonic structure analysis, and new acoustic features are proposed and various probabilistic techniques were applied for note detec-

tion. This chapter is based on the research published in IEICE Transactions on Information and Systems [61].

In most related studies, a musical note is defined by three components: onset, duration, and pitch. Since the late 1990s, many approaches have been proposed to detect the onset time, defined by the exact time when a note starts [14]. In general, onsets can be categorized as hard and soft onsets depending on the attack time, which is the time taken for initial run-up of the amplitude envelope. Soft onsets that commonly appear in singing voices or sustained string instruments such as the violin, are usually more difficult to detect, because the changes in acoustic features such as the energy envelope are very gradual and insignificant. Duration refers to the time for which the note is played; therefore, it is equal to the offset time minus the onset time of a note. Pitch is a quantitative value representing how high or low a sound is. Pitch detection algorithms estimate a sequence of successive pitch values at the frame level, which are typically defined by the fundamental frequency (F0 henceforth) in Hz or are given by MIDI note numbers. For monophonic music signals, the accuracy of pitch estimation algorithms has already reached a high level. One of the most popular pitch trackers called YIN [21] achieved an average gross error rate of 1.03%, which is still competitive today. When the input signal is a human voice such as speech or singing, it can be more reliable by using the bone-conducted signal [62].

Although the human voice is a type of musical instrument to “perform” in the easiest way, automatic transcription for the singing voice still needs improvement. According to the Music Information Retrieval Evaluation eXchange (MIREX), the F-measures in the singing voice onset detection for the last five

years have been around 0.6, which is 30% less than the results of other solo instruments. Compared to general solo instruments, some difficulties in note detection are commonly found in singing voice signals. Note events often arise in very unpredictable ways, and it is difficult to define a single acoustic pattern. From various singing voice signals, it is observed that this unpredictability is mostly caused by two factors: loudness inconsistency and spectral heterogeneity. In singing, the dynamic range of loudness is not stable; rather, it varies among singers and their singing styles. In addition, the spectral distribution in singing depends on the pronunciation, whereas other instruments have their own timbral characteristics.

Despite all these difficulties, singing voice signals have a clear benefit for transcription. F0 estimation for the singing voice has reached a reliable level because it is basically monophonic. A precise tracking of the F0 sequence can give useful information to identify not only the pitch but also important temporal attributes such as the onset and offset. Since McNab introduced a simple segmentation method for singing transcription using the pitch and amplitude [29], many approaches have been based mostly on the discontinuity in the F0 sequence. An auditory-model-based method uses the pitch continuity, together with the loudness and voicing patterns [63]. Rynnänen combined two probabilistic models to detect natural notes in a musicological sense [64]. More recently, Gómez and Bonada proposed an iterative note-consolidation technique using low-level features related to the pitch, duration, voicing and stability [35]. Molina presented a note segmentation method based on pitch-time hysteresis, making use of the dynamic average of the pitch curve [65].

However, the pitch-based approach has a problem that it cannot detect

smoothly continued notes with the same pitch. Frequently observed in singing and humming, these notes can be detected using instantaneous changes in other acoustic properties. In this respect, this study begins with a hypothesis that both the beginning and the end of a note are recognized by the local homogeneity in the harmonic structure. The basic idea of using temporal changes in the harmonic structure was first attempted by using the regularity of the harmonic-related cepstrum [32]. We extend a similar approach here to make full use of the harmonic structure as an important cue for detection of note boundaries. The goal of this work is to propose a comprehensive transcription system that converts a singing voice recording into a western music score. The proposed system is presented in a unified framework, which includes extraction of a novel acoustic feature reflecting the harmonic structure, a probabilistic model for classifying the intrinsic harmonic structure, and transcription schemes for identifying the musical attributes.

In the proposed system, a *stream* is defined by a region with continuous voiced F0s, which is divided by unvoiced frames. A stream may contain several notes smoothly continued, or may consist of only one note. There are two strategic benefits when the transcription process is allocated for each stream. It enables an efficient mixture model (described in Section 5.4) as it does not necessarily consider the whole range of an input audio. In addition, the system can be composed in a clear and unified framework because it does not need any exceptional treatments for unvoiced regions. The overall workflow of the entire proposed system is shown in Fig. 5.1.

The rest of this chapter is organized as follows. Section 5.2 explains a front-end stage for F0 tracking, and Section 5.3 describes the extraction of an acoustic



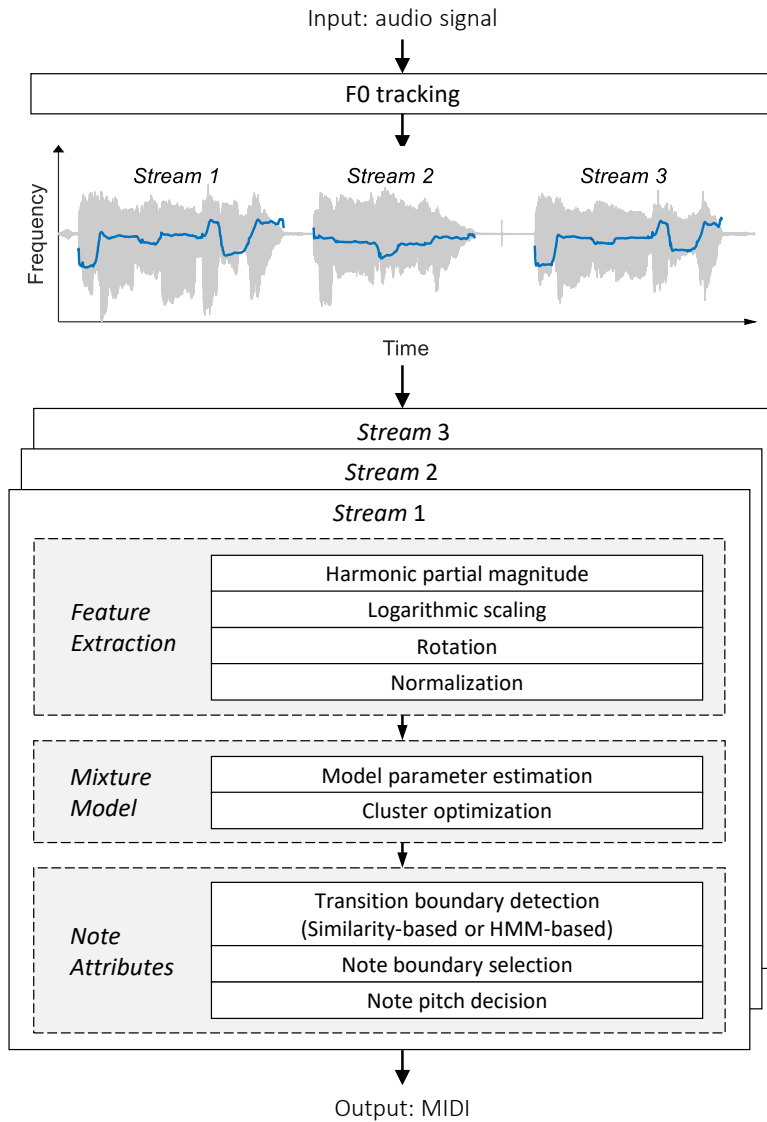


Fig. 5.1 Schematic flow underlying the proposed transcription system.

feature to signify the harmonic structure. A probabilistic model to characterize the feature is also presented in Section 5.4, followed by a transcription of note attributes such as the onset, offset, and note pitch in Section 5.5. Section 5.6 presents the evaluation methodology to assess the proposed system, and the experimental results including the comparison with other systems are shown in Section 5.7. Finally, the conclusions of this chapter are drawn in Section 5.8.

## 5.2 F0 tracking

Before the local homogeneity in the harmonic structure is directly mentioned, a precise F0 tracking should precede it to identify the harmonic partials. In this work, it is implemented by a robust pitch tracker called pYIN [52]. This algorithm is chosen as a front-end F0 tracker of the entire transcription system due to its strength against “octave errors,” which means that estimates are sometimes doubled (or half) frequencies. In order to enhance the original YIN algorithm, pYIN selects a few F0 candidates by taking valleys in the difference function of the input signal. After that, the probability of each candidate is calculated by observations in a hidden Markov model (HMM) for temporal smoothing of the F0 track, which is determined by the optimal path of pitch state decoded by the Viterbi algorithm.

The pitch space was defined from 65 Hz (C2) to 830 Hz (G#5) to cover the vocal pitch range of non-professional singers. It was divided in a step of 1/4 semi-tones, yielding 140 voiced pitch states in total. The same number of unvoiced pitch states were concatenated with these voiced pitch states to construct the HMM. In the tracking result, some frames could be labeled as unvoiced if their corresponding path indicated an unvoiced state (weak probabilities of F0 candi-

dates) or if the root-mean-square value was less than 0.1 (weak signal energy). Observation probabilities were calculated using a parameter prior modeled by the beta distribution with means 0.25, which is slightly greater than the parameter configuration that the original authors used. This is because the priority of the proposed system is a high recall, which means it aims to estimate as many frames as possible of the voiced F0s.

### 5.3 Feature extraction

Extraction of an acoustic feature that reflects the harmonic structure begins with the magnitude of the harmonic partials. The use of harmonic partials has been introduced in many previous works for different tasks, such as music source separation [66] and vocal activity recognition [67]. In this work, we focus on the point that the relative ratio between the harmonic energies remains constant, regardless of the external factors including the pitch and loudness. The feature extraction process consists of the two following steps: (1) Extraction of harmonic partial magnitudes and (2) Vector transformation such as scaling, rotation, and normalization.

The first step of feature extraction is a time-frequency representation of an input signal using the short-time Fourier transform. The input signal is downsampled to 22.05 kHz for a better computation time, and a Blackman window of 32 ms is used to split the signal into frames. Only the magnitude spectrum is considered, and the phase information ignored.

Instead of taking the magnitude at particular harmonic frequency bins, the tracking of harmonic partials is realized by a dynamic filter bank, whose frequency response is dynamically characterized by the estimated F0. The used

filter bank is a series of overlapping triangular band-pass filters, so that the center frequency of one filter is equal to the lower boundary of the next filter. The center frequency of each filter is obtained from the multiple integers of the estimated F0. All the filters show a maximum response of unity at their center frequency.

The use of a filter bank offers advantages in two aspects. First, it compensates the errors arising from insufficient frequency resolution. Some algorithms [34, 68] use a multi-resolution FFT to enhance both the time and frequency resolution. However, a recent study has shown there is no significant benefit in locating the spectral peak frequency [69]. Second, frequencies slightly deviating from the exact integer multiples of the F0 can be considered. Inharmonic partials are rarely ever observed in cases of singing, but the spectral peak width can be relatively wide when the pitch is sharply changing within a frame.

The harmonic partial magnitude is not refined enough to be used as a feature vector for the harmonic structure in two respects: energy dynamics and imbalance in dimensions. The deviation in energy is too large to be characterized, and most of the spectral energy is concentrated in the first few harmonic partials. Therefore, the harmonic partial magnitude is transformed into a more refined form of a feature called the Harmonic Structure Coefficient (HSC), by the three following steps of scaling, rotation and normalization.

Let a column vector  $\mathbf{u} = [u_1 \ u_2 \ \dots \ u_h]^T$  denotes the magnitude for up to the  $h$ -th harmonic partial at a time instance. The logarithmic scaling

$$\mathbf{x} = \log_{10}(\mathbf{u} + 1) \tag{5.1}$$

converts the magnitudes into non-negative values in a limited range, thereby

making the data more stable for abrupt events. One example of this is the mel-scale filterbank cepstral coefficient (MFCC), which is the most popular acoustic feature that represents the timbral texture.

Log-scaled magnitudes are then rotated in such a way that the eigenvector with the minimum eigenvalue is parallel to the mean vector of a stream. This vector rotation allows the data distribution to be grouped easily when it is projected onto a unit hypersphere. Given a sequence of log-magnitude vectors  $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_N]$  with a stream length of  $N$ , its distribution can be expressed by the mean vector  $\mu_{\mathbf{x}}$  and the covariance matrix  $\mathbf{C}_{\mathbf{x}} = \text{Cov}(\mathbf{X})$ , reflecting the center point and the spreadness in the  $h$ -dimensional Euclidean space, respectively. Since  $\mathbf{C}_{\mathbf{x}}$  is a  $h \times h$  square matrix, eigen decomposition  $\mathbf{C}_{\mathbf{x}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}$  can be applied to find the eigenvectors and eigenvalues of  $\mathbf{C}_{\mathbf{x}}$ . Then, the eigenvector  $\mathbf{q}_{\min}$  with the minimum eigenvalue is chosen to determine the rotation angle.

The generalized form of the rotation matrix between two arbitrary vectors  $\mathbf{a}$  and  $\mathbf{b}$  is defined as [70]

$$\mathbf{R} = \mathbf{I} - \mathbf{u}\mathbf{u}^T - \mathbf{v}\mathbf{v}^T + \begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \mathbf{u} & \mathbf{v} \end{bmatrix}^T \quad (5.2)$$

where

$$\begin{aligned} \mathbf{u} &= \frac{\mathbf{a}}{\|\mathbf{a}\|}, \\ \mathbf{v} &= \frac{\mathbf{b} - (\mathbf{u} \cdot \mathbf{b}) \mathbf{u}}{\|\mathbf{b} - (\mathbf{u} \cdot \mathbf{b}) \mathbf{u}\|}, \\ \theta &= \arccos \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|}. \end{aligned}$$

The first three terms in Eq. (5.2) find a projection onto the rotation subspace

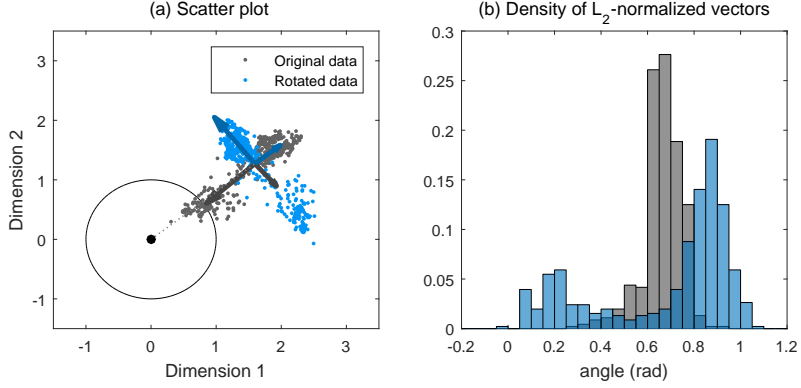


Fig. 5.2 A two-dimensional example of the vector rotation. (a) A scatter plot of the original and the rotated data. Eigenvectors and eigenvalues are depicted by the direction and the length of arrows. (b) A density plot of angles for both data when normalized onto the unit circle.

using the orthonormal basis  $\mathbf{u}$  and  $\mathbf{v}$ . The last term performs a two-dimensional rotation on a plane generated by two vectors  $\mathbf{a}$  and  $\mathbf{b}$ , and maps it back to the original dimension. By substituting with  $\mathbf{a} = \mathbf{q}_{\min}$  and  $\mathbf{b} = \mu_{\mathbf{x}}$ , the rotation is fixed with the angle between  $\mathbf{q}_{\min}$  and the mean vector  $\mu_{\mathbf{x}}$ . This allows that the feature vectors are widely dispersed when projected onto the hypersphere, by keeping the basis with the lowest spreadness parallel to the mean vector. Figure 5.2 illustrates a graphical example of the two-dimensional vector rotation, showing two distinct groups in the rotated data when the normalization is applied.

As the final step, the HSC is defined by the rotation around the mean vector followed by normalization:

$$\mathbf{y} = \mathbf{R}(\mathbf{x} - \mu_{\mathbf{x}}) + \mu_{\mathbf{x}} \quad (5.3)$$

$$\text{HSC} = \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad (5.4)$$

The rotation enables to find the best perspective to interpret the clustered data, while preserving the relative information between dimensions. Besides, the normalization removes absolute information about the energy, thus the HSC only includes the relative information between the harmonic partials. In other words, the HSC eventually contains only the essential information to represent the harmonic structure, regardless of other acoustic properties such as pitch and loudness.

## 5.4 Mixture model

As mentioned in the previous section, it is assumed that perception of a note boundary is closely related to a significant transition of the harmonic structure. If a stream contains several notes with different pronunciations that can be clearly distinguished, the HSCs will form several clusters on the surface of the unit hypersphere. Ideally, the number of clusters would be equal to the number of vowel pronunciations. Unsupervised classification is known as a standard solution for identifying these clusters; however, clustering methods such as the K-means or Gaussian mixture model are not suitable for the data in this particular distribution. Alternatively, a mixture model based on the von Mises–Fisher distribution is used here.

The von Mises–Fisher (vMF) distribution provides a suitable model to fit the data on the surface of a multidimensional unit sphere. The vMF distribution is applied in recent topics of information retrieval such as text mining, allowing it not to have a huge bias towards only a few words with highly frequent occurrence [71]. It is parametrized by the mean direction  $\mu$  and the concentration parameter  $\kappa$ , which refers to the spread of the distribution around the mean. Its

probability density function (pdf) for the  $h$ -dimensional unit vector  $x$  is defined by

$$p(x|\mu, \kappa) = \frac{\kappa^{h/2-1}}{(2\pi)^{h/2} I_{h/2-1}(\kappa)} e^{\kappa x^T \mu} \quad (5.5)$$

where  $I_r(\kappa)$  is the modified Bessel function of the first kind at order  $r$ .

In the mixture model, the Expectation-Maximization (EM) algorithm is used to estimate the mean and concentration parameters of each vMF distribution as formulated by Banerjee [72]. In a general EM framework, the model may converge to a local maximum of the likelihood function depending on setting the initial point, and it does not guarantee that the model is correctly converged to the global maximum. To avoid this, all the steps of parameter estimation are repeated ten times with different initial points, and the iteration is selected for which the log-likelihood sum is maximized. The mean vector of randomly selected samples, for which the mixing proportions are uniform, gives the initial point.

As all vectors belong to the  $(h-1)$ -sphere, the mean vector should be calculated in the  $h-1$  dimensional angular coordinate, instead of the Euclidean space. The angular coordinates  $\phi_i$  can be converted from the Cartesian coordinates  $x_1, \dots, x_h$  as

$$\phi_i = \arccos \frac{x_i}{\sqrt{x_h^2 + x_{h-1}^2 + \dots + x_i^2}} \quad (5.6)$$

where  $i = 1, 2, \dots, h-1$ . For a special case of  $x_h < 0$ ,  $\phi_{h-1} = 2\pi - \arccos \frac{x_{h-1}}{\sqrt{x_h^2 + x_{h-1}^2}}$ . Given  $N$  sample vectors, the mean angle of each coordinate  $\bar{\phi}_i$



is calculated by

$$\bar{\phi}_i = \text{atan2}(\text{Im}(\bar{z}_i), \text{Re}(\bar{z}_i)) \quad (5.7)$$

$$\text{where } \bar{z}_i = \frac{1}{N} \sum_{n=1}^N e^{j\phi_i}. \quad (5.8)$$

The mean vector  $\bar{\mathbf{x}} = [\bar{x}_1 \ \bar{x}_2 \ \dots \ \bar{x}_h]^T$  is finally obtained by the inverse transformation from the angular coordinates as follows:

$$\bar{x}_i = \begin{cases} \sin(\bar{\phi}_1) \cdots \sin(\bar{\phi}_{h-2}) \cos(\bar{\phi}_{h-1}) & \text{if } i < h \\ \sin(\bar{\phi}_1) \cdots \sin(\bar{\phi}_{h-2}) \sin(\bar{\phi}_{h-1}) & \text{if } i = h \end{cases} \quad (5.9)$$

Meanwhile, estimating the optimal number of mixture components (i.e. clusters) is not a simple issue, especially when the statistical characteristic of the data is not specified. In this work, fortunately, it is possible to assume roughly that the number of notes is proportional to the length of the stream. A heuristic regression approximated the correlation between the stream length and the note count. Using the ground truth in the dataset (see details in section 5.6.1), streams were first segmented so that each stream was divided by a short interval ( $> 0.1$  s). By counting the notes for each stream, it was noticed that the maximum note count could be roughly approximated to five times the stream length in seconds. To contain unnecessary clusters for a short transition, the maximum number of clusters is limited to five so that clusters are generated for only significant harmonic structures. Figure 5.3 shows the approximation of the initial number of clusters using the actual note counts.

In practical cases, streams may contain fewer notes than the maximum number. Moreover, the number of intrinsic harmonic structures can be even lower when some notes have the same vowel pronunciation. To this end, an

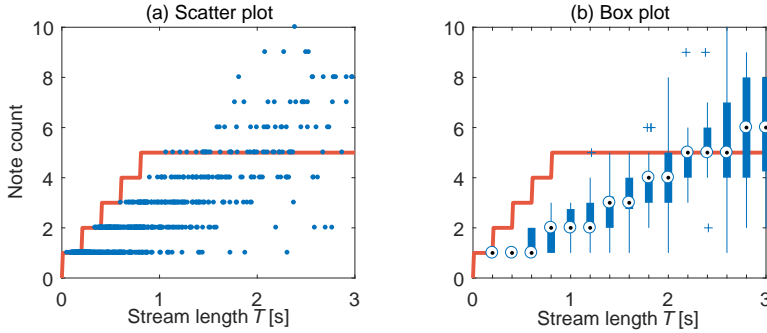


Fig. 5.3 Note counts by different stream lengths and the heuristic regression of the maximum number of clusters. Each dot in the scatter plot represents a stream. Variances in the box plot are shown with stream groups divided in a step of 0.2 s. The regression function  $g(T) = \min(\lceil 5T \rceil, 5)$  is depicted by the red line.

iterative method is developed to optimize the number of clusters as shown in Fig. 5.4, using the regression function of the maximum number of clusters.

Once the maximum number of clusters is initially determined by the stream length, the largest number of clusters that the mixture model converges within 100 EM iterations is found first. Next, by decreasing the number of clusters  $K$ , the EM algorithm is repeated to estimate the model parameters  $\Theta = \{\mu_{1\dots K}, \kappa_{1\dots K}\}$ , as long as the distance between the means of two clusters is shorter than a threshold  $d_{\min}$ . Since all the cluster means are located on the  $(h-1)$ -sphere, the distance is defined by the arc length between two points on the unit hypersphere,

$$d = \arccos \boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j, \quad 0 < d \leq \pi. \quad (5.10)$$

A close pair of clusters is merged by taking the mean vector of the two cluster means, and the initial points of the next vMF model are determined by the

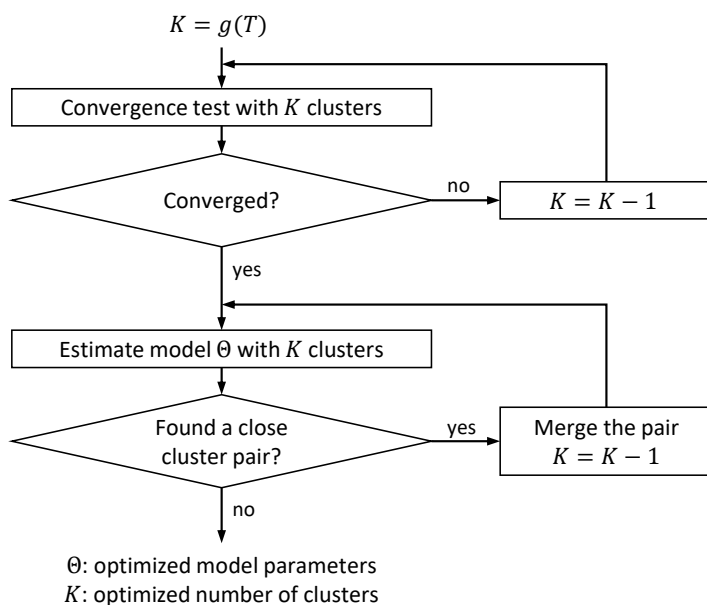


Fig. 5.4 Flowchart on the cluster optimization.

mean vector and all the other cluster means. This method is based on the agglomerative clustering, a bottom-up approach to merging pairs that are closely formed. It is advantageous to make the final clusters as distant to each other as possible.

## 5.5 Note detection

This sub-section describes the methods for determining the three basic attributes of a note: the onset, offset, and note pitch. Significant transitions in the harmonic structure are primarily detected to identify note boundaries. Then, the actual onsets and offsets are selected from the harmonic structure transitions, and a single pitch that represents a note will be finally decided.

### 5.5.1 Transition boundary detection

Detection of the harmonic structure transition is achieved by two different methods. The first builds a detection function representing the degree of local changes in the feature, using the self-similarity (or self-distance) analysis. The self-similarity analysis has been used mainly for music segmentation since early studies [73, 74]. The purpose of these works is to automatically find some points of significant structural transitions in music, such as a chorus after verses. In this work, a similar technique is applied at the note level to detect onsets instead of segments. A self-similarity matrix is obtained by subtracting from one the cosine distance between two HSC vectors, i.e.,

$$S_{i,j} = 1 - \text{HSC}_i \cdot \text{HSC}_j \quad (5.11)$$

where  $\text{HSC}_n$  denotes a row vector of the harmonic structure coefficient at the  $n$ -th frame. Note that the denominator of the cosine distance formula is removed since the  $\ell_2$ -norm of the HSC is unity. The novelty function is determined by

$$\text{Novelty}(n) = \sum_{i=-N/2}^{N/2} \sum_{j=-N/2}^{N/2} W_{i,j} \cdot S_{n+i,n+j} \quad (5.12)$$

where  $W$  is a Gaussian-tapered checkerboard kernel [75] which slides alongside the diagonal elements of the self-similarity matrix. A small kernel allows the detection of short notes but increases the chance of false positives. Conversely, a large kernel can be considered when the transcription system should avoid detecting spurious notes. In order to locate the transition boundaries, all the peaks (i.e., local maxima) in the novelty function are found first, and only the peaks higher than a peak-picking threshold  $\delta_{\text{peak}}$  are chosen. Note that this similarity-based method does not use the mixture model.

Although this approach is quite simple and easy to understand, choosing a proper peak-picking threshold heavily affects the transcription performance. Thus, another approach based on the hidden Markov model (HMM) is proposed as well, applying the parametric mixture model. The proposed HMM consists of a transient state and the same number of sub-HMMs as clusters from the mixture model. Each sub-HMM contains several one-way states to model a harmonic structure with a minimum duration constraint. This constraint prevents the state path from fluctuating instantaneously, as the state path is forced to stay in a cluster for  $T_{\min}$  seconds at least. All transition probabilities are determined by an input parameter  $\alpha$ , which decides the probability of staying in the current cluster or the transient state. This “self-transition probability” parameter controls the sensitiveness of the note event detection. If they become closer to unity, the transition is less likely to occur, thus less number of notes will be detected.

Observation probabilities are given by a function of the likelihood  $p(x|\mu, \kappa)$  of each cluster as defined in Eq. (5.5). Since the pdf can be greater than unity by its definition, the pdf is so normalized that the probabilities sum to unity at every instance of time. Given the normalized pdf  $p_{k,n}$  for all  $K$  clusters, the observation probabilities are calculated in the range between 0 and 1 as

$$b_{k,n} = \begin{cases} p_{k,n} \cdot \exp(p_{k,n} - 1) & \text{(sustain state)} \\ \sum_{k=1}^K \frac{\Delta p_{k,n+1} + \Delta p_{k,n}}{2} & \text{(transient state)} \end{cases} \quad (5.13)$$

where  $\Delta p_{k,n} = |p_{k,n} - p_{k,n-1}|$ . The observation probabilities of the transient state are determined by changes in the pdf of the clusters. At the end, the prior probability is uniformly given to all clusters and the transient state. After the three HMM parameters are determined for all the states, the optimal state path

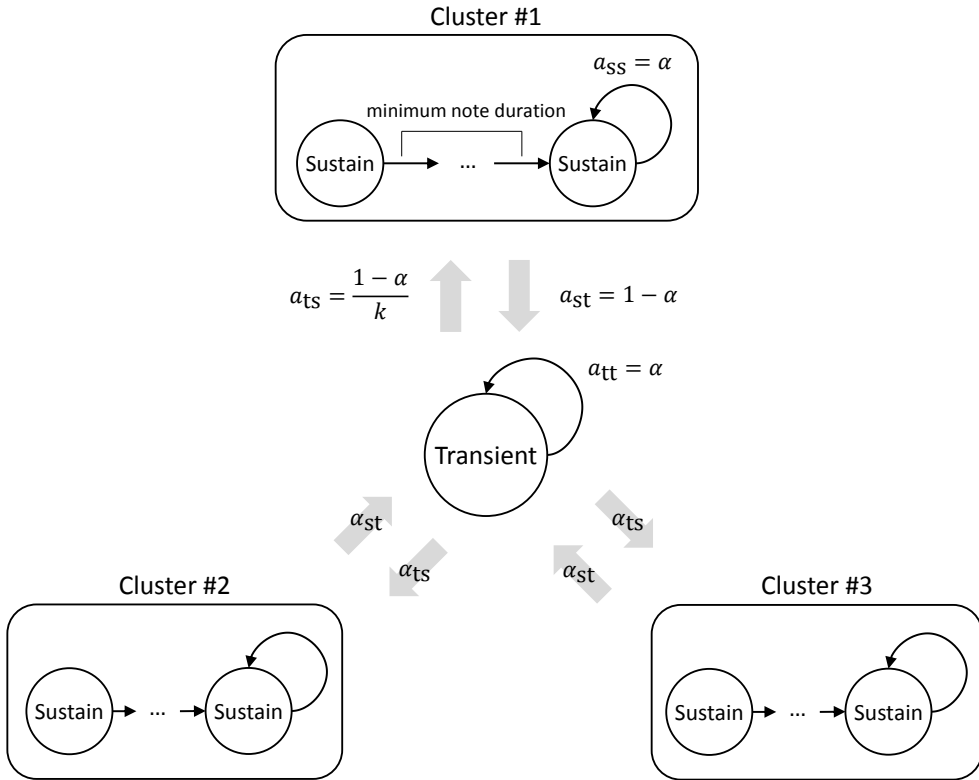


Fig. 5.5 Transitions in the hidden Markov model.

$v = \{v_1, \dots, v_N\}$  is decoded by the Viterbi algorithm. Accordingly, transition boundaries, at which the state path changes from the transient state to a sustain state, are simply detected.

### 5.5.2 Note boundary selection

It is clear that the transition boundaries indicate the points at where the harmonic structure significantly changes. However, not all transitions are directly converted into the note onset, because some voiced consonants such as [l], [m] and [ŋ] can be included. These voiced consonants, commonly observed in hum-

ming, may cause low detection accuracy if they are detected as independent notes. Therefore, it is necessary to exclude the voiced consonants from the note boundary, using their distinguishing spectral characteristic due to the nasal sound.

Let  $x_{i,t}$  denotes the log-magnitude of the  $i$ -th harmonic partial at a time instance  $t$ . Mean height  $\bar{\delta}_\tau$  at a transition boundary time  $\tau$  is defined by

$$\bar{\delta}_\tau = \frac{1}{h} \sum_{i=1}^h \left( \max_{t \in (\tau, \tau+T)} x_{i,t} - \min_{t \in (\tau-T, \tau)} x_{i,t} \right) \quad (5.14)$$

where  $T = T_{\min}/2$ . When a voiced consonant is followed by a normal vowel, the harmonic partial magnitude decreases except in the first few partials. A note boundary is selected at  $\tau$  only if  $\bar{\delta}_\tau > \delta_{\text{note}}$ , and determines *onset* and *offset*.

### 5.5.3 Note pitch decision

When the note boundary and F0s are given, the simplest way to decide the note pitch would be by taking a mean or median value of the F0s between the onset and the offset. In singing, however, it is sometimes difficult to specify a single value of the F0s within a note. Singing voices often include musical expressions and ornaments such as a grace note, which is a separate pitch prefixed to a principal note. The longest region for which the pitch deviations are kept below a tolerance of 50 cents (100 cents = 1 semitone) is selected, and the pitch at the beginning of the region decides the note pitch. In doing so, a note pitch that is most likely to be perceived can be chosen. The detailed algorithm for note pitch decision is presented in the form of a pseudo-code in Algorithm 2.

Figure 5.6 summarizes the whole transcription process of a female singing voice signal. Panel (b) illustrates the HSC and eight detected transition bound-

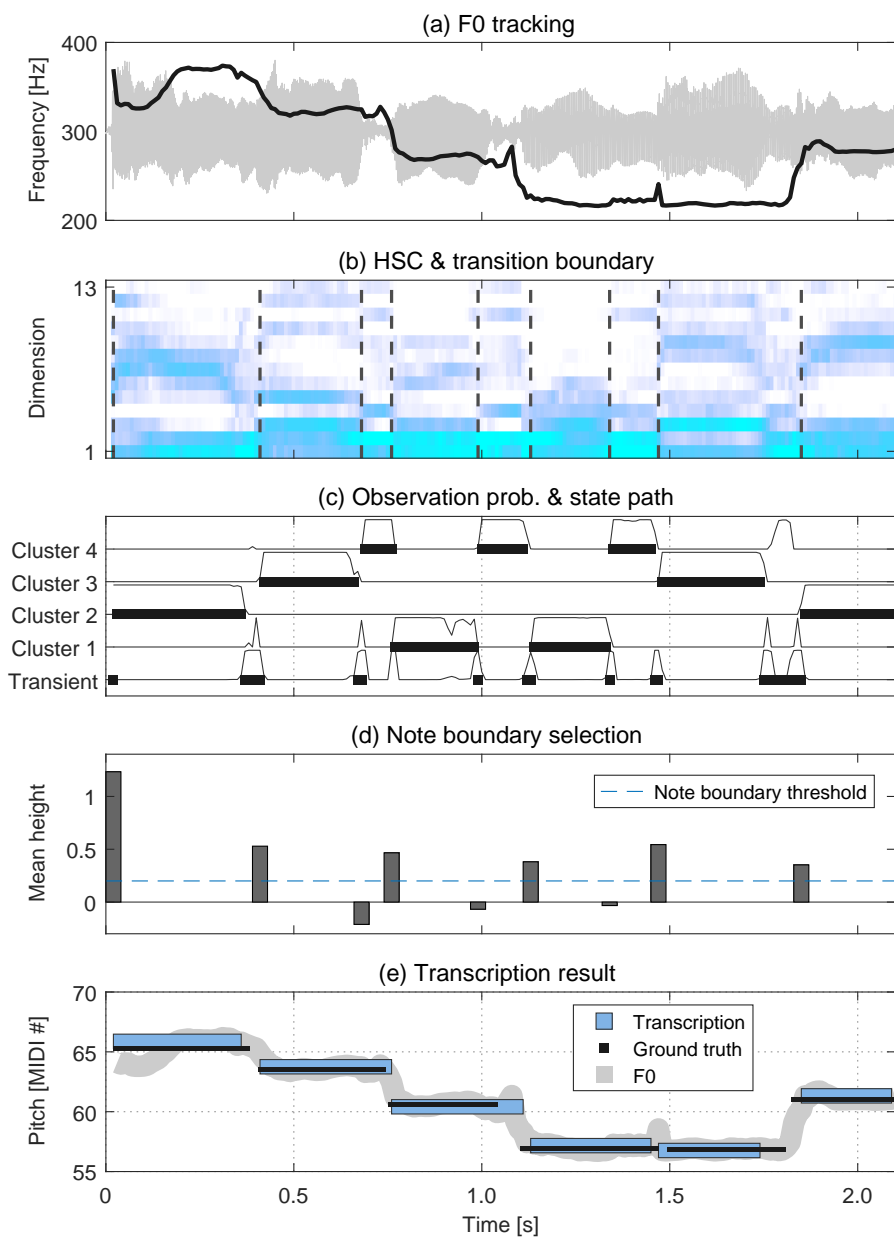


Fig. 5.6 Transcription result from an excerpt of `afemale10.wav` in the dataset.



---

**Algorithm 2** Note pitch decision.

---

```
1: Input:  
    $p_n$  : F0s in MIDI number for  $n = 1, \dots, N$   
    $T_p$  : Tolerance of pitch deviation  
2:  $l_{\max} \leftarrow 0$   
3: for  $n = 1$  to  $N - 1$  do  
4:    $m \leftarrow n$   
5:    $l \leftarrow 1$   
6:   while  $m < N, |p_{m+1} - p_n| < T_p$  do  
7:      $m \leftarrow m + 1$   
8:      $l \leftarrow l + 1$   
9:   end while  
10:  if  $l > l_{\max}$  then  
11:     $l_{\max} \leftarrow l$   
12:     $pitch \leftarrow p_n$   
13:  end if  
14: end for  
15: return  $pitch$ 
```

---

aries. Panel (c) shows the observation probabilities of the HMM and the corresponding state path. Mean height for each transition boundary is depicted in the panel (d), showing six of them are selected as note onset. In the last panel, the transcription result is displayed in the form of a piano-roll representation. It is notable that two connected notes with the same pitch (the fourth and the fifth note) are correctly transcribed.

## 5.6 Evaluation

### 5.6.1 Dataset

Evaluations have been conducted using a publicly available dataset [76, 77], released for the purpose of evaluation on singing transcription. The dataset consists of 38 audio recordings of monophonic singing, recorded with a sample rate of 44.1 kHz and a 16-bit resolution. All the singings are in English, but

a few hummings are also contained. Singers are categorized into three classes: adult males (13 recordings), adult females (11 recordings), and children (14 recordings). The male and female recordings are randomly chosen from the public dataset MTG-QBH<sup>1</sup> [76], and the melodies came from several excerpts of popular songs such as The Beatles and Aerosmith. In the case of children recordings, traditional children songs were originally sung by eight different children. The pitch and loudness are quite unstable as the singers are untrained. The duration of the whole dataset is up to 19 minutes and 15 seconds in total. All the recordings were very freely performed with musical articulations and ornaments.

The dataset also contains the note-level ground truth by manual annotations. The ground truth provides annotation of the onset, offset, and note pitch for all the 2,154 notes in the dataset. The onset and offset are given by their exact time in seconds, and the note pitch is by a MIDI number with two decimal places. The MIDI number is converted from the frequency in Hz by  $12 \log_2(\text{frequency}/440) + 69$ .

### 5.6.2 Criteria and measures

Precision and recall have been commonly considered the standard measures for binary classification such as the onset detection. Combining the precision and the recall, the F-measure is the most representative measure of the overall performance. However, a note transcription system needs to adopt more extensive criteria, because it includes the overall evaluation for the three note attributes. Thus, recent criteria were extended particularly for singing transcription [65].

---

<sup>1</sup>The MTG-QBH dataset is available at <http://mtg.upf.edu/download/datasets/MTG-QBH>.

The qualitative meanings in the criteria are described as follows:

- COnPOff (correct onset, pitch and offset): The most restrictive criterion, meaning the correct rate of onset ( $\pm 50$  ms), offset ( $\pm 20\%$  of the ground-truth note duration or 50 ms, whichever is larger) and pitch ( $\pm 0.5$  semi-tones). A note is correctly transcribed only if its onset, offset, and pitch satisfy the criteria simultaneously.
- COnP (correct onset and pitch): A less restrictive criterion, accounting for both the onset and pitch, using the same size of tolerance window as above.
- COn (correct onset): Similar to the above two criteria, but only onset is considered in this case. This is equal to the traditional metric for onset detection.
- Split: The rate of ground truth notes incorrectly segmented into consecutive notes by transcription.
- Merge: The rate of ground truth notes merged as they are transcribed into the same note (complementary to Split).
- Spurious: The rate of transcribed notes not having any overlap with ground truth notes (neither in time nor pitch domain).
- Non-detected: The rate of ground truth notes not having any overlap with transcribed notes (neither in time nor pitch domain).

Note that COnPOff, COnP, and COn are chosen as major criteria for the overall performance of note transcription. Each criterion has its numerical measures such as precision, recall and F-measure. For other criteria such as Split,

Merge, Spurious and Non-detected, the measures are expressed by the rate of incorrectly transcribed notes that each criterion defines, emphasizing the more specific points of wrong transcription.

### 5.6.3 Experimental setup

Two evaluations were conducted in various aspects of singing transcription at the note level, rather than the assessment for the front-end pitch tracker at the frame level. This is because the existing algorithms for monophonic pitch tracking have already accomplished a reliable performance, and the proposed note transcription system is based on the assumption that the F0 is known.

The first evaluation assessed the transcription performances among two methods for transition boundary detection, and to examine the influence of different parameter configurations. By comparing the results, the best method and the most optimized parameter were determined. On the other hand, the second evaluation shows the improvement of the proposed system compared to other systems including the state-of-the-arts. For a fair comparison, the experiment was conducted under an identical experimental setup including dataset and metrics. The default parameter configuration in all the experiments is summarized in Table 5.1.

All the experiments were conducted on a personal computer with a 3.3 GHz CPU and 8 GB RAM. The computational time for the entire transcription system depends on whether the probabilistic models are included or not, and most of the time was spent on F0 tracking. The detailed time taken for all input signals with a total length of 1,155 seconds is displayed in Table 5.2.

Table 5.1 Parameter configuration.

Parameter	Description	Value
$h$	Number of harmonic partials	13
$d_{\min}$	Minimum cluster distance (rad)	0.25
$\delta_{\text{peak}}$	Peak picking threshold	0.03
$\alpha$	Self-transition probability	0.5
$\delta_{\text{note}}$	Note boundary threshold	0.2
$T_{\min}$	Minimum note duration (s)	0.1

Table 5.2 Computational time of the proposed system.

	Using the given F0 track		Including F0 tracking	
	Similarity-based	HMM-based	Similarity-based	HMM-based
Time (s)	37	110	296	374

## 5.7 Results and discussions

As the first evaluation, the overall performance was compared by using different parameters, including the number of harmonic partials and the detection sensitivity. This experiment was conducted using the two methods for transition boundary detection, the similarity analysis and the HMM-based note event model. As shown in Fig. 5.7, the performance improvement was saturated in both methods with more than 11 partials, and the highest F-measure of 0.82 was achieved by the HMM-based method. As the number of partials increases, the performance of the similarity-based method slightly decreases while the HMM-based method does not change. It is also noticeable that the similarity-based method scored a very low performance when only a few partials were

used. This result implies that the HMM-based method is more robust than the similarity-based method.

To verify the robustness of the HMM-based method more clearly, the precision-recall curve for both detection methods is reported in Fig. 5.8, showing the trade-off between precision and recall. The precision and the recall were obtained by varying the parameters  $\delta_{\text{peak}}$  and  $\alpha$ , which determine the detection sensitivity of the similarity-based and the HMM-based method, respectively. While the HMM-based method achieved a reliable performance for various detection sensitivities, the precision rapidly decreased in the similarity-based method as the peak-picking threshold increased. In most cases, it was reported that the recall tends to be greater than the precision.

Both experimental results show that the use of the mixture model not only improves the overall performance, but also accomplishes the robustness of the system. The similarity-based method is heavily influenced by the parameters and the characteristic of the input signal, since it is difficult to choose a proper threshold for peak picking. Whereas, the mixture model is effective for classifying the intrinsic harmonic structures in a stream, even when a limited number of partials are given. Nonetheless, the overall performance of the similarity-based method is still higher than the recent average results of the onset detection for the singing voice class in the MIREX. This infers that the HSC is a very effective feature to represent the harmonic structure, and is suitable for singing transcription even without the mixture model.

The second evaluation was conducted to compare the system performance with five other methods. All the results are excerpts from the original papers [65, 77] that use the same dataset and criteria. The results attained by

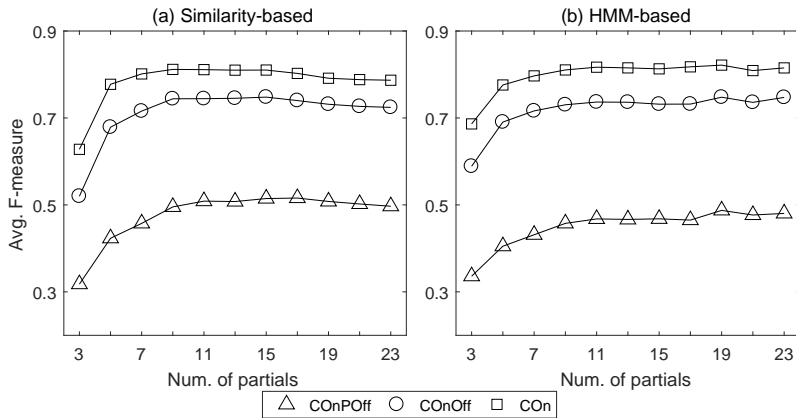


Fig. 5.7 Average F-measures in three criteria by different number of harmonic partials.

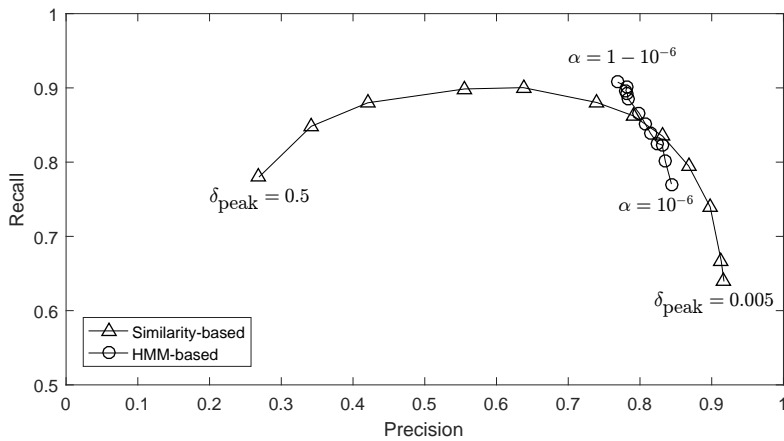


Fig. 5.8 Precision-recall curves in CO<sub>n</sub> criterion for two transition detection methods.

Rynnänen’s note event model approach [64], Gómez & Bonada’s method [35], a commercial system named Melotranscript [78] were cited from Molina’s evaluation framework [77]. The SiPTH system has only one overall performance about CO<sub>n</sub>POff, since the authors do not mention the result on CO<sub>n</sub>P and CO<sub>n</sub> in

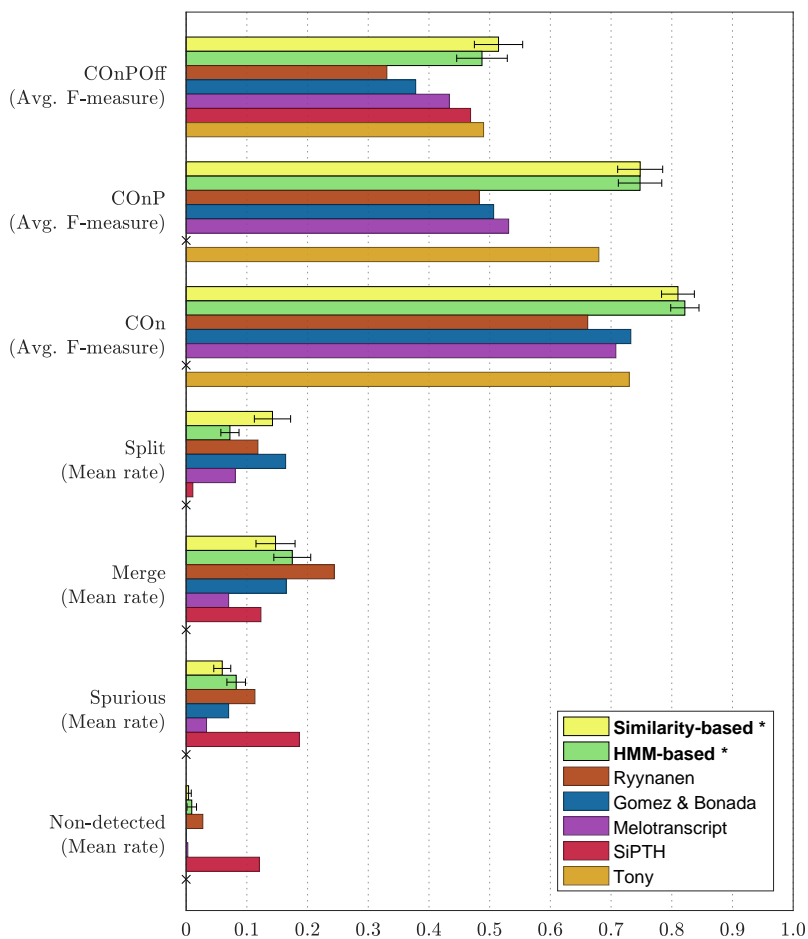


Fig. 5.9 Evaluation comparison of the proposed system (marked by asterisk) and other algorithms. Labels on the y-axis indicate the criteria and their numerical measure. Items marked by crosses are not publicly announced.

their paper [65]. In case of Tony [79], their best result was chosen (reported as pYIN  $s=0.8$ ,  $prn=0.10$ ) among different parameter configurations.

As shown in Fig. 5.9, the overall performance of the proposed system outperforms others including the state-of-the-art methods (SiPTH and Tony). In terms of COOn, the best performance (average F-measure 0.82, 95% confidence



interval 0.80 to 0.84) was achieved using the HMM-based method. The performance improvement on COnP becomes more significant compared to the first three systems. It implies that the local homogeneity within the harmonic structure, which is the most distinguishing point to other approaches, can be an effective feature for singing transcription, as it has an advantage that connected notes with the same pitch can be detected.

However, the proposed system did not improve much when the offset detection is included. The relatively low improvement on COnPOff can be explained by two factors. First, even with the feature normalization to remove the influence of the loudness, it cannot reflect the changes in harmonic structure as the singing becomes softer at the end of a note. Second, it may be caused by the ambiguity in the offset annotation for the singing voice.

Split and Merge are complementary to each other. As the detection sensitivity becomes higher, Merge decreases and Split increases. In the proposed system, the detection sensitivity mainly depends on the note boundary threshold  $\delta_{\text{note}}$ . When it increases from 0.2 to 0.3, it was observed the system produces only Splits less than 0.05% of the entire ground truth notes, while the overall performance is still higher than others (over 80% COn). Since it cannot say that either Split or Merge is more critical, it is required to use appropriate settings depending on the purposes of transcription.

Although the proposed system accomplished the best overall performance, it is not always the best approach for all cases. One example is a stepwise pitch change with the same pronunciation, which can be easily detected by pitch-based systems. It is expected that the system can be further improved when the time-pitch curve is also considered.

### 5.7.1 Failure analysis

Although the proposed system has improved overall performance compared to existing algorithms, it was reported that the note detection still needs to improve for some particular cases. As such, we analyze three typical cases of incorrect transcription as follows:

- **Long-tail release:** As opposed to attack, the term ‘release’ means the ending part of a note. A more relaxed criterion is applied to the tolerance window for offset detection (as we have previously described), since it is difficult to specify ambiguous offset points of singing voice. A long release, which means gradual changes in volume, is expressed as *decrescendo* or *diminuendo* in musical terms. Split occurs frequently in notes with long release because the harmonic structure is not homogeneous from onset to offset. As singers do not maintain the pronunciation and relax the tension for singing at the end of the note, it is often observed that the pronunciation changes to voiced consonants such as [m] or [ŋ] (see Fig. 5.10). Some detection errors can be corrected during the note boundary selection process, but if these errors occur frequently, the harmonic structure coefficients of the long-tail release will form an individual cluster, which may cause the note detection error. An exceptional treatment on the pre-defined harmonic structure of the voiced consonants is considerable to reduce this error.
- **Vibrato:** Vibrato is a musical technique by a regular and pulsating change of pitch (and loudness). In singing, it occurs by movement in different parts of the vocal tract, and has a frequency of about 5 Hz [80]. If only

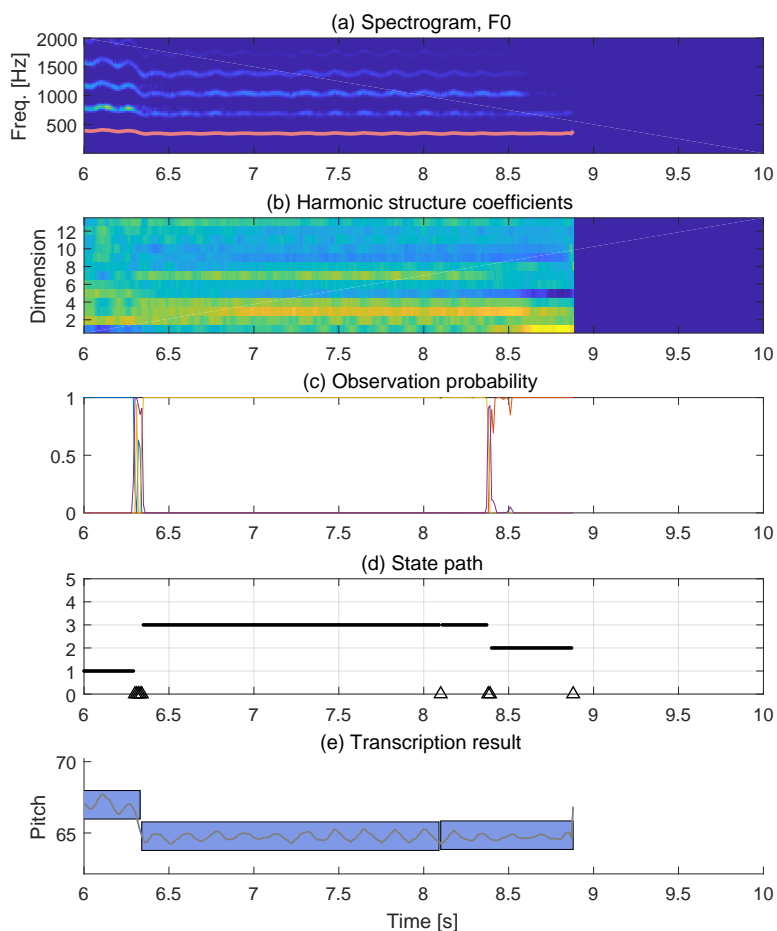


Fig. 5.10 Failure analysis for a case of incorrect transcription caused by long-tail release. The ending part of the note formed a different cluster (from 8.5 to 9 seconds).

the pitch curve are used as features, vibrato can be a hard obstacle for singing transcription. For non-professional singers, fortunately, it has been observed in our system that the fluctuation in the harmonic structure coefficient by vibrato is not extreme to be classified into other clusters. However, since a strong vibrato of professional singers affects not only

the pitch and loudness but also the timbre, it is also possible to consider performing the vibrato suppression [81] in the pre-processing step.

- **Pitch changes in a single harmonic structure:** Although not often observed, notes with different pitches while maintaining the same harmonic structure cannot be detected by the proposed approach. As a solution to this problem, existing pitch-curve-based algorithms are simple to implement and have low complexity. In order to prevent such errors, it is expected that the performance can be improved by applying existing algorithms together with the proposed system.

## 5.8 Summary

We presented a singing transcription system based on the analysis of harmonic structure. Given the estimated F0 sequence, a novel acoustic feature named the harmonic structure coefficient (HSC) was derived by extracting the harmonic partial magnitude with several refinement steps of vector transformation. In doing so, the HSC is defined on the surface of a unit hypersphere, representing the relationship between harmonic partials.

A parametric mixture model based on the von Mises–Fisher distribution was used to characterize the feature space. Further, an optimization technique was proposed to determine the optimal number of clusters, so that the intrinsic harmonic structure could be correctly classified.

To detect significant transition boundaries in the harmonic structure, two different methods were presented based on the self-similarity analysis and the HMM. Then, note attributes were finally determined by excluding the voiced

consonant from the detected transition boundaries.

The proposed system was evaluated using the latest evaluation methodology for singing transcription. Comparing results of the two proposed methods for transition boundary detection showed that the mixture model and the note event model improve the transcription performance and robustness. When comparing with the existing systems, the evaluation results confirm that the proposed transcription system significantly outperforms other systems including the state-of-the-art systems.

## Chapter 6

# Conclusion and Future Work

### 6.1 Contributions

As mentioned in the introduction, the final goal of this thesis was to develop a fully-automatic singing transcription system that outperforms the state-of-the-art methods. This requires a broad understanding and multilateral techniques for pitch estimation and note segmentation, which are the fundamental topics of a transcription system. We analyzed their strengths and limits through a comprehensive review of existing methods (Chapter 2), and proposed a novel approach to improve the accuracy and efficiency of pitch tracking algorithms (Chapter 3). Also, focusing on the observation that the homogeneity of the harmonic structure is maintained locally in a note, we presented an onset detection method based on cepstral analysis (Chapter 4), which improves the detection of soft onset which was difficult in traditional approaches. By extending these works to the transcription of singing voices, we have shown that our approach

is applicable to construct a robust system that can be applied to singing voice signals with complex acoustic characteristics (Chapter 5).

The main contributions of this thesis can be summarized in the following points:

- **A broad range of review covering transcription-related topics:** A comprehensive review of related topics was provided to help understand the problems and strategies for the purpose of music transcription. This includes a variety of approaches for pitch estimation, onset detection, and singing transcription. We also introduced the standard evaluation framework and methods in the field of music transcription, and listed highly relevant datasets to aid in future research.
- **Improved F0 estimation method using data sampling:** We exploited a data sampling method to improve the performance of F0 estimators. This method can be used in the time and frequency domain, and can be applied to many conventional algorithms. The data sampling method was not only computationally efficient, but also effective for estimation accuracy. The performance evaluation using a large-scale singing voice dataset showed that estimation accuracy was improved compared to the original detection functions. In addition, an iterative F0 refinement technique was also proposed.
- **Note onset detection based on cepstral analysis:** A novel approach for note onset detection for pitched instruments was presented. We derived the detection function that quantifies the regularity of the harmonic structure. This cepstrum-based method achieved a significant improve-

ment, especially in soft onset detection. Whereas the conventional methods depend heavily on the amplitude changes, this method enabled to detect onsets in various types of music signals such as strings and singing voices.

- **Robust singing transcription system:** A robust singing transcription system was presented in a unified framework. This includes (i) a novel acoustic feature, (ii) a parametric mixture model, and (iii) note transcription schemes. The feature was defined to represent the relationship between harmonic partials, regardless of pitch and loudness. It was characterized by a mixture model based on the von Mises–Fisher distribution. Further, an optimization technique was proposed to determine the optimal number of clusters, so that the intrinsic harmonic structure in a music signal can be properly classified. Finally, a note event model based on the hidden Markov model was also designed.

The singing transcription system presented in this thesis was implemented as an application named STAM (Singing Transcription App for Matlab). STAM is an app with graphical user interface, developed using Matlab App Designer. It can be run on multiple operating systems (Windows, Mac OS, and Linux) with the Matlab version after R2016b, or can be executable standalone with the Matlab Runtime. Its transcription method uses the vMF mixture model and the HMM-based note detection methods, which were reported to be the most robust algorithms in our experiments. Users may record their own song through a microphone, or import a prepared audio file. The transcription result is displayed in the form of piano-roll illustration, and can be exported to a MIDI format file for the further uses. A screenshot of STAM is shown in Fig. 6.1.



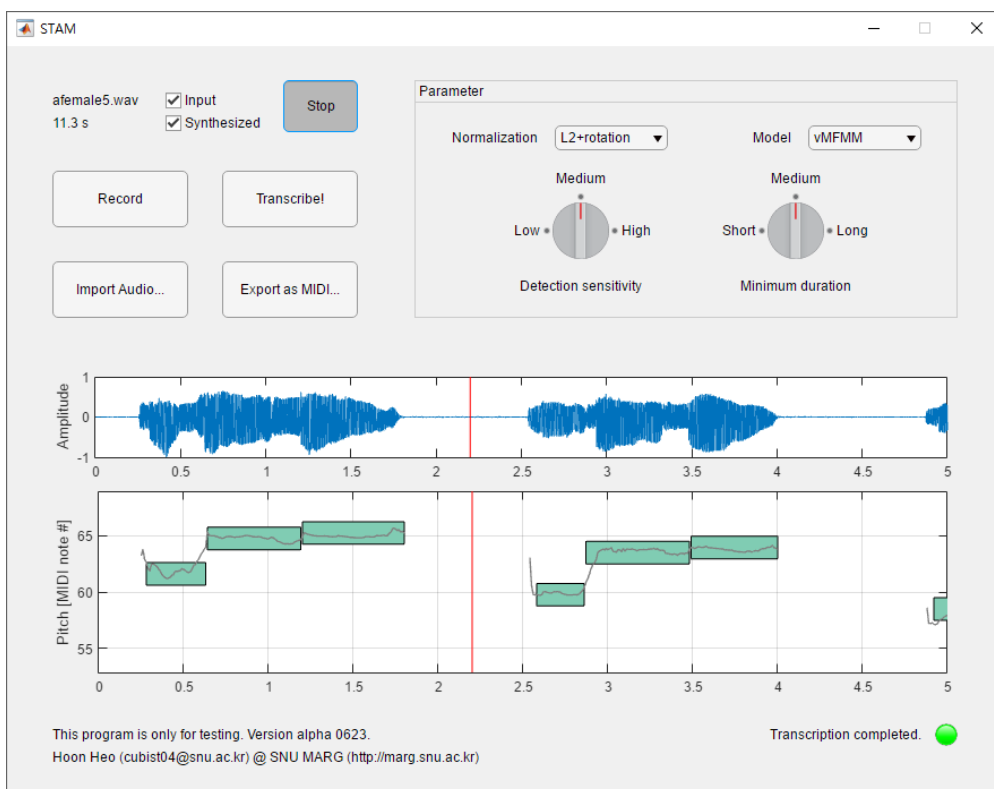


Fig. 6.1 A screenshot of the program implementation of the proposed system.

Despite the contributions to singing transcription and its relevant technologies in this study, there are some limitations for the following parts. First, this study has a very limited consideration for the extensibility to polyphonic music transcription. Regarding the most popular instruments such as piano and guitar allow polyphonic notes, this study needs to be extended to polyphonic transcription for a wider range of musical instruments. Secondly, the pitch tracking method presented in Chapter 3 was not used in the final system implementation, but was replaced by a newer pitch tracking algorithm with higher performance. Consequently, this has weakened the logical flow and connection of the entire work. For better configuration of the overall system, a new

pitch tracking method needs to be originally developed and optimized for this system. The next section describes an approach to track the F0 and its harmonic partials more precisely. We expect that this approach is more suitable for polyphonic pitch tracking. Thirdly, we did not take full advantage of lyrics, which is another useful information in the singing voice. Finally, the transcription of musical expressions such as articulations and ornaments has not yet been addressed in this thesis. Commonly found in other studies for singing transcription, these limitations draw the future direction. In the next section, we provide some good points that further studies can begin and go deeper.

## **6.2 Future work**

### **6.2.1 Precise partial tracking using instantaneous frequency**

Although a method to improve the pitch tracking accuracy was discussed in Chapter 3, we did not use it for the final implementation of the proposed singing transcription system. Instead, we constructed the system by employing a recent pitch tracker such as pYIN algorithm [52] in the front end of the system. However, this is not a desirable configuration for the following reasons.

Currently, harmonic partial tracking in our system is done by taking the magnitude spectrum corresponding to the integer multiples of the F0 estimated by a pitch tracker. However, not only F0 but also harmonic partials can also be utilized in the pitch tracking stage. For example, in the pYIN algorithm, the observation probability is determined from the YIN function (the lower this function, the higher the probability), and an average of four (voiced) pitch states with a significant probability are calculated per frame. As the front-end pitch tracking is independently working, only a single F0 estimate is preserved,

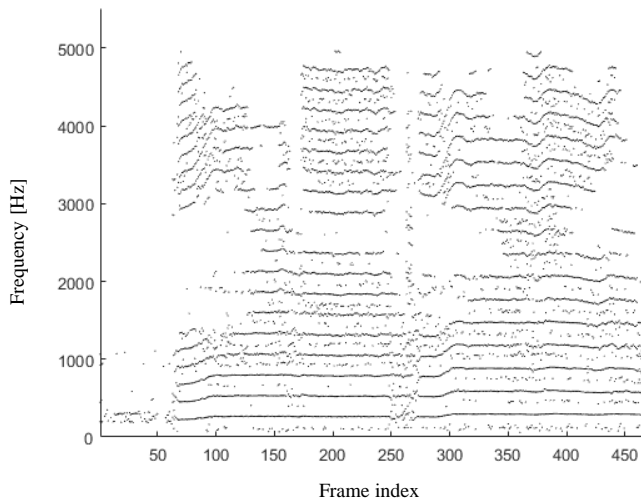


Fig. 6.2 Instantaneous frequency of harmonic components in a singing voice signal.

and useful information on the rest of harmonics would be lost in the main transcription stage. This causes redundant calculation on harmonic partials, and therefore, may worsen the efficiency of the overall system.

Can we realize both F0 tracking and multiple partial tracking at once? One possible idea is tracking using instantaneous frequency (IF). Since the IF is defined by a time-varying function, it has the advantage that the frequency at a certain time can be precisely calculated. After tracking all harmonic components including higher-order partials, if we estimate the F0 using the iterative F0 refinement (see Section 3.5), we expect that more precise partial tracking and efficient system configuration can be accomplished. In particular, this approach may be effective for tracking harmonic partials that are not exactly the integer multiples of the F0. We describe the derivation of the instantaneous frequency of a music signal in Appendix.

Figure 6.2 shows the instantaneous frequency of harmonic components obtained from the above formulations. In a singing voice signal, energy distribution of harmonic partials varies depending on the pronunciation. Although not shown in the figure, precise magnitudes of partials with strong energy can also be obtained, as well as their frequency. This precise tracking of harmonic partials is expected to improve the harmonic structure coefficient, and consequently lead to better transcription performance.

### **6.2.2 Linguistic model for note segmentation**

In the current transcription system, we regard the note segmentation of singing voice as an unsupervised classification problem, and decompose the singing voice by vowels (and voiced consonants). Although this approach improves the detection performance for legato notes, lyrics of the song is not yet utilized. Lyrics can be useful to detect syllables in a singing/speech signal. In situations where very high performance is required, we expect a great performance improvement by allowing the system to know in advance what song the user will sing.

According to International Phonetic Alphabet (IPA), the standard in linguistics for classifying the pronunciations of various languages, vowels can be plotted on a two-dimensional plane of the ‘closeness’ and ‘backness.’ This plot is called a vowel chart or a vowel diagram [82] (see Fig. 6.3). In speech recognition, there have been many studies to analyze the phoneme based on this phonetic criterion [83, 84].

Note segmentation based on the linguistic model will re-define the transition probability in our system. Assuming the one-to-one mapping between a

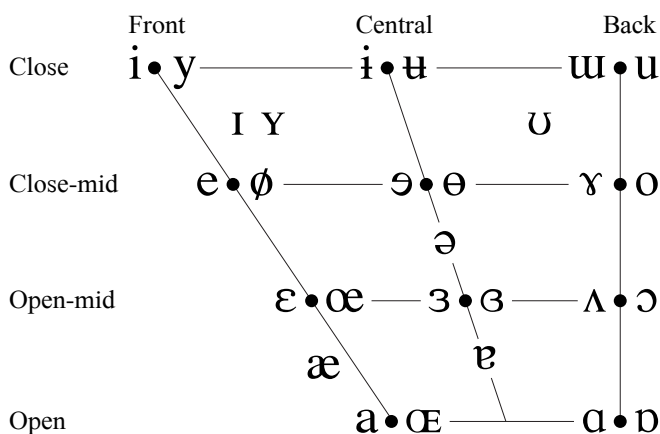


Fig. 6.3 IPA vowel chart.

musical note and a syllable, the transition probability between vowels is statistically determined by analyzing a large-scale text corpus of a specific language. According to the frequency of vowel transition in the Korean case we have investigated [85], eight most frequent vowels are chosen from the IPA vowel chart: [i], [ɨ], [u], [e], [ə], [æ], [a], and [o]. The authors examined the observed and expected frequencies of these vowels. Similarly, there are studies on ‘vowel harmony’ in other languages such as English and Finnish [86, 87]. The term ‘harmony’ implies that there are constraints on which vowels may be found near each other.

As a pilot test, using a pronunciation data set of 31 vowels, we extracted the HSC features from the eight most commonly used vowels in Korean. The 13-dimensional HSC feature vectors were clearly distinguishable from each vowel. Next, we assigned the clusters obtained by the mixture model to the vowels with the nearest HSC vector distance, and then used the transition probability between the vowels based on the statistics. However, transcription performance

was not improved as expected compared to our original method based on unsupervised classification.

The main issue to be solved to utilize the linguistic model for singing transcription is to refine the transition probability appropriately. The transition frequency between vowels obtained from linguistic literature is given at the phone level, whereas audio signal processing is usually done at the frame level. At the phone level, it is impossible to grasp the transition within a very short time (commonly observed in a diphthong). Unlike speech recognition, fortunately, singing transcription does not require a strict identification of every phoneme. If only a few common vowels are modeled, acquiring training data directly from an audio signal rather than a text corpus can be a possible solution.

# Appendix

## Derivation of the instantaneous frequency

For a single sinusoid  $s(t) = Ae^{j(\omega t + \theta)}$ , the instantaneous phase  $\varphi(t)$  and instantaneous frequency  $\lambda(t)$  are determined by their definition as

$$\varphi(t) = \omega t + \theta \quad (6.1)$$

$$\lambda(t) = \frac{1}{2\pi} \frac{d\varphi(t)}{dt} = \frac{\omega(t)}{2\pi}. \quad (6.2)$$

Real world music signals have multiple harmonic components, of course, so it is necessary to extend this concept to a two-dimensional time-frequency plane. This can be done by analyzing the original signal using a filter bank, as described in literature [88, 89].

Given  $x_k(t)$ , the output of the  $k$ th filter, the original signal  $x(t)$  can be expressed as

$$x(t) \cong \sum_{k=1}^K x_k(t). \quad (6.3)$$

Let the impulse response of the  $k$ th filter be  $g_k(t) = h(t) \cos(\omega_k t)$  where  $h(t)$  is the impulse response of a physically-realizable low-pass filter (normally realized by a window function). By using some basic properties of the Fourier transform

$$G_k(\omega) = \frac{1}{2} [H(\omega - \omega_k) + H(\omega + \omega_k)], \quad (6.4)$$

the filter bank can be explained by the frequency shifting of the low-pass filter.

Then, the output of the  $k$ th filter is the convolution of  $x(t)$  with  $g_k(t)$ ,

$$x_k(t) = \int_{-\infty}^t x(\tau)h(t-\tau) \cos[\omega_k(t-\tau)] d\tau. \quad (6.5)$$

By substituting  $\cos[\omega_k(t-\tau)]$  into  $e^{j\omega_k(t-\tau)}$  and taking the real part,

$$x_k(t) = \operatorname{Re} \left\{ e^{j\omega_k t} \int_{-\infty}^t x(\tau)h(t-\tau)e^{j\omega_k \tau} d\tau \right\} \quad (6.6)$$

$$= \operatorname{Re} \left\{ e^{j\omega_k t} X(\omega_k, t) \right\} \quad (6.7)$$

$$= |X(\omega_k, t)| \cos[\omega_k t + \varphi(\omega_k, t)]. \quad (6.8)$$

Each  $x_k(t)$  may be described as the simultaneous amplitude and phase modulation of a carrier  $\cos(\omega_k t)$  by the short-time amplitude and phase spectra of  $x(t)$ , both evaluated at frequency  $\omega_k$ . The instantaneous frequency  $\lambda$  at the point  $(\omega_k, t)$  is then defined as

$$\lambda(\omega_k, t) = \frac{d\varphi(\omega_k, t)}{dt}. \quad (6.9)$$

Instead of a direct calculation of phase, its time derivatives can be expressed by

$$\lambda(\omega_k, t) = \frac{a \frac{db}{dt} - b \frac{da}{dt}}{a^2 + b^2} \quad (6.10)$$

where

$$X(\omega_k, t) = a(\omega_k, t) + jb(\omega_k, t) \quad (6.11)$$

$$a(\omega_k, t) = \int_{-\infty}^t x(\tau)h(t-\tau) \cos(\omega_k \tau) d\tau \quad (6.12)$$

$$b(\omega_k, t) = - \int_{-\infty}^t x(\tau)h(t-\tau) \sin(\omega_k \tau) d\tau. \quad (6.13)$$



# Bibliography

- [1] S. Frankel and D. Gervais, *The Evolution and Equilibrium of Copyright in the Digital Age*, ser. Cambridge Intellectual Property and Information Law. Cambridge University Press, 2014. [Online]. Available: <https://books.google.com.au/books?id=KghQBAAAQBAJ>
- [2] C. Stutz. (2014) The average american listens to four hours of music each day. SPIN. Accessed: 23-May-2017. [Online]. Available: <http://www.spin.com/2014/06/average-american-listening-habits-four-hours-audio-day>
- [3] RealWire. (2012) Official study reveals that the average person will spend 13 years of their lives listening to music. RealWire. Accessed: 23-May-2017. [Online]. Available: <http://www.realwire.com/releases/Official-study-reveals-that-the-average-person-will-spend-13-years-of-their-lives-listening-to-music>
- [4] Beatport, LLC. Beatport. [Online]. Available: <http://beatport.com>
- [5] Pandora Media, Inc. Pandora. [Online]. Available: <http://pandora.com>
- [6] Last.fm Ltd. Last.fm. [Online]. Available: <http://last.fm>

- [7] The Echo Nest. The echo nest. [Online]. Available: <http://the.echonest.com>
- [8] A. Klapuri, “Signal processing methods for the automatic transcription of music,” Ph.D. dissertation, Tampere University of Technology Finland, 2004.
- [9] Songquito UG. Songs2See. [Online]. Available: <http://songs2see.com/en>
- [10] Microsoft Research. Songsmith. [Online]. Available: <http://songsmith.ms>
- [11] I. Simon, D. Morris, and S. Basu, “Mysong: automatic accompaniment generation for vocal melodies,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2008, pp. 725–734.
- [12] Yousician. Yousician. [Online]. Available: <http://yousician.com>
- [13] I. R. Titze and D. W. Martin, “Principles of voice production,” *The Journal of the Acoustical Society of America*, vol. 104, no. 3, pp. 1148–1148, 1998.
- [14] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, “A tutorial on onset detection in music signals,” *IEEE Transactions on speech and audio processing*, vol. 13, no. 5, pp. 1035–1047, 2005.
- [15] W. A. Schloss, “On the automatic transcription of percussive music: from acoustic signal to high-level analysis,” Ph.D. dissertation, Stanford University, 1985.
- [16] P. Smaragdis and J. C. Brown, “Non-negative matrix factorization for polyphonic music transcription,” in *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2003, pp. 177–180.

- [17] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *Proceedings of the 2011 Advances in Neural Information Processing Systems*, 2001, pp. 556–562.
- [18] S. Böck and M. Schedl, “Polyphonic piano note transcription with recurrent neural networks,” in *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2012, pp. 121–124.
- [19] S. Leglaive, R. Hennequin, and R. Badeau, “Singing voice detection with deep recurrent neural networks,” in *Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2015, pp. 121–125.
- [20] J. Nam, J. Ngiam, H. Lee, and M. Slaney, “A classification-based polyphonic piano transcription approach using learned feature representations,” in *Proceedings of the 2011 International Symposium on Music Information Retrieval*, 2011, pp. 175–180.
- [21] 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.
- [22] A. M. Noll, “Pitch determination of human speech by the harmonic product spectrum, the harmonic sum spectrum, and a maximum likelihood estimate,” in *Proceedings of the Symposium on Computer Processing Communications*, vol. 779, 1969.

- [23] D. J. Hermes, “Measurement of pitch by subharmonic summation,” *The Journal of the Acoustical Society of America*, vol. 83, no. 1, pp. 257–264, 1988.
- [24] M. Goto and Y. Muraoka, “A beat tracking system for acoustic signals of music,” in *Proceedings of the 2nd ACM International Conference on Multimedia*, ser. MULTIMEDIA '94. New York, NY, USA: ACM, 1994, pp. 365–372. [Online]. Available: <http://doi.acm.org/10.1145/192593.192700>
- [25] P. Masri, “Computer modeling of sound for transformation and synthesis of musical signal,” Ph.D. dissertation, University of Bristol, Bristol, United Kingdom, 1996.
- [26] A. Klapuri, “Sound onset detection by applying psychoacoustic knowledge,” in *Proceedings of the 1999 IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 6, Mar. 1999, pp. 3089–3092 vol.6.
- [27] C. Duxbury, M. Sandler, and M. Davies, “A hybrid approach to musical note onset detection,” in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, 2002, pp. 33–38.
- [28] E. Benetos and S. Dixon, “Polyphonic music transcription using note onset and offset detection,” in *Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2011, pp. 37–40.

- [29] R. J. McNab, L. A. Smith, I. H. Witten *et al.*, “Signal processing for melody transcription,” *Australian Computer Science Communications*, vol. 18, pp. 301–307, 1996.
- [30] E. Pollastri, “A pitch tracking system dedicated to process singing voice for music retrieval,” in *Proceedings of the 2002 IEEE International Conference on Multimedia and Expo*, vol. 1. IEEE, 2002, pp. 341–344.
- [31] T. De Mulder, J.-P. Martens, M. Lesaffre, M. Leman, B. De Baets, and H. De Meyer, “Recent improvements of an auditory model based front-end for the transcription of vocal queries,” in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. IEEE, 2004, pp. iv–iv.
- [32] H. Heo, D. Sung, and K. Lee, “Note onset detection based on harmonic cepstrum regularity,” in *Proceedings of the 2013 IEEE International Conference on Multimedia and Expo*, Jul. 2013, pp. 1–6.
- [33] J.-S. R. Jang. MIR-QBSH Corpus. MIR Lab, CS Dept, Tsing Hua Univ, Taiwan. Accessed: 23-May-2017. [Online]. Available: <http://mirlab.org/jang>
- [34] C. L. Hsu and J. S. R. Jang, “On the improvement of singing voice separation for monaural recordings using the mir-1K dataset,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 310–319, Feb. 2010.
- [35] E. Gómez and J. Bonada, “Towards computer-assisted flamenco transcription: An experimental comparison of automatic transcription algorithms

- as applied to a cappella singing,” *Computer Music Journal*, vol. 37, no. 2, pp. 73–90, 2013.
- [36] Z. Duan, B. Pardo, and C. Zhang, “Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 2121–2133, 2010.
- [37] P. Leveau and L. Daudet, “Methodology and tools for the evaluation of automatic onset detection algorithms in music,” in *Proceedings of the 2004 International Symposium on Music Information Retrieval*. Citeseer, 2004.
- [38] ODB. Pattern Recognition and Artificial Intelligence Group, University of Alicante (PRAIg-UA). Accessed: 23-May-2017. [Online]. Available: <http://grfia.dlsi.ua.es/cm/projects/prosemus/database.php>
- [39] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, “Rwc music database: Music genre database and musical instrument sound database,” in *Proceedings of the 2003 International Symposium on Music Information Retrieval*, Oct. 2003, pp. 229–230.
- [40] M. Goto *et al.*, “Development of the rwc music database,” in *Proceedings of the 18th International Congress on Acoustics (ICA 2004)*, vol. 1, 2004, pp. 553–556.
- [41] M. Ramona, G. Richard, and B. David, “Vocal detection in music with support vector machines,” in *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2008, pp. 1885–1888.

- [42] H. Heo, H. Lim, and K. Lee, “시간/주파수 영역의 샘플링을 통한 주기성 분석 기법,” in *2015년도 한국음향학회 추계학술발표대회 논문집*, vol. 34, no. 2, 2015.
- [43] J. W. Xu and J. C. Principe, “A pitch detector based on a generalized correlation function,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1420–1432, Nov. 2008.
- [44] J. Tabrikian, S. Dubnov, and Y. Dickalov, “Maximum a-posteriori probability pitch tracking in noisy environments using harmonic model,” *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 1, pp. 76–87, 2004.
- [45] A. Klapuri, “Multiple fundamental frequency estimation by summing harmonic amplitudes.” in *Proceedings of the 2006 International Symposium on Music Information Retrieval*, 2006, pp. 216–221.
- [46] P. Boersma, “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound,” in *Proceedings of the Institute of Phonetic Sciences*, vol. 17, no. 1193. Amsterdam, 1993, pp. 97–110.
- [47] R. Goldberg and L. Riek, *A practical handbook of speech coders*. CRC press, 2000.
- [48] T. Drugman and A. Alwan, “Joint robust voicing detection and pitch estimation based on residual harmonics,” in *Proceedings of Interspeech 2011*, 2011, pp. 1973–1976.

- [49] Gloat (glottal analysis toolbox). Accessed: 27-April-2015. [Online]. Available: <http://tcts.fpms.ac.be/~drugman/Toolbox>
- [50] Computepitch/harmonic product spectrum. Accessed: 27-April-2015. [Online]. Available: <http://www.audiocontentanalysis.org/code>
- [51] Yin pitch estimator. Accessed: 27-April-2015. [Online]. Available: <http://audition.ens.fr/adc/sw/yin.zip>
- [52] M. Mauch and S. Dixon, "Pyin: A fundamental frequency estimator using probabilistic threshold distributions," in *Proceedings of the 2014 IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2014, pp. 659–663.
- [53] pYIN plugin. Accessed: 27-April-2015. [Online]. Available: <https://code.soundsoftware.ac.uk/projects/pyin/files>
- [54] Sonic annotator. Accessed: 27-April-2015. [Online]. Available: <http://www.vamp-plugins.org/sonic-annotator>
- [55] L. Rabiner, M. Cheng, A. Rosenberg, and C. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 399–418, 1976.
- [56] O. Babacan, T. Drugman, N. d'Alessandro, N. Henrich, and T. Dutoit, "A comparative study of pitch extraction algorithms on a large variety of singing sounds," in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7815–7819.



- [57] C. C. Toh, B. Zhang, and Y. Wang, “Multiple-feature fusion based onset detection for solo singing voice,” in *Proceedings of the 2009 International Symposium on Music Information Retrieval*, Kobe, Japan, 2009.
- [58] J. T. Foote, “Content-based retrieval of music and audio,” in *Proceedings of the Voice, Video, and Data Communications*. International Society for Optics and Photonics, 1997, pp. 138–147.
- [59] R. E. Berg and D. G. Stork, *The physics of sound*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2005.
- [60] O. Lartillot and P. Toivainen, “A matlab toolbox for musical feature extraction from audio,” in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Bordeaux, 2007.
- [61] H. Heo and K. Lee, “Robust singing transcription system using local homogeneity in the harmonic structure,” *IEICE Transactions on Information and Systems*, vol. E100-D, no. 5, pp. 1114–1123, 2017.
- [62] M. S. Rahman and T. Shimamura, “Pitch determination from bone conducted speech,” *IEICE Transactions on Information and Systems*, vol. E99-D, no. 1, pp. 283–287, Jan. 2016.
- [63] L. Clarisse, J.-P. Martens, M. Lesaffre, B. De Baets, H. De Meyer, and M. Leman, “An auditory model based transcriber of singing sequences,” in *Proceedings of the 2002 International Symposium on Music Information Retrieval*. Citeseer, 2002.

- [64] M. P. Rynänen and A. P. Klapuri, “Modelling of note events for singing transcription,” in *ISCA Tutorial and Research Workshop (ITRW) on Statistical and Perceptual Audio Processing*, 2004.
- [65] E. Molina, L. J. Tardón, A. M. Barbancho, and I. Barbancho, “Sipth: Singing transcription based on hysteresis defined on the pitch-time curve,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 252–263, Feb. 2015.
- [66] Z. Duan, Y. Zhang, C. Zhang, and Z. Shi, “Unsupervised single-channel music source separation by average harmonic structure modeling,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 4, pp. 766–778, May 2008.
- [67] M. Mauch, H. Fujihara, K. Yoshii, and M. Goto, “Timbre and melody features for the recognition of vocal activity and instrumental solos in polyphonic music.” in *Proceedings of the 2011 International Symposium on Music Information Retrieval*, 2011, pp. 233–238.
- [68] K. Dressler, “Audio melody extraction for mirex 2009,” *5th Music Information Retrieval Evaluation eXchange (MIREX)*, vol. 79, pp. 100–115, 2009.
- [69] J. Salamon and E. Gomez, “Melody extraction from polyphonic music signals using pitch contour characteristics,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1759–1770, Aug. 2012.

- [70] S. Montgomery-Smith. Finding the rotation matrix in n-dimensions. Mathematics Stack Exchange. Accessed: 23-May-2017. [Online]. Available: <http://math.stackexchange.com/q/598782>
- [71] J. Reisinger and R. J. Mooney, “Multi-prototype vector-space models of word meaning,” in *Human Language Technologies: The 2010 Ann. Conf. the North American Chapter of the Association for Computational Linguistics*, ser. HLT '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 109–117. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1857999.1858012>
- [72] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra, “Clustering on the unit hypersphere using von mises-fisher distributions,” *Journal of Machine Learning Research*, vol. 6, pp. 1345–1382, Sep. 2005.
- [73] M. Cooper and J. Foote, “Summarizing popular music via structural similarity analysis,” in *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2003, pp. 127–130.
- [74] J. Foote, “Visualizing music and audio using self-similarity,” in *Proceedings of the 7th ACM International Conference on Multimedia*, ser. MULTIMEDIA '99. New York, NY, USA: ACM, 1999, pp. 77–80. [Online]. Available: <http://doi.acm.org/10.1145/319463.319472>
- [75] —, “Automatic audio segmentation using a measure of audio novelty,” in *Proceedings of the 2000 IEEE International Conference on Multimedia and Expo*, vol. 1, Aug. 2000, pp. 452–455.

- [76] J. Salamon, J. Serra, and E. Gómez, “Tonal representations for music retrieval: from version identification to query-by-humming,” *International Journal on Multimedia Information Retrieval*, vol. 2, no. 1, pp. 45–58, 2013.
- [77] E. Molina, A. M. Barbancho, L. J. Tardón, and I. Barbancho, “Evaluation framework for automatic singing transcription.” in *Proceedings of the 2014 International Symposium on Music Information Retrieval*, 2014, pp. 567–572.
- [78] T. D. Mulder, J. P. Martens, M. Lesaffre, M. Leman, B. D. Baets, and H. D. Meyer, “Recent improvements of an auditory model based front-end for the transcription of vocal queries,” in *Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, May 2004, pp. iv–257–iv–260 vol.4.
- [79] M. Mauch, C. Cannam, R. Bittner, G. Fazekas, J. Salamon, J. Dai, J. Bello, and S. Dixon, “Computer-aided melody note transcription using the tony software: Accuracy and efficiency,” in *Proceedings of the 1st International Conference on Technologies for Music Notation and Representation*, 2015, pp. 23–30.
- [80] J. Sundberg, “Acoustic and psychoacoustic aspects of vocal vibrato,” *Vibrato*, pp. 35–62, 1995.
- [81] S. Böck and G. Widmer, “Maximum filter vibrato suppression for onset detection,” in *Proceedings of the 16th International Conference on Digital Audio Effects (DAFx)*, 2013.

- [82] “IPA Chart,” International Phonetic Association, 2015. [Online]. Available: <http://www.internationalphoneticassociation.org/content/ipa-chart/>
- [83] L. Burget, P. Schwarz, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. Goel, M. Karafiát, D. Povey, A. Rastrow, R. C. Rose, and S. Thomas, “Multilingual acoustic modeling for speech recognition based on subspace gaussian mixture models,” in *Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2010, pp. 4334–4337.
- [84] T. Schultz and A. Waibel, “Language-independent and language-adaptive acoustic modeling for speech recognition,” *Speech Communication*, vol. 35, no. 1, pp. 31–51, 2001.
- [85] S.-H. Hong, “Gradient vowel cooccurrence restrictions in monomorphemic native Korean roots,” *음성음운형태론연구*, vol. 16, no. 2, pp. 279–295, 2010.
- [86] A. C. Baker, “Two statistical approaches to finding vowel harmony,” Cite-seer, Tech. Rep., 2009.
- [87] J. Goldsmith and J. Riggle, “Information theoretic approaches to phonological structure: the case of Finnish vowel harmony,” *Natural Language and Linguistic Theory*, vol. 30, no. 3, pp. 859–896, 2012.
- [88] T. Abe and M. Honda, “Sinusoidal model based on instantaneous frequency attractors,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1292–1300, 2006.

- [89] J. L. Flanagan and R. Golden, "Phase vocoder," *Bell Labs Technical Journal*, vol. 45, no. 9, pp. 1493–1509, 1966.

## 초 록

음악정보검색 분야의 가장 오래된 문제 중 하나인 자동 음악 전사는 오디오 신호로부터 음표 등의 음악적 속성을 기호화된 수준으로 자동적으로 추출하는 기술을 의미한다. 악보 등의 형태로 변환된 음악 데이터는 소비자와 창작자 모두에게 보다 고차원적인 정보를 제공하여 음악 교육, 제작 및 편집 등의 목적으로 다양하게 활용될 수 있다. 그 중 노래하는 목소리는 가장 쉽게 연주할 수 있고 또 일상적으로 들을 수 있는 음악 신호이지만, 음색의 불균일성과 복잡한 신호적 특징으로 인하여 일반적인 악기들에 비해 기존의 전사 기법 적용에 어려움이 존재한다. 본 논문의 궁극적인 목표는 이러한 노래 신호의 특성을 고려하여 최고 성능을 넘는 자동화된 노래 전사 시스템을 개발하는 것이다. 이를 위하여 우리는 음악 전사의 요소 기술들인 음고 추적과 시작점 검출에 대한 기존의 접근들을 조사하고, 각 기술들의 성능을 개선하는 방법을 제안한다. 본 논문은 음고 추적의 측면에서는 시계열 데이터의 부분 샘플링이 음악 신호의 주기성 분석 성능 향상에 미치는 영향을 살펴보고, 시작점 검출의 측면에서는 칩스트럼 분석과 비지도 분류 기법 등을 이용하여 배음 구조의 국지적 균질성을 검출에 활용한다. 최종 전사 시스템은 이에 필요한 배음 구조의 특징 벡터화 기법과 확률 모형, 그리고 음의 전이를 표현한 은닉 마르코프 모형 등을 다루며, 음 수준의 성능 평가에서 82%의 최고 성능을 보인다.

**주요어:** 자동 음악 전사, 음악 정보 분석, 음고 추적, 시작점 검출, 노래, 배음 구조  
**학 번:** 2011-31243