UNIVERSITAT POMPEU FABRA

Computational Analysis of Expressivity in Classical Guitar Performances

Tan Hakan Özaslan

TESI DOCTORAL UPF / 2013

Director de la tesi:

Dr. Josep Lluis Arcos
Artificial Intelligence Research Institute (IIIA-CSIC)

Prof. Xavier Serra i Casals
Dept. of Information and Communication Technologies
Universitat Pompeu Fabra, Barcelona, Spain

Dissertation submitted to the Deptartment of Information and Communication Technologies of Universitat Pompeu Fabra in partial fulfillment of the requirements for the degree of

DOCTOR PER LA UNIVERSITAT POMPEU FABRA,

Music Technology Group (http://mtg.upf.edu), Dept. of Information and Communication Technologies (http://www.upf.edu/dtic), Universitat Pompeu Fabra (http://www.upf.edu), Barcelona, Spain.

Anneme

Acknowledgements

For the first and foremost I wish to express my appreciation and gratitude to Josep Lluis Arcos who was my primary guide and source of motivation all through the M.Sc. and Ph.D.
I would like to thank all of the Music Technology Group and specially Xavier Serra giving me the opportunity to collaborate with MTG people.
It is also important to recognize all recent and former IIIA people. Specially Sergio Manzano for his delightful discussions, Albert Vilamala, Isaac Cano, Atılım Güneş Baydın and Joan Serrà for his insights and guidance. I would like to acknowledge Hendrick Purwins. I doubt he could remember but five years ago he was the one who inspired me about machine learning and showed me complex structures can be intuitive. I wish to thank Rafeal Ramirez, Barış Bozkurt and all the GuitarLab people, Enric Guaus, Eric Palacios, Lars Fabig. I would like to mention my spring 2012 office friends in MTG, Sankalp Gulati, Gopala Koduri, Mohamed Sordo and specially Sertan Şentürk (will miss your jokes).
And more people have been important to me, as friends by being there, Onur Özdemir, Hakan Karaoğuz, Tahir Öztürk and Bengi Öztürk. Also I would like to thank my new friend Eren, for his warm smile.
Last but not least, I wish to thank to my mother Nejla Özaslan and my sister Ayça Ceylan Özaslan, without them I can not even imagine going this far.

This thesis has been carried out at the Artificial Intelligence Research Institute of Spanish Research Council, (IIIA-CSIC) and Information and Communication Technologies Department of Universitat Pompeu Fabra (UPF) in Barcelona, Spain from Sep. 2009 to Sep. 2013. This work has been supported by Spanish projects NEXT-CBR (TIN2009-13692-C03-01), IL4LTS (CSIC-200450E557), European Commission PRAISE project (ICT-2011-8-318770), and by the Generalitat de Catalunya under the grant 2009-SGR-1434.

Abstract

The study of musical expressivity is an active field in sound and music computing. The research interest comes from different motivations: to understand or model musical expressivity; to identify the expressive resources that characterize an instrument, musical genre, or performer; or to build synthesis systems able to play expressively. To tackle this broad problem, researchers focus on specific instruments and/or musical styles. Hence, in this thesis we focused on the analysis of the expressivity in classical guitar and our aim is to model the use of expressive resources of the instrument.
The foundations of all the methods used in this dissertation are based on techniques from the fields of information retrieval, machine learning, and signal processing. We combine several state of the art analysis algorithms in order to deal with modeling the use of the expressive resources.
Classical guitar is an instrument characterized by the diversity of its timbral possibilities. Professional guitarists are able to convey a lot of nuances when playing a musical piece. This specific characteristic of classical guitar makes the expressive analysis is a challenging task.
The research conducted focuses on two different issues related to musical expressivity. First, it proposes a tool able to automatically identify expressive resources such as legato, glissando, and vibrato, in commercial guitar recordings.
Second, we conducted a comprehensive analysis of timing deviations in classical guitar. Timing variations are perhaps the most important ones: they are fundamental for expressive performance and a key ingredient for conferring a human-like quality to machine-based music renditions. However, the nature of such variations is still an open research question, with diverse theories that indicate a multi-dimensional phenomenon. Our system exploits feature extraction and machine learning techniques. Classification accuracies show that timing deviations are accurate predictors of the corresponding piece.
To sum up, this dissertation contributes to the field of expressive analysis by providing, an automatic expressive articulation model and a musical piece prediction system by using timing deviations. Most importantly, it analyzes the behavior of proposed models by using commercial recordings.

Resum

L'estudi de l'expressivitat musical és un camp molt actiu en la computació musical. El seu interès ve donat per diverses motivacions: entendre i modelitzar l'expressivitat musical; identificar els recursos expressius que caracteritzen un instrument, gènere musical o intèrpret; i construir sistemes de síntesi amb la capacitat de reproduir música expresivament. Per abordar aquest problema tan ampli, la literatura existent tendeix a focalitzar-se en instruments o gèneres musicals concrets. En aquesta tesi, ens hem focalitzat en l'anàlisi de la expressivitat en la guitarra clàssica y el nostre objectiu serà modelitzar l'ús de recursos expressius en aquest instrument.

Els fonaments de tots els mètodes utilitzats en aquesta tesi estan basats en tècniques de búsqueda y recuperació de la informació, aprenentatge automàtic y processament del senyal. Concretament, combinem diversos algorismes de l'estat de l'art per fer una proposta de caracterització de l'ús dels recursos expressius.

La guitarra clàssica és un instrument que es caracteritza per la diversitat de les seves possibilitats tímbriques. Els guitarristes professionals són capaços de transmetre molts matisos durant la interpretació d'una peça musical. Aquesta característica específica de la guitarra clàssica fa que l'anàlisi d'aquest instrument sigui una tasca difícil.

Dividim el nostre anàlisi en dues línies de treball principals. La primera línia proposa una eina capaç d'identificar automàticament recursos expressius en el context d'una gravació comercial. Construim un model amb l'objectiu d'analitzar i extreure automàticament els tres recursos expressius més utilitzats: legato, glissando i vibrato. La segona línia proposa un anàlisi integral de desviacions de tempo en la guitarra clàssica. De les variacions, potser les més importants siguin les variacions de tempo: són fonamentals per a la interpretació expressiva i un ingredient clau per conferir una qualitat humana a interpretacions basades en ordinador. No obstant, la naturalesa d'aquestes variacions és encara un problema d'investigació que no ha estat resolt, amb diverses teories que apunten a un fenòmen multi-dimensional. El nostre sistema utilitza tècniques d'extracció de característiques i aprenentatge automàtic. La precisió de la classificació mostra que les desviacions de tempo són predictors precisos de la peça musical corresponent.

Para recapitular, aquesta tesi contribueix al camp de l'anàlisi expressiu proveint un model automàtic d'articulació expressiva i un sistema predictor de peces musicals que analitza les desviacions de tempo. Finalment, aquesta tesi analitza el comportament dels models proposats utilitzant gravacions comercials.

Resumen

El estudio de la expresividad musical es un campo muy activo en la computación musical. El interés en investigar ésta área tiene distintas motivaciones: entender y modelar la expresividad musical; identificar los recursos expresivos que caracterizan un instrumento, género musical, o intérprete; y construir sistemas de síntesis con la capacidad de reproducir música expresivamente. Para abordar este problema tan amplio, la literatura existente tiende a enfocarse en instrumentos o géneros musicales específicos. En esta tesis nos enfocaremos en el análisis de la expresividad en la guitarra clásica y nuestro objetivo será modelar el uso de recursos expresivos en este instrumento.

Los fundamentos de todos los métodos usados en esta tesis están basados en técnicas de búsqueda y recuperación de la información, aprendizaje automático y procesamiento de señales. Combinamos varios algoritmos del estado del arte para lidiar con el modelado del uso de los recursos expresivos.

La guitarra clásica es un instrumento que se caracteriza por la diversidad de sus posibilidades tímbricas. Los guitarristas profesionales son capaces de transmitir muchos matices durante la interpretación de una pieza musical. Esta característica específica de la guitarra clásica hace que el análisis de este instrumento sea una tarea difícil.

Dividimos nuestro análisis en dos líneas de trabajo principales. La primera línea propone una herramienta capaz de identificar automáticamente recursos expresivos en el contexto de una grabación comercial. Construimos un modelo con el objetivo de analizar y extraer automáticamente los tres recursos expresivos más utilizados: legato, glissando y vibrato. La segunda línea propone un análisis integral de desviaciones de tiempo en la guitarra clásica. De las variaciones, quizás las más importantes sean las variaciones de tiempo: son fundamentales para la interpretación expresiva y un ingrediente clave para conferir una cualidad humana a interpretaciones basadas en ordenador. No obstante, la naturaleza de tales variaciones es aún un problema de investigación que no ha sido resuelto, con diversas teorías que apuntan a un fenómeno multi-dimensional. Nuestro sistema utiliza técnicas de extracción de características y aprendizaje de automático. La precisión de la clasificación muestra que las desviaciones de tiempo son predictores precisos de la pieza musical correspondiente.

Para recapitular, esta tesis contribuye al campo del análisis expresivo proveyendo un modelo automático de articulación expresiva y un sistema predictor de piezas musicales que emplea desviaciones de tiempo. Finalmente, esta tesis analiza el comportamiento de los modelos propuestos utilizando grabaciones comerciales.

Contents

List of figures

List of tables

"If I had to live my life again, I would have made a rule to read some poetry and listen to some music at least once every week. The loss of these tastes is a loss of happiness, and may possibly be injurious to the intellect, and more probably to the moral character, by enfeebling the emotional part of our nature."

- Charles Darwin

CHAPTER 1

Introduction

"Music is what feelings sound like"

- Anonymous

Motivation

Music is a complex human activity where multiple purposes are involved. First, it is a communication act where some actors, the composer and the performers, use specific tools, musical instruments, to transmit a message to an expected audience. Music can be considered as an activity where the cultural and social context play a preeminent importance. Moreover, music is one of the most important human ways to express and convey feelings and emotions.

In western music, composers use the score as the way to codify and prescribe the instructions to the performers. However, a musical score only codifies partially the way performers must play. There are many of these instructions that are implicit and that come from different origins such as the type of music that is played, the instruments used, the piece interpretation of the performers, or the emotional intention of the performers. Many of these implicit aspects of music are known as musical expressivity.

Thus, designing computational tools able to help in this complex human activity require of the understanding and modeling of these "*implicit*" knowledge. There are several studies focused on the understanding of musical expressivity from the analysis of written compositions in the field of musicology and music theory (Copland, 1939; Dahlhaus, 1989; Levy, 1970; Meyer, 1989). However, computational expressive analysis of music is relatively a new field and there are several questions that are waiting for their answers. In this work, we are providing new computational tools to analyze the expressive resources used in classical guitar.

1.1 Musical Expressivity

People are using score to compose and play music for many years, but the music that we hear when musicians play includes much more elements than the written in the score. Then, scores codify only a small portion of the musical experience (Saunders et al., 2004). For instance, if we convert the written score into an audio form using a computer and listen the result, we will realize that there is a lack of a lot of features such as phrase emphasis, lengthening and shortening of notes, vibratos, glissandos, etc.
Related to this, we often say that a performer plays with great expression. But, what exactly do we mean? What are the decisions a performer takes when playing? Understanding these decisions is a broad research area in Music Information Retrieval (MIR) community and the study of musical expressivity is still an active field. The research interest comes from different motivations:

- Understanding and/or modeling musical expressivity.
- Identifing the expressive resources that characterize an instrument, musical genre, or performer.
- Building synthesis systems able to play expressively.

When different performers play the same score, the result is never the same and the output is never a mechanical rendering of the score. Moreover, even if the performer tries to play mechanically, the output has noticeable differences and variances (Begston & Gabrielson, 1980). Furthermore, different performances of the same piece played by the same performer have noticeable differences (Henderson, 1937). Thus, the study of this phenomenon has a high research interest to understand the foundations of musical expressivity.
Expressivity can be described as the differences (deviations) between a musical score and its execution. According to the literature, these deviations are mainly motivated by two purposes: to clarify the musical structure (Gabrielsson, 1987; Palmer, 1996; Sloboda, 1983) and as a way to communicate affective content (Gabrielsson, 1995; Juslin, 2001; Lindström, 1992). Moreover, these expressive deviations vary depending on the musical genre, the instrument, and the performer. Specifically, each performer has his/her own unique way to add expressivity by using the instrument.
Gracthen (Grachten, 2006) analyzed different perspectives of older studies and classified the definition of expressivity in to three main perspectives.

1. According to Repp, expressivity is everything that is not specified in the score and named this definition as **Expressivity as Microstructure** (Repp, 1990). According to this definition score only represents the macro elements of the music and does not give clue about the micro elements.

2. Another view conceive expressivity as the deviations from musical score (Gabrielsson, 1987) and named as **Expressivity as Deviation from the Score**. Different from the Expressivity as Microstructure, in this definition expressivity is defined as the differences between the facts that are defined in the written score in terms of, pitch, timbre, timing and dynamics.

3. Last category of expressivity definition that Gracthen gave place in his dissertation is **Expressivity as Deviation within the Performance** (Timmers & Honing, 2002), which defines expressivity as a hierarchical description of the musical piece. For example according to this definition a beat duration can be related to the deviation of the duration of the enveloping bar.

In the scope of these three different perspectives, what is clear is that expressivity is a complex phenomenon that can be studied from different approaches. Our position regarding expressivity is that is a a combination all of above. As we will discuss in Chapter 3, although some expressive articulations are explicitly marked in the score, the addition of new ones is common phenomenon. Furthermore, score does not give any clue how an expressive articulation should be executed. This fact falls into the *Expressivity as Microstructure* definition. In Chapter 4, we analyze the onset deviations from the score reference. We will see that there are places in the score where performers tend to apply the same kind of deviations whereas there are other places where they do not agree. These findings fall into *Expressivity as Deviation from the Score* and *Expressivity as Deviation within the Performance*.

1.2 Guitar

In our research guitar is selected as the instrument of the study because it is one of the most popular instruments in western music, most genres include guitars (such as classical, folk, flamenco, jazz) and is played in many cultures. Although plucked instruments and guitar synthesis have been studied extensively (Erkut et al., 2000; Janosy et al., 1994; Laurson et al., 1999; Norton, 2008; Traube & Depalle, 2003), expressive analysis from real guitar recordings has not been fully tackled.

1.3 Objectives and Problem Definition

Our objective is to propose new methods in order to analyze musical expressivity with a focus on the classical guitar. Specifically, we focused our research on expressive articulations and timing deviations, which we believe are the most important building blocks of guitar expressivity.

1.3.1 Expressive Articulations

In music, an expressive articulation is the manner in which the performer applies her technique to enrich the sounds or notes such as, vibrato, glissando or legato. Expressive articulations are often described rather than quantified. Furthermore, in some cases they are not even described in the score. In both cases there is room to interpret how to execute and where to execute each expressive articulation. Therefore, expressive articulations are one of the key factors that makes each performance unique. Analysis of expressive articulations is curial to understand the musical expressivity. The general description of expressive articulation in music that is agreed by several different authors (Blatter, 1980; Chew, 2008; Duncan, 1980; Norton, 2008) is:

> *"The difference between the notation of music and its actual performance".*

More specifically for the guitar expressive articulation stands for:

> *"The term articulation refers in guitar music, to the manner in which tones are attacked and released and is distinct from phrasing, which pertains more to how they are grouped for expressive purpose" (Duncan, 1980).*

Supporting the definition of Duncan, as a concrete example of how an expressive articulation should be used in order to enrich the timbral performance of the guitar, Aguado defines vibrato as:

> *"The left hand can prolong the sound using vibrato. If after a string is stopped sufficiently firmly it is plucked and the finger holding it down moves immediately from side to side at the point of pressure, the vibration of the string and, consequently, its sound is prolonged; but the finger must move as soon as the string has been plucked in order to take advantage of the first vibrations which are the strongest, maintaining at least the same degree of pressure on the string. These movements should not be too pronounced nor involve the left arm, only the wrist.*
>
> *A successful vibrato does not depend so much on the amount of the pressure as on how it is applied. The last joint must be pressed perpendicularly down on the string, parallel to the frets, ensuring that the weight of the hand on the point of pressure, offset by the thumb behind, maintains and prolongs the vibrations better than the excess pressure which may result if the arm is used." (Aguado, 1981)*

In guitar playing, both hands are used. Some guitarists use the right hand to pluck the strings whereas others use the left hand. For the sake of simplicity,

in the rest of the dissertation we consider the hand that plucks the strings as the right hand and the hand that presses the frets as the left hand.
Moreover, in the guitar strings can be plucked using a single plectrum called a flat-pick or by directly using the tips of the fingers. The hand that presses the frets is mainly determining the notes while the hand that plucks the strings is mainly determining the note onsets and timbral properties. However, left hand is also involved in the creation of a note onset or different expressive articulations like legatos, glissandos, and vibratos.
In this thesis, we focus on three most-widely applied expressive guitar articulations, legato, glissando and vibrato. Although all three of them share common analysis methodologies, (see Chapter 3), they need significantly different approaches for further extraction and classification purposes.

1.3.2 Onset Deviations

Timing variations are perhaps the most important ones: they are fundamental for expressive performance and a key ingredient for conferring a human-like quality to machine-based music renditions. However, the nature of such variations is still an open research question, with alternative and sometimes contrasting theories that indicate a multi-dimensional phenomenon. We bring a complementary view to the origin of timing variations, showing that sequences of note onset deviations are robust and reliable predictors of the musical piece being played, irrespective of the performer. In our study we are analyzing well-known pieces from different eras of music. Moreover, we are analyzing commercial recordings of famous virtuoso guitarists. In our approach, we take 10 different interpretations of 10 different pieces. By using machine learning principles we formulate our study as a classification problem. We suggest that onset deviations are reliable predictors of the musical pieces.

1.4 Outline of the Thesis

This thesis is structured as follows: Chapter 2, briefly talks about the current state of the art. In Chapter 2, we choose to mention about the references that has the most impact in the field of guitar and expressivity. We leave the more detailed (specific) references to the corresponding chapters. The main contribution of the thesis is described in Chapters 3, and 4. In the first part of Chapter 3, Section 3.2, we describe our feature extraction methodology that will be also exploited in Chapter 4. In the rest of Chapter 3, we explain our methodology for extraction of expressive articulations and for their classification. Chapter 4 presents our approach and experiments for the extraction and analysis of onset deviations from commercial audio files. Finally, Chapter 5 discusses open issues and future work, as well as summarizes the contributions of our research.

CHAPTER 2

State of the Art

General View

Musical expressivity rely on the intentional manipulation of sound properties such as pitch, timbre, dynamics, or timing (Gabrielsson, 1999, 2003). Studies on musical expressivity go back to the early twentieth century. In 1913, Johnstone (1913) analyzed piano performers. Johnstone's analysis can be considered as one of the first studies focusing on musical expressivity. Recently, advances in audio processing techniques risen the opportunity to analyze audio recordings in a finer level (see Gouyon et al. (2008) for an overview). Up to now, there are several studies focused on the analysis of expressivity of different instruments (Bresin & Battel, 2000; Dobrian & Koppelman, 2006; Grachten, 2006; Norton, 2008; Solis et al., 2007). Although the instruments analyzed differ, most of them focus on analyzing monophonic or single instrument recordings. For instance in his dissertation Norton, focused on jazz saxophone, Grachten analyzed monophonic violin recordings and Bresin & Battel focused on the articulation analysis of piano performances.

Musical expressivity has been studied from different perspectives. Although each view has intersections with others and does not exist crisp borders, with our best intention, we have tried to sum up studies related on musical expressivity in four main research areas: Physical Modeling, Symbolic Analysis, Gestural Analysis, and Expressive Performance Analysis. Additionally, one of the fields that have contributed to the study of musical expressivity is Artificial Intelligence. Interested readers can find a complete survey of computer music systems based on Artificial Intelligence techniques in (de Mantaras & Arcos, 2002). Other more recent studies combine machine learning techniques (Giraldo & Ramirez, 2012; Grachten et al., 2006; López de Mántaras & Arcos, 2012; Ramirez & Hazan, 2005) to model expressive jazz performances and violin performance (Ramirez et al., 2010).

To introduce the studies having direct or indirect connection with musical expressivity, we have follow a chronological order. Each section has connections with each other, for instance symbolic analysis studies have models which are

similar with physical modeling studies, or, since each study has connection with expressivity, they could be thought as an expressive performance analysis. However, we have organized the chapter presenting first the existing literature in three main related studies such as physical modeling, gesture analysis, and symbolic analysis. These approaches have been then usually used as low-level models for the study of expressive performance analysis, the last main section of this chapter.

2.1 Physical Modeling

One way to study musical expressivity is by means of the analysis and modeling of the instruments and the physical laws associated to the generation of the sound. That is, by indirectly extract musicians intentions from modeling their instruments (e.g. resonance of the body, strings) or elements used by the performers (such as the violin bow). In this scope, a lot of work was done in the early 1980's on modeling the string behavior. Karplus & Strong (1983) presented a model of string vibrations based on their physical behavior which was the basis for further work by Jaffe D. & Smith J. (1983). For deeper knowledge we refer to McKay (2003) which is a detailed survey about classical guitar string modeling between 1983 and 2001.

Researches on string analysis constructed a strong base for further and more complete works which include the strings, the body, the finger action, and the interactions between them (Cuzzucoli & Lombardo, 1999). The models of Cuzzucoli & Lombardo include parameters that can influence the tonal response, such as the body, the string characteristics, and the static and dynamic parameters of fingers. In order to simulate the behavior of the instrument quantitatively, Cuzzucoli & Lombardo evaluated a set of plausible values for the instrument characteristics and for the finger parameters through a number of simple experiments. Their system takes as input, the physical parameters of each string, the resonator and the finger, together with a description of a musical score that includes each position of the note on the guitar fret board. The final output of the model is a sound file, which helps in evaluating the effectiveness of the model.

Same year Laurson et al. (1999) developed a new notation package called Expressive Notation Package (ENP) to control the model-based synthesis engine. ENP is written in Lisp and CLOS (Common Lisp Object System), Figure 2.1. A real-time synthesis engine has been developed based on earlier results in digital waveguide modeling. Laurson et al. also described an analysis of vibrato in acoustic guitar tones, which provides control information for realistic synthesis.

Different from other instruments, such as piano, in guitar to produce sound, player can pluck different places of the string (Traube & Depalle, 2003). Thus, different timbres can be produced from the exact same guitar. Orio (1999),

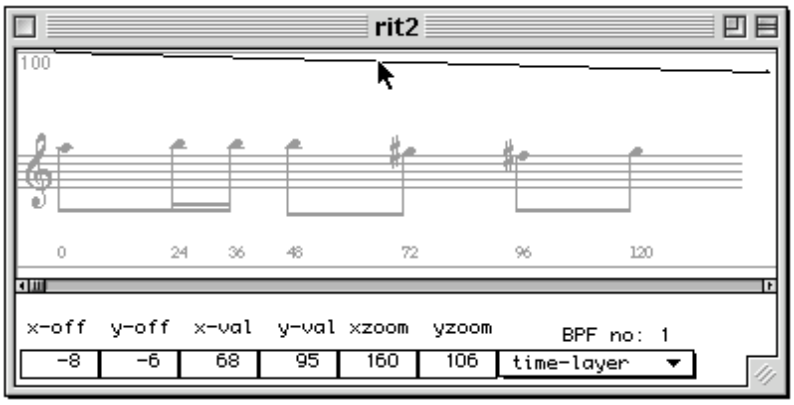


Figure 2.1: A view from Expressive Notation Package, ENP - Tempo editing. The line on the top is for modifying the tempo. 100 means at that point tempo starts from 100% of the real tempo.

conducted a research in order to analyze the timbre space of the classical guitar and its relationship with the different plucking techniques. The analysis were done with 28 audio samples which were played by hand(using fingers or nails). In each sample, player changed playing position on the string from 12^{th} fret to the bridge[1]. In each playing position player changed;

- Inclination between the finger and string.
- Inclination between the hand and string.
- The degree of the relaxation of the plucking finger.

After the recording sessions, two analyzes applied to recorded samples;

1. **Time-frequency analysis :** Each sample is divided in to $100ms$ samples beginning from the onset, in order to compare and see the evaluation of the sample. They divided this analysis in to two measure of the harmonics and attack time. The author concluded that in the case of inclination (both finger and hand), normal position has a high pass effect. In normal position attack time is also longer.

2. **Psychoacoustical analysis :** For this analysis two features were extracted from audio sample; Center of Gravity of the Spectrum (CGS) and Irregularity of the Spectrum (IRR). CGS has a linear when the finger moves along the string and has a symmetric trend for the inclinations. For the IRR, the relevance is when the finger is moved along the string.

[1] very end of the guitar body where the strings are attached

Orio has enlightened the studies that focused on the detection of the plucking point of the guitar string, (Traube & Depalle, 2003). In guitar the plucking point is a key factor for controlling the timbre. The plucking point of a guitar string affects the sound envelope and influences the timbral characteristics of notes, Figure 2.2. For instance, plucking close to the guitar hole produces more mellow and sustained sounds where plucking near to the bridge (end of the guitar body) produces sharper and less sustained sounds. Traube & Depalle proposed a method for estimating the plucking point on a guitar string by using a frequency-domain technique applied to acoustically recorded signals. They also proposed an original method to detect the fingering point, based on the plucking point information.

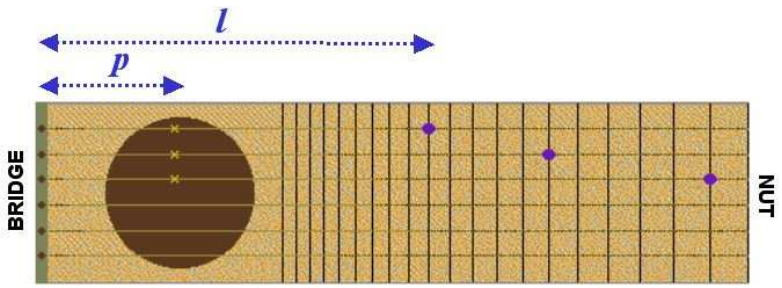


Figure 2.2: Plucking point distance representation of the guitar.

The guitar sound is complex as other instruments. In order to artificially produce the guitar sound, nonlinearities should be examined (Bader, 2005). According to Bader, there are three main types of nonlinearities; the coupling of transversal and longitudinal vibrations in the guitar body, the complicated guitar geometry and the coupling of the strings to the plate. In his research all three factors were examined.

More recently, Lee N. & Smith J. (2007) proposed a new method for extraction of the excitation point of an acoustic guitar signal. Before explaining the method, three state of the art techniques were examined in order to compare with the new one. The techniques analyzed were *matrix pencil inverse-filtering*, *sinusoids plus noise inverse-filtering*, and *magnitude spectrum smoothing*. After describing and comparing these three techniques, Lee N. & Smith J. proposed a new method in order to extract the excitation point of a acoustic guitar signal. The proposed method, *statistical spectral interpolation*, is based on removal of spectral peaks, followed by statistical interpolation to reconstruct the excitation spectrum in frequency intervals occluded by partial overtones. After comparing four methods, results show that the method presented outperforms previous methods in removing tonal components in the resulting excitation signal while maintaining a noise-burst like quality.

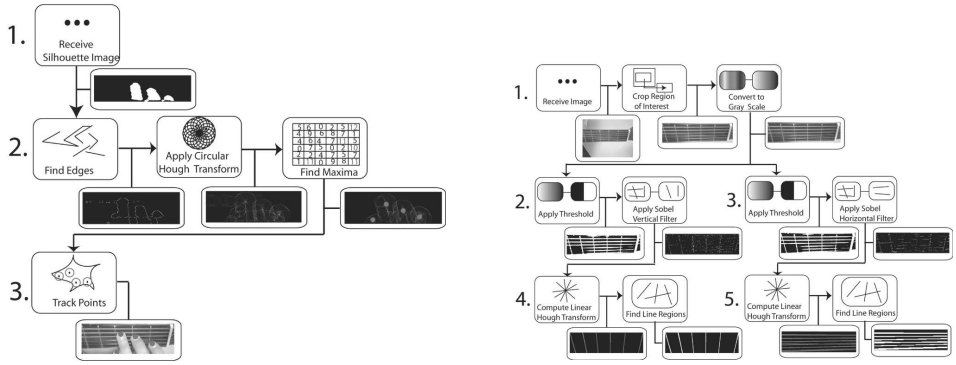
Research on physical modeling is very relevant for the analysis of musical ex-

pressivity. First, it provides clues about the constraints and capabilities of each instrument, i.e. the resources musicians have to convey musical expression and emotions. Second, it helps to bring closer low-level sound control phenomena with the high level musical language, the knowledge level used by humans when talk about the way to perform/understand music performances.

2.2 Gestural Analysis

Musical expressivity is directly related to the gestures performed by the musicians when playing. These gestures are composed of several parallel movements occurring a very diverse time scale. Specifically, these gestures may range from the body movements of performer, and possible also movements of the instrument, to micro gestures such as the fingering stress while producing a sound or the choice of fingers position to play a guitar a chord. All these elements have important effects on the generation and control of the sounds. Thus, they have important effects on timing and timbral characteristics of the played music. Therefore, many of these gestural movements fall into the definition of expressivity as deviation from the Score, Section 1.1.
There have been several studies regarding the gestural information from different aspects. Gestural information related to guitarists may refer from analysis of the general movements of the guitar body (Quested et al., 2008) to the detailed study of specific finger movements (Burns & Wanderley, 2006; Guaus et al., 2010; Radicioni & Lombardo, 2007). Moreover there are studies that combine gestural and audio analysis techniques (Heijink & Meulenbroek, 2002). In the study of Heijink & Meulenbroek, audio and camera recordings of six professional guitarists playing the same song were used to study the complexity of the left hand fingering of classical guitar from the perspective of behavioral point of view. In their music collection for the experiments, different guitarists have performed the same song. They state that *"The problem of finding a suitable left-hand fingering for a note sequence is closely related to the inverse kinematics problem that individuals continuously and effortlessly solve in everyday motor tasks such as pointing, reaching, and grasping."* Heijink & Meulenbroek have briefly reviewed the different techniques of finding the optimal fingering positions. Among these different techniques, authors have focused on as they called, *joint coupling and intrinsic movement dynamics*. They were trying to find optimal places and fingerings for the notes. Several constraints were introduced to calculate cost functions such as; *minimization of jerk*, *torque change*, *muscle-tension change*, *work*, *energy* and *neuromotor variance*. As a result of the study, they found a significant effect on timing. However, authors stated that, "while calculating the fingering of the left hand not only the distance factor is considered but also acoustic dimensions should be taken into account". But since in their study they were using simple scales, they did not consider the acoustic dimensions, they have only focused on the

left-hand fingering distance cost functions.
Burns & Wanderley (2006) proposed a method to visually detect and recognize fingering gestures of the left hand of a guitarist by using affordable camera. Burns & Wanderley conducted studies about the three important aspects of a complete fingering system. First one was the finger tracking, second one was, strings and frets detection, and the last one was the movement segmentation. They used Hough Transform Theory in order to detect the position of the finger tips, strings and frets Figure 2.3.



(a) Fingertips detection **(b)** String and fret detection

Figure 2.3: Burns' Algorithms

In guitar the same note can be played in different positions of the fretboard. Finding the proper fingering position and the transition to the next note is an ability that is gained by practice. Radicioni & Lombardo (2007) calculated the weights of the finger transitions between finger positions by using the weights of Heijink & Meulenbroek (2002). Furthermore, they defined different structures for the explanation of their model such as; **MEL** as melody block, **CHO** as chord block, **HN** as held note and **MIX** as mix of MEL, CHO or HN, Figure 2.4. Authors also defined; *comfortable span* as, two fingers can press their respective positions with minimum effort, and this effort must be valid for both vertical and horizontal movements. Also the model computed the finger strengths. Briefly, Radicioni & Lombardo (2007) calculated the weight of two finger transposition. They were using three different variables for calculating the general weight; finger movement, finger strength and neck anatomy of the guitar. After calculating the all possible distances, directed acyclic graph (DAG) was used for finding the possible minimum distance, starting from the first note to the last note. They called this technique, *relaxation*. As authors stated, for the MEL blocks, finding the shortest path was relatively easy. However for the CHO and MIX block they have needed constraint based approach.
Different from the left-hand fingering analysis studies (Burns & Wanderley, 2006; Guaus et al., 2010; Heijink & Meulenbroek, 2002; Radicioni & Lom-

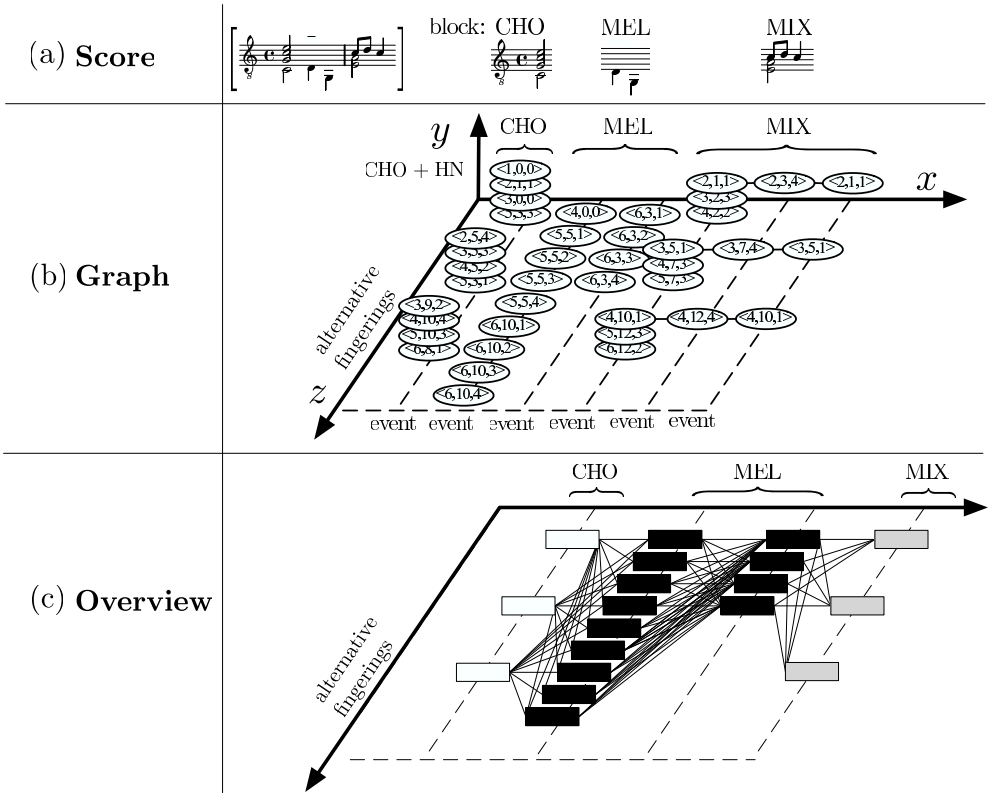
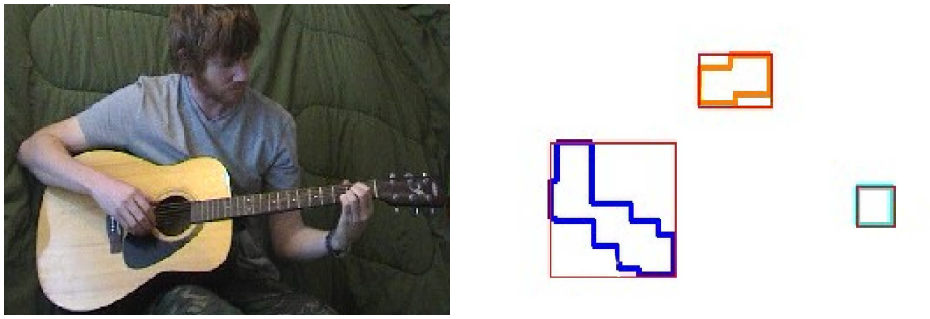


Figure 2.4: Plucking point distance representation of the guitar.

bardo, 2007; Radisavljevic & Driessen, 2004), Trajano et al. (2004) focused on right hand. The approach that is used is similar to left-hand analysis. They represented each chord as a 6-tuple with the finger names such as (x, T, F, M, R, x). They used two cost function transition and application which were highly similar to study of Radisavljevic & Driessen (2004). By using these definitions and cost functions, optimal way of chord transition is calculated.
Quested et al. (2008) focused on note recognition of acoustical guitar. They used a camera pointed at the performer rather than the mounted on the instrument. At the end they achieved to detect the performer location, guitar neck location and fretting hand/finger location from the camera view. Results can be seen on Figure 2.5 .
They were using non-negative matrix factorization.

- **Performer Location :** Musician was located by using skin color and movement detection techniques. For skin detection Gaussian Mixture Models were used. Also Stauffer Grimson adaptive mixture model were used in order to eliminate back ground.



(a) String and fret detection **(b)** Fingertips detection

Figure 2.5: Segmentation of face and hands using fore- ground detection and skin colour detection

- **Guitar Neck Location :** Neck position was assumed between the hands of the guitarist. Line detection techniques were used. Also Luthiers *seventeen rule* was used in order to determine the bridge position.

- **Fretting Hand/Finger Location :** Skin recognition as described in Performer Location section was good but not enough for accurate contour. **Expectation Maximization** was used to update **GMM**. Once the neck was located then within this area, skin pixels were searched and after determining, it is cropped. This search is lies on RGB clusters.

One complete study about the gestural analysis of guitar performances is the dissertation of Norton (Norton, 2008). His aim was to identify the gestures required for the generation of each guitar sound and their analysis. Norton used a motion capture system in order to capture the gestures. Specifically, 30 markers were placed on specific areas of the player, which were thumb, fingers, wrist joints of the both hands, the elbows and the shoulders. Then, the guitarist was asked to repeat each articulation four times at three different separate dynamic levels. Norton contributions are very interesting to catalog and characterize guitar expressive resources, but his work did not analyzed their use in real musical pieces.

2.3 Symbolic Analysis

In guitar same note, and even the same chord, can be played in different positions of the fretboard. Unless cited the explicit position, it is the personal choice of the performer. As we explained in Section 2.1, together with the plucking point, playing the same note at different places of the fretboard may have important effects on the timbral characteristics of the note. Therefore, the choice of the note places can be clearly considered as musical expressive choices.

As an example, when designing systems for musical synthesis, determining the finger places of the notes can help a lot in generating the appropriate sound nuances.

In addition to standard notation, guitarist also use tablature notation. Tablature is a form of musical notation indicating instrument fingering rather than musical pitches (see Figure 2.6). Tablature notation is mainly used in string instruments. Each string of the instrument is represented with a line and synchronously note positions are marked with the fret numbers.

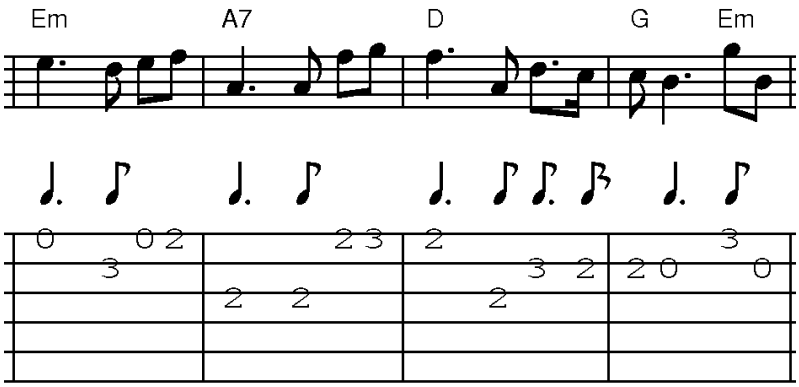


Figure 2.6: Each line on the bottom of the standard notation corresponds to a string in the guitar and each number corresponds to the fret position.

Since same note can be played in different positions of the fretboard, conversion from a standard notation to tablature has its own constraints. Tuohy D. & Potter W. (2005) conducted a study that produces tablature from the standard notation. They used genetic algorithms (GA) in order to produce different tablatures, Figure 2.7. As the fitness function, they used the finger position complexities from the study of Heijink & Meulenbroek (2002).

One year later, in his master thesis Daniel R. Tuohy (Tuohy, 2006) in addition to GA he also includes neural networks. He used a previously constructed guitar tablature data set Classtab.org (2006). 75 classical guitar tablatures were used in order to train the system. From this 75 pieces 1853 patterns were created.

Similar to Tuohy D. & Potter W., Radisavljevic & Driessen (2004) investigate the optimal fingering position for a given set of notes. Their method, *path difference learning*, uses tablatures and AI techniques in order to obtain fingering positions and transitions. However rather than using the cost function of Heijink & Meulenbroek (2002), they defined two cost functions; transition and static. Transition cost function defines the cost transition between different fingerings and static cost function computes the difficulty of alternative fingerings, Figure 2.8.

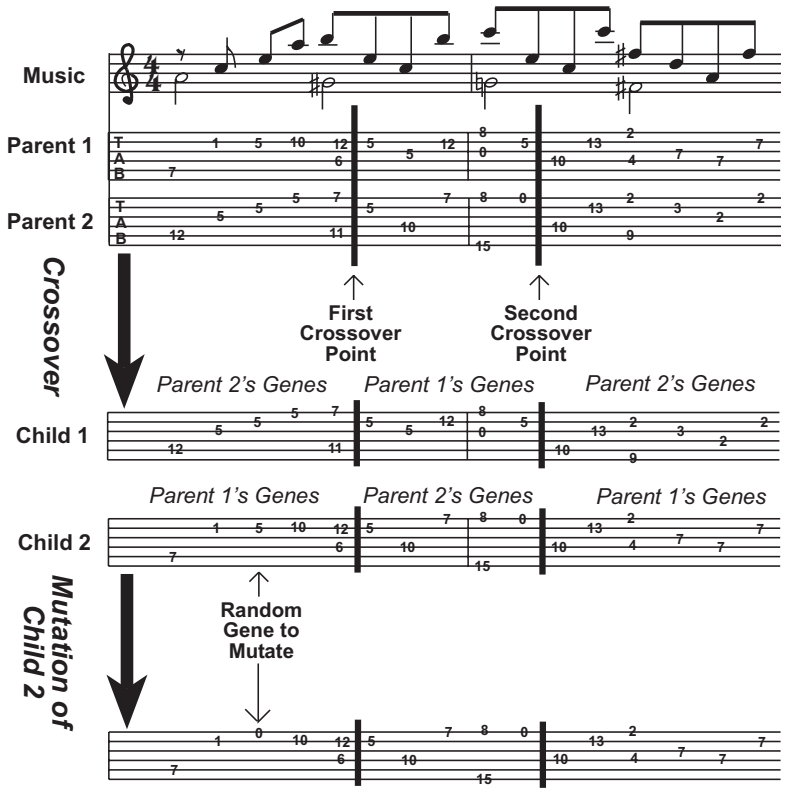


Figure 2.7: Genetic algorithm schematic. Music from "Stairway to Heaven" by Jimmy Page and Robert Plant.

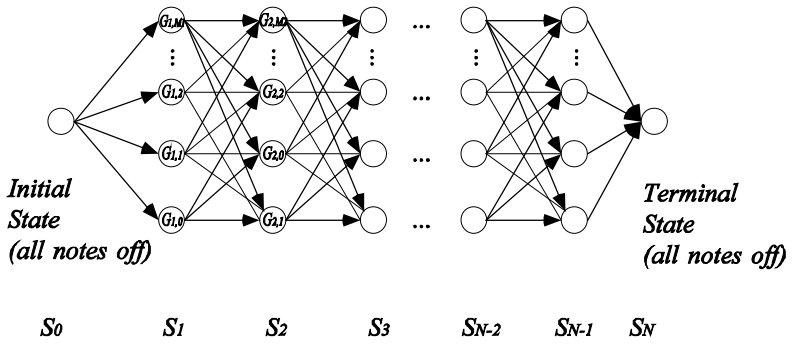


Figure 2.8: Finding the optimal path. Each G_i is a state (note or set of notes)

Scholz & Ramalho (2008) proposed a new approach to recognize chords, from symbolic guitar data, called COCHONUT (Complex Chords Nutting). As they stated, there are more studies about symbolic guitar chord recognition, but in

this study their challenge was to deal with non quantized data. Their test set includes 270 hand label chords. More importantly, their system could be able to recognize 7^{th}, 9^{th} and even 13^{th} chords.
Costalonga & Miranda (2008) investigated what is intentional during a performance and what is unintentional. Authors used biomechanical models. This paper is a further study of Heijink & Meulenbroek (2002) by focusing the symbolic information that was gathered from a MIDI guitar. Heijink & Meulenbroek investigated the right hand positions only considering the single notes, this time Costalonga & Miranda included chords. Five male guitarists were used. Rather than a real guitar, experiments was done with Yamaha EZ-AG, Figure 2.9.

Figure 2.9: Yamaha EZ-AG is an electronic guitar with lights in the frets and different sound options such as 8 different guitars, banjo, piano. Yamaha commercializes this product as self-teaching instrument.

Each performer recorded a sequence of chords in the different positions of the fretboard. They were trying the measure the speed of the change in chord positions. Performers played the same chords in different positions of the fretboard. For instance first position was between fret [1..4], second position is [2..5], third position is [3..6] etc.They also determined starting reference points for all the chord. After that, they measured the time from the last reference point released to the final shape of the chord. This task was repeated for ten chords, C, A, G, E, D, Am, Dm, F, D and Bm. The overall speed of the chord shape was calculated as the mean of the three performers. At the end Costalonga & Miranda (2008) came up with below results;

1. Bar chords longer to perform.
2. First position G chord is one of the slowest to perform.
3. The average time for performers to perform a chord is 350ms.

Recently Barbancho et al. proposed a system for the extraction of the tablature of guitar musical pieces using only the audio waveform (Barbancho et al., 2012).

Their model uses inharmonicity analysis to find the simultaneous string/fret combinations used to play each chord.

2.4 Expressive Performance Analysis

There are recent studies, (Aho & Eerola, 2013), that investigate the similarities between different expressive segments of audios by using audio alignment tools, such as MATCH (Dixon & Widmer, 2006). However, our aim is to study the expressive resources, the resources used by musicians to convey expressivity. As we defined in the introduction (see Chapter 1), expressivity can be defined as the deviations from the actual score. However, these deviations could be subjective. Therefore, in order to analyze and understand expressive performances we need to find objective ways to measure them. Core expressive resources in guitar can be grouped in two classes: expressive articulations and timing deviations.

2.4.1 Analysis of Guitar Articulations

One of the most important steps when analyzing guitar expressivity is to identify and characterize the way notes are played, known as expressive articulations. The analysis of expressive articulations has been previously performed with image analysis techniques as we mentioned in the Section 2.2. One of the few studies about guitar articulations, which we also cited in Section 2.2 is the dissertation of Norton. According to Norton (2008), guitar expressive articulations can be divided into three main groups related to the place of the sound where they act: *attack*, *sustain*, and *release* articulations. Besides from Norton there is a recent study devoted on articulation analysis of guitar (Migneco, 2012). As the Migneco titled his dissertation, *Synthesis of the expressive articulations of guitar*, the main aim of study was to synthesize the classical guitar sound rather than the articulations that Norton defined. Majority of the dissertation covers the audio synthesis techniques such as resonant string response.
Except these two, in the Music Information Retrieval (MIR) literature there are few studies about articulations as a whole. However, by its own, vibrato has been analyzed in several studies. Rossignol et al. (1999a) proposed 5 different algorithms for detection, estimation, extraction and modification of vibrato. The methods proposed vary from spectral envelope to F0 trajectory analysis. F0 trajectory approaches were tested with continuous excitation examples like singing voice and flute. Järveläinen (2002) reported four listening experiments for exploring the perception of vibrato and proposes rules for synthesizing high-quality vibrato sounds. There are three main important features in vibrato: rate, extend, and intonation. Pang & Yoon (2005), used probability modeling existence using vibrato rate, extend and intonation. Tested instruments were of continuous excitation like oboe, trumpet, cello, or viola. In a

similar direction, Verfaille et al. (2005) performed listening experiments where different vibrato models are compared. Their generalized vibrato model combines modulations of amplitude, frequency, and spectral envelope. Martens & Marui (2006) analyzes the perception of vibrato, flange and chorus, which are the three most often used digital audio effects. In their study, 25 listeners were asked to make categorical judgments regarding their perception. More recently, Wen & Sandler (2008) proposed and test several features to model vibrato. Their model contains two modules, analyzer and synthesizer. Analyzer describes the frequency variations of a vibrato using a period-synchronized parameter set, and the accompanying amplitude variations using a source-filter model. Synthesizer reconstructs a vibrato from a given set of parameters.

2.4.2 Timing Deviation

Timing is often considered to be the most important expressive resource, and is perhaps the only variable over which any performer has practically complete control, regardless of the instrument used (Gabrielsson, 2001). Timing generally refers to variations in note duration, onset delays, or onset anticipations, introduced by a performer as compared to the strict adherence to tempo and notated score values. Research on timing deviations has a long history, dating back to the beginnings of the twentieth century (for pointers to such early works we refer to (Gabrielsson, 1999)). Overall, the wealth of existing literature confirms that performers make "systematic and significant deviations from strict metricality" but, at the same time, indicates that "it is hard to make generalizations about the nature of [such] deviations" (Gabrielsson, 2001).
In the literature we find contrasting and complementary views on the origin of timing deviations. There is evidence that timing deviations help the listener to clarify phrasing (Istók et al., 2013; Repp, 1998; Todd, 1992), metrical accents (Sloboda, 1983), musical form (Liem et al., 2011), and harmonic structure (Palmer, 1996; Repp, 1990). Complementarily, different note patterns or groups exhibit some common timing "tendencies" (Gabrielsson, 2001). All these works point towards musical structure as a source for timing deviations, what constitutes the basis of the so-called generative approach (Clarke, 2001). However, to the best of our knowledge, there is yet no systematic, compelling, and large-scale study in this direction (e.g., involving multiple pieces, performers, instruments, styles, and epochs). Moreover, timing deviations might not arise solely from music structure. It has been also shown that they can be idiosyncratic of a performer's style (Liem et al., 2011; Repp, 1990, 1992), to the point that machines can identify such performers using automatically-extracted timing information (Grachten & Widmer, 2009; Stamatatos & Widmer, 2005). Emotional expression is also assumed to play an important role (Juslin & Sloboda, 2001, 2013). Besides, we also find the so-called perceptual hypothesis (Penel & Drake, 1998), in which some observed variations would be due to functional constrains of the auditory system. This way, some time intervals

would be heard shorter and thus played longer as a phenomenon of perceptual compensation (Penel & Drake, 1998). Additionally, some timing deviations may be shaped in accordance to patterns of biological motion (Juslin, 2003) or instrument-related motion (Gabrielsson, 1999, 2003). Last, but not least, one could always attribute timing deviations, to some extent, to random variability (Goebl & Palmer, 2013; Juslin, 2003).

Chen et al. (2001) analyzes the human sensorimotor error with 8 human participants. Each participant attempt to coordinate finger tapping with a computer generated metronome. They made two types of experiments, synchronization and syncopation. In each case error was defined as the timing difference between key press and the associated metronome onset. They presented that for each case long-range correlations are exist. Delignieres et al. (2009) successfully replicate the results of Chen et al. (2001) with two additional test subjects. Moreover they further showed by ARFIMA/ARMA modeling that in both synchronization and syncopation, series contained genuine long-range correlation.

Hennig et al. (2011) analyzed deviations from metronome beat positions of hand, feet and vocal performances by both amateur and professional performers. In their paper they show that these deviations are are not random and long-range correlations are much more pleasing and preferable for listeners. However in the scope of expressive analysis, their study cannot go beyond just analyzing error of controlled studio recordings of performers who try to follow an exact tempo.

CHAPTER 3

Expressive Articulation Extraction

3.1 Introduction

Our research on musical expressivity aims at developing a system able to identify and analyze the expressive resources exploited by classical guitar performers. To achieve our goal, we have divided the process in three stages:

- Low-level feature extraction (onset, pitch, amplitude, periodicity etc.).
- Higher-level feature extraction models build on top of low-level features (chroma, interpolation etc.).
- Analysis of the resources that we identified by using both low and high level features.

In this chapter we focus the analysis on three of the most-widely applied expressive guitar articulations: legato, glissando, and vibrato. Specifically, we present a system that combines several state of the art analysis algorithms to identify and characterize guitar legatos, glissandos, and vibratos. Besides from the analysis, we also propose a tool able to identify expressive resources automatically in the context of real classical guitar recordings.
Although all three guitar articulations share common feature extraction methodologies (see extraction module in Figure 3.1), explained in Section 3.2, they need significantly different approaches for further extraction and classification purposes. Briefly, our method first distinguishes between non-articulated, vibrato, and legato or glissando. Further details will be provided all through next sections.
After describing the components of our system, we report the experiments conducted with recordings containing single articulations, short melodies performed by a professional guitarist, and also commercial recordings of Villa Lobos Prelude Number 4 performed by different guitarists.

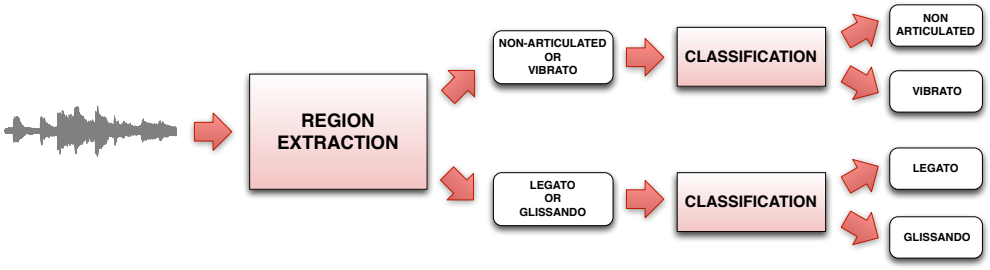


Figure 3.1: Main diagram for the proposed expressive articulation extraction and classification methodology.

3.2 Feature Extraction

As we stated above, our aim is to extract, analyze and classify three main guitar articulations: legato, glissando, and vibrato. For this purpose, our model has two main modules: extraction and classification (see Figure 3.1). The extraction module constructs a model according to our needs. After describing the process of extraction of low level features such as onset detection (section 3.3.1), pitch detection (section 3.3.2), and detection of attack and release points (section 3.3.3), a detailed description of the extraction module is provided in Section 3.3. Each of these low-level extraction tasks is an area of research (Bello et al., 2005; Brossier, 2006; de Cheveigné, 2005; de Cheveigné & Kawahara, 2002). Therefore, we did not aim to construct or propose new extraction algorithms for any of the listed features. Rather, we come up with robust optimization solutions by using existing algorithms according to our needs.

3.2.1 Onset Detection

Onset detection (or segmentation) is the process by which we can divide the musical signal into smaller units of sound. Onset detection still is an active area of research[1].
In music information retrieval (MIR), onset detection is perhaps one of the most important tools. It can be used as the core method for several high-level systems, such as identifying the beats from a recording of polyphonic music by looking for the drum onsets (Alghoniemy & Tewfik, 1999) or for melody detection on a monophonic signal, to determine when an instrument is actually playing (Thornburg et al., 2007) .
In music signal processing, there exist different techniques of varying the complexity for automatic onset detection (Bello et al., 2005; Müller et al., 2011). These techniques usually work on the time domain, the frequency domain, or both (Bello et al., 2005; Brossier, 2006; Collins, 2005). More specifically,

[1] http://www.music-ir.org/mirex/wiki/2013:Audio_Onset_Detection

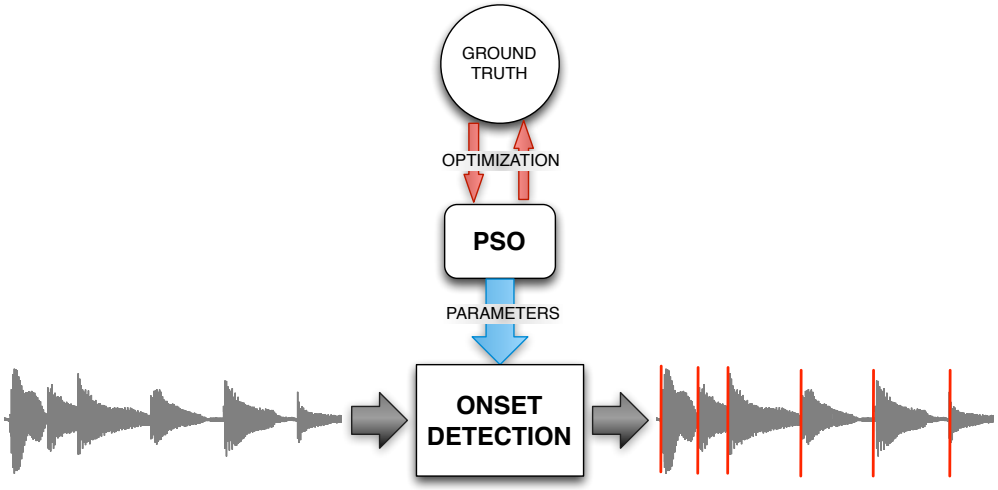


Figure 3.2: Onset detection method.

there are approaches including finding abrupt changes in the energy envelope (Dixon, 2001), in the phase content (Bello & Sandler, 2003), in pitch trajectories (Rossignol et al., 1999b), in audio similarities (Foote & Cooper, 2000), in autoregressive models (Jehan, 1997), in spectral frames (Gouyon et al., 2003), and using psychoacoustic knowledge (Klapuri, 1999). In addition, there are also high-level techniques which combine different methods such as neural networks (Bock et al., 2012) or hidden markov models (Abdallah & Plumbley, 2003) .

Limitations

It is not expected that a single method or parameter combination should work for any recording (Bello et al., 2005). In our research, as we stated previously, we do not aim to come up with a new onset detection technique. Rather our approach was aimed to find an appropriate function and tune its parameters according to guitar characteristics. In order to be able to test several state of the art methods we chose Aubio library (Brossier, 2006). It gave us all the functionality by having different onset detection functions. However, considering that Aubio library has 7 different onset detection functions and each function has 4 continuos parameters, choosing by trail and error was not an option. Then, our approach was to exploit an optimization algorithm to find the best fitting algorithm and parameters.

Particle Swarm Optimization

For selecting the best algorithm and parameters we developed a particle swarm optimization (PSO) algorithm (Kennedy & Eberhart, 2001) and tuned our parameters according to PSO. PSO iteratively tries to improve a candidate solution with regard to a given measure of quality. In a PSO algorithm there are three main concepts: population (swarm), candidate solutions (particles) and the cost function. In each iteration, for each particle p_i, a fitness value is computed from the cost function $f(p_i)$. The best position (according to the fitness) of the particle is stored and called the personal best β_i. Also best of all personal bests is stored and called global best β_{global}. The algorithm starts by randomly assigning particles. These particles are moved around in the search-space according to a pre-determined formula. The movements of the particles are guided by their own personal best as well as the global best. When improved fitnesses are being discovered, they are used to update particle and global bests. The process is iterated many times with the goal, not fully guaranteed, that a satisfactory solution will eventually be discovered. In our case our optimum solution is the global best at the end of the PSO process.
Our problem was, not having an optimized algorithm and its parameters for onset localization in commercial guitar recordings. Our ground truth was 6 commercial guitar recordings. Specifically, from a set of 12 commercial recordings, we selected 6 of them as training for PSO and use the rest of the pieces for expressive articulation detection. We hand annotated the onset positions of each recording. Our aim was to maximize the result of the *Fitness Function*. Each PSO implementation either needs a termination condition such as a fitness threshold or maximum number of iterations. In our implementation it was a pre-determined maximum number of iterations.
In our case a particle, p_i, is a configuration in the solution space. Specifically, each onset function and its corresponding parameters was modeled as a dimension $d_1, d_2, d_3 \dots d_n$. Thus, a particle was represented as a tuple $p_i = (d_1, d_2, d_3 \dots d_n)$. We had 5 dimensions and each dimension can take values between 0 and 1. Actual names of the dimensions are summarized in Table 3.1. As we stated previously, the PSO algorithm starts by randomly assigning the positions of each particle. Each particle p_i owns a velocity vector, $\vec{v_i}$, which influences position updates according to a simple discretization of particle motion:

$$p_i(t+1) = p_i(t) + \vec{v_i}(t+1) \tag{3.1}$$

$$\vec{v_i}(t+1) = \chi(\vec{v_i}(t) + U_\phi(\beta_g - p_i(t)) + U_\phi(\beta_i - p_i(t)) \tag{3.2}$$

where p_i, $\vec{v_i}$, and t are particle position, particle velocity, and time (iteration counter) respectively; U_ϕ represents a vector of random numbers uniformly distributed in $[0, \phi]$; β_i and β_g are, respectively, particle best position and

global best position; and χ and ϕ are constants that take the standard values $\chi = 0.729843788$ and $\phi = 2.05$ (Clerc & Kennedy, 2000).

Dimension	Name	Actual Values	PSO Range
d_1	Algorithm	Complex Domain	0.0 - 0.14
		High Frequency Content	0.14 - 0.28
		Phase	0.28 - 0.42
		Spectral Difference	0.42 - 0.56
		Energy	0.56 - 0.70
		Kullback Liebler	0.70 - 0.84
		Modfied Kullback Liebler	0.84 - 1.00
d_2	Window Size (ω)	128	0.00 - 0.14
		256	0.14 - 0.28
		512	0.28 - 0.42
		1024	0.42 - 0.56
		2048	0.56 - 0.70
		4096	0.70 - 0.84
		8192	0.84 - 1.00
d_3	Hop Size	$\omega/4$	0.00 - 0.25
		$\omega/3$	0.25 - 0.50
		$\omega/2$	0.50 - 0.75
		$\omega/1$	0.75 - 1.00
d_4	Silence Threshold	-90 - 0	0 - 1
d_5	Peak-Picking Threshold	0 - 1	0 - 1

Table 3.1: Summary of each dimension and its corresponding PSO value.

We used out of sample audio files in PSO evaluation. If the onset that is detected by the chosen algorithm is between the range of $100ms$ of the ground truth onset, we marked this onset as a True Positive (TP), otherwise it is marked as False Positive (FP). For each ground truth onset there can be only one onset candidate. If there are more than one onset candidates that are in the range of $100ms$ only the nearest one is considered and others are marked as FPs:

$$f(p_i) = \frac{\#TP(p_i)}{\#GTOnsets} - \frac{\#FP(p_i)}{\#AllOnsets} \tag{3.3}$$

Our fitness function takes values from -1 to 1. The complete PSO algorithm for maximizing a fitness function f is summarized in Algorithm 3.2.1.

// Initializations **for** *each particle* **do**
  Randomly assign particle position p_i Randomly assign a first velocity vector, $\vec{v_i}$
end
while *termination criterion reached* **do**
  for *each iteration* t **do**
    for *each particle* i **do**
      $p_i(t+1) = p_i(t) + \vec{v_i}(t+1)$;
      if $f(p_i(t+1)) > f(\beta_i)$ **then**
        $\beta_i = p_i(t+1)$;
      end
      if $f(p_i(t+1)) > f(\beta_g)$ **then**
        $\beta_g = p_i(t+1)$;
      end
    end
  end
end

Algorithm 1: Implementation of the PSO algorithm. p_i refers to each particle, β_i refers to particle best, and β_g refers to the best of all particles (global best).

We run our PSO algorithm with 10 particles and 50 iterations. As shown in Figure 3.3, at the end, according to our fitness function we reached a measure of 0.98 with the Kullback-Liebler onset detection algorithm (Hainsworth & Macleod, 2003), a window length of 1024 samples, a hop size of 512 samples, a peak-picking energy threshold of 0.53, and a silence threshold of -67 dB. For each audio the sampling rate was 44.1 KHz. For further explanations of the parameters we refer to the `Aubio documentation`[2] and also the dissertation of Brossier (2006).

Kullback-Liebler Onset Detection Function

Kullback-Liebler onset detection function works on frequency domain (Hainsworth & Macleod, 2003). Basically it computes the distance of the amplitudes of each point between two consecutive frames by using the *Kullback-Liebler* distance function:

$$D_{K-L}(n) = \sum_{k=0}^{\frac{N}{2}-1} (|X_n(k)|) \log_2 \left(\frac{|X_n(k)|}{|X_{n-1}(k)|} \right) \tag{3.4}$$

[2] http://aubio.org/doc/onsetdetection_8h.html

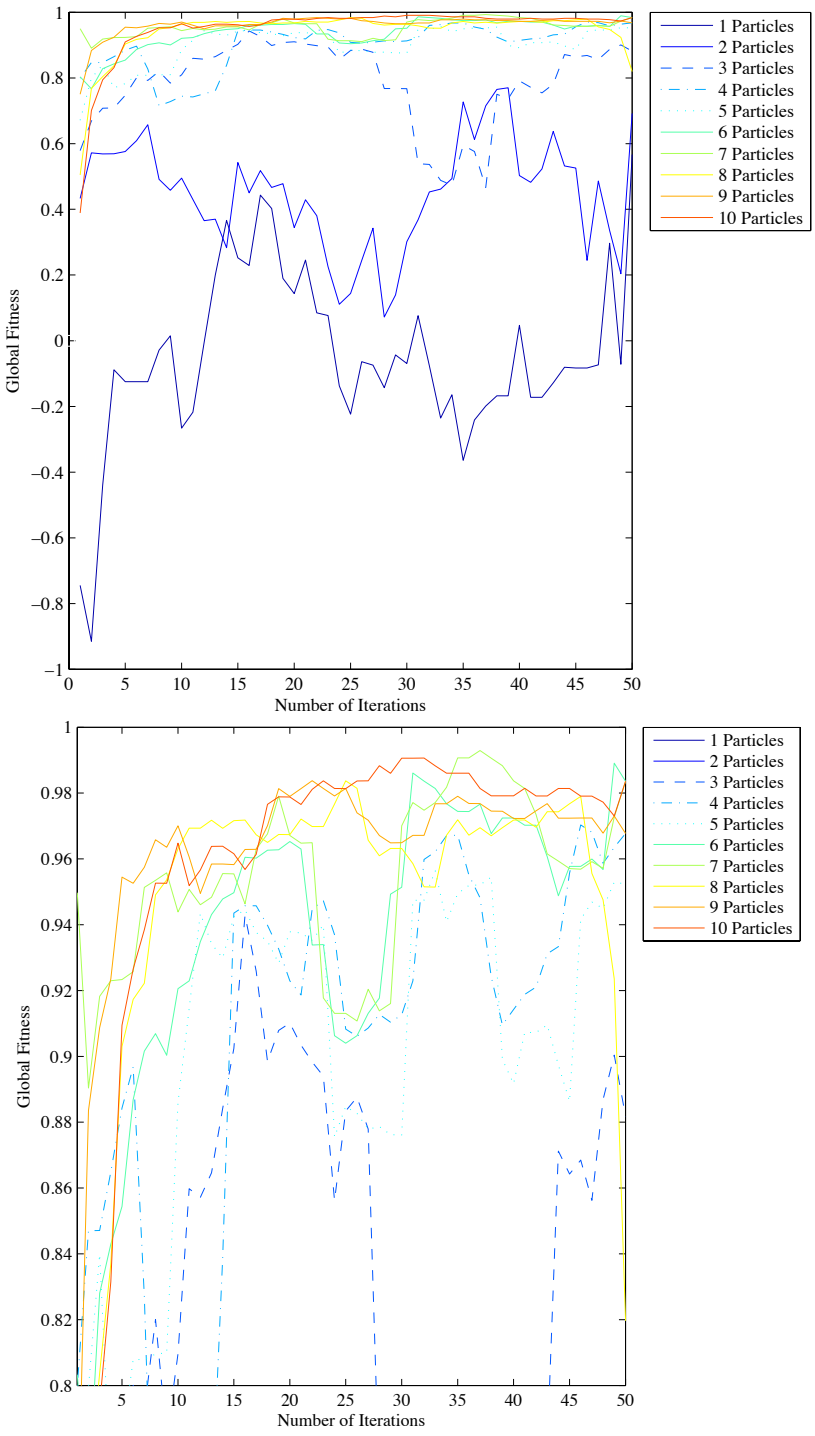


Figure 3.3: PSO Fitness.

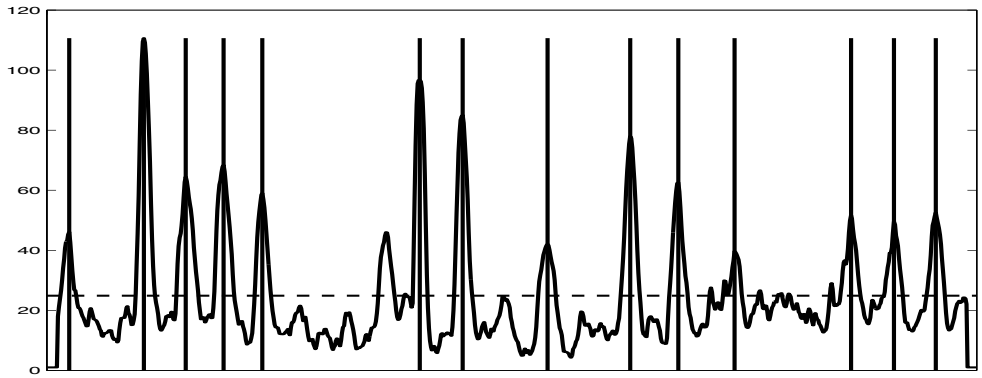


Figure 3.4: Kullback-Liebler distances of each frame of an example audio sample.

where X_n is the Short Time Fourier Transform of the n^{th} frame of the data with the length of N bins. An example is shown in Figure 3.4, where each peak corresponds to an onset candidate in the example audio sample.

3.2.2 Pitch Detection

Estimating pitch or fundamental frequency from a signal is called pitch detection. This detection technique is widely used in speech and music studies. Computational studies for the understanding of pitch perception goes back to 1970's (de Boer, 1976). There are two interesting studies for the readers who are passionate about making progress in understanding state of the art pitch detection algorithms (PDA). First one is the survey of de Cheveigné (2005), in which he explains both the methodologies and practical uses of PDAs which are published before 2005. Second one is shorter but one of the most recent surveys upon the written date of this dissertation (Babacan et al., 2013). Both surveys cite Yin algorithm as one of the most popular and accurate PDA.
Yin is an algorithm, which is presented for the estimation of the fundamental frequency of speech or musical sounds (de Cheveigné & Kawahara, 2002). It is based on the well-known autocorrelation method with a number of modifications that are combined to prevent errors. Yin algorithm has three outputs: aperiodicity, fundamental frequency, and energy. Furthermore, its analysis parameters can be modified, with fields explained in Table 3.2. By this way Yin algorithm can be tuned according to desired input.

3.2.3 Attack and Release Points

Sounds that are produced by musical instruments have a regular pattern that repeats in time. Although these sounds are rather complex, they share a common characteristic with sinusoids: the periodicity. This periodicity leads

	Parameter	Explanation
Input	f0-min	minimum expected F0 (default: 30 Hz)
	f0-max	maximum expected F0 in Hz
	threshold	threshold (default: 0.1)
	relfag	if 0, thresh is relative to min of difference function
	window	integration window size (defaut: SR/minf0)
	hop	hopsize((default: 32/sampling rate (P.sr)))
	buffer size	size of computation buffer (default: 10000)
	sampling rate	sampling rate (usually taken from file header)
	low pass	intial low-pass filtering (default: SR/4)
	shift	0: shift symmetric, 1: shift right, -1: shift left (default: 0)
Output	f0	fundamental frequency
	aperiodicty	aperiodic content of the input
	energy	energy content of the input

Table 3.2: Input and output parameters of YIN algorithm

humans the perception of pitch (Goldstein, 2001). In other words, as the sound gets more periodic, the pitch content increases as the aperiodic content decreases. In order to find the portion that contains the most of the pitch content we need to find the portion that contains most of the periodic information. Therefore, to find the sound fragment with the highest periodic content between two onsets, we first find the fragments with the highest amount of aperiodic content and omit these fragments. Previous to a detailed explanation, improvement is exemplified in Figure 3.5. From top to bottom first graph on Figure 3.5 is the audio fragment that we are analyzing, second graph is the amplitude and third graph is the aperiodicity of this audio portion.
Before deciding to use aperiodicity, our first attempt was to use amplitude in order to decide the attack and release points. However as shown in Figure 3.5, what actually we were doing was omitting a useful portion because of a wrong detection of attack finish point and also introducing more noise to the system because of determining a late release start point.
As shown in the upper first graph in Figure 3.6, our aperiodicity data was noisy. Therefore, our first attempts to determine attack-finish and release-start points failed. We needed to clean our data. In order to avoid noise and obtain a smoother data first we applied envelope approximation.

Envelope Approximation

After obtaining a smoother data, an envelope approximation algorithm was applied. The core idea of the envelope approximation is to obtain a fixed length representation of the data, specially considering the peaks and also

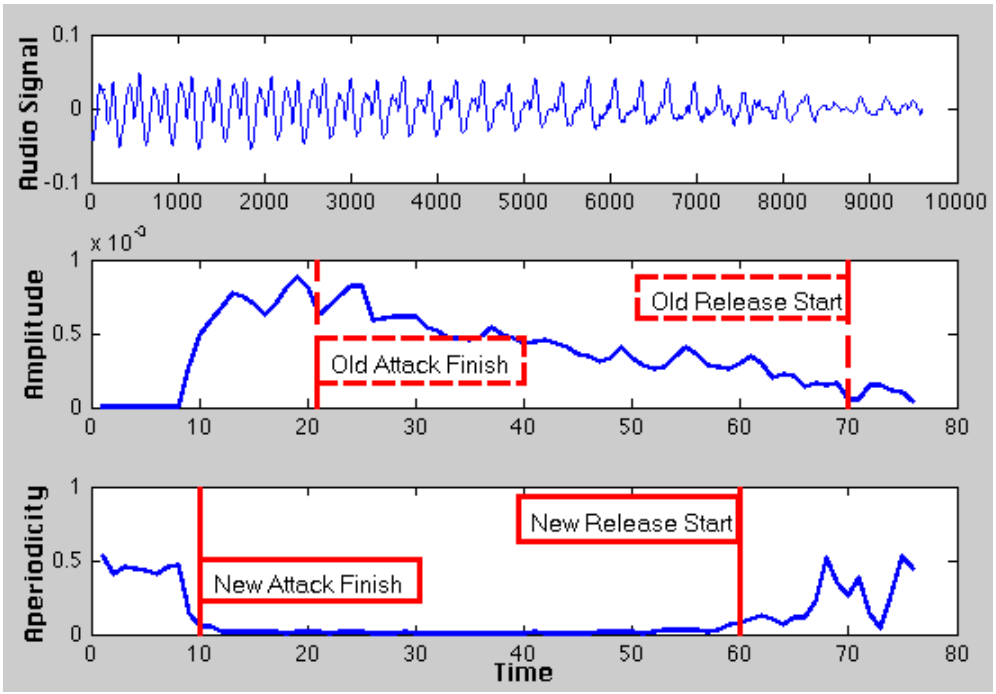


Figure 3.5: Comparison of different attack and release points.

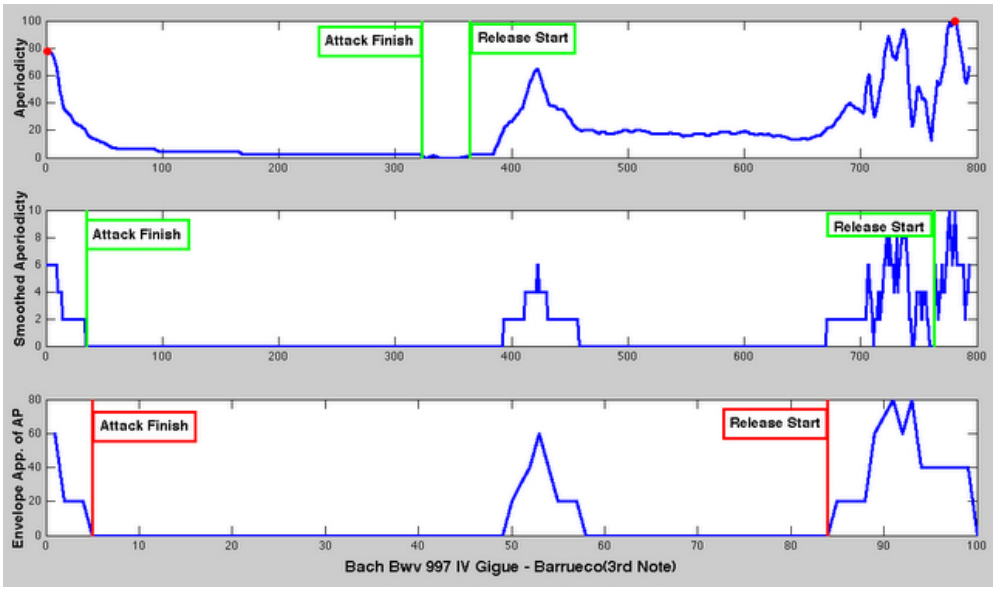


Figure 3.6: Envelope approximation of Aperiodicity Graph of a Legato Note.

avoiding small deviations by connecting these peak approximations linearly. The envelope approximation algorithm has three parts: *peak picking*, *scaling*, and *linear interpolation*. After the envelope approximation, all data regions to be analized had the same length, i.e., regions were compressed or enlarged depending on their initial size.

- **Peak Picking**
  All the peaks higher than a pre-determined threshold were collected. After a normalization step, the threshold applied was 0.1.

- **Scaling**
  We scale all peak positions. For instance, imagine that our data includes 10000 bins and we want to scale this data to 1000. And lets say, our peak positions are : 1460, 1465, 1470, 1500 and 1501. What our algorithm does is to scale these peak locations dividing all peak locations by 10 (since we want to scale 10000 to 1000) and round them. So they become 146, 146, 147, 150 and 150. Since after the scaling process we have 2 peaks in 146 and 150, we have to remove duplicates. In order to fix this duplicity, we choose the ones with the highest peak value.

- **Linear interpolation**
  After collecting and scaling peak positions, the peaks are linearly interpolated.

At the end of envelope approximation, we obtained a smother data as shown in the middle graph in Figure 3.7. However, we were still introducing noise because of the error in detection of release start point. In the last stage, what we did was to apply smoothing with value of 50 bins. At the end, we obtained a perfect estimation for the aperiodicity data, third graph in Figure 3.6, and successfully determined attack finish and release start points.

3.3 Region Extraction

We have two main modules for the expressive articulation extraction, region extraction, and classification. Differently from the classification modules, region extraction module is common for all the expressive articulation analysis and extraction. Region extraction module also includes low level feature extraction. We extract onsets, attack-release points, and pitch. The parameters that we are using for each feature extraction method were explained in the previous section (Section 3.2).

- **Onset Detection**
  Vertical lines in Figure 3.9 represent the onset positions. Again in the same Figure 3.9, in the bottom picture it can be observed that although there are pitch changes both in the vibrato and legato part, we managed

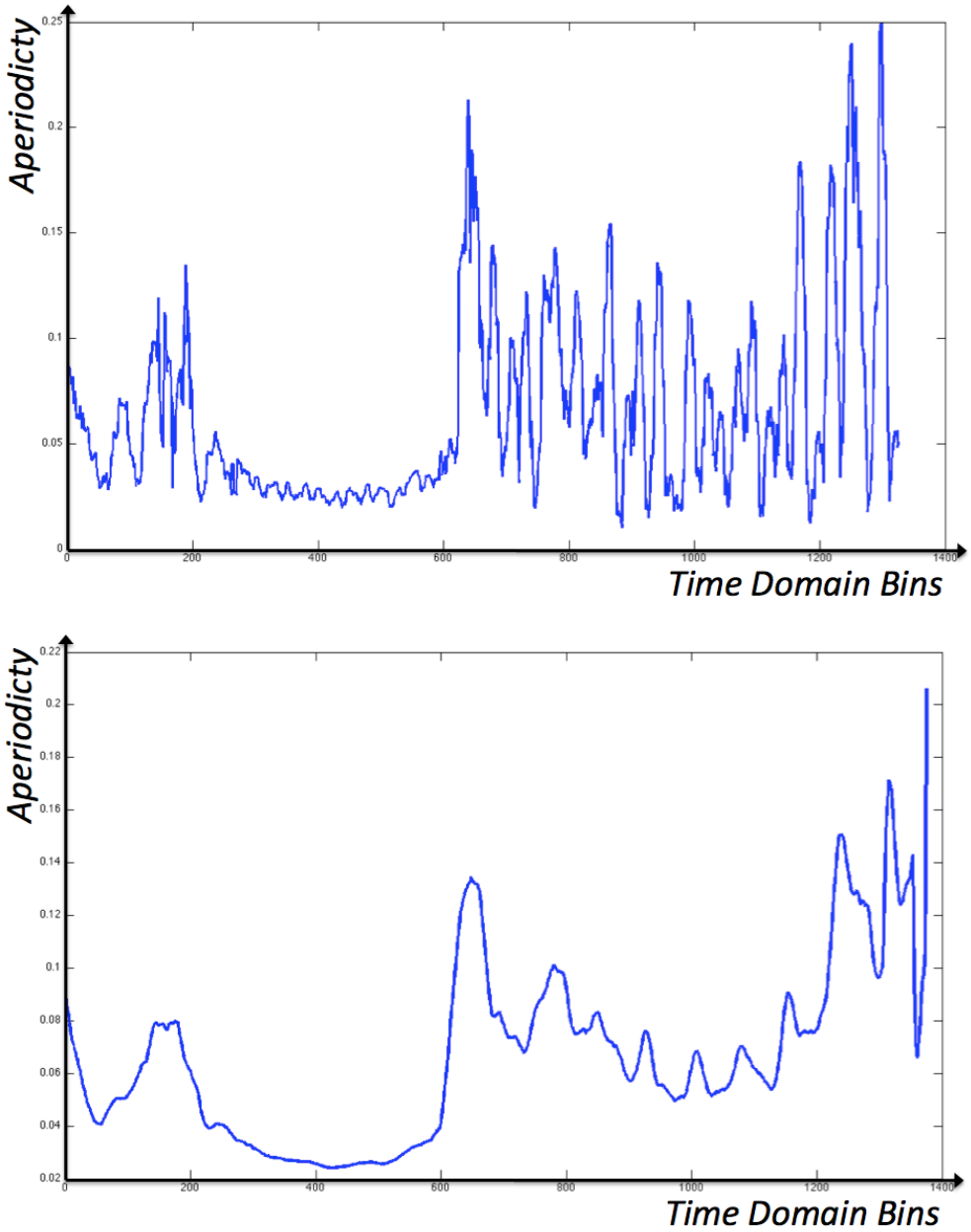


Figure 3.7: The bottom figure is an approximation of the top figure. As shown, linear approximation helps the system to avoid consecutive small tips and dips.

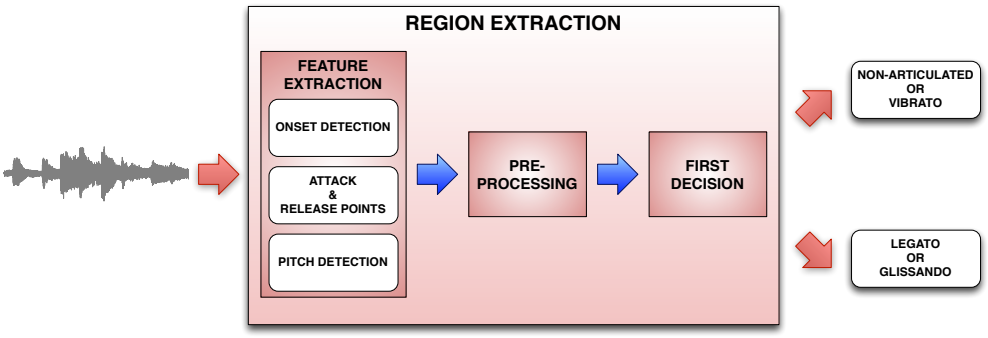


Figure 3.8: Main diagram for our expressive articulation extraction and first decision methodology.

to optimize the parameters of our onset detection algorithm, enough less sensitive for not considering these pitch changes as real onsets (More details can be found in Section 3.2.1).

- **Attack and Release Points Determination**
  Attack and Release Points are determined as explained in Section 3.2.3.

- **Pitch Detection**
  After successfully determining the attack-finish and release-start points, our second task was to analyze the sound fragment between these two points. Specifically, we analyzed the sound fragment between attack ending point and release starting point (because the noisiest part of a signal is the attack part and the release part of a signal contains unnecessary information for pitch detection (Dodge & Jerse, 1985)). Therefore, for our analysis we take the fragment between attack and release parts where pitch information is relatively constant.

  We tuned our parameters as: minimum expected frequency $80Hz$, maximum expected frequency $1500Hz$ window size 2048 bins, and hop size 1024 bins. Our silence threshold was $-70db$. According to our observations, other parameters have little effect on the outputs. Rather than searching parameters for better Yin results, we designed pre-processing techniques for cleaning the Yin outputs.

3.3.1 Pre-Processing

After feature extraction, in order to avoid octave errors and to analyze the changes in notes rather than in frequency, we converted our pitch information to its corresponding 12 step chroma representation. In this representation, each step corresponds to a semitone. In guitar since each fret is separated

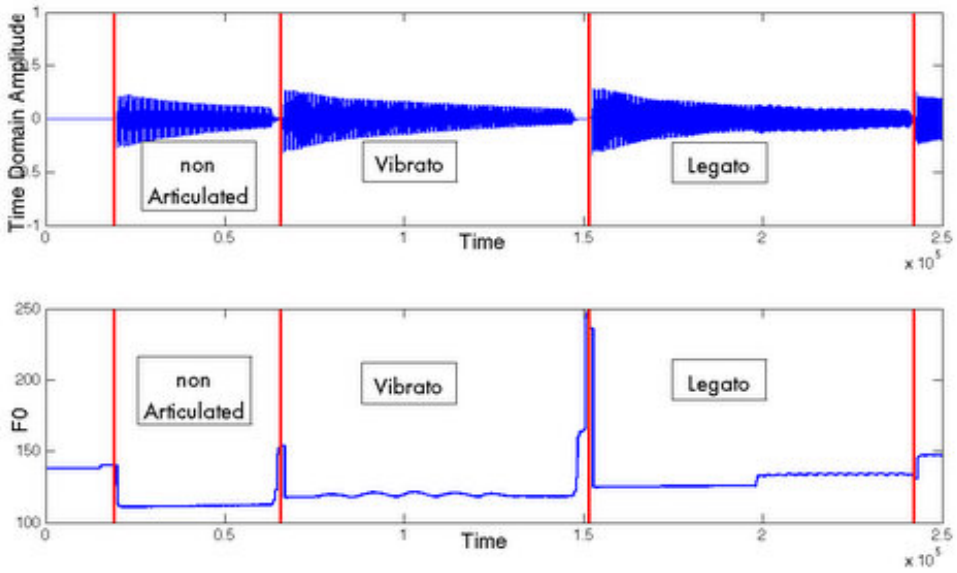


Figure 3.9: Onsets and Pitch from the Extraction module.

with a semitone, each step corresponds to a fret. Therefore, this conversion also makes the changes in semitones(frets) more abrupt. In the case of legatos and glissandos in guitar, the change in 12 step chroma representation is always greater than one step. So, after the 12 step chroma conversion, if the note is legato or glissando, the pitch values should be represented by at least with two different steps. If the note is a non-articulated or a vibrato note the pitch values should be represented by a single step.
In Figure 3.10, each graph presents the bar representation of the chroma steps occurrence percentage of the note. From left to right, first graph corresponds to a non articulated note, second graph is a vibrato note, and third graph is a legato or glissando note. As shown in Figure3.10, graphs one and two have similar characteristics compared to third graph. The reason is that most of the time the vibrato in the classical guitar falls in the same chroma step. In other words, in the 12 step chroma representation, there are no distinctions between a non-articulated note and a vibrato note.

First Decision

After obtaining the chroma occurrence percentages, for each note we search for the peaks equal or greater than 80% . This means that, 80% of the frequency frames of the note that is investigated corresponds to the same chroma step. We classify these notes as non-articulated or vibrato notes. The rest are classified as Legato or Glissando notes. After this first decision we run two different classification algorithms for both classes. Our Legato and Glissando classifi-

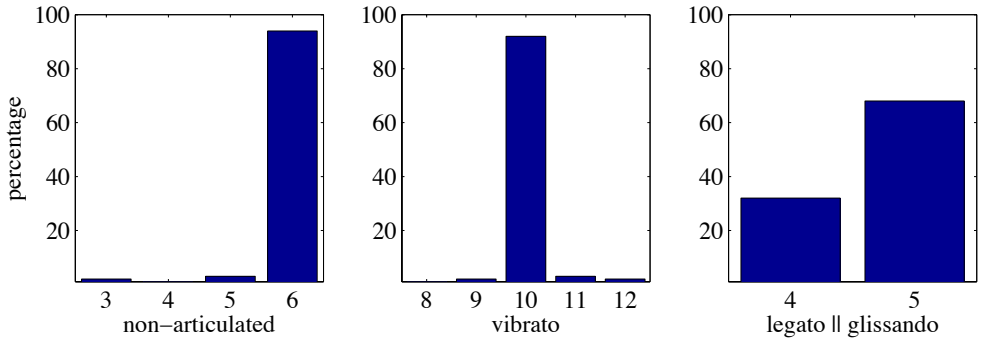


Figure 3.10: Chroma representation of different notes. X-axis corresponds to 12-step chroma value.

cation algorithm is going to be explained in Section 3.5. In the next section we will explain our algorithm for differentiating between non articulated and vibrato notes.

3.4 Vibrato Extraction

The two main characteristics of a vibrato are the *vibrato amplitude*; the range of pitch oscillation, and the *vibrato frequency*; the velocity of oscillation. Vibrato plays an important role in phrasing. Depending on the musical structure, vibrato varies in amplitude, frequency or dynamicity (Timmers & Desain, 2000).

Guitar sounds are unique, like other impulsive generated sounds, because their overtones do not present an exact harmonicity relation. Moreover, because of the complex resonances produced by the guitar body, the analysis of vibrato in classical guitar present specific difficulties that are addressed in our proposal. For instance, the periodicity of the vibrato is not regular. The behavior of vibrato in guitar is much more different from the instruments that were investigated in previous studies. Therefore rather than trying to model the vibrato, we check how the region is close to an ideal vibrato.

In this section we present the new capabilities of our system for vibrato detection. We propose the use of a 120 step chroma representation to analyze guitar vibrato together with a measure of the distance to a model of perfect periodicity on pitch oscillation. We will describe the complexity of vibrato detection analyzing 10 different guitarists playing the beginning of Villa Lobos Prelude Number 4 and a recording of Pepe Romero's adaptation of a J. S. Bach Cello Suite.

The structure of the section is as follows: Section 3.4.1 describes our methodology for vibrato determination. First we applied preprocessing techniques to obtain clearer and smoother data for further analysis. Then we used descrip-

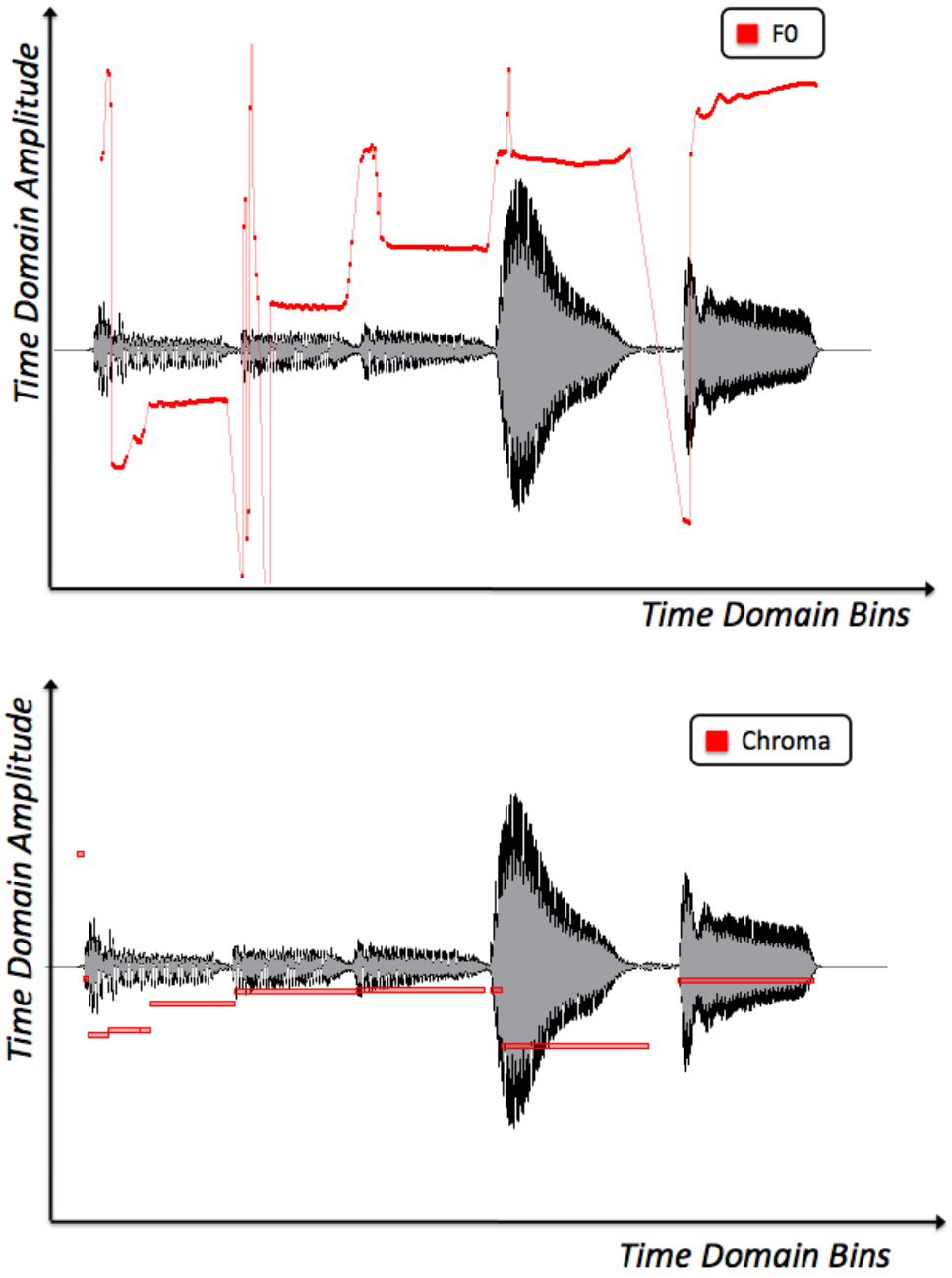


Figure 3.11: Comparison of note extraction without chroma(top), and with chroma features(bottom)

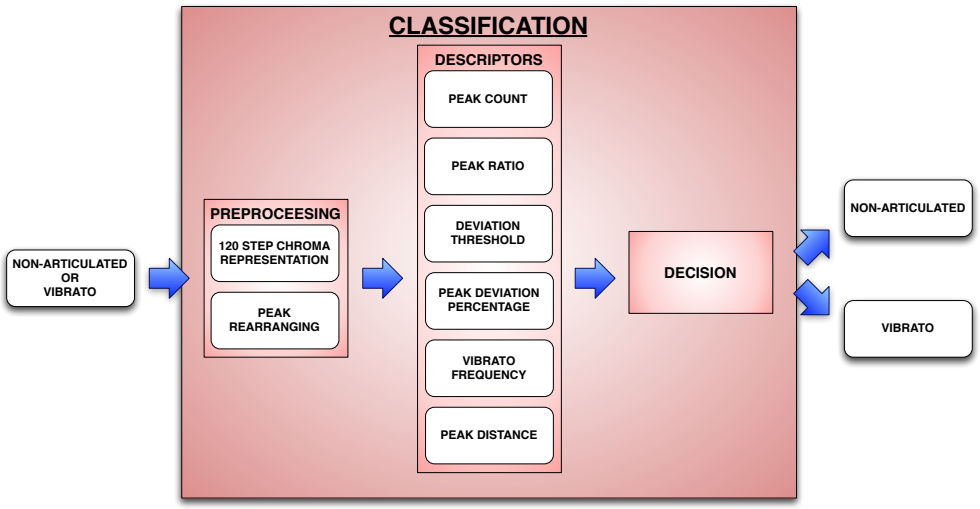


Figure 3.12: Vibrato classification chart.

tors to classify the regions as vibratos. Section 3.4.4 focuses on the experiments conducted to evaluate our system and summarizes current results.

3.4.1 Classification

In classification module we are using the output of region extraction module. Therefore our input is either a non-articulated note or a vibrato note. Our task is to differentiate between these two. To that purpose, we first apply a pre-processing method and then we take a decision by using the descriptors that we defined.

3.4.2 Pre-Processing

120 Step Chroma Representation

As shown in Figure 3.13, in twelve step chroma representation there are no distinctions between non articulated and vibrato notes, hence we cannot rely on twelve step chroma representation for differentiating between non articulated and vibrato notes.
Although in Figure 3.13 there is significant visual distinction between frequency representations of non-articulated and vibrato notes, tiny frequency deviations in both cases cause false positive peak occurrences. Also in some cases, octave errors can result in false positive peaks. In the case of vibrato, since we are interested in the frequency deviations in the borders of a semitone, we came up with a solution. We divided one semitone into 10 equal chroma steps. In

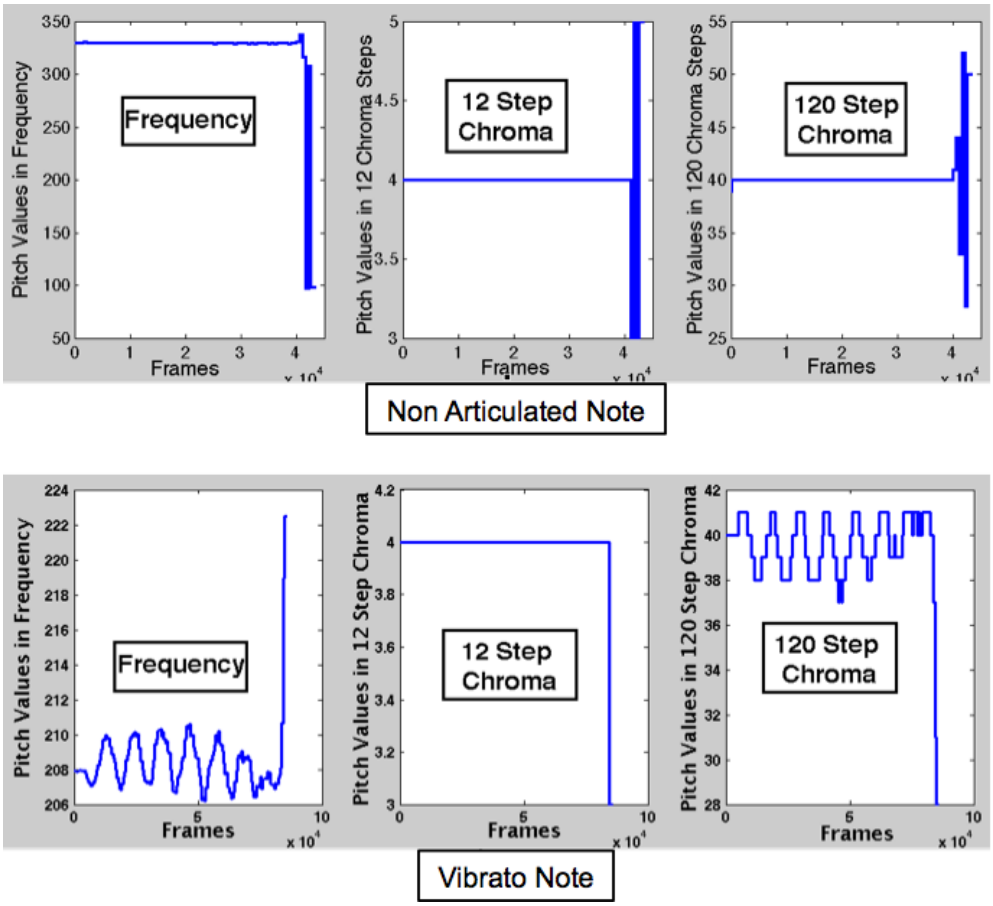


Figure 3.13: Different F0 representations of a non-articulated and a vibrato note with non-chroma, 12 step chroma and 120 step chroma, respectively.

this way we can only concentrate on the deviations one tenth of a semitone. As a result of this, we obtained 120 step chroma representation.
After converting each note into 120 step chroma, we applied preprocessing techniques. As shown in Figure 3.13, at the end of all graphs there is a leap. As shown also in chroma representations this leap can not be avoided, thus it is not an octave leap. We consider this frequency change as an error for our system, because since we are analyzing between two onsets and we know that the note is a candidate for either a non articulated or a vibrato note, there should not be a frequency change more than one semitone. However neither our system nor the guitar players are perfect, because of that, these changes can be due to the beginning of the next note, a string buzz, a wrong estimation of Yin etc. Therefore, we consider these changes as an error for our system.
According to our previous investigations, we know that the occurrence percentage of these errors are less than 5% all through the frequency bins. In

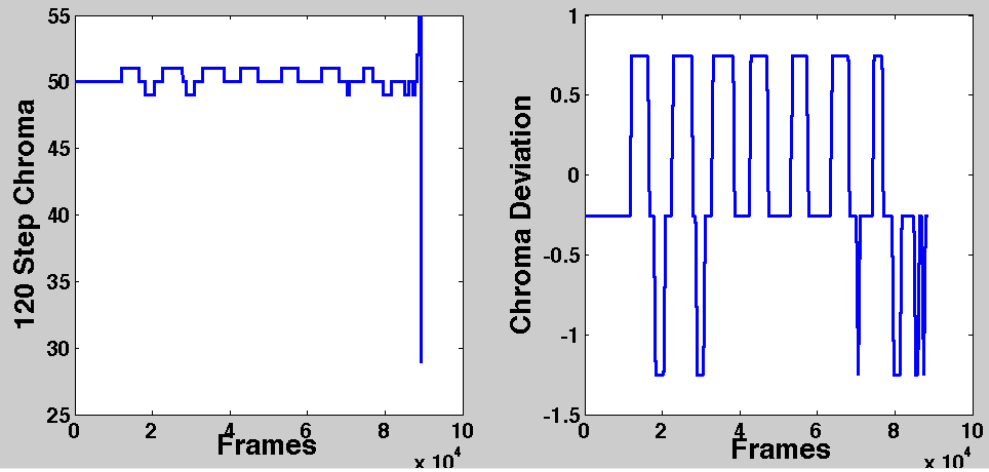


Figure 3.14: Comparison of Vibrato.

order to clean these frequency leaps, we constructed the occurrence percentage histograms of all the chroma bins, as we did in Figure 3.10, and cleaned the ones that occur less than 5% frequency. After cleaning the errors, to obtain the deviation in chroma, we simply calculated the average of all chroma bins and subtracted from the 120 step chroma values. The before and after graphs can be shown in Figure 3.14.

Peak Rearranging

All through our model, we used the chroma deviation values. Last preprocessing that we applied to our data is *Peak Rearranging*. As shown in the left graph in Figure 3.15, in some cases chroma deviation data could contain peaks that are too close to each other. In order to determine and rearrange these peaks, first we calculated the average distance between all peaks. Then we determined the peaks which distance between them is lower than one fourth of the average distance. Finally, we deleted those peaks and created a new one, having an *x coordiate* value as the average of the deleted peaks and the value for the *y coordinate* the corresponding value in chroma deviation values.
After the last preprocessing step, we may define our *Descriptors* (see diagram in Figure 3.12). These descriptors are used to determine whether the audio portion is a non-articulated or vibrato note.

3.4.3 Vibrato Detection Descriptors

In order to define vibrato, we need to understand and identify its behavior in an audio content. For this purpose, we recorded basic exercises played on an electric guitar. The reason to use an electric guitar is that we could record by

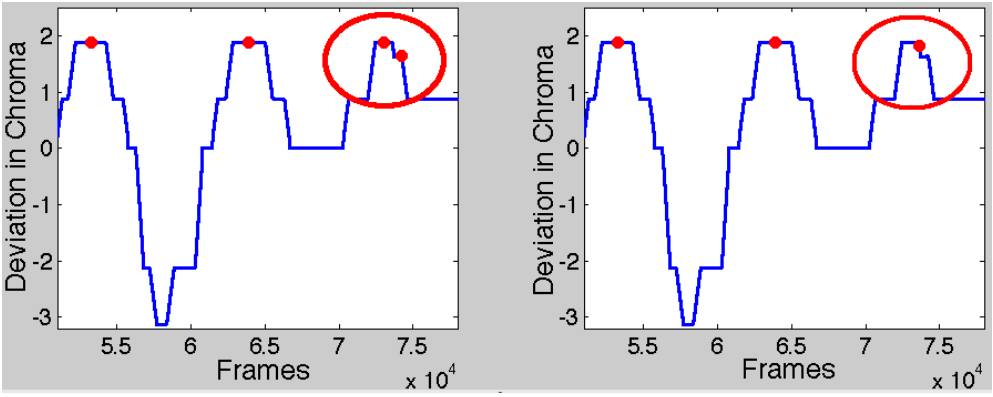


Figure 3.15: Cleaning and rearranging the peaks.

using line signal which has the lowest artifacts like the resonance of the body, reverberation and microphone character.
For our training set, we recorded basic chromatic exercises at three positions as shown in Figure 3.17. Each position contains 6 mixtures of legato and glissando and 6 vibrato notes. Since we repeated the exercise at three different positions, we obtained 54 notes with 18 legato or glissando examples and 18 vibrato examples for our electric guitar test set. We used this training set for initial understanding and the definition of the descriptors. Also we used the training set to determine the values for Section 3.4.3. In this section we will explain each descriptor we used for the classification of vibrato notes.

Peak Count

Our peak count feature contains two fields. Count of *Maxima* (red dots in Figure 3.16) and also *Minima* (green dots in Figure 3.16). As we will explain, minimum vibrato frequency for guitar is 4hz. Considering this value, there needs to be a 4 full cycle deviation, for the perception of vibrato. In other words there needs to be at least 4 peaks both in Maxima and Minima.

In the case of Figure 3.16, Maxima Count is 7 and Minima Count is 7. By the help of Peak Count feature, we can be sure that there are enough positive and negative peaks for vibrato decision.

Peak Ratio

Peak ratio is the ratio of maxima to minima.

$$PeakRatio = \frac{\#Maxima}{\#Minima} \tag{3.5}$$

For the perfect periodic deviation, there should be equal number of Maxima and Minima. Therefore, peak ratio value should be 1. However ac-

cording our observation, considering the errors that we can not avoid, the peak ratio can be between 0.3 and 3.

Deviation Threshold

As we stated in Section 3.4.2, in the means of *F0* frequency deviation, vibrato occurs between one semitone. Since we divided a semitone into ten equal steps, in our analysis chroma deviation of a vibrato portion should occur in the borders of -5 and +5. Therefore, we also check for this deviation value.

Peak Deviation Percentage

In string instruments, the tonal variation of pitch in vibrato is between 0.7Hz and 3Hz, (Järveläinen, 2002). However our observations pointed that the tonal variation value changes logarithmically as the perception of notes. Therefore, rather than relying on directly the change in frequency, we examined the change in 120 step chroma. According to 120 step chroma scale we observed that the tonal variation is between 0.5 and 3 chroma steps.

In order to test tonal variation in an audio portion, we calculate the occurrence percentage histogram of all peak values. Then, we sum the percentage of values of the peaks that have values greater than 0.5. The result provides a percentage estimation for the values of the peaks. For instance, if the *Peak Deviation Percentage* value is 40%, it means that from all the peaks, 40% percent of them have deviation values greater than 0.5.

Peak Distance

Perception of vibrato relies on the periodic deviation of *F0* (Järveläinen, 2002). In other words, in a vibrato note the deviation of *F0* should be evenly distributed all through audio portion. In this feature we are not dealing with the amplitude of the deviation values. Our interest is only the location of the peaks, which should be evenly distributed.

First we constructed a vibrato model with the peaks we gathered from the audio portion. This model contains the peak positions of an optimal vibrato that can occur with the number of peaks that we gathered. Then we calculated the distance of each peak location with the closest peak in the model. In Figure 3.16 green and red dots are the peaks that we gathered from audio portion and dashed lines are the peak positions that we construct. The *Peak Distance* value is;

$$d = abs(y_{realPeak} - y_{modelPeak}) \tag{3.6}$$

$$PeakDistance = \sum_{i=1}^{PeakCount} d_i \tag{3.7}$$

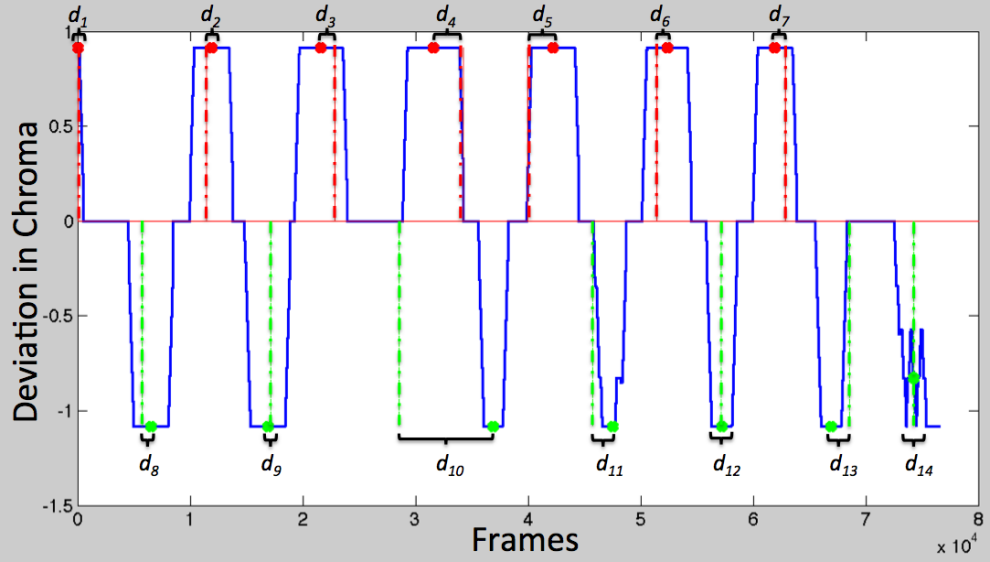


Figure 3.16: Distances between real peaks and vibrato model peaks.

Vibrato Frequency

For string instruments, vibrato rate is typically $5Hz$ (Järveläinen, 2002). However from to the analysis performed in our test set, in guitar vibrato rate is between $4Hz$ and $14Hz$. Therefore, in our model we took Vibrato Frequency borders as $4Hz$ to $14Hz$.

Decision

We are aware of that some features are correlated with other features. For instance, if we know that *Vibrato Frequency* is higher than a given threshold, we don't need to count peaks or calculate peak ratios because in a lower level *Vibrato Frequency* feature actually uses both the *Peak Count* and *Peak Ratio* features. However, from a cost of computation point of view, calculating *Peak Count*, *Peak Peak Ration* and *Deviation Threshold* features are much cheaper than computing *Peak Deviation Percentage*, *Peak Difference* and *Vibrato Frequency* features. Therefore to increase our model's performance first we classify, whether the note is a vibrato candidate or not.

3.4.4 Experiments

The purpose of the experiments was to test the accuracy of the model that we presented. We were interested in automatically detecting vibratos in context which contains, non-articulated, legato or glissando and vibrato notes. Also we tested our model both with chromatic exercises and real recordings.

if *Peak Count* $>$ *4 and 0.3* $<$ *Peak Ratio* $<$ *3 and Deviation Threshold* $<$ *10* **then**
  Note(i).type = vibrato candidate;
else
  Note(i).type = non-articulated;
end
Then we test for computationaly more expensive features;
if *Note(i).type == vibrato candidate and Peak Deviation Percentage* $>$ *40 and Peak Difference* $<$ *4000 and 4* $<$ *Vibrato Frequency* $<$ *14* **then**
  Note(i).type = vibrato ;
end

Algorithm 2: Final Decision Algorithm. In this algorithm, in the first step we can pick the obvious non-articulated notes from possible vibrato candidates. In the second step we can run for our second set of features and pick the vibrato notes.

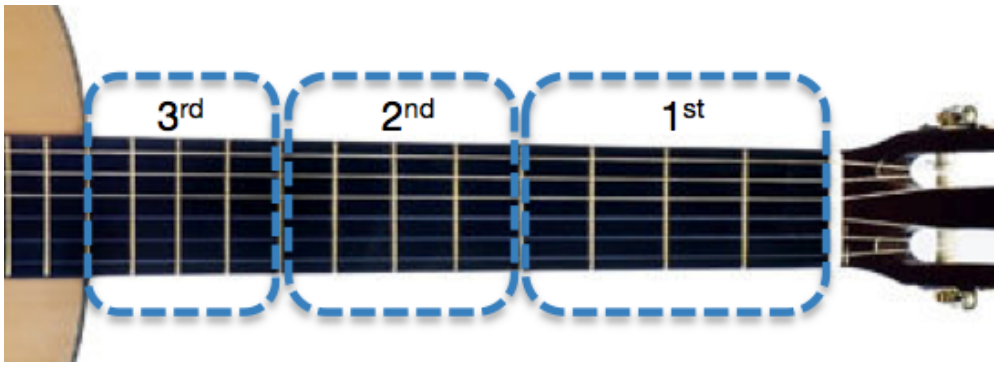


Figure 3.17: Representation of our recording regions on a guitar fretboard.

Simple Exercises

Our chromatic exercises are the modified version of the Carlvaro's guitar exercises (Carlevaro, 1974). We recorded a collection of ascending chromatic scales that contain non-articulated, legato or glissando and vibrato notes.
The performer was asked to play chromatic scales in three different regions of the guitar fretboard. Specifically, we recorded notes from the first 12 frets of the fretboard. As shown in Figure 3.17 we divided 12 frets into 4 fret equal regions.
For our second set, again we recorded exercises at three positions. But this time rather than mixing legato and glissando, first we recorded all three positions with legato, vibrato and non-articulated notes and then recorded all three positions with glissando, vibrato and non-articulated notes. At the end we had 108 notes with 18 legato, 18 glissando and 36 vibrato examples.

Classical Guitar Recordings			
Recordings	**Legato**	**Glissando**	**Vibrato**
Legato&Vibrato	18/17	|	18/10
Glissando&Vibrato	|	18/18	18/10

Table 3.3: Performance of our model applied to second test set.

For the test set, for the legato & vibrato recordings we determined 17 legato articulations out of 18. In addition to 17 we had one false positive result for legato excerpts, which is a vibrato note classified as a legato. Also for the glissando&vibrato recordings we had one false positive which is actually a vibrato note. In the second set the correct hit rate of vibrato is lower than the first set. The reason was that in the first set, the vibratos were much more obvious and exaggerated.

The vibratos that we missed had a sonically less obvious vibrato character than the ones that we determined correctly. The fact that the ones that we missed have *Vibrato Frequency* value between 3 and 4, and *Peak Deviation* value between 30% and 40% also proves our previous statement; our system successfully determined the obvious vibratos.

Guitar strings have different tension strengths in the parts of the freeboard. Parts that are closer to the neck and bridge have higher tension than to the middle parts of the strings. So, in our case the string parts in first region have the strongest tension value, second region has lighter and third region has the lightest tension value. Therefore applying vibrato is harder in first section compared the second and third sections. Our results are consistent with these properties. In the first set we determine 3 out of 6 in first region and 10 out of 12 in the second and third regions. In the second set, in legato&vibrato recordings we determine 2 out of 6 in first region and 8 out of 12 in the second and third regions. In glissando&vibrato recordings we determine 2 out of 6 in first region and 8 out of 12 in the second and third regions.

Commercial Recordings

We also tested our model with commercial recordings. We used 7 different professional guitarists' recordings of first 9 bars of Villa Lobos Prelude Number 4. The recordings were stereo. Although in all of them guitars were recorded in mono, because of the reverberation there was a big stereo field. Also most of the recordings has noticeable amount of noise. Our first challenge was to decrease reverb and noise as much as possible without harming the audio content. In order not to introduce phase information, rather than summing two channels, we used only the left channel. We applied -6db of noise reduction and a low pass filter with cut off frequency of 8200Hz, and Q value of 0.71. By the help

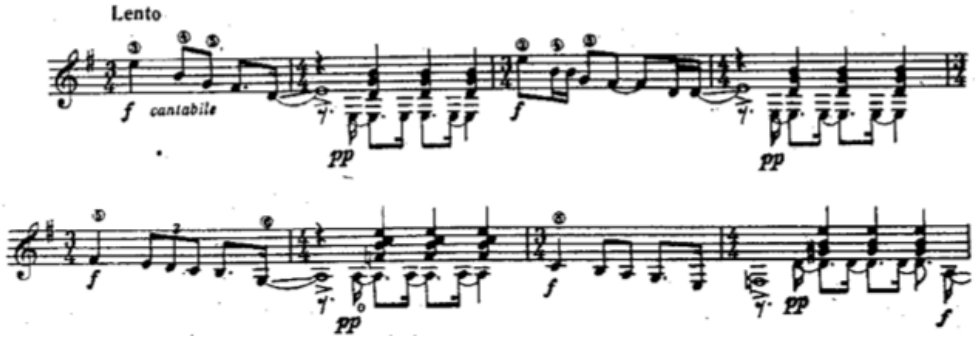


Figure 3.18: Villa Lobos' Prelude Number IV.

of low pass filter we managed to decrease reverb in audio files.
Another difficulty that we faced was the polyphony of the guitar. Our model is mainly builded for the analysis of melodies in the monophonic way. Because of the polyphonic content in some cases our model failed. However in almost all monophonic sections our model worked perfectly. Moreover, most of the *False Positives* were in the polyphonic regions.
As shown in Figure 3.18, in the score, vibratos were not marked. Therefore for each piece we annotated the places of the vibratos by hand. Although the number of notes are constant in the score, the number notes that were played by each performer also varied.

Villa Lobos 5 Preludes, Prelude IV				
Performer	**#Notes**	**#Vib.**	**T.P.**	**F.P.**
Alvaro Pierri	28/61	2	2	1
Gerald Garcia	53/61	5	3	1
Joseph Bacon	63/61	5	2	2
Kevin McCormic	56/61	-	-	2
Marcelo Kayath	58/61	6	4	3
Micheal A. Nigro	57/61	8	6	3
Pedro Ibanez	61/61	1	1	1

Table 3.4: Performance of our model applied to Villa Lobos' Prelude Number IV

Another important observation was that the vibratos that we detected from commercial recordings sound much more expressive than the ones that we detected in basic exercises. The reason was that in the basic exercises, performer played in an isolated studio environment by looking at the score that contains basic chromatic exercises where the places of the vibratos were marked. However, in the commercial recordings, most likely the performer was mastered

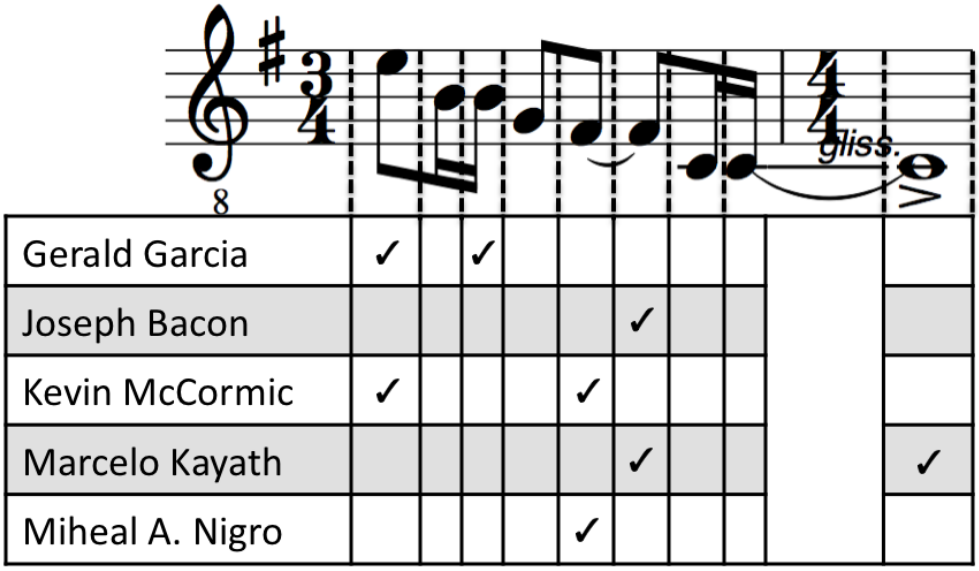


Figure 3.19: Places where performers applied vibrato.

the piece and also could apply vibrato where he/she thought appropriate, expressive etc. Second situation is much more comfortable than the first one, also since when the vibratos are in the context of a melody or harmony, they have much more meaning than the ones in a chromatic scale. In other words, vibratos does not make sense without a context.

We realized that different performers applied vibratos in different places of the melody. Figure 3.19 shows one melody from our test score. Each column represents the corresponding note in the melody and each row represents one performer. 'Check' symbol means that corresponding performer applied vibrato in the note above. We also realized that different performers applied different kinds of vibratos. For instance, Kevin McCormic's vibratos were faster (higher vibrato frequency), than Gerald Garcias, or Pedro Ibanez started vibrato little bit later than the other performers.

In this section we presented a system to identify vibratos on classical guitar recordings. We successfully determined the places of vibratos both from a collection of chromatic exercises recorded by a professional guitarist and commercial recordings. There are two main parts in our model, first one is the region extraction, and second one is the decision.

Our two evaluation experiments, shows us that vibrato has different characteristics according to context. Compared to chromatic exercises, vibratos in Villa Lobos Prelude Number 4 were easier to determine.

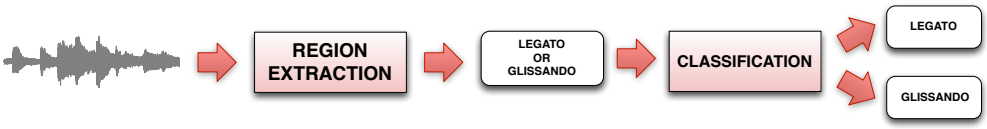


Figure 3.20: Our diagram for the legato and glissando analysis and automatic detection.

3.5 Legato and Glissando Extraction

In this section, as we did in vibrato extraction, we are focusing on the classification of legatos and glissandos. Specifically, we present an automatic classification system that uses the output of the extraction region input. We propose a new system able to determine and classify two expressive articulations from audio files. For that purpose, we analyzed the regions that were identified as candidates of expressive articulations by the extraction module (see Section 3.3).

In both, legato and glissando, left hand is involved in the creation of the note onset. In the case of an *ascending legato*, after plucking the string with the right hand, one of the fingers of the left hand (not already used for pressing one of the frets), presses a fret causing another note onset. *Descending legato* is performed by plucking the string with a left-hand finger that was previously used to play a note (i.e. pressing a fret).

The case of *glissando* is similar but this time after plucking one of the strings with the right hand, the left hand finger that is pressing the string is slipped to another fret also generating another note onset. When playing legato or glissando on guitar, it is common for the performer to play more notes within a beat than the stated timing enriching the music that is played. A powerful legato and glissando can be differentiated between each other easily by ear. However, in a musical phrase where legato and glissando are not isolated, it is hard to differentiate among these two expressive articulations.

Figure 3.21 shows fundamental frequency values and right hand onsets. X-axis represents the time domain bins and Y-axis represents the frequency. Also in Table 3.5 F0 values are shown. In Figure 3.21, vertical lines depict the attack and release parts respectively. In the middle there was a change in frequency, which was not determined as an onset by the first module. Although it seems an error, it was a success result for our model. Specifically, in this phrase there was a glissando, which was a left hand articulation, and was not identified as an onset.

In Figure 3.21, the first portion of the Figure 3.21 was zoomed. The first and the last lines were the plucking onsets identified by onset detection algorithm. The first line was the place where attack finishes. The second dashed line was the place where release starts.

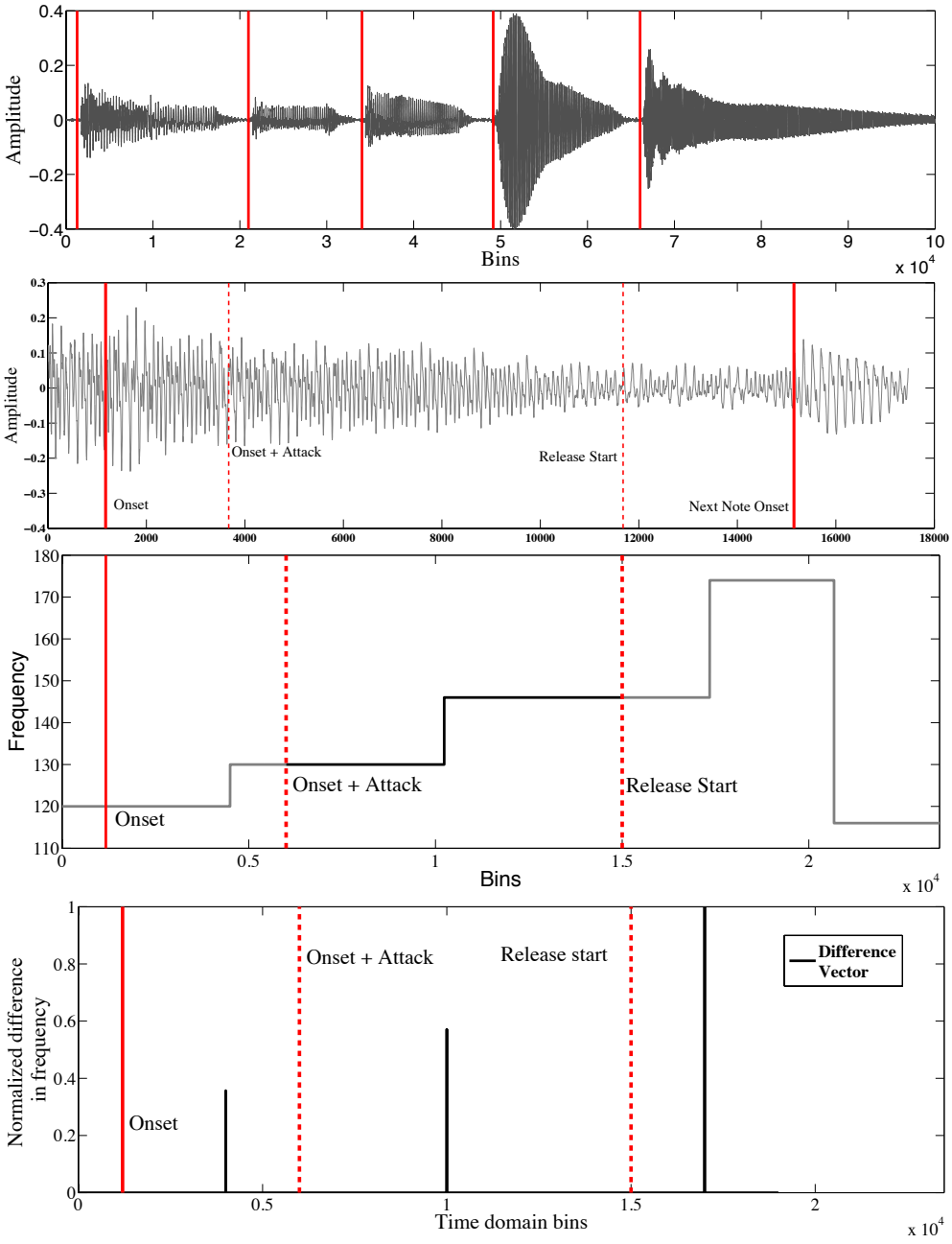


Figure 3.21: Top Figure - Onsets that were detected in the plucking detection section. **Middle Figure** - Features of the portion between two onsets. **Middle Figure** - Example of a glissando articulation. **Bottom Figure** Difference vector of pitch frequency values of fundamental frequency array.

Note Start (ms.)	Fundamental Frequency
0.02	130
0.19	130
0.37	130
0.46	146
0.66	146
0.76	146
099	146
1.10	146
1.41	174
1.48	116

Table 3.5: Output of the pitch-change detection module.

3.5.1 Classification

The *classification* module analyzes the regions identified by the extraction module and labels regions as legato or glissando. A diagram of the classification module is shown in Figure 3.22. In this section, first, we describe our research to select the appropriate descriptor to analyze the behavior of legato and glissando. Then, we explain the new two components, *Models Builder* and *Detection*.

Selecting a Descriptor

After extracting the regions which contain legato or glissando candidates, the next step was to analyze them. Because each one should present different characteristics in terms of changes in amplitude, aperiodicity, or pitch (Norton, 2008), we focused the analysis on comparing these deviations.
Specifically, we built representations of these three features (amplitude, aperiodicity, and pitch). Representations helped us to compare different data with different length and density. As we stated above, we were mostly interested in changes: changes in fundamental frequency, changes in amplitude, etc. Therefore, we explored the peaks in the examined data because peaks are the points where changes occur.
As an example, Figure 3.23 shows, from top to bottom, amplitude evolution, pitch evolution, and changes in aperiodicity for both legato and glissando. As both figures show, *glissando* and *legato* examples, the changes in pitch are similar. However, the changes in amplitude and aperiodicity present a characteristic slope.
Thus, as a first step we concentrated on determining which descriptor could be used. To make this decision, we built models for both aperiodicty and amplitude by using a set of training data. As a result, we obtained two models

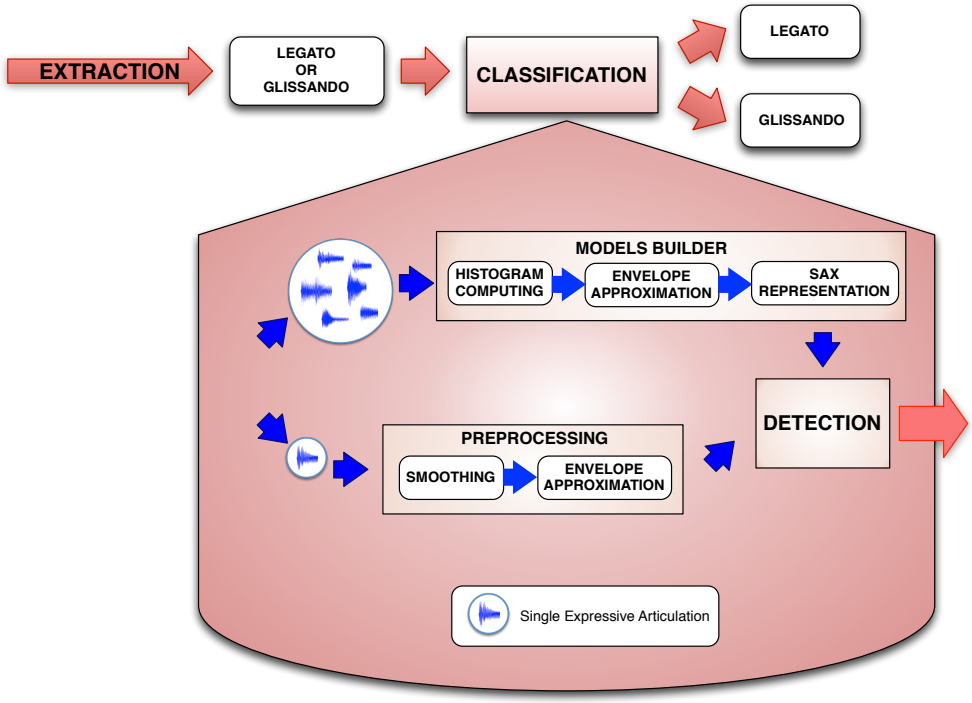


Figure 3.22: Classification module diagram.

(for amplitude and aperiodicity) for both legato and glissando as is shown in Figure 3.23. Analyzing the results, amplitude was not a good candidate because the models behave similarly. In contrast, aperiodicity models presented a different behavior. Therefore, we selected aperiodicity as the descriptor. The details of this model construction will be explained in next section.

Preprocessing

Before analyzing and testing our recordings, we applied two different preprocessing techniques to the data in order to make them smoother and ready for comparison: *Smoothing* and *Envelope Approximation*.

Smoothing

As expected, the aperiodicity signal of the audio portion we are examining includes noise. Our first concern was to avoid this noise and to obtain a nicer representation. In order to do that, first we applied a 50 step running median smoothing. Running median smoothing is also known as median filtering. Median filtering is widely used in digital image processing because under certain conditions, it preserves edges whilst removing noise. In our situation since we were interested in the edges and

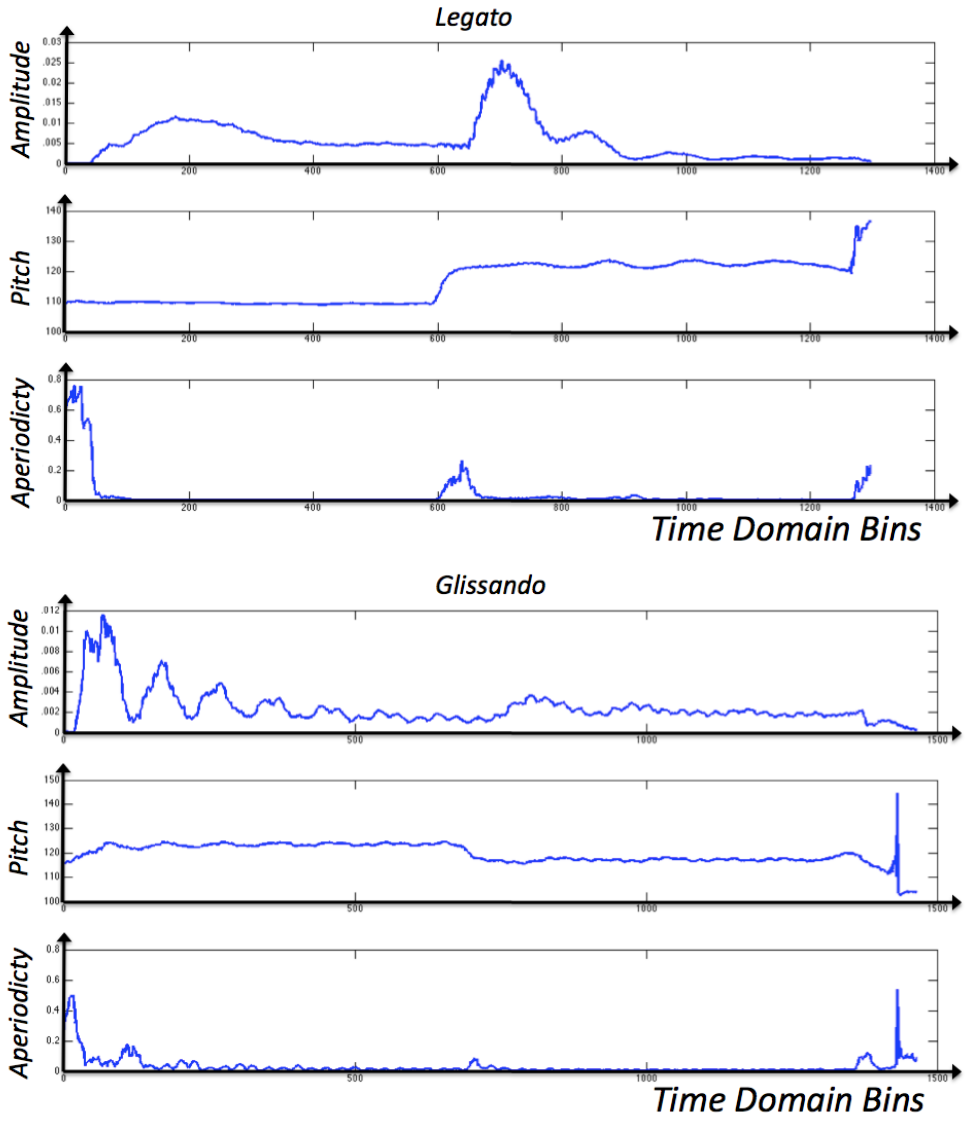


Figure 3.23: From top to bottom, representations of amplitude, pitch and aperiodicty of the examined regions.

in removing noise, this approach fitted our purposes. By smoothing, the peak locations of the aperiodicity curves become more easy to extract. In Figure 3.7, the comparison between aperiodicity and smoothed aperiodicity graphs exemplify the smoothing process and shows the results we pursue.

Envelope Approximation

We used the same implementation as we explained in Section 3.2.3.

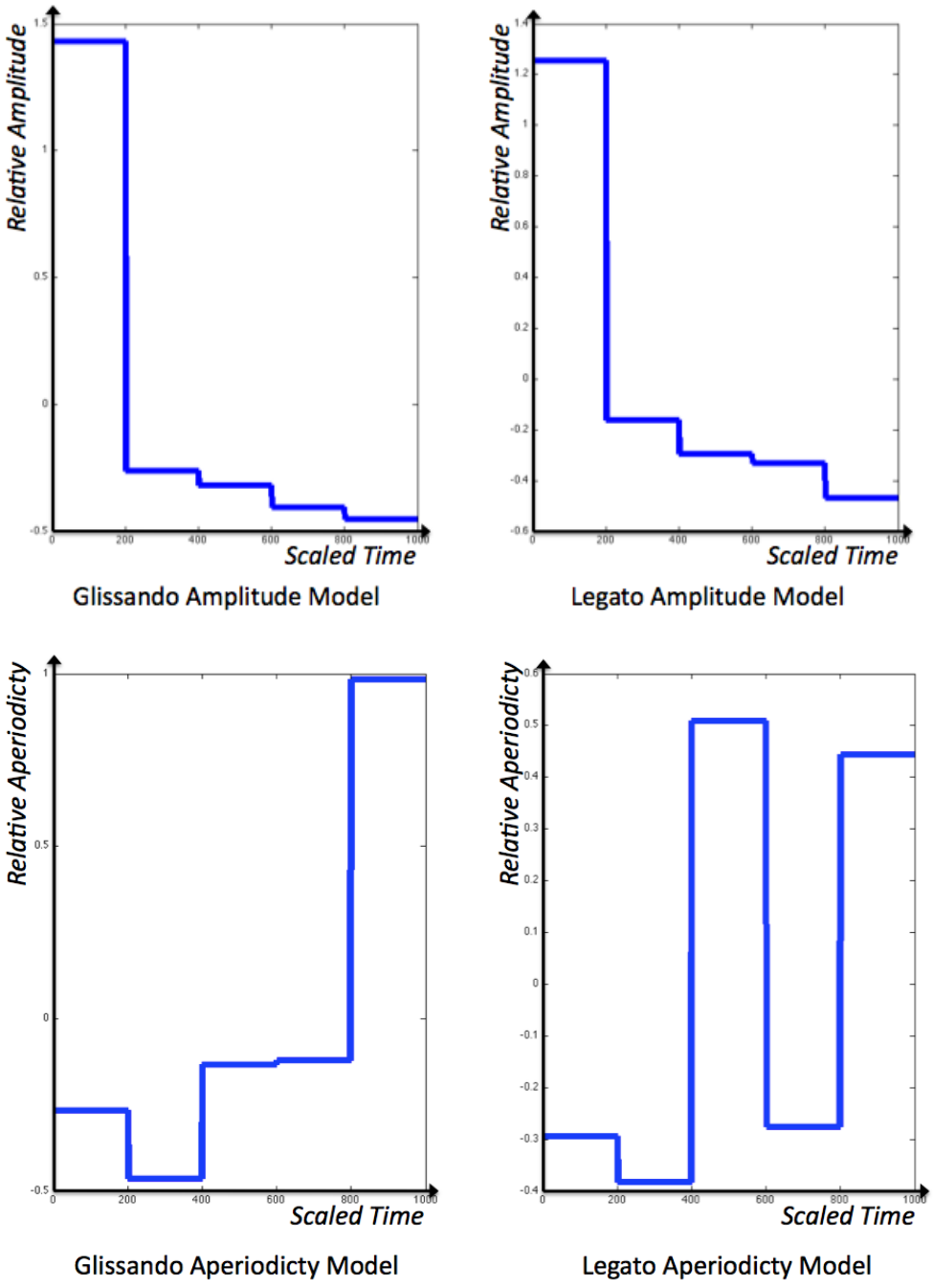


Figure 3.24: Models for Legato and Glissando

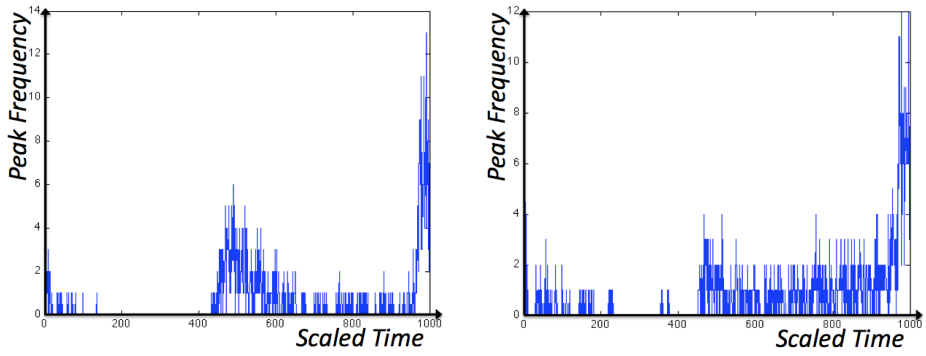


Figure 3.25: Peak histograms of legato and glissando training sets.

In our case all the recordings were performed at 60bpm and all the notes in the recordings are 8^{th} notes. That is, each note takes half a second, and each legato or glissando portion takes 1 second. We recorded with a sampling rate of 44100, and we did our analysis by using a hop size of 32 bins, i.e. $44100/32 = 1378$ bins. We knew that this was our highest limit. For the sake of simplicity, we scaled our x-axis to 1000 bins.

Building the Models

After applying the pre-processing techniques, we obtained equal length aperiodicity representations of all our legato and glissando portions. Next step was to construct models for both legato and glissando by using these data. In this section we describe how we constructed the models shown in Figure 3.24. The following steps were used to construct the models: *Histogram Calculation*, *Smoothing* and *Envelope approximation* (explained in Section 3.2.3), and finally, *SAX representation*.

Histogram Calculation

We used this technique to calculate the peak density of a set of data. Specifically, a set of recordings containing 36 legato and 36 glissando examples (recorded by a professional classical guitarist) was used as training set. First, for each legato and glissando example, we determined the peaks. Since we wanted to model the places where condensed peaks occur, this time we used a threshold of 30 percent and collect the peaks with amplitude values above this threshold. Notice that the threshold is different than we used in envelope approximation. Then, we used histograms to compute the density of the peak locations. Figure 3.25 shows the resulting histograms.

After constructing the histograms, as shown in Figure 3.25, we used our envelope approximation method to construct the envelopes of legato and

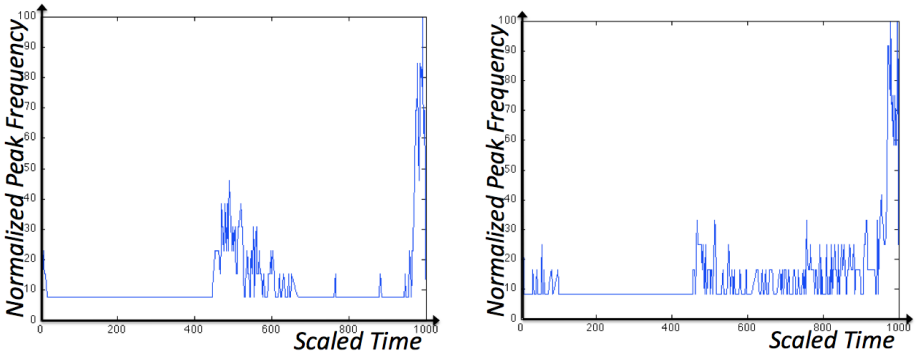


Figure 3.26: Final envelope approximation of peak histograms of legato and glissando training sets.

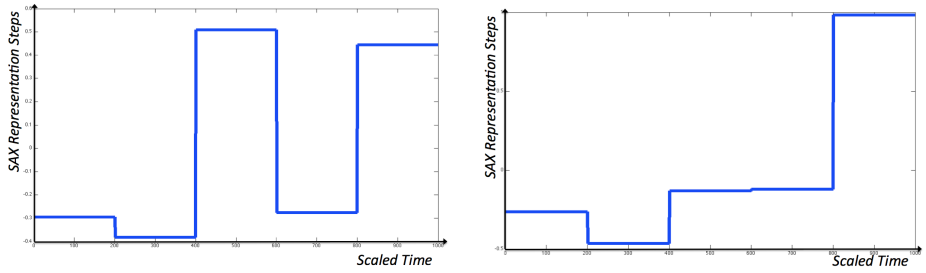


Figure 3.27: SAX representation of legato and glissando final models.

glissando histogram models (see Figure 3.26).

SAX: Symbolic Aggregate Approximation

Although the histogram envelope approximations of legato and glissando in Figure 3.26 were close to our purposes, they still included noisy sections. Rather than these abrupt changes (noises), we were interested in a more general representation reflecting the changes more smoothly. SAX (Symbolic Aggregate Approximation) is a symbolic representation used in time series analysis that provides a dimensionality reduction while preserving the properties of the curves (Lin et al., 2007). Moreover, SAX representation makes the distance measurements easier. Then, we applied the SAX representation to histogram envelope approximations.

As we mentioned in envelope approximation, Section 3.2.3, we scaled the x-axis to 1000. We made tests with step sizes of 10 and 5. As we will report in the experiments, Section 3.5.2, an step size of 5 gave better results. We also tested with step sizes lower than 5, but the performance clearly decreased. Since we were using an step size of 5, each step be-

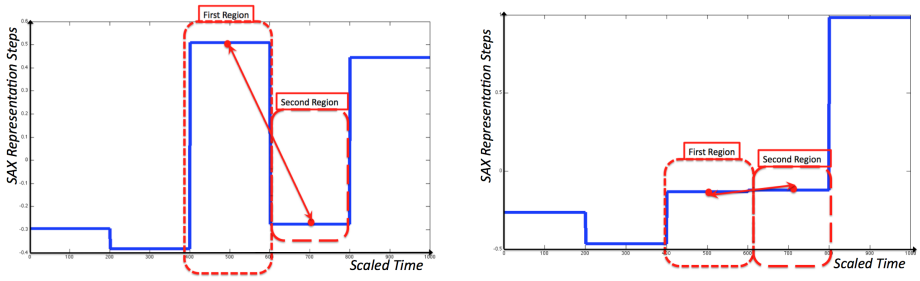


Figure 3.28: Peak occurrence deviation.

comes 200 bins in length. After obtaining the SAX representation of each expressive articulation, we used our distance calculation algorithm which we are going to explain in the next section.

Detection

After obtaining the SAX representation of glissando and legato models, we divided them into 2 regions, a first region between bins 400 and 600, and a second region between bins 600 and 800 (see Figure 3.28). For the expressive articulation excerpt, we had the envelope approximation representation with the same length of the SAX representation of final models. So, we could compare the regions. For the final expressive articulation models (see Figure 3.27) we took the value for each region and compute the deviation (slope) between these two regions. We performed this computation for both legato and glissando models separately.

We also computed the same deviation for each expressive articulation envelope approximation (see Figure 3.29). But this time, since we did not have SAX representation, for each region we did not have single values. Therefore, for each region we computed the local maxima and took the deviation (slope) of these two local maximas. After obtaining this value, we may compare this deviation value with the numbers that we obtained from both final models of legato and glissando. If the deviation value was closer to the legato model, the expressive articulation would be labeled as a legato and vice versa.

3.5.2 Experiments

The goal of the experiments realized was to test the performance of our model. Since different modules have been designed, and they work independently of each other, we tested Extraction and Classification modules separately. After applying separate studies, we combined the results to assess the overall performance of the proposed system.

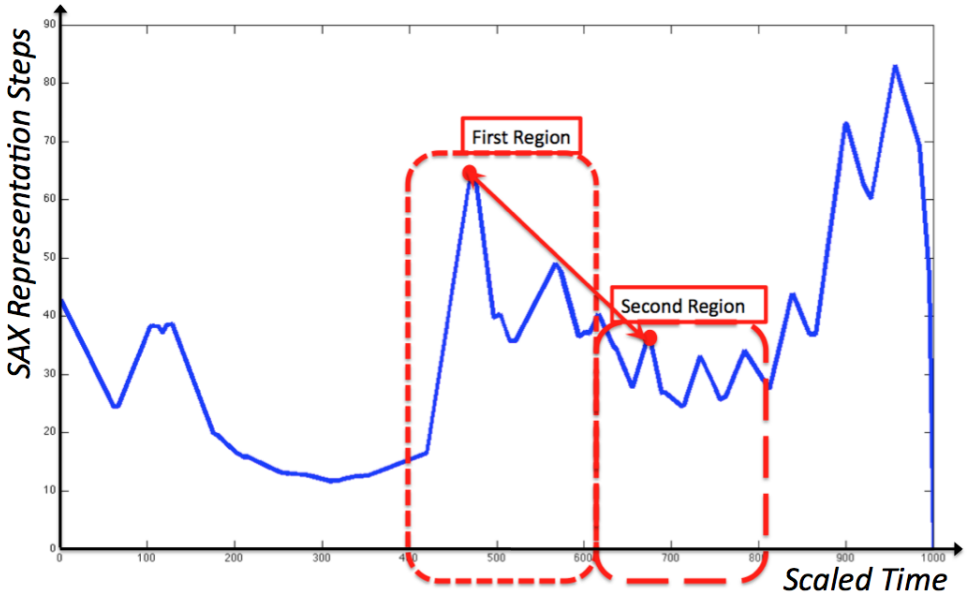


Figure 3.29: Expressive articultion difference.

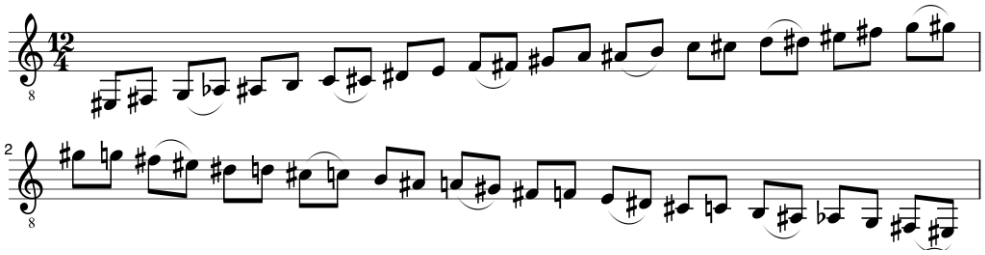


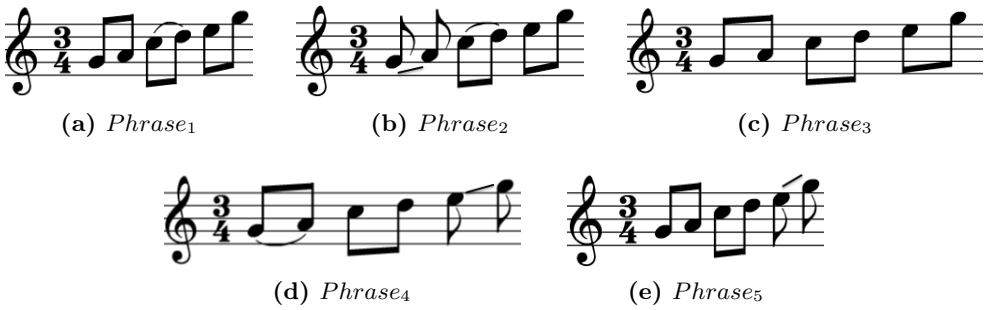
Figure 3.30: Legato Score in first position.

Legato and glissando could be played in ascending or descending intervals. Thus, we were interested in studying the results distinguishing among these two movements. Additionally, since in a guitar there are three nylon strings and three metallic strings, we also studied the results taking into account these two sets of strings.

Recordings

Borrowing from Carlevaro's guitar exercises (Carlevaro, 1974), we recorded a collection of ascending and descending chromatic scales. Legato and glissando examples were recorded by a professional classical guitar performer[3]. The

[3] http://www.iiia.csic.es/guitarLab/gallery



(a) $Phrase_1$ **(b)** $Phrase_2$ **(c)** $Phrase_3$

(d) $Phrase_4$ **(e)** $Phrase_5$

Figure 3.31: Short melodies.

performer was asked to play chromatic scales in three different regions of the guitar fretboard. Specifically, we recorded notes from the first 12 frets of the fretboard where each recording concentrated on 4 specific frets. The basic exercise from the first fretboard region is shown in Figure 3.30.

Each scale contains 24 ascending and 24 descending notes. Each exercise contains 12 expressive articulations (the ones connected with an arch, Figure 3.30). Since we repeated the exercise at three different positions, we obtained 36 legato and 36 glissando examples. Notice that we also performed recordings with a neutral articulation (neither legatos nor glissandos). We presented all the 72 examples to our system.

We also recorded a small set of 5-6 note phrases. They include different articulations in random places (see Figure 3.31). As shown in Table 3.7, each phrase includes different combinations of expressive articulations varying from 0-2. For instance, $Phrase_3$ (see Figure 3.31c) does not have any expressive articulation and $Phrase_4$ (see Figure 3.31d) contains the same notes of $Phrase_3$ but including two expressive articulations: first a legato and next a glissando.

Experiment Results

As explained in *building the models* section, we performed experiments applying different step sizes for the SAX representation. Specifically (see results reported in Table 3.6), we may observe that a step size of 5 is the most appropriate setting. This result corroborates that a higher resolution when discretizing was not required and demonstrates that the SAX representation provides a powerful technique to summarize the information about changes.

The overall performance for legato identification was 83.3% and the overall performance for glissando identification was 80.5%. Notice that identification of ascending legato reached a 85% of accuracy whereas descending legato achieved only a 53.6%. Regarding glissando, there was no significant difference between ascending or descending accuracy (58.3% versus 54.4%). Finally, analyzing

Recordings	Accuracy
Ascending Legato	85.0 %
Descending Legato	53.6 %
Ascending Glissando	58.3 %
Descending Glissando	54.4%
Legato Nylon Strings	68.0 %
Legato Metallic Strings	69.3 %
Glissando Nylon Strings	58.3 %
Glissando Metallic Strings	54.4 %

Table 3.6: Performance of our model applied to chromatic exercises.

Excerpt Name	Ground Truth	Detected
$Phrase_1$	1	2
$Phrase_2$	2	2
$Phrase_3$	0	0
$Phrase_4$	2	3
$Phrase_5$	1	1

Table 3.7: Results of extraction module applied to short phrases.

the results when considering the string type, the results presented a similar accuracy on both nylon and metallic strings.

We also tested the performance of our model with short melodies. Analyzing the results, the performance of our model was similar to the previous experiments, i.e. when we analyze single articulations. However, in two phrases where a note was played with a soft right-hand plucking, these notes were proposed as legato candidates ($Phrase_1$ and $Phrase_4$).

The final step of the model is to annotate the sound fragments where a possible legato or glissando was detected. Specifically, to help the system's validation, the whole recording was presented to the user and the candidate fragments to expressive articulations were colored. As example, Figure 3.32 shows the annotation of $Phrase_2$ (see score in Figure 3.31b). $Phrase_2$ has two expressive articulations that correspond with the portions colored in black.

Our proposal was to use aperiodicity information to identify the articulation and a SAX representation to characterize articulation models. Applying a distance measure to the trained models, articulation candidates were classified as legato or glissando.

We conducted experiments to validate our proposal by analyzing a collection of chromatic exercises and short melodies recorded by a professional guitarist. Although we were aware that our current system might be improved, the re-

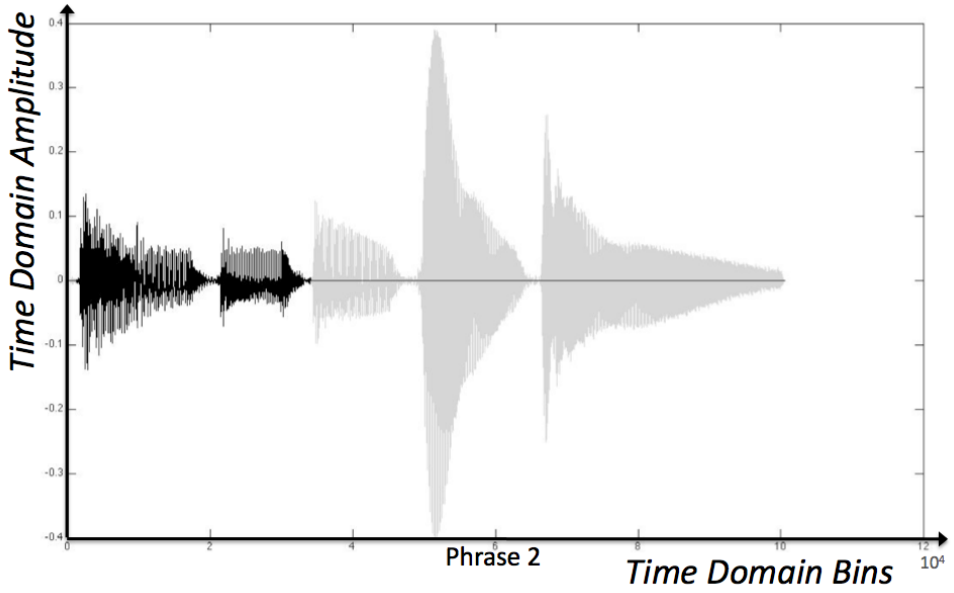


Figure 3.32: Annotated output of $Phrase_2$

sults showed that it was able to identify and classify successfully legatos and glissandos. As expected, legato were more easy to identify to glissando. Specifically, the short duration of a glissando was sometimes confused as a single note attack.

3.6 Conclusion

In this chapter we presented a system that combines several state of the art analysis algorithms to identify guitar left hand expressive articulations such as legatos, glissandos, and vibratos. In the first part we explained our feature extraction methodology. Then, by using the extracted features, we reported experiments to validate our proposal in an initial set of chromatic exercises, in a collection of short melodies recorded by a professional guitarist, and in a the context of commercial recordings.

CHAPTER 4

Onset Deviation Analysis

Performance of a musical piece include several deviations from the written score. One of the most common ones are the timing deviations, i.e. temporal anticipations or delays of notes.

4.1 Introduction

In this chapter we study the analysis of musical expressivity from the perspective of timing variations in the context of classical guitar pieces. The choice of guitar recordings represents an interesting test corpus, as almost no studies (Aho & Eerola, 2012) on timing deviations consider this instrument. The use of a semi-automatic approach to onset detection (see Section 3.2 and Section 4.3.3) allows us to go from the analysis of single, experiment-specific performances to medium-scale real-world music collections (see Section 4.3.1). Since music has a structure, timing deviations can be analyzed at different musical levels. We have analyzed deviations in two different levels. First one is the analysis of onset deviations at note level. Second level is more global, the analysis of onset deviations at a measure level. The results from each of the two analyzed onset deviation levels suggest that the predictive power of onset deviations are statistically significant than the chance. Moreover, analyses (see Section 4.2), suggest that timing variations are reliable predictors of the musical pieces.
By formulating our hypothesis as a classification problem and, thus, within a strong statistical framework (Hastie et al., 2009; Mitchell, 1997; Witten & Frank, 2005), we gain objective and quantitative evidence for the piece-dependent nature of onset deviations. To show that the predictive power of onset deviation sequences is generic and not biased towards a specific classification scheme, we consider five different machine learning principles (Hastie et al., 2009; Mitchell, 1997; Witten & Frank, 2005): decision tree learning, instance-based learning, linear regression, Bayesian learning, and support vector machines (for more details see Section 4.3.8).

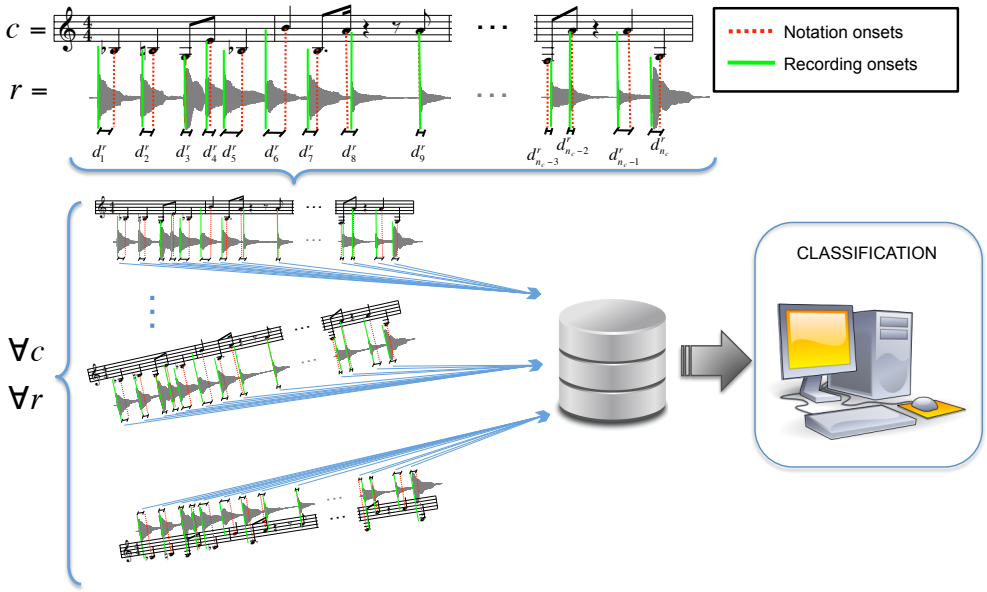


Figure 4.1: Methodology overview. The difference $d^r_{i_c}$ between notation and recording onsets is computed, $\mathbf{d}^r_{i_c} = \{d^r_1, d^r_2, \dots d^r_{i_c}\}$.

In our experiments we prove that onset deviations are characteristic of a given composition, up to the point of allowing the automatic identification of the musical piece they belong to. This hypothesis is validated at different levels of onset deviation sequences used as the feature vector for the classifiers.

4.2 Levels of Onset Deviation

A music piece is structurally organized at different levels. With the purpose of determining how different levels influence timing deviations, in our study we consider two different levels. First level of analysis, as well as the most basic, is the note level. Our second level of analysis is the measure. A musical measure is a segment of time defined by a given number of beats, and constitutes one of the basic regular structures in a given piece.

The simplest way of analyzing onset deviations in a sequence of notes is to count them and compute their distributions, i.e. using histograms. We use histograms because we believe that the distributions of the deviations provide a simple and powerful summary of timing deviations. Furthermore, as a higher level, we may consider pairs of adjacent consecutive n elements, i.e. N-Grams. In our analysis we are using bigrams, i.e. n-grams with the $n = 2$. For a brief understanding we refer to Figure 4.2. We will explain how we compute histogram and bi-gram models for each audio file in Section 4.3.7.

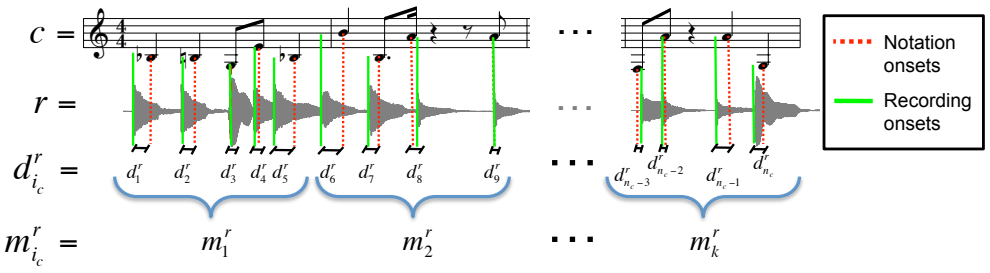


Figure 4.2: Levels of deviations. c is the composition, r is the performance of this composition, d is the note level onset deviation and m is the musical measure level onset deviation.

As a summary, we extract 2 different levels of deviations: note level (n) and measure level (m) , see Figure 4.2. Furthermore each level includes 2 additional cases: histogram and bi-gram models. So for each level, by counting the base analysis we have 3 different models, n_0, n_1, n_2, and m_0, m_1, m_2. For instance, for the note level deviations, n_0 will be pure note deviations (see Figure 4.2, $\{d^r_{i_c}, d^r_{i_c+1}, \dots d^r_{i_c+l}\}$), n_1 will be histogram model, and n_2 will be the bi-gram model. Our aim was to analyze note deviations both from a micro and macro level, with the purpose of determining how each approach enlightens new ways of understanding onset deviations.

Note Level Deviations - n_0

Our first level of analysis is the single note level. We conduct our analysis with the set of consecutive note onsets. As shown in the top Figure 4.3, each d_i represents the corresponding deviation value of a note. Then, level n_0 is the set of all deviation values represented as a time series, $\mathbf{d}^r_{i_c} = \{d^r_1, d^r_2, \dots d^r_{i_c}\}$.

Histogram Model - n_1

After extracting the deviation values, we may construct a b-bin equal step histogram from the these values. Each audio file in our music collection is represented with a b-bin histogram. An example representation is shown in top figure of Figure 4.3. The graph corresponds to the onset deviation values of each note in the score. We divide y-axis (numerical values of the deviations) into b equal steps. The histogram gathers the counts of each onset deviation that fall into the corresponding bin. We call this model as n_1.

N-gram Model - n_2

Each b-bin histogram is a representation of the onset deviation distribution.

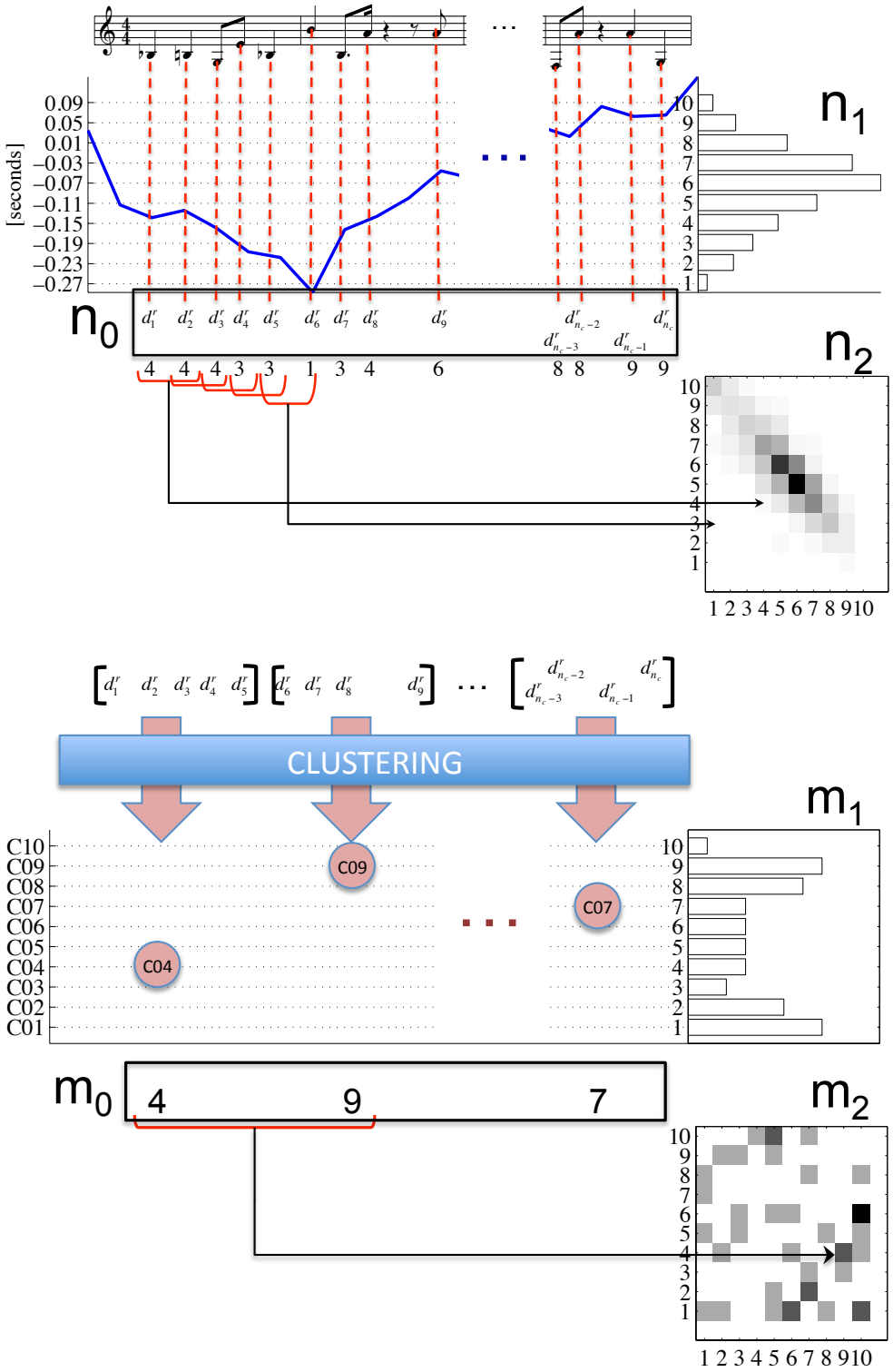


Figure 4.3: Methodology of all levels and models

Since each onset deviation value d_i can be mapped to a bin value b_i, we can represent our onset deviation sequence, $\mathbf{d}^r_{i_c} = \{d^r_1, d^r_2, d^r_3, \dots d^r_{i_c}\}$ as a sequence of their corresponding bin numbers, $\mathbf{b}^r_{i_c} = \{b^r_7, b^r_4, b^r_1, \dots b^r_{i_c}\}$. Then, the sequences of bin values will be used to construct a matrix gathering the bi-gram information. Specifically, since we have b-bins, we have $b \times b$ transition possibilities for each bi-gram, i.e. a bi-gram matrix of a given audio is modeled as follows:

$$\begin{bmatrix} (1,1) & (1,2) & (1,3) & \dots & (1,b-2) & (1,b-1) & (1,b) \\ (2,1) & (2,2) & (2,3) & \dots & (2,b-2) & (2,b-1) & (2,b) \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ (b-1,1) & (b-1,2) & (b-1,3) & \dots & (b-1,b-2) & (b-1,b-1) & (b-1,b) \\ (b,1) & (b,2) & (b,3) & \dots & (b,b-2) & (b,b-1) & (b,b) \end{bmatrix}$$

Then, each cell (i, j) in the matrix will gather the number of consecutive notes in the recording where first note presents a deviation i and second note presents a deviation j. For instance, if in our sequence of note deviations we have b^r_7 followed by b^r_4, in our bi-gram matrix we will increase the count of $(7, 4)$ by 1. A more complete example of the construction of the n-gram model is shown in Figure 4.3. For instance, in our note sequence n_0 (top of the figure), first four notes are represented as 4, 4, 4 and 3. Thus, in our bigram matrix, for the first three steps we increase the indexes of bi-gram cells $(4, 4)$ by 2 and $(4, 3)$ by 1. We proceed this action by sliding one note in each step till to the end of the piece. So if we had total number of n notes, in our bi-gram matrix we had $n - 1$ counts.

Musical Measure Level Onset Deviations - m_0

The second level of deviations considered is the measure level, m. In order to extract measure level deviations we used our validated hand-annotations of measure starting positions (see **??**). In this level, rather than analyzing each note's deviation, we model the deviation of each single musical measure. We use clustering techniques to model measure level deviations. We will explain the technical details in a specific section, Section 4.3.7, but basically we define k different deviation behaviors and we map each measure to its corresponding deviation behavior. As a final representation, we represent each measure with it's corresponding cluster index. Specifically, first we represent each audio file as $\mathbf{Comp}^r_k = \{m_1, m_2, m_3, \dots m_k\}$, see Figure 4.2, where $Comp$ is the composition, r is the performance of this composition, and each m is the onset deviation vector of the corresponding notes inside the measure. These vectors are used as input of a clustering step to obtain a clustering index for each measure , $\mathbf{Comp}^r_k = \{C_1, C_4, C_6, \dots\}$.

Histogram Model - m_1

After representing each measure with it's corresponding cluster index, we

applied the same analysis approach we performed at note level. However, at a measure level we have a discrete number of values, k, that cannot be interpreted as numerical values. They are only the names of cluster indices. Therefore, we could not treat the vector of clustering indexes of a composition, $\mathbf{Comp}_k^r = \{C_1, C_4, C_6, \dots\}$ as a time-series as we did in n_0. Thus, the histogram represents the frequency of occurrence of each cluster index.

Bi-gram Model - m_2

Similarly to the note level, bi-gram models at the measure level can be constructed by analyzing pairs of consecutive measures. Specifically, since the clustering process provides k-measure patterns, there are $k \times k$ transition possibilities for the bi-gram representation at the measure level. That is:

$$\begin{bmatrix} (1,1) & (1,2) & (1,3) & \dots & (1,b-2) & (1,k-1) & (1,k) \\ (2,1) & (2,2) & (2,3) & \dots & (2,k-2) & (2,k-1) & (2,k) \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ (k-1,1) & (k-1,2) & (k-1,3) & \dots & (k-1,k-2) & (k-1,k-1) & (k-1,k) \\ (k,1) & (k,2) & (k,3) & \dots & (k,k-2) & (k,k-1) & (k,k) \end{bmatrix}$$

For instance, from the measure sequence m_0 in Figure 4.3 (bottom of the figure), where first two measures are represented as C_{04} and C_{09}, in the bi-gram matrix the index of $(4, 9)$ is increased by 1. This process is repeated by sliding one measure in each step till to the end of the piece.

4.3 Experiment Setup

The purpose of the experiments is to analyze the predictive power of onset deviations at the different representation levels. Up to now we explained briefly our different levels of onset deviations. In this Section we will go deeper and will explain how we construct our setup for the experiments. Before the detailed explanation of the experiment setup, we will describe first the music collection used in the experiments. Furthermore, at the end of this section, we will present the classification algorithms used in the experiments and the methodology adopted to analyze the results.

4.3.1 Music Collection

In our music collection we have 10 different compositions, and each composition is performed by 10 different guitar players, thus yielding a total of 100 recordings. However, some performances of different compositions have been interpreted by the same player. In total, we have 82 different guitar players, with some of them playing between 2 and 5 pieces (a Table with player and recording details can be seen in Appendix B - Music Collection Details). The collection includes well-known guitar players such as Andrés Segovia, John

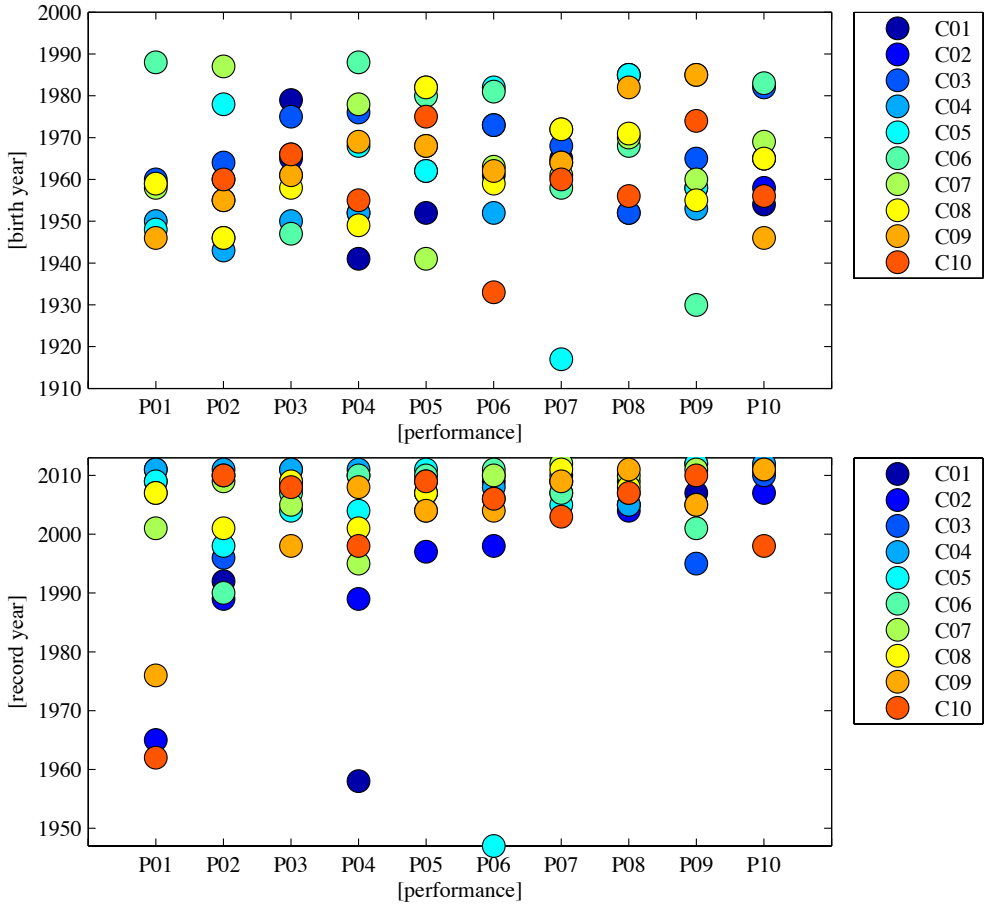


Figure 4.4: Information about music collection. In our music collection there 10 compositions, C01,C02...,C10 and each composition has 10 different performance, X-axis, P01,P02,...,P10. Each color represents a composition and each circle represents a performance.

Williams, Manuel Barrueco, Rey de la Torre, Robert Westaway, and Stanley Myers. In order to encompass different epochs, we chose compositions spanning four different periods: baroque, classical, romantic, and modern (Table 4.1). Recording years go from 1948 to 2011, Figure 4.4, and the number of onsets per score measure varies between 1 and 16 Table 4.1.

ID	Composition	Composer	Composition year	Historical period	Tempo	Number of notes	Shortest note	Longest note
C01	BWV 999	J.S. Bach	1720	Baroque	Allegretto	505	1/4	3
C02	BWV 1007	J.S. Bach	1720	Baroque	Andante	641	1/4	2
C03	La Catedral, Prelude	A. Barrios	1921	Modern	Lento	379	1/4	2
C04	C minor Prelude	A. Barrios	1920	Modern	Moderato	361	1/6	1
C05	Cavatina	S. Myers	1970	Modern	Andante	515	1/8	3
C06	Romance	Anonymous	ca. 1800	Classical	Andante	680	1/3	2
C07	Adelita	F. Tarrega	ca. 1880	Romantic	Moderato	173	1/4	2
C08	Lagrima	F. Tarrega	ca. 1880	Romantic	Andante	219	1/8	2
C09	Moonlight Sonata	L.V. Beethoven	1801	Classical	Adagio	794	1/3	4
C10	Etude B minor	F. Sor	1828	Romantic	Allegretto	27 9	1/2	2

Table 4.1: Information about compositions.

4.3.2 Audio-Score Synchronization

In our study we hand annotated[1] measure positions of all audio files in our music collection. By this way we could synchronize each corresponding score-measure position with the audio-measure position. The reason of not using audio-score synchronization tools was, although they are working with an acceptable accuracy (Devaney & Ellis, 2009), they our onset imputation process and measure-level onset deviation extraction systems are based on measure positions and, even little accuracy looses, can cause exponential growth of error rates in experiments. Moreover, false-positive measure positions could cause wrong analysis outcomes as we will discuss in the next section. Therefore, we decided to hand annotate all measure starting positions in all recordings.

4.3.3 Refined Onset Detection

After detecting the onsets in our collection using the algorithm and parameters that we obtained from PSO (Section 3.2.1), we refined our onset detection by adding an additional step. For that purpose, we again used the manually annotated measure positions (see Section 4.3.2). This way, we could synchronize each measure in the audio file with the corresponding measure positions in the written score, and check whether there were missing onsets (Fig. 4.5). If a score onset $\hat{o}^r_{i_c}$ did not match an audio onset $o^r_{i_c}$, we imputed the temporal location corresponding to 7 milliseconds before the highest audio signal magnitude (absolute values) closest to $\hat{o}^r_{i_c}$ and within a short-time window. Each onset detection function marks onsets in different implementations. Some of them annotate onsets as the highest peak as the transient reach, and some define the starting point of the transient as the onset. In our case *Kullback-Liebler (K-L)* algorithm, see Section 3.2.1, marks the transient starting point as the onset position. Therefore, after detecting the amplitude peak we need to go

[1] A most probable future study is to propose an automatic Audio-Score synchronization algorithm. If so, all the process could be automatic and we can run our analysis with bigger corpus of data.

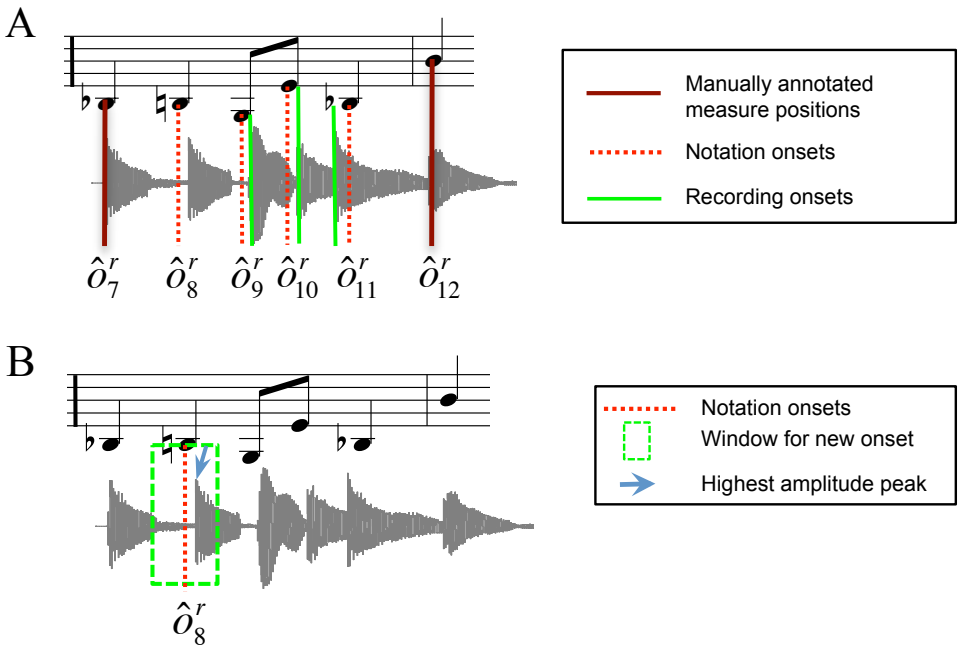


Figure 4.5: A) After synchronizing the audio with the score we have matches for all score onsets except $\hat{o}^r_4$ and $\hat{o}^r_9$. (B) For these two, we look at possible onset candidates inside the green windows

back and find the transient starting point. In order to find this point, we manually checked 360 correct onsets detected by K-L and calculated the distance between the transient start and maximum peak. 7 milliseconds correspond to the median of this distance. We used a window centered at $\hat{o}^r_{i_c}$ whose length corresponded to the 90^{th} percentile value of the composition's note durations.

4.3.4 Onset Validation

To check the accuracy of the obtained $o^r_{i_c}$, we manually validated 223 random onsets from the whole data set. Specifically, we annotated the temporal differences between what we considered to be the true onset location and the one determined by our approach (Fig. 4.7A). The vast majority of the inspected onsets were at their correct locations. Using a threshold evaluation strategy to determine the percentage of correct onset placements (Brossier, 2006), we estimated that only a 6.7% of them were not placed on the exact location they should be. This number drops down to 2% if we consider a threshold of 150 milliseconds (Fig. 4.7B). Also we validated the additional amount of accuracy that we obtained with our refined onset detection method, (see Figure 4.6). This validation was also appropriate for musical measure positions.

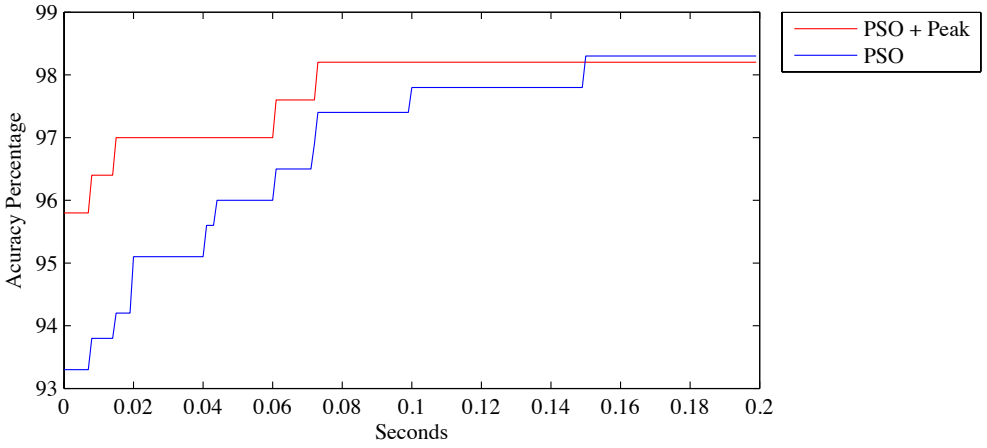


Figure 4.6: Onset detection accuracy in seconds.Y axis is the onset accuracy in the up-limits of the values in X axis. X axis is the onset Section 3.2.1

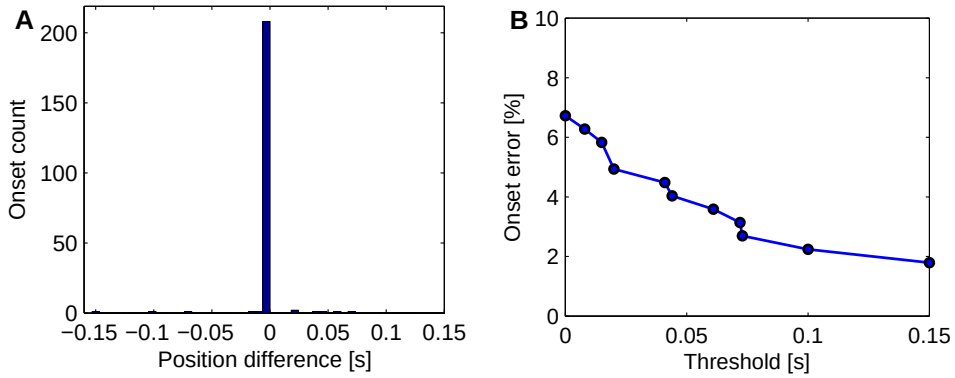


Figure 4.7: Semi-automatic onset detection accuracy. (A) Histogram of onset temporal differences. (B) Onset deviation error rate as a function of a threshold (see text).

Besides from manual validation of each measure position, we also considered them as onset positions and while choosing random onsets, measure positions were also likely to be chosen.

4.3.5 Onset Deviation Extraction

After extracting onset positions $o^r_{i_c}$, we follow an approach as we did in *Refined Onset Detection*, Section 4.3.3. This time we synchronized audio with score in the window of 4 measures rather than 1. The reason of using 4 measures rather

	C01	C02	C03	C04	C05	C06	C07	C08	C09	C10
Mean	0.043	-0.004	0.031	0.031	-0.004	0.014	0.038	0.018	0.024	0.024
Standard deviation	0.164	0.220	0.192	0.182	0.124	0.148	0.259	0.232	0.260	0.259
Maximum anticipation	-0.420	-2.584	-0.947	-0.597	-1.288	-0.621	-1.401	-0.975	-1.220	-1.217
Maximum delay	2.225	0.864	1.616	1.360	0.573	1.026	0.950	1.262	1.636	1.363

Table 4.2: Summary statistics for onset deviations for all performances of a given composition. Each **C** refers to a musical composition. For the names of the compositions see Table 4.1. All values are given in seconds.

than 1 was that, due to the manual synchronization of each score measure with the audio signal, in 1-measure synchronization the first onset of each measure would result in $d^r_{i_c} = 0$, i.e. loosing several meaningful onset deviations. After the synchronization we defined the theoretical onset positions that should be played in this audio fragment. To calculate the theoretical onset positions we were using the MIDI information. In MIDI protocol, starting point of each note is defined according to specified tempo (bpm) value. We scaled all MIDI-onsets inside the synchronized 4-measure and we slid this 4-measure window by 1-bar in each iteration. After calculating the theoretical positions of the onsets we computed the arithmetic difference between the recording onsets and the theoretical onsets in each iteration. At the end, with the exception of the onsets at the beginning and end of the piece, $d^r_{i_c}$ would be obtained as the average over four deviation values, (see Figure 4.2). However, we made a further refinement and avoided the extremes, i.e., the maximum and minimum values, and compute the average between the two central ones.

Onset deviations were computed as in Eq. 4.1, obtaining a sequence $d^r_{i_c} = \{d^r_1, d^r_2, \dots d^r_{n_c}\}$, for a composition with n_c note onsets. All compositions provided similar numbers for the statistics of raw onset deviation values $d^r_{i_c}$ (Table 4.2).

$$d^r_{i_c} = \hat{o}^r_{i_c} - o^r_{i_c}, \tag{4.1}$$

Additionally, we confirmed that maximal anticipations/delays generally corresponded to full cadences, usually ritardandos found in piece endings or strong structural locations (cf. Grachten & Widmer (2009); Liem et al. (2011); Palmer (1996); Repp (1990)). For instance, in the middle of the twenty-first measure of C02 (J.S. Bach, BWV 1007), for all performances, we observed a long pause between 0.5 and 1 seconds, which does not correspond to any existing annotation in the written score. Also, in C03 (A. Barrios, La Catedral–Prelude), the notes corresponding to the melody in the arpeggios are significantly delayed in most of the performances.

4.3.6 Onset Deviation Pre-Analysis

In pre-analysis, we inspected whether the onset deviations could be inferred somehow from score notation. Specifically, we are investigating two main questions:

1. Is there a correlation between pitch intervals and relative note durations with onset deviations ?
2. What is the distribution of the onset deviations ?

The results suggested that the considered onset deviations are rather independent of their associated relative note duration, expressed with relation to the beat (e.g., 1/2 beat, 1/3 beat, 1/4 beat) or their associated pitch interval size, expressed in semitones (e.g., +1 semitone, +2 semitones, −3 semitones). Very low, marginal, non-significant correlations were found (Figs. 4.8 and 4.9). Overall, we found no compelling evidence of the relation between onset deviations and the most fundamental short-time score elements, i.e., the single notes.
We also observed that the distribution of onset deviations conforms to a stretched Gaussian (Fig. 4.10A), an aspect that, as far as we know, had not been formally assessed yet. Interestingly, such distributions seem to qualitatively agree with data from neurological interval timing studies (Buhusi & Meck, 2005). A further interesting aspect that relates the obtained onset deviations with existing literature is the observation of long-range correlations (Fig. 4.10B). The fact that there exist long-range temporal correlations suggests that, as with the case of basic rhythm tapping (Hennig et al., 2011), onset deviation sequences of $l > 1$ can be characterized as memory processes (Baddeley, 2003), and thus may have the potential to contain non-trivial information of their context.

4.3.7 Data Structuring

As we explained in Section 4.2, we were analyzing onset deviations by using different levels of data. In this section we will explain how we formed our feature vectors for the classifiers. Mainly, we analyzed 2 musical levels and 3 models of each level.

Note Level Onset Deviations, n_0

Our first level is the note level onset deviations, n_0. We are constructing our feature vector as consecutive note onset deviations (see Figure 4.1, $\mathbf{d}^r = \{d_1^r, d_2^r, \dots d_{n_c}^r\}$). Each d_i corresponds to a feature for the classifiers. For the note level onset deviation analysis we choose different subsequences from the whole onset deviation sequence. For each composition c (the one to which the r-th recording corresponds to), an integer note index i_c is uniformly chosen, $i_c \in$

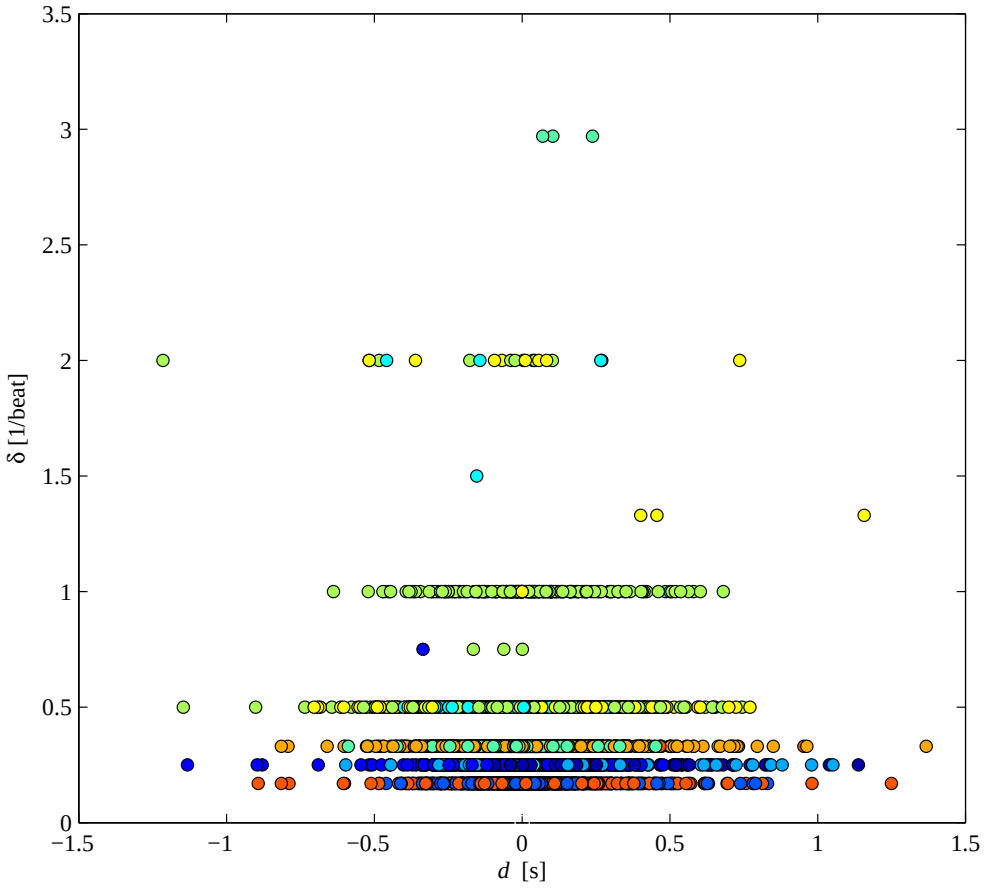


Figure 4.8: Scatter plot of relative note durations from the score δ versus onset deviations d (sample of 50 values per performance; different colors correspond to different scores). Kendall τ rank correlation coefficients between δ and d were low across all possible 10×10 comparisons between score and performance: $\tau \in (-0.24, 0.24)$, $\bar{p} = 0.41 \pm 0.42$.

$[1, n_c - l]$. This, together with a predefined length l, determines a subsequence $\bar{\mathbf{d}}^r_{i_c:l} = \{\bar{d}^r_{i_c}, \bar{d}^r_{i_c+1}, \ldots \bar{d}^r_{i_c+l}\}$. The final data $\mathcal{D}$ that serves as input for the classifier consists of the union of feature sequences plus the composition labels across all recordings. Formally,

$$\mathcal{D} = \bigcup_{r=1}^{100} \{\bar{\mathbf{d}}^r_{i_c:l}, c\} \ , \tag{4.2}$$

where $\bigcup$ denotes the union operator and, as mentioned, c indicates the composition index of the r-th recording. Notice that, due to the random choice of $i_c \in [1, n_c - l]$ and the fact that $n_c \geq 170$ (Table 4.1), the i_c for each composition might be different. However, notice also that i_c is the same for every

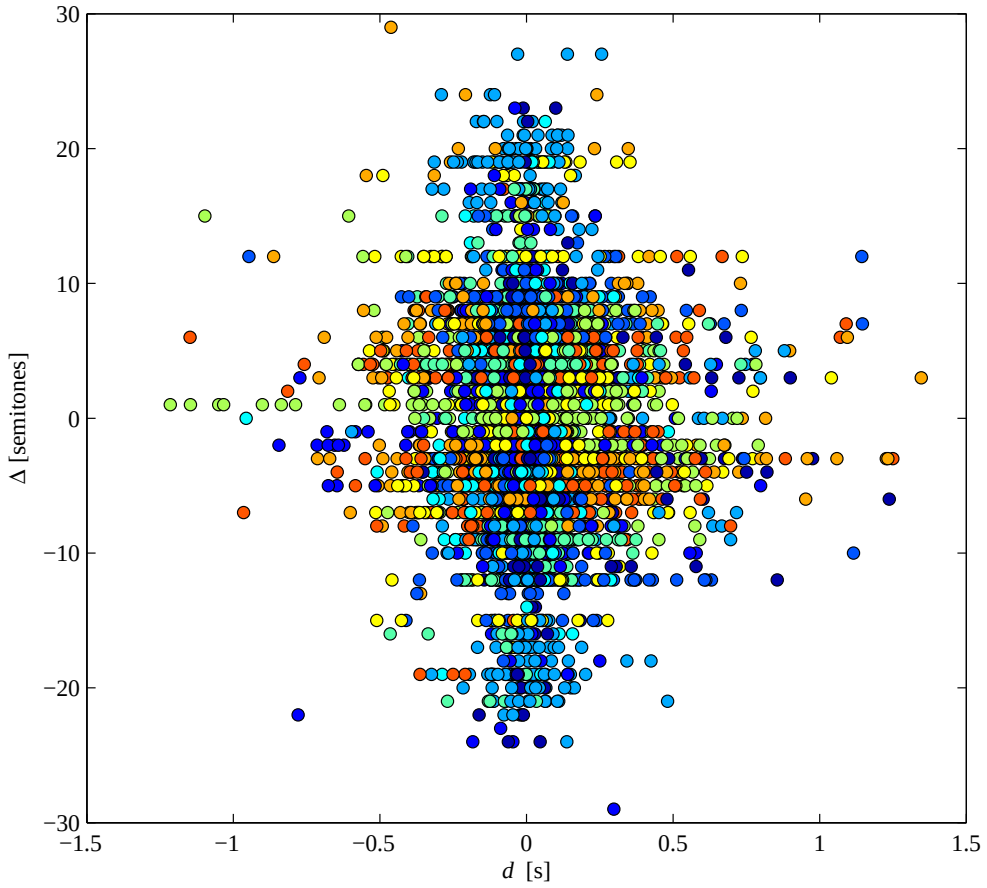


Figure 4.9: Scatter plot of note intervals from the score Δ versus onset deviations d (sample of 50 values per performance; different colors correspond to different scores). Kendall τ rank correlation coefficients between Δ and d were low across all possible 10×10 comparisons between score and performance: $\tau \in (-0.11, 0.1)$, $\bar{p} = 0.49 \pm 0.30$.

recording of composition c. Hence, the same subsequence position is taken for each composition.

The entire sequences $\mathbf{d}^r = \{d_1^r, d_2^r, \ldots d_{n_c}^r\}$ for each recording r are normalized to have zero mean and unit variance, $\bar{\mathbf{d}}^r = (\mathbf{d}^r - \mu)/\sigma$, where μ and σ correspond to the mean and standard deviation of all n_c values in $\mathbf{d}^r$ (recall that n_c is the number of note onsets deviations for a given composition c, Eq. 4.1).

Musical Measure Level Onset Deviations - m_0

The second level of analysis is the measure level, m. In each measure there is n number of notes, so there are n number of deviation values. Our intention is to model onset deviations of a composition in the unit of a musical measure. Each

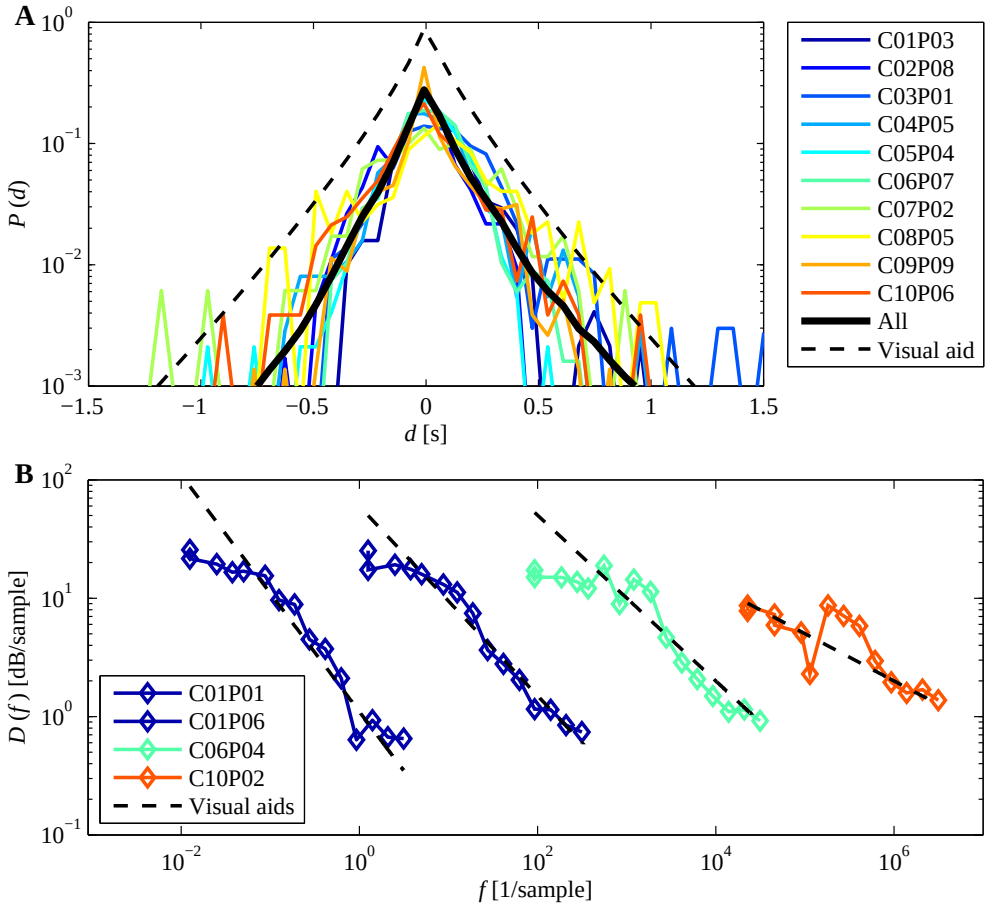


Figure 4.10: (A) Examples of onset deviation distributions $P(d)$. The visual aid corresponds to a stretched exponential of the form $P(d) = e^{-a|d|^{\alpha}}$, where a is a constant, d is the onset deviation, and α is the stretching exponent. In the plot we use $a = 6$ and $\alpha = 0.8$. To the best of our knowledge, this is the first quantitative, mathematically-oriented characterization of real-world note onset deviations. (B) Examples of power spectral densities $D(f)$ from the full onset deviation sequences. The visual aids correspond a power law of the form $D(f) = b/f^{\beta}$, where b is a constant, $f \in [0, \pi]$ is the normalized frequency, and β is the power law exponent. Frequencies f are linearly scaled for ease of visualization. From left to right $b = \{1, 1.1, 2, 2\}$ and $\beta = \{1, 0.8, 0.7, 0.4\}$. These exponents are akin the ones obtained rhythmic tapping tasks Hennig et al. (2011). For (A) and (B) the color-coded legends correspond to recording identifiers, C`XX`P`YY`, where `XX` corresponds to composition number, `XX` $\in [1, 10]$, and `YY` corresponds to performance number, `YY` $\in [1, 10]$.

musical composition has different number of notes in each measure. However, we model the general onset deviation behavior of performers in each musical measure (see Section **??**), independent of the number of notes. Briefly, our task is to find a representation to make each performance be comparable with the rest of the performances. In this sense, our principal components of the methodology are:

1. Represent onset deviation of each musical measure with an equal number of data units, i.e. *data re-sampling*.

2. Represent similar structures (musical measures) with the same representation by using clustering techniques, specifically, *k-means clustering*.

Data Re-Sampling: Onset deviations of a musical measure were represented by the onset deviation of notes inside each measure. We could interpret this note onset deviations as data points of it's corresponding measure. By having different number of data units, first, it was not possible to compare measures with each other, and second, it was not possible (most of the cases) to apply any kind of data analysis techniques. Therefore, we needed to represent each measure with equal number of data units.
Given that each performance representation contained k number of measures and each measure m_j has n_j number of notes, i.e. n_j number of onset deviations, each measure could be represented as consecutive onset deviation values d_i, from 1 to n_j. Then, a musical measure m_j could be represented as $m_j = \{d_1, d_2, d_3, \dots d_n\}$ and each r^{th} performance of a compostion, $\mathbf{C}_k^r$, with k number of measures, could be represented as consecutive measures $\mathbf{C}_k^r = \{m_1, m_2, m_3, \dots m_k\}$, see Figure 4.2.
In order to re-sample onset deviation values inside each musical measure to a matching number, first we up-sampled all the onset deviation values of each measure, $m_j = \{d_1, d_2, d_3, \dots d_n\}$, of all performances to the *least common multiple* of all the measure onset deviation values. After that, we down-sampled to 100 data units, see Figure 4.11. For instance, if we had different number of notes in different measures such as $\{2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 14, 16\}$ this leads to a *least common multiplier* of 55440. Furthermore, since we had k measures for each performance, we would have $55440 \times k$ data unit feature vector. In the sense, any kind of further computation (clustering, classification etc.) $55440 \times k$ data units could be unnecessary and too big to compute. Therefore, we down-sampled it to 100 data units. To summarize our approach for the data re-sampling, first each measure is up-sampled to 55440 data units and then each one down-sampled to 100 data units.
K-Means Clustering: With the clustering process we were interested in grouping measures with a similar deviation behavior. That is, measures with a similar deviation behavior will be clustered together and can be identified with as a single index unit. Specifically, given a set of observations $\{x_1, x_2, \dots, x_n\}$, where each observation is a f-dimensional real vector, $x_i = \{d_1, d_2, d_3, \dots, d_f\}$,

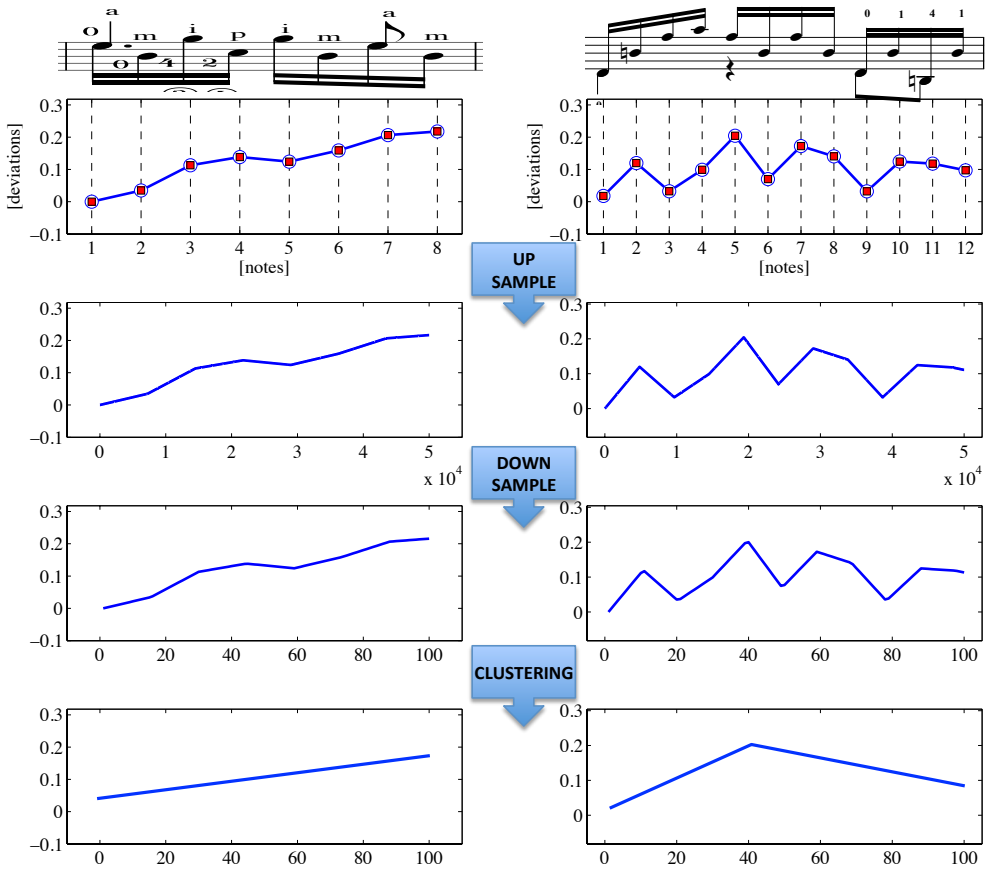


Figure 4.11: Re-Sampling of data units.

k-means clustering aims to partition the n observations into k sets ($k \leq n$). In our case observations were measures and dimensions were the data units inside each measure (see Data Re-sampling).
After clustering, we obtained k number of centroids $\mathcal{C}_k = \{\mathcal{C}_1, \mathcal{C}_2 \ldots, \mathcal{C}_k\}$ that represented deviation behavior of all the measures in the music collection. Thus, we could model each measure with its nearest cluster centroid, see Figure 4.12. Moreover, we could represent a performance as consecutive cluster indexes. The outcome of the clustering procedure is illustrated in the bottom picture of Figure 4.3.

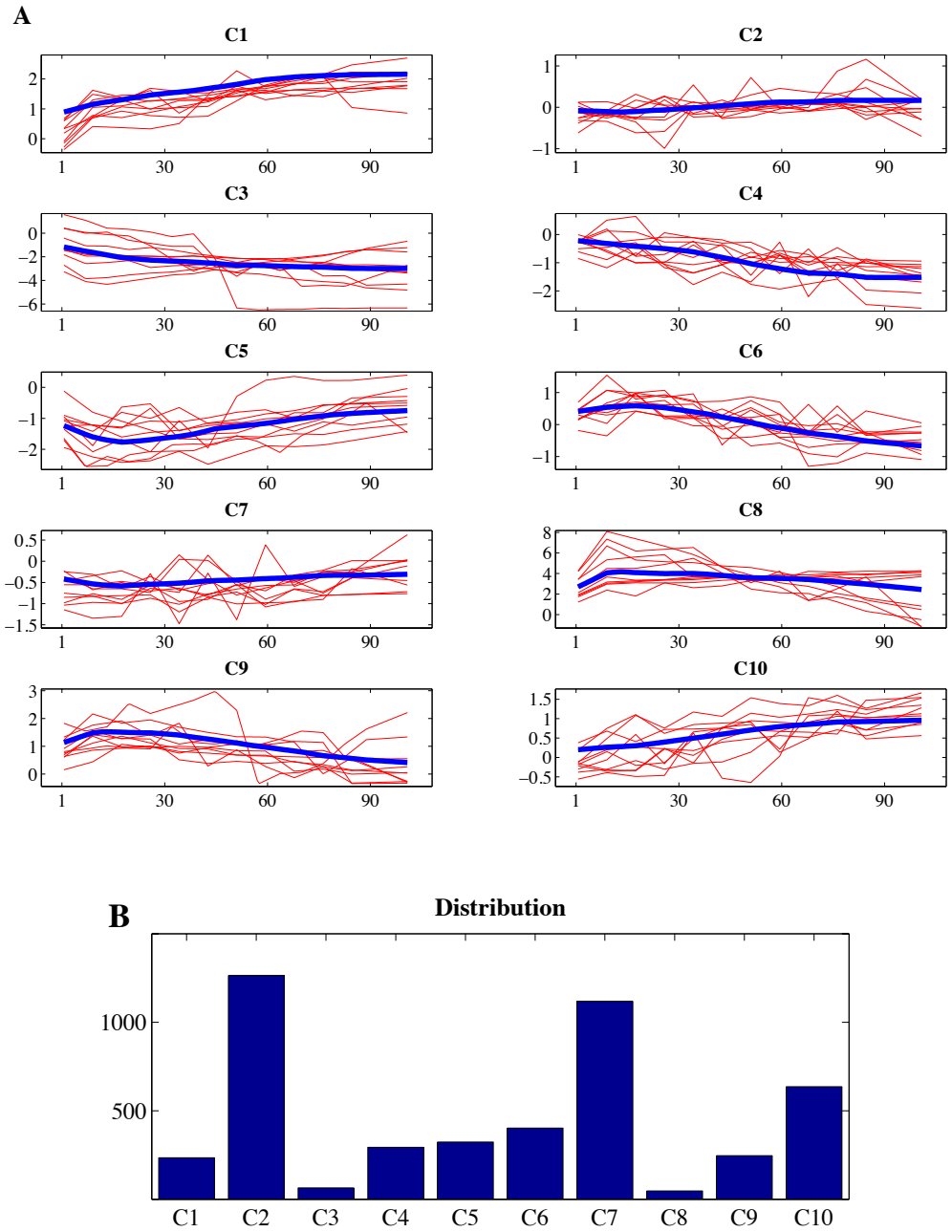


Figure 4.12: A - In each graph blue line represent a centroid. The red cloud around the red lines are the members of the corresponding cluster. For the visualization we only included 10 members. B - distribution of the cluster members

Data models for classification

From the multiples data representations of onset deviations introduced previously, we can evaluate the predictive power of each of them or evaluate a combination of features from them. Specifically, in the experiments we evaluated also the combinations of them. That is, the representation levels evaluated were:

n_0 **:** Note level onset deviations. For the note level onset deviation analysis we chose different sub-sequences from the whole onset deviation sequence (see Section 4.3.7 - Note Level Onset Deviations).

n_1**:** Note level onset deviations distribution. We represent distributions as 10 bin histograms. Each bin value serves as a feature, i.e. it has $b = 10$ features plus the composition labels.

n_2**:** Bi-gram of note onset deviations. The $\mathcal{D}$ consists of all the elements of the bi-gram matrix. There are $b \times b$, 100 features plus the the composition labels.

n_{12}**:** Combination of n_1 and n_2. There are $10 + 100 = 110$ features plus the the composition labels.

m_1**:** Measure level cluster count distributions. In k-means clustering, we used 10 clusters. The probability distribution of belonging to each cluster serves as a feature. Thus, there are $k = 10$ features plus the the composition labels (see Figure 4.13 for an example).

m_2 : Bi-gram of the cluster index representation. The $\mathcal{D}$ consists of all the elements of the bi-gram matrix. There are $k \times k$, 100 features plus the the composition labels.

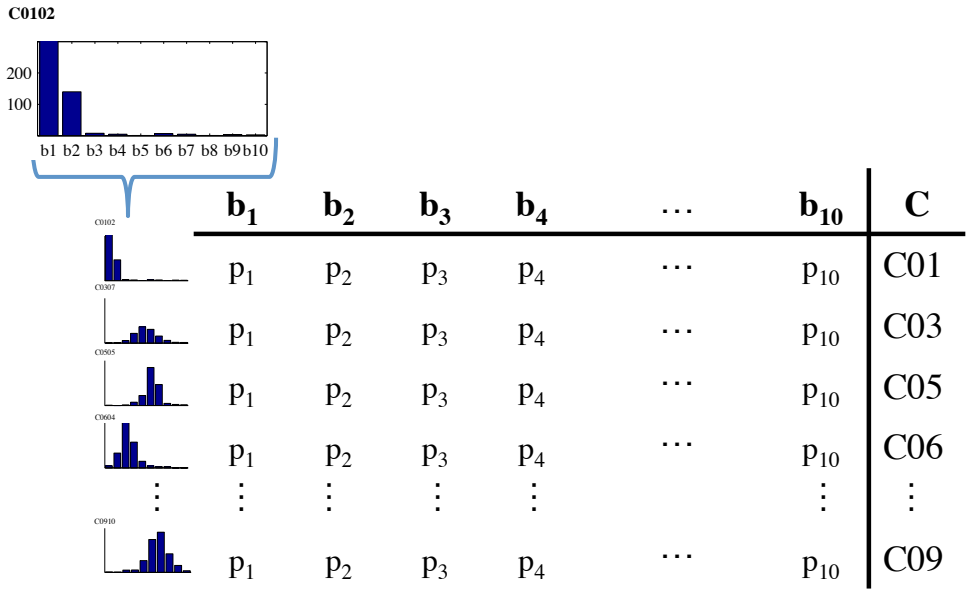
m_{12}**:** Combination of m_1 and m_2. There are $10 + 100 = 110$ features plus the the composition labels.

nm_{12}**:** Combination of n_{12} and m_{12}. There are $110 + 110 = 220$ features plus the the composition labels.

Both histogram and bi-gram feature vectors are normalized separately in order to avoid over-fitting.

4.3.8 Classification

Since we have 10 different compositions in our music collection (see Section 4.3.1) for the each level of analysis we cast the problem of identifying the piece from its onset deviations as a 10-class classification problem (Hastie et al., 2009; Mitchell, 1997; Witten & Frank, 2005).



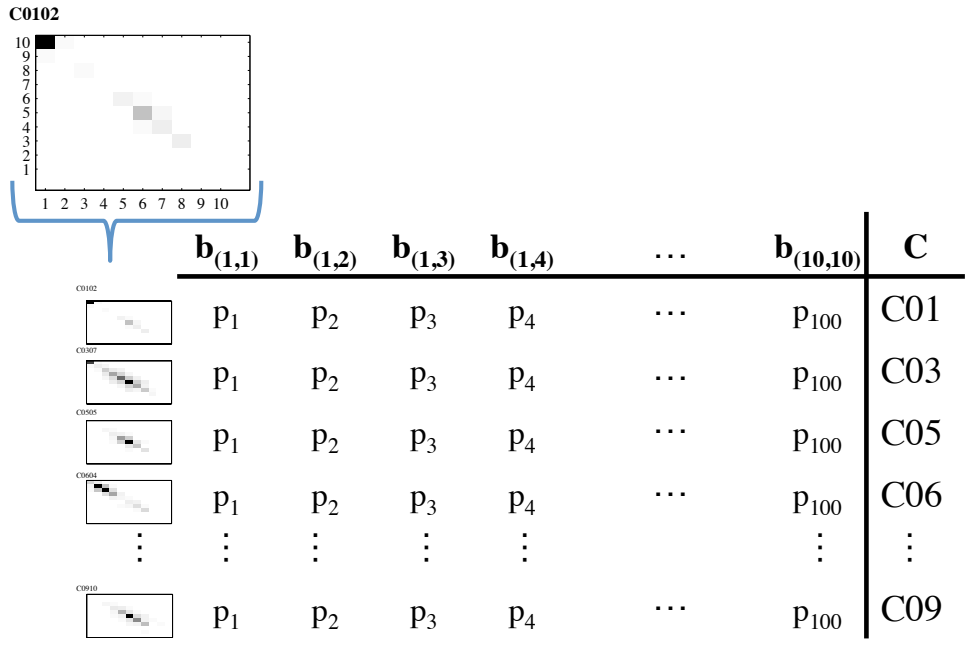


Figure 4.13: Feature Vectors

We investigated whether different levels of onset deviations are related to the musical piece, i.e., whether onset deviations have some predictive power of the composition being interpreted. Notice that, in the case of relative note durations, and in contrast to onset deviations, there will be no differences between performances, (see Section 4.3.6). This makes it a specially strong competitor against which the predictive power of onset deviations can be compared.
To show that the predictive power of the considered feature sequences is generic and not biased towards a specific classification scheme, we employ basic algorithms exploiting five different machine learning principles (Hastie et al., 2009; Mitchell, 1997; Witten & Frank, 2005): decision tree learning, instance-based learning, logistic regression, probabilistic learning, and support vector machines. The implementations we use come from the `weka java libraries`[2](Hall et al., 2009; Witten et al., 1999) and, unless stated otherwise, their default parameters are taken. Since our focus is on assessing the predictive power of onset deviation sequences rather than obtaining the highest possible classification accuracies, we make no tuning of the classifiers' parameters. In total for the note level sequences, n_0 we use 7 and for the rest of the models we use 5 (except NN-D and LR) implementations (Hastie et al., 2009; Mitchell, 1997; Witten & Frank, 2005) plus a random classifier:

- NN: k-nearest neighbor classifier. We use the Euclidean distance (NN-E) and dynamic time warping similarity (NN-D). For dynamic time warping we use a standard implementation with a global corridor constraint of 10% of the sequence length (Gusfield, 1997). The number of neighbors is arbitrarily set to $k = 1$.

- Tree: classification and regression tree classifier. We use the Gini coefficient as the measure of node impurity and arbitrarily set a minimum number of 2 instances per leaf.

- NB: naive Bayes classifier. We employ a Gaussian function to estimate the likelihood of each onset deviation.

- LR: logistic regression classifier. We use L2-regularized logistic regression with automatically-scaled intercept fit.

- SVM: support vector machine. We consider a linear kernel (SVM-L) and a radial basis function kernel (SVM-R).

- Random: random classifier. We additionally consider a random classifier as the baseline. It outputs a randomly selected class from the pool of all available training labels.

For each data set $\mathcal{D}$ we performed standard 10-times, 10-fold, out-of-sample cross-validation Hastie et al. (2009); Mitchell (1997); Witten & Frank (2005).

[2]Version 3.6.9: `http://www.cs.waikato.ac.nz/ml/weka/index.html`

We not only applied classification to whole music collection but also all possible subsets of our music collection, $[2, 3, 4, 5, 6, 7, 8, 9, 10]$ and for our music collection with 10 composition the number of possible music collection subsets for each pair would be $[45, 120, 210, 252, 210, 120, 45, 1]$ respectively. We have a total of 1003 possible sub-music collection possibilities. With 10-times, 10 fold cross validation we needed to run our classifiers $1003 \times 10 \times 10 = 100300$ times. With our hardware, each run computationally cost around 1 minute (we are not considering the previous k-means clustering cost) and 100300 minutes corresponded around 69 days. In order to decrease this time complexity and also represent each sub-music collection with equal number, we randomly chose 25 unique pairs from all possible sub-music collection combination. We also forced each subset to be represented with balanced number of elements. Such as in our 2 pair sub-music collection, 25 pairs included equal number of elements for each composition.

Even if our music collection was already balanced (10 performances per piece), we forced internal training and testing data sets to be balanced as well. Hence, we train with 9 performances per piece and test with 1. We additionally force that all classifiers see the same training/testing sets. As different selections of i_c could affect the results, we repeated the whole process 100 times, in order to obtain a reliable estimation of all possible accuracies (not only for average accuracies and their standard deviations, but also for having an proper idea of maximum/minimum values and reliably assessing statistical significance). In summary with for single pair in the sub-music collection, with 10-times 10-fold, we generated 100 data sets $\mathcal{D}_1, \mathcal{D}_2, \ldots \mathcal{D}_{100}$ and test each classifier with them. This yields a total of 100×7 accuracy values computed from 10×10 folds for each pair element in pair-subsets. For the 10 pair set we had only one possible combination, all the compositions, thus it has $100 \times 7 \times 1$ accuracy values. For the rest, we have balanced sets with the 25 elements, yielding $100 \times 7 \times 25$ accuracy values.

4.3.9 Statistical Tests

As we use matched samples $\mathcal{D}_i$ in all our models, we assessed statistical significance with the well-known Wilcoxon signed-rank test (Hollander & Wolfe, 1999). The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used when comparing two matched samples (or related samples, or repeated measurements) in order to assess whether their population mean ranks differ. It is the natural alternative to the Student's t-test for dependent samples when the population distribution cannot be assumed to be normal (Hollander & Wolfe, 1999). We use as input the accuracy values obtained for one classifier and the random baseline.

4.4 Results

We report our results into two main sections. The experiments that we have done with note onset deviation sequences, n_0, and the others (combination of the different models). Note level sequence experiments showed that have a higher predictive power than other tests. It could be expected since with histograms and bi-grams we were analyzing the distribution regardless of the time information. However, even these distributions have higher classification accuracies than the random classifier. Unless stated, we only used the default parameters of the classifiers (see Section 4.3.8). Our aim is to demonstrate the predictive power of different levels of onset deviations rather than reaching the highest classification accuracies. However, even with default parameters in all classifiers, in all levels and models we achieved accuracies clearly above the random classifier.

4.4.1 Note Onset Deviation Model

If we plot the classification accuracies Ψ as a function of l we see that all classifiers perform on a similar range, with NB and SVM-R generally achieving the best accuracies (Fig. 4.14). As expected, NN-E and NN-D perform relatively similarly, thus indicating that no strong sequence misalignments (Gusfield, 1997) were present, thanks to the semi-automatic measure-based synchronization between score and recordings mentioned before (see Section 4.3.2). Trees achieved the lowest accuracies and seem to had some difficulties in learning from the considered sequential information. Nevertheless, for $l > 5$, all obtained accuracies lie far beyond the random baseline, always increasing with l. Importantly, we saw that statistically significant accuracies could be reached with very short sequences $\mathbf{d}^r_{i_c:l}$ (Fig. 4.15). Specifically, it turns out that a single sample $\mathbf{d}^r_{i_c:1} = \{d^r_{i_c}\}$ was sufficient for characterizing a piece statistically significantly beyond the random baseline, but with a low accuracy ($l = 1$, Fig. 4.15A). This difference increases with l, until no single accuracy across 100 trials goes below the ones achieved by the baseline ($l = 5$, Fig. 4.15B). Obviously, the longer the deviation sequence, the better (e.g., $l = 170$, Fig. 4.15C). To check whether the predictive power of onset deviation sequences was robust with respect to the size of the music collection, we could plot the accuracies Ψ as a function of the number of compositions m (Fig. 4.16). With this we observed that the obtained accuracies decrease at a much lower rate than the ones provided by our random baseline, independently of l (see also Fig. 4.16). This shows that Note Level sequences can be a reliable predictor of a musical piece. Additionally, we confirm that accuracies are balanced across compositions, with no exceptional confusion between pairs of them (Fig. 4.17). In fact, we see that confusions substantially depend on the classifier. This suggests that a specific confusion may not be largely due to the onset deviations themselves and, furthermore, that a strategy based on the combination of clas-

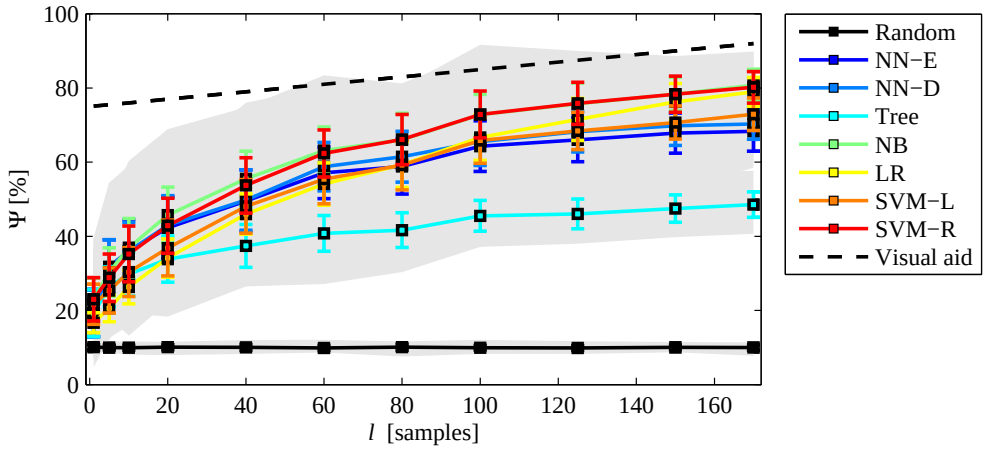


Figure 4.14: Average classification accuracy as a function of the length of the onset deviation sequence. The error bars correspond to the standard deviation and the shaded area denotes the range of all possible values (including minimum and maximum). The visual aid corresponds to a straight line of the form $\Psi(l) = a + bl$, where a is the intercept, b is the slope of the straight, and l is the sequence length. In the plot $a = 75$ and $b = 0.1$.

sifiers could potentially increase the overall accuracy. As our objective here is more focused on showing the predictive power of onset deviations rather than achieving very high accuracies on a music classification task. Interestingly, the best performing classifiers, NB and SVM-R, were also the ones where such difference was more clearly observable. Notice that, as mentioned above, relative note durations were found to be independent of onset deviations. (Figures 4.8 and 4.9).

4.4.2 Alternative Onset Deviation Models

We have 7 alternative configurations of onset deviation models, n_1, n_2, n_{12}, m_1, m_2, m_{12}, nm_{12}, (see Section 4.3.7). For the sake of computation we run with only 5 different classifiers, omitting NN-D and LR. Experiments with alternative models performed with less classification accuracy and more variance compared to model n_0 (see Figure 4.21). Similar to n_0, we reached relatively better accuracies with with NB and SVM-R. As in note sequence classification, trees achieved the lowest accuracies (Figures 4.18, 4.19 and 4.20). We reached statistically significant accuracies with all the different levels. For each level as the number of features in the feature vector increases, SMV-R reaches better accuracies. For instance, among the note level experiments the order of SVM classification accuracy was, $n_{12} > n_2 > n_1$ and $m_{12} > m_2 > m_1$, (see Figure 4.21). As we did for n_0, we could plot the accuracies Ψ as a function of the number of compositions m for each level of analysis (Figures 4.18, 4.19

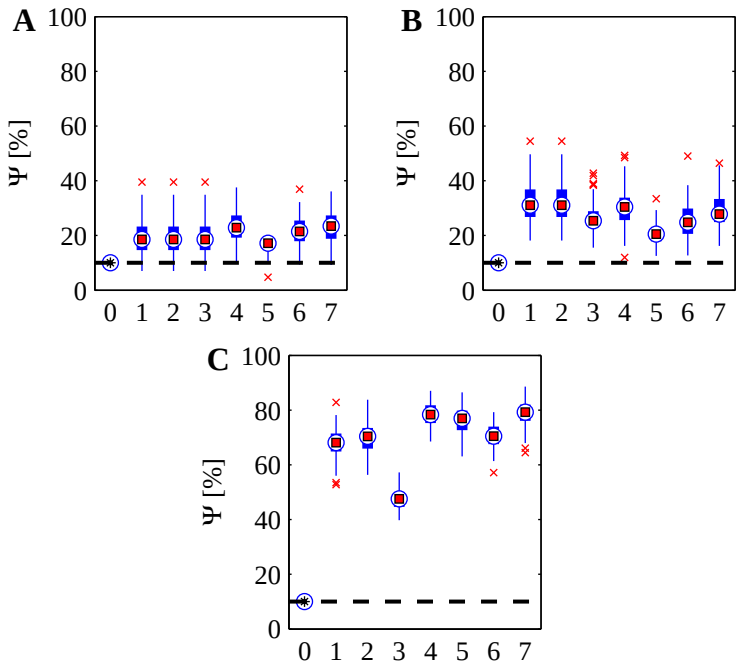


Figure 4.15: Box plot of classification accuracies using different sequence lengths. These are $l = 1$ (A), $l = 5$ (B), and $l = 170$ (C). The labels in the horizontal axis correspond to classification algorithms: Random (0), NN-E (1), NN-D (2), Tree (3), NB (4), LR (5), SVM-L (6), and SVM-R (7). In all plots, all medians are statistically significantly higher than the random baseline ($p < 0.01$).

and 4.20). With these figures we observe that the obtained accuracies decreased at a lower rate than the ones provided by our random baseline, independently of the model.

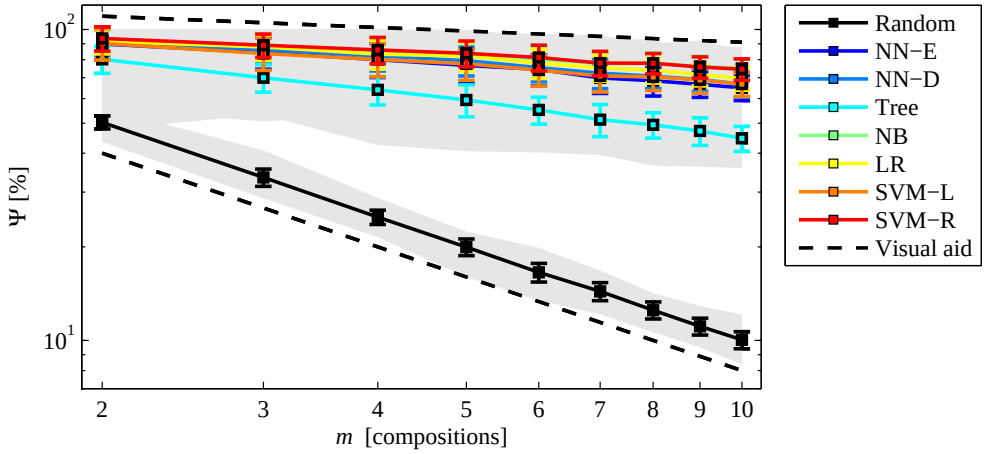


Figure 4.16: Average classification accuracy as a function of the number of compositions. Results obtained using a sequence length $l = 120$. The error bars correspond to the standard deviation and the shaded area denotes the range of all possible values (including minimum and maximum). The visual aids correspond to a power law of the form $\Psi(m) = b/m^{\beta}$, where b is a constant, $m \in [2, 10]$ is the number of compositions, and β is the power law exponent. The upper one is plotted with $b = 128$ and $\beta = 0.12$, and is associated with classification accuracies. The lower one is plotted with $b = 80$ and $\beta = 1$, and corresponds to the random baseline. The exponent associated with classification accuracies is much smaller than the one for the random baseline, what suggests that the absolute difference between the two increases with the number of considered compositions and, therefore, with the size of the data set.

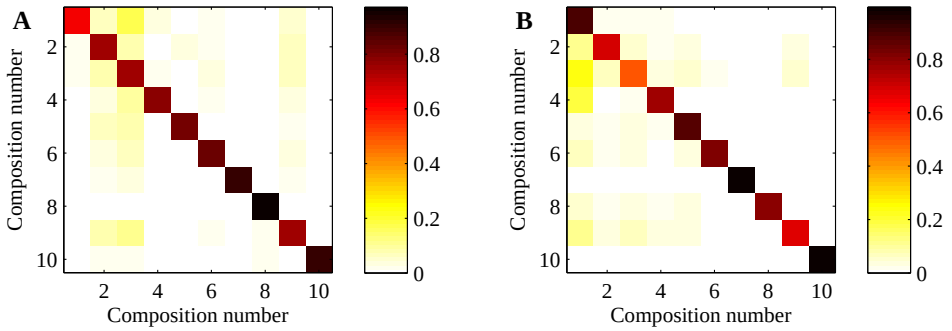


Figure 4.17: Confusion matrices for two different classifiers. These are NB (A) and SVM-R (B). The color code indicates average accuracy per composition (the higher, the darker). Compositions 7, 8, and 10 seem to be generally well-classified. For NB, compositions 2 and 3 attract many of the confusions while, for SVM-R, composition 1 takes that role.

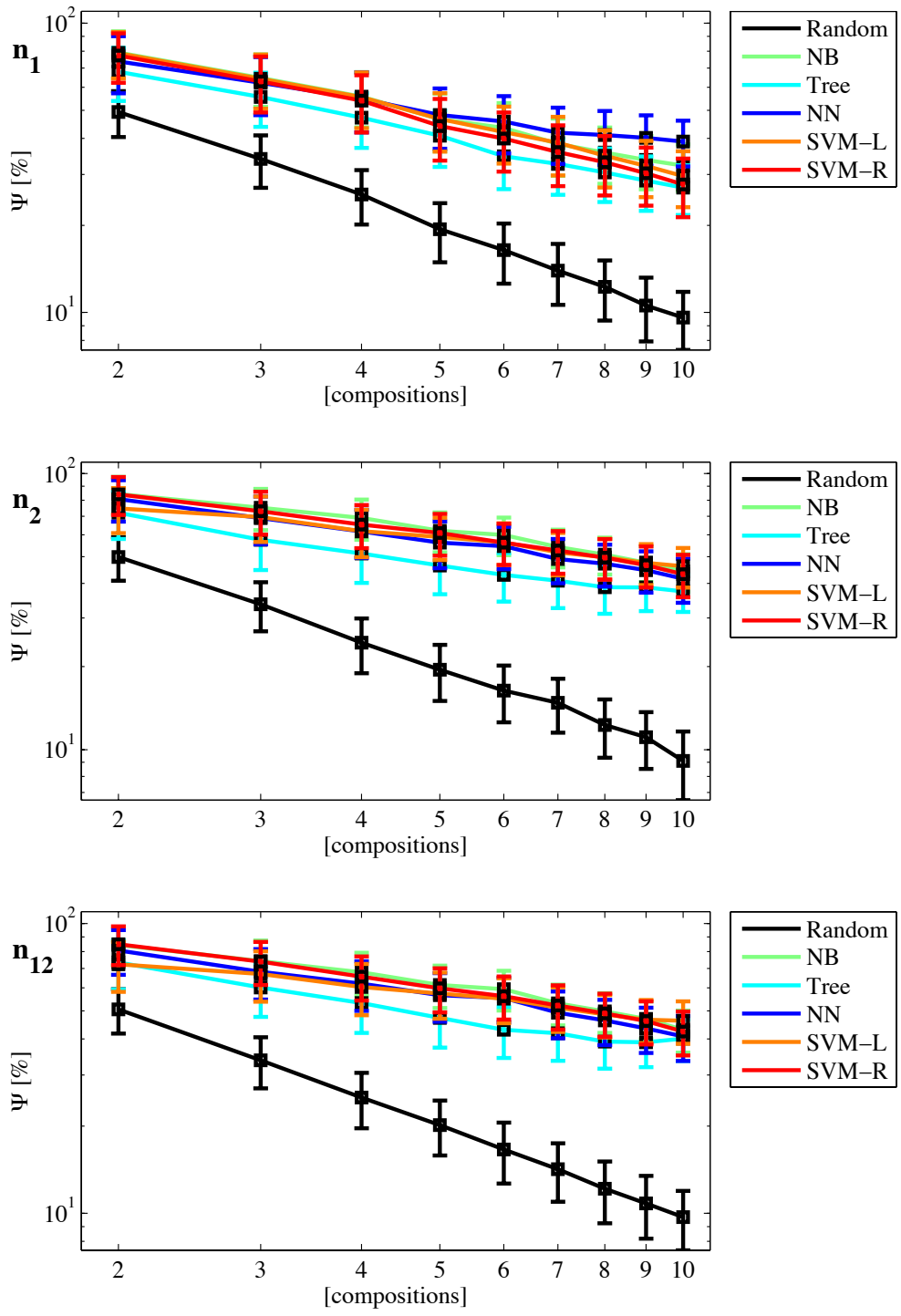


Figure 4.18: Average classification accuracy as a function of the number of compositions. The error bars correspond to the standard deviation.

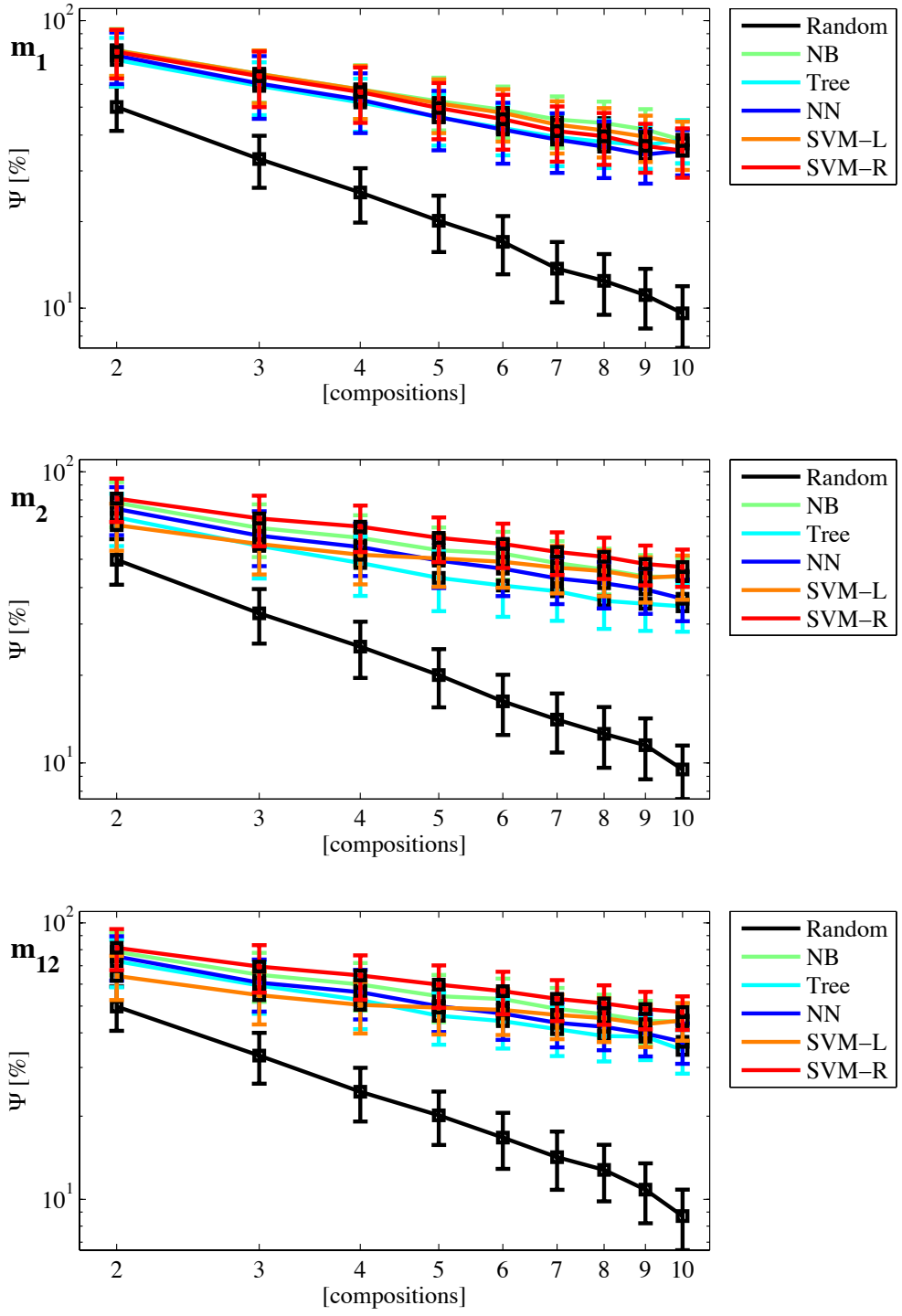


Figure 4.19: Average classification accuracy as a function of the number of compositions. The error bars correspond to the standard deviation.

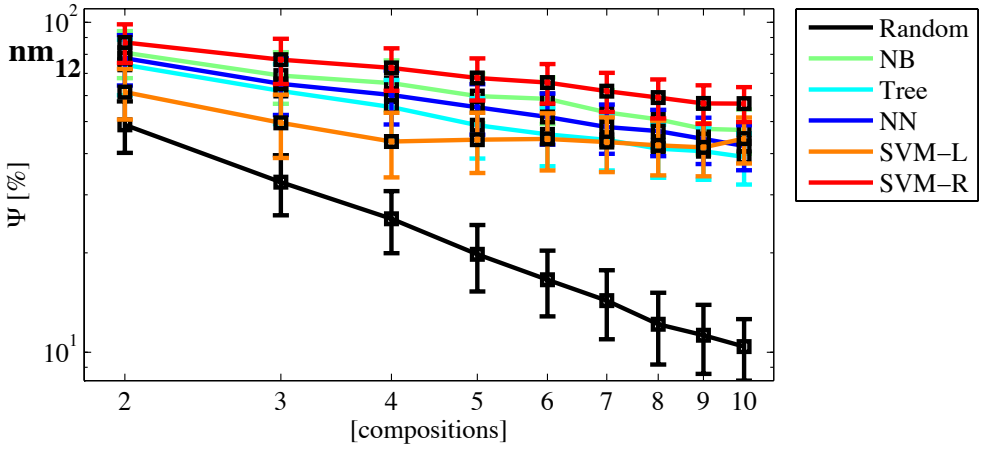


Figure 4.20: Average classification accuracy as a function of the number of compositions. The error bars correspond to the standard deviation.

4.5 Conclusions & Discussions

The obtained results show that:

1. All models of onset deviations are a powerful predictor of the musical piece being played. However, with less variance and higher medians, note level onset deviation sequences have far better predictive power compared to other levels.

2. Note level onset deviation sequences, n_0, are at least as powerful as direct music score information corresponding the relative note durations, if not better.

3. Predictive power is robust to classification scheme choices, to different levels of models, to the size of the considered data set and to the length of the considered sequences.

4. Even very short note level onset deviation sequences and basic deviation distributions provide statistically significant accuracies.

5. The obtained raw onset deviations conform and complement recent findings reported in the literature (Hennig et al., 2011).

In the light of these quantitative results, and re-taking the multi-dimensional perspective on onset deviations, we can now open some qualitative discussiosn. First, evidence suggests that randomness is a very minor component of the considered onset deviations. Indeed, if we substitute the onset deviations by random noise, the accuracies dramatically drop down to the considered baseline.

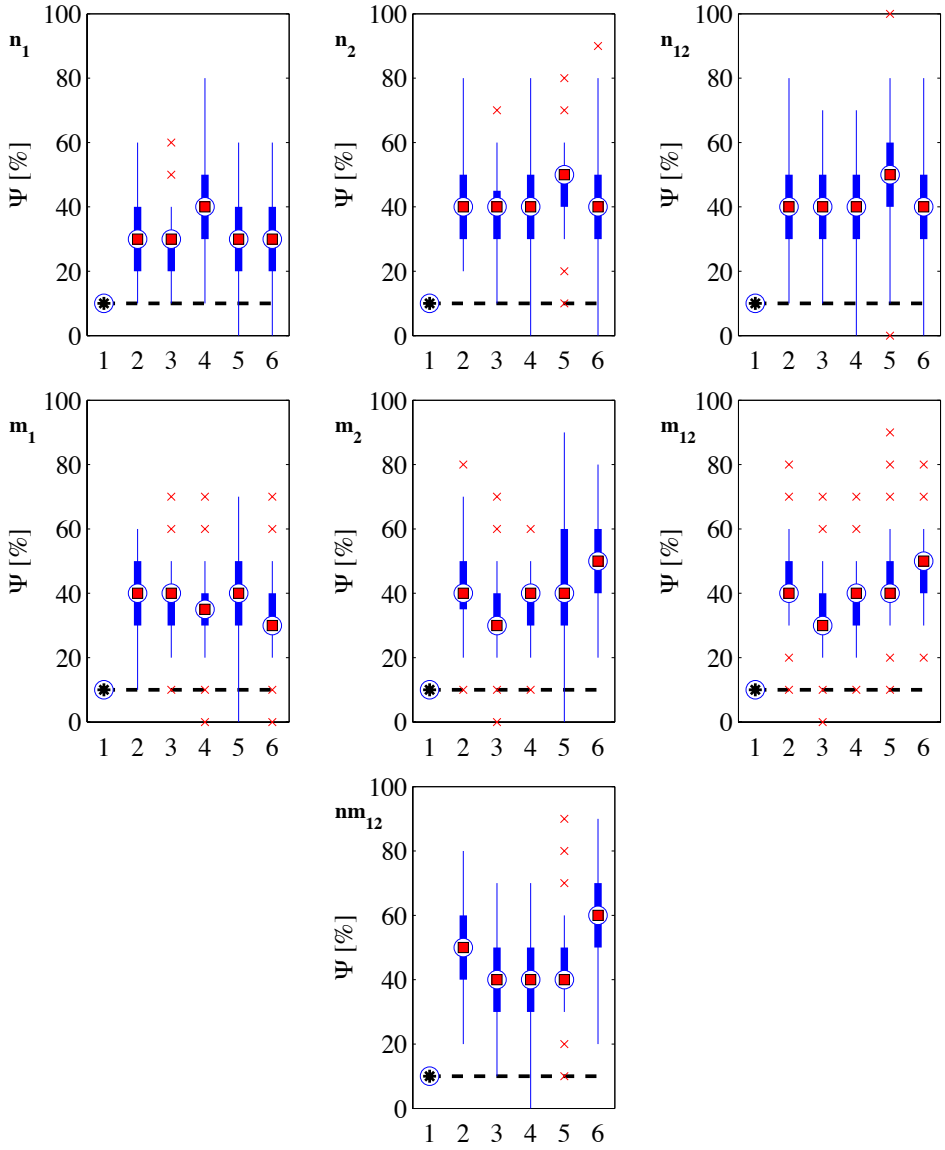


Figure 4.21: Box plot of classification accuracies using different models of analysis for the 10 pair data set. The labels in the horizontal axis correspond to classification algorithms: Random (1), NB (2), Tree (3), NN (4), SVM-L (5), and SVM-R (6). Black line is the random base line. n all plots, all medians are statistically significantly higher than the random baseline ($p < 0.01$).

Second, we can conjecture that the effect of performer-specific deviations on the raw onset sequences must be small, compared to the effect of other dimensions. As mentioned, the considered music collection contains a number of recordings of different pieces by the same performer. Hence, if performer-specific deviations dominated the raw onset sequences, we would expect much worse piece identification accuracies, as recordings would tend to cluster around performers and not around pieces. As some works indicate, performer-specific aspects may be better studied after subtracting a global, average performance template (Grachten & Widmer, 2009; Liem et al., 2011; Stamatatos & Widmer, 2005).

Third, a similar but weaker argument could potentially hold for emotional-specific onset deviations. If among the 10 different recordings of a piece we had many contrasting emotional expressions which were directly determining the magnitude of raw onset deviations, we would not be able to classify pieces with the accuracies obtained here ('emotional clusters' would be present). Of course, this leaves us with the possibility that all performers could adhere to similar emotional expressions for a given composition. Thus, one could argue that those emotional expressions may be, to some extent, 'dictated' by the composition itself (Juslin & Sloboda, 2013). Moreover, there exists the possibility that other performance resources different than timing are the responsible of such emotional expressions (Juslin & Sloboda, 2001). In this sense, a more holistic study involving multiple musical facets is needed.

The previous discourse leave us with two hypotheses regarding the origin of onset deviations: that onset deviations may be due to biological or instrument-related motion, and that they may respond to the musical piece structure and its psycho-perceptual consequences for interpretation. Assessing the first hypothesis would require a different collection containing recordings of the same piece played with different instruments. In the present study, we wanted to focus on classical guitar, as this is an unexplored area. Nevertheless, from our point of view, the second hypothesis seems to be the responsible for most of the variations in onset location. As mentioned, there is evidence that phrasing, metrical accents, musical form, and harmonic structure can determine timing deviations. However, no quantitative and deterministic rules are yet able to explain all available case studies (cf. (Gabrielsson, 2001)). In our experiments, mixing different compositions and their interpretations, we found scarce evidence for the dependence of onset deviations on individual score elements. Specifically, no clear analogies could be drawn between onset deviations and note durations or intervals. It is true that the existence of long-range temporal correlations (Hennig et al., 2011) suggests that note or interval sequences may provide a more deterministic character to onset deviations. However, the power law decay of such temporal correlations indicates that dependencies are exponentially weaker with time. Finally, the fact that onset deviations perform similarly or slightly better than relative note durations, combined with the fact that the former were independent and uncorrelated with the latter,

suggests that onset deviations could encapsulate information that goes beyond duration/temporal aspects of the score.
Apart from contributing to the discussion on the origins of note onset deviations, and as a main objective, we want to provide a new and fresh view to such a topic. We believe that by taking quantitative medium-scale approaches, considering real-world commercial recordings, and different instruments apart from piano is a necessary step towards a better understanding of it.

CHAPTER 5

Conclusion

5.1 Summary of the Thesis

At the outset of the thesis, two questions were posed that shaped the direction of the research. First one was, what is expressivity in guitar music? In a general view, as we argued in Chapters 1 and 3, musical expressivity can be conceived as the difference between the written, defined, and quantified music with the actual music that we finally listen. Thus, musical expressivity can be studied by analyzing additions and deviations performers apply to the written score. Our second questions was, how to extract and characterize musical expressivity in guitar music? For this purpose, we have proposed and applied feature extraction mechanisms, optimization techniques, and machine learning models. In this thesis we have explored two types of expressive resources: expressive articulations and timing deviations.

In Chapter 3, we designed a system able to identify the most used three expressive articulations of classical guitar: legato, glissando, and vibrato. Also, we reported experiments to validate our proposal by analyzing a collection of chromatic exercises and short melodies recorded by a professional guitarist. Although we are aware that our current system may be improved, the results showed that it was able to identify and classify successfully the defined expressive articulations.

Chapter 4 provides a new and fresh view on the topic of music timing variations. We believe that by taking quantitative medium-scale approaches, considering real-world commercial recordings, and a different instrument apart from piano is a necessary step towards a better understanding of timing deviations. In Chapter 4, the focus was on the utility of onset deviation sequences as musical piece signatures, and on the predictive power of those sequences from different levels of analysis. Hence, our analysis showed that all levels of onset deviations have predictive power.

5.2 Discussion

We chose to spread the description of existing literature through the thesis rather than concentrating all in the state of the art chapter. Our aim was to provide a general perspective of the field in the state of the art chapter and to introduce more specific literature in the chapters where we describe our proposals. By this way, we believe that readers could establish the connections among the previous work and our study more properly.
In Chapter 3 we proposed a system that automatically classifies guitar expressive articulations. We started our model by analyzing legato and glissando articulations. After that, we conducted vibrato analysis. To the best of our knowledge there were no studies that have worked on automatic legato and glissando detection on classical guitar. Our biggest challenge was to propose a model to differentiate between legato and glissando. For humans it is very hard to differentiate them, if not specified in the score. The key feature that we have used to differentiate them was aperiodicity. We realized that legato includes a small finger punch on fingerboard. Therefore, during the transition moment the periodic content drastically decreases compared to glissando, where aperiodic content increases (see Chapter 3 for a detailed description).
Although the conclusions we reached in Chapter 4 sound intuitive, to the best of our knowledge there were no studies working on that big number of commercial recordings. We know that a musical collection with 100 performances is far beyond enough to come up with universal conclusions about timing deviations. But we believe that our onset deviation extraction methodology enlightens new paths in order to analyze classical recordings which were recorded without a metronome. Furthermore, our analysis of different levels of onset deviations brings up new discussions for the performance analysis.

5.3 Contributions

This thesis contributes to the understanding of the musical expressivity. It contributes to the field of expressive analysis which is a sub-field of music information retrieval and general data analysis:

- It is, to the best of author knowledge, the first thesis that is entirely devoted to the topic of expressive analysis of guitar by using real-world commercial audio recordings.

- It makes a strong link with music information retrieval and current data analysis approaches.

- It demonstrates a multi-model approach to expressive analysis. In particular, it is shown that by combining the existing low-level algorithms but with a high-level of understanding, an effective analysis could be done.

- It shows how to successfully use machine learning algorithms for the classification and optimization.

Although guitar is chosen as the instrument of analysis, all the techniques and analysis could be easily applicable to other instruments. For instance, similar feature extraction algorithms were used in a completely different Ney instrument (Özaslan et al., 2012).
The outcomes of the research carried out in this thesis have been published in the form of several papers in international peer reviewd conferences, a journal and a book chapter. The full list of the author's publications are provided in annex to this thesis (Appendix B).

5.4 Limitations and Future Directions

The analysis of musical expressive is far from being solved. Musical expressivity is a so complex phenomenon which understanding still requires years of research. We may think about how a student learns to play an instrument and how after many years of hard training she becomes the master of her instrument. Mastery is the point where written notes become music. To be able understand this unique human creativity activity, we need more robust algorithms, much more data, and better understanding of interactions between performers.

Better feature extraction algorithms. As we discussed in Chapter 3, most of the MIR algorithms, such as source separation, onset detection, instrument identification are still hot research topics. There is a big amount of room for the improvement. As researchers, we know that we need better accuracies and results from existing low-level feature extraction algorithms such as onset detection to be able analyze high-level phenomenon such as expressivity. Nevertheless, we showed in our study that with existing algorithms and with the careful choice of optimization methods, it is possible to obtain acceptable results.

More data, much more data. We are in a era of data. By analyzing big amounts of data even simple, most primitive Machine Learning algorithms can come up with accurate findings. As Anand Rajaraman, a data mining professor in Stanford University stated *'more data usually beats better algorithms'* [1]. Of course this statement does not falsify our previous point. Furthermore, first we need accurate algorithms and then we need data. As Rajaraman pointed out, having more data gives better understanding compared to improving the algorithms. The perfect circumstances about music study is that we have the data, we have huge amount of recorded performances and each moment we are producing

[1] http://anand.typepad.com/datawocky/2008/03/more-data-usual.html

more content. However, as a future direction of MIR, we need tools and techniques to be able to analyze this big amount of data.

Performer interaction There are some attempts for not analyzing interaction but creating interaction by using supervised learning (Pachet, 2003). However, analysis of interaction between performers is, to the best of author knowledge, not investigated due to the limitations of current state of the art signal analysis and source separation techniques. As a future direction, the improvement of source separation algorithms could lead to the analysis of interaction between performers during recording sessions and live performances. It can provide important insights about human decisions and as of expressivity.

5.5 Final Thoughts

I believe that expressive analysis will be an important field in music information retrieval. However now our biggest obstacle is to find techniques that we can run with big amounts of data. From the perspective of a performer, in the end all that matters is the time that she spends with her instrument, the experience. In the sense of computers this experience means data. As the field advances, we will be able to analyze big amounts of performances and this will be the point where we will be able to understand this incredible creative human input to music, expressivity. I would be more than happy, with this thesis, if I could put a brick to the progress of expressive analysis.

"It is good to have an end to journey toward; but it is the journey that matters, in the end."

- Ernest Hemingway

Tan Özaslan, Berlin, October 13, 2013.

Bibliography

Abdallah, S. & Plumbley, M. D. (2003). Unsupervised onset detection: a probabilistic approach using ICA and a hidden markov classifier. In *Cambridge Music Processing Colloquium*. Cambridge, UK.

Aguado, D. (1981). *New guitar method*. London, UK, tecla edit edn.

Aho, M. & Eerola, T. (2012). Expressive performance cues in gypsy swing guitar style: a case study and novel analytic approach. *Journal of Interdisciplinary Music Studies*, *6*(1), 12060101.

Aho, M. & Eerola, T. (2013). Expressive performance cues in Gypsy swing guitar style: A case study and novel analytic approach. *Interdisciplinary Music Studies*, *6*(1), 01–21.

Alghoniemy, M. & Tewfik, A. H. (1999). Rhythm and periodicity detection in polyphonic music. In *Multimedia Signal Processing, 1999 IEEE 3rd Workshop*, pp. 185–190.

Babacan, O., Drugman, T., D'Alessandro, N., Henrich, N., & Dutoit, T. (2013). A comparative study of pitch extraction algorithms on a large variety of singing sounds. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. Vacouver, Canada.

Baddeley, A. (2003). Working memory: looking back and looking forward. *Nature Reviews Neuroscience*, *4*(10), 829–839.

Bader, R. (2005). Nonlinearities in the sound production of the classical guitar. In *Forum Acusticum*.

Barbancho, I., Tardon, L. J., Sammartino, S., & Barbancho, A. M. (2012). Inharmonicity-Based Method for the Automatic Generation of Guitar Tablature. *Audio, Speech, and Language Processing, IEEE Transactions on*, *20*(6), 1857–1868.

Begston, I. & Gabrielson, A. (1980). Methods for analyzing performance of musical rhythm. *Scandinavian Journal of Psychology*, *21*, 257–268.

Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M., & Sandler, M. B. (2005). A tutorial on onset detection in music signals. *IEEE Trans. on Speech and Audio Processing*, *13*(5), 1035–1047.

Bello, J. P. & Sandler, M. (2003). Phase-based note onset detection for music signals. proceedings of the IEEE. In *International Conference on Acoustics, Speech, and Signal Processing*.

Blatter, A. (1980). *Instrumentation/orchestration*. New York.

Bock, S., Artz, A., Krebs, F., & Schedl, M. (2012). Online real-time onset detection with recurrent neural networks. In *International Conference on Digital Audio Effects*, pp. 15–18. York, UK.

Bresin, R. & Battel, G. U. (2000). Articulation Strategies in Expressive Piano Performance Analysis of Legato, Staccato, and Repeated Notes in Performances of the Andante Movement of Mozart's Sonata in G Major (K 545). *Journal of New Music Research*, *29*(3), 211–224.

Brossier, P. (2006). *Automatic annotation of musical audio for interactive systems*. Ph.D. thesis, Centre for Digital music, Queen Mary University of London.

Buhusi, C. V. & Meck, W. H. (2005). What makes us tick? Functional and neural mechanisms of interval timing. *Nature Reviews Neuroscience*, *6*, 755–765.

Burns, A. & Wanderley, M. (2006). Visual methods for the retrieval of guitarist fingering. In *NIME '06: Proceedings of the 2006 conference on New interfaces for musical expression*, pp. 196–199.

Carlevaro, A. (1974). Serie Didactica para Guitarra. Barry Editorial.

Chen, Y., Ding, M., & Kelso, J. (2001). Origins of timing errors in human sensorimotor coordination. *Motor Behavior*, *33*, 3–8.

Chew, G. (2008). Articulation and phrasing. *Grove Music Online.*

Clarke, E. F. (2001). Generative principles in music performance. In J. A. Sloboda (Ed.) *Generative Processes in Music: the Psychology of Performance, Improvisation and Composition*, chap. 1, pp. 1–26. Oxford, UK: Oxford University Press.

Classtab.org (2006). Classical guitar tablature. \url{http://www.classtab.org}.

Clerc, M. & Kennedy, J. (2000). The particle swarm: explosion, stability and conver- gence in a multi-dimensional space. In *IEEE transactions on Evolutionary Com- putation*, pp. 158–173.

Collins, N. (2005). A change discrimination onset detector with peak scoring peak picker and time domain correction. In *Mirex*.

Copland, A. (1939). *What to listen for in music*. University of California Press.

Costalonga, L. & Miranda, E. R. (2008). Equipping artificial guitar players with biomechanical constraints: a case study of precision and speed. In *International Computer Music Conference*.

Cuzzucoli, G. & Lombardo, V. (1999). A physical model of the classical guitar, including the player's touch. *Computer Music Journal*, *23*(2), 52–69.

Dahlhaus, C. (1989). *19th-Century Music*. University of California Press.

de Boer, B. (1976). On the "residue" and auditory pitch perception. In W. Keidel & W. Neff (Eds.) *Handbook of Sensory Physiology*, pp. 479–583. Berlin, Germany: Springer.

de Cheveigné, A. (2005). Pitch perception models. In C. Plack, A. Oxenham, R. Fay, & A. Pooper (Eds.) *Pitch - Neural coding and perception.*, chap. 6, pp. 169–233. New York: Springer-Verlag.

de Cheveigné, A. & Kawahara, H. (2002). YIN, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, *111*(4), 1917–1930.

de Mantaras, R. L. & Arcos, J. L. (2002). AI and music from composition to expressive performance. *AI Mag.*, *23*(3), 43–57.

Delignieres, D., Torre, K., & Lemoine, L. (2009). Long-range correlation in synchronization and syncopation tapping: a linear phase correction model. *PloS one*, *4*(11), e7822.

Devaney, J. & Ellis, D. (2009). Handling asynchrony in audio-score alignment. In *International Computer Music Conference*. San Francisco, California.

Dixon, S. (2001). Automatic extraction of tempo and beat from expressive perfomances. *Journal of New Music Research*, *30*(1).

Dixon, S. & Widmer, G. (2006). MATCH: A Music Alignment Tool Chest. In *International Conference on Music Information Retrieval (ISMIR)*.

Dobrian, C. & Koppelman, D. (2006). The 'E' in NIME: musical expression with new computer interfaces. In *Conference on New Interfaces for Musical Expression*, pp. 277–282. Paris, France.

Dodge, C. & Jerse, T. A. (1985). *Computer Music: Synthesis, Composition, and Performance*. Macmillan Library Reference.

Duncan, C. (1980). *The art of classical guitar playing*. Miami, FL: Summy-Birchard, Inc.

Erkut, C., Valimaki, V., Karjalainen, M., & Laurson, M. (2000). Extraction of physical and expressive parameters for model-based sound synthesis of the classical guitar. In *108th AES Convention*, pp. 19–22.

Foote, J. & Cooper, M. (2000). Automatic audio segmentation using a measure of audio novelty. In *IEEE International Conference on Multimedia and Expo*, pp. 452–455. New York, USA.

Gabrielsson, A. (1987). Once again: The theme from Mozart's piano sonata in A major (K. 331). A comparison of five performances. In A. Gabrielsson (Ed.) *Action and perception in rhythm and music*, pp. 81–103. Stockholm: Royal Swedish Academy of Music.

Gabrielsson, A. (1995). Expressive Intention and Performance. In R. Steinberg (Ed.) *Music and the Mind Machine*, pp. 35–47. Berlin: Springer-Verlag.

Gabrielsson, A. (1999). The performance of music. In D. Deutsch (Ed.) *The Psychology of Music*, chap. 14, pp. 501–602. Waltham, USA: Academic Press, 2nd edn.

Gabrielsson, A. (2001). Timing in music performance and its relations to music experience. In J. A. Sloboda (Ed.) *Generative Processes in Music: the Psychology of Performance, Improvisation and Composition*, chap. 2, pp. 27–51. Oxford, UK: Oxford University Press.

Gabrielsson, A. (2003). Music performance research at the millenium. *Psychology of Music*, *31*(3), 221–272.

Giraldo, S. & Ramirez, R. (2012). Modeling Embellishment, Timing and Energy Expressive Transformations in Jazz Guitar. In *International Workshop on Machine Learning and Music*. Edinburgh.

Goebl, W. & Palmer, C. (2013). Temporal control and hand movement efficiency in skilled music performance. *PLoS ONE*, *8*(1), e50901.

Goldstein, E. B. (2001). *Blackwell Handbook of Perception*. Wiley-Blackwell.

Gouyon, F., Fabig, L., & Bonada, J. (2003). Rhythmic expressiveness transformations of audio recordings: swing modifications. In *6th International Conference on Digital Audio Effects (DAFx)*. London, UK.

Gouyon, F., Herrera, P., Gómez, E., Cano, P., Bonada, J., Loscos, A., Amatriain, X., & Serra, X. (2008). *Content Processing of Music Audio Signals*, chap. 3, pp. 83–160. Berlin: Logos Verlag Berlin GmbH.

Grachten, M. (2006). *Expressivity-aware tempo transformations of music performances using case based reasoning*. Ph.D. thesis, Pompeu Fabra University.

Grachten, M., Arcos, J., & de Mántaras, R. L. (2006). A case based approach to expressivity-aware tempo transformation. *Machine Learning*, *65*(2-3), 411–437.

Grachten, M. & Widmer, G. (2009). Who is who in the end? Recognizing pianists by their final ritardandi. In *Proc. of the Int. Soc. for Music Information Retrieval Conf. (ISMIR)*, pp. 51–56.

Guaus, E., Ozaslan, T., Palacios, E., & Arcos, J. L. (2010). A Left Hand Gesture Caption System for Guitar Based on Capacitive Sensors. *Interfaces*, (Nime), 238–243.

Gusfield, D. (1997). *Algorithms on strings, trees, and sequences: computer science and computational biology*. Cambridge, UK: Cambridge University Press.

Hainsworth, S. & Macleod, M. (2003). Onset detection in music audio signals. In *Proc. of the Int. Computer Music Conf. (ICMC)*, pp. 163–167.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, *11*(1), 10–18.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning*. Berlin, Germany: Springer, 2nd edn.

Heijink, H. & Meulenbroek, R. G. J. (2002). On the Complexity of Classical Guitar Playing:Functional Adaptations to Task Constraints. *Journal of Motor Behavior*, *34*(4), 339–351.

Henderson, M. T. (1937). Rhytmic organization in artistic piano performance. *University of Iowa studies in the psychology of music*, *4*, 281–385.

Hennig, H., Fleischmann, R., Fredebohm, A., Hagmayer, Y., Nagler, J., Witt, A., Theis, F. J., & Geisel, T. (2011). The nature and perception of fluctuations in human musical rhythms. *PLoS ONE*, *6*(10), e26457.

Hollander, M. & Wolfe, D. A. (1999). *Nonparametric statistical methods*. New York, USA: Wiley, 2nd edn.

Istók, E., Friberg, A., Huotilainen, M., & Tervaniemi, M. (2013). Expressive timing facilitates the neural processing of phrase boundaries in music: evidence from event-related potentials. *PLoS ONE*, *8*(1), e55150.

Jaffe D., A. & Smith J., O. (1983). Extensions of the Karplus-Strong plucked string algorithm. *Computer Music Journal*, *7*(2), 56–69.

Janosy, Z., Karjalainen, M., & Välimäki, V. (1994). Intelligent Synthesis Control with Applications to a Physical Model of the Acoustic Guitar. In *Proc.\ of the 1994 International Computer Music Conference*, pp. 402–406. Aarhus, Denmark.

Järveläinen, H. (2002). Perception-based Control of Vibrato Parameters in String Instrument Synthesis. In *International Computer Music Conference.(ICMC)*, pp. 287–294.

Jehan, T. (1997). *Music signal parameter estimation*. Ph.D. thesis, Berkeley.

Johnstone, J. A. (1913). *Phrasing in piano playing*. Withmark New York.

Juslin, P. N. (2001). Communicating emotion in music performance: a review and a theoretical framework. In P. N. Juslin & J. A. Sloboda (Eds.) *Music and emotion: theory and research*, pp. 309–337. New York: Oxford University Press.

Juslin, P. N. (2003). Five facets of musical expression: a psychologist's perspective on music performance. *Psychology of Music*, *31*(3), 273–302.

Juslin, P. N. & Sloboda, J. A. (2001). *Music and emotion: theory and research*. Oxford, UK: Oxford University Press.

Juslin, P. N. & Sloboda, J. A. (2013). Music and emotion. In D. Deutsch (Ed.) *The Psychology of Music*, chap. 15, pp. 583–645. Waltham, USA: Academic Press, 3rd edn.

Karplus, K. & Strong, A. (1983). Digital synthesis of plucked string and drum timbres. *Computer Music Journal*, *7*(2), 43–55.

Kennedy, J. & Eberhart, R. (2001). *Swarm intelligence*. Burlington, USA: Morgan Kaufmann.

Klapuri, A. (1999). Sound onset detection by applying psychoacoustic knowledge. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 3089–3092.

Laurson, M., Hiipakka, J., Erkut, C., Karjalainen, M., Valimaki, V., & Kuuskankare, M. (1999). From Expressive Notation to Model-Based Sound Synthesis: a Case Study of the Acoustic Guitar. In *In the International Computer Music Conference*.

Lee N., D. Z. & Smith J., O. (2007). Excitation signal extraction for guitar tones. In *International Computer Music Conference (ICMC)*.

Levy, E. (1970). Compositional Technique and Musical Expressivity. *Journal of Research in Music Education*, *18*(1), 3–15.

Liem, C. C. S., Hanjalic, A., & Sapp, C. S. (2011). Expressivity in musical timing in relation to musical structure and interpretation: a cross-performance, audio-based approach. In *Proc. of the Audio Engineering Soc. Conf. (AES), paper no. 6-1.*

Lin, J., Keogh, E., Wei, L., & Lonardi, S. (2007). Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, *15*(2), 107–144.

Lindström, E. (1992). 5 x "Oh, my darling Clementine". The influence of expressive intention on music performance.

López de Mántaras, R. & Arcos, J.-L. (2012). Playing with cases: Rendering Expressive Music with Case-Based Reasoning. *AI Magazine*, *33*(4), 22–32.

Martens, W. L. & Marui, A. (2006). Categories of Perception for Vibrato, Flange, and Stereo Chorus: Mapping out the Musically Useful Ranges of Modulation Rate and Depth for Delay Based Effects. In *International Conference on Digital Audio Effects.*

McKay, C. (2003). A survey of physical modeling techniques for synthesizing the classical guitar. Tech. rep., McGill University.

Meyer, L. (1989). *Style and music : theory, history, and ideology.* University of Pennsylvania.

Migneco, R. V. (2012). *Analysis and synthesis of expressive guitar performance.* Ph.D. thesis, Drexel University.

Mitchell, T. M. (1997). *Machine Learning.* New York, USA: McGraw-Hill.

Müller, M., Ellis, D. P. W., Klapuri, A., & Richard, G. (2011). Signal processing for music analysis. *IEEE Journal of Selected Topics in Signal Processing*, *5*(6), 1088–1110.

Norton, J. (2008). *Motion capture to build a foundation for a computer-controlled instrument by study of classical guitar performance.* Ph.D. thesis, Stanford University.

Orio, N. (1999). The timbre space of the classical guitar and its relationship with the plucking techniques. In *International Computer Music Conference.*

Özaslan, T. H., Serra, X., & Arcos, J. L. (2012). Characterization of Embellishments in Ney Performances of Makam Music in Turkey. In *The International Society for Music Information Retrieval.*

Pachet, F. (2003). The Continuator: Musical interaction with style. *Journal of New Music Research*, *32*(3), 333–341.

Palmer, C. (1996). Anatomy of a performance: Sources of musical expression. *Music Perception*, *13*(3), 433–453.

Pang, H.-S. & Yoon, D.-H. (2005). Rapid and brief communication: Automatic detection of vibrato in monophonic music. *Pattern Recognition*, *38*(7), 1135–1138.

Penel, A. & Drake, C. (1998). Sources of timing variations in music performance: a psychological segmentation model. *Psychological Research*, *61*, 12–32.

Quested, G., Boyle, R. D., & Ng, K. (2008). Polyphonic note tracking using multimodal retrieval of musical events. In *International Computer Music Conference*. Belfast.

Radicioni, D. P. & Lombardo, V. (2007). A constraint-based approach for annotating music scores with gestural information. *Constraints*, *12*(4), 405–428.

Radisavljevic, A. & Driessen, P. (2004). Path difference learning for guitar fingering problem. In *International Computer Music Conference(ICMC)*.

Ramirez, R. & Hazan, A. (2005). Modeling expressive music performance in jazz. In *International Florida Artificial Intelligence Research Society Conference*.

Ramirez, R., Perez, A., Kersten, S., Rizo, D., Roman, P., & Inesta, J. M. (2010). Modeling violin performances using inductive logic programming. *Intelligent Data Analysis*, *14*(5).

Repp, B. H. (1990). Patterns of expressive timing in performances of a Beethoven minuet by nineteen famous pianists. *Journal of the Acoustical Society of America*, *88*(2), 622–641.

Repp, B. H. (1992). Diversity and commonality in music performance: an analysis of timing microstructure in Schumann's "Träumerei". *Journal of the Acoustical Society of America*, *92*(5), 2546–2568.

Repp, B. H. (1998). A microcosm of musical expression. I. Quantitative analysis of pianists' timing in the initial measures of Chopin's Etude in E major. *Journal of the Acoustical Society of America*, *104*(2), 1085–1100.

Rossignol, S., Depalle, P., Soumagne, J., Rodet, X., & Collette, J. (1999a). Vibrato: Detection, Estimation, Extraction, Modification. In *Digital Audio Effects Workshop*.

Rossignol, S., Rodet, X., Soumagne, J., L, C. J., & Depalle, P. (1999b). Automatic characterization of musical signals: Feature extraction and temporal segmentation. *Journal of New Music Research*, *28*(4), 281–295.

Saunders, C., Hardoon, D. R., Shawe-taylor, J., & Widmer, G. (2004). Using string kernels to identify famous performers from their playing style. In *In Proceedings of the 15th European Conference on Machine Learning*, pp. 384–395.

Scholz, R. & Ramalho, G. (2008). COCHONUT: Recognizing complex chords from MIDI guitar sequences. In *Proceedings of the The International Society for Music Information Retrieval*. Belfast.

Sloboda, J. A. (1983). The communication of musical metre in piano performance. *Quarterly Journal of Experimental Psychology*, *35A*, 377–396.

Solis, J., Suefuji, K., Taniguchi, K., Ninomiya, T., Maeda, M., & Takanishi, A. (2007). Implementation of Expressive Performance Rules on the WF-4RIII by modeling a professional flutist performance using NN. In *IEEE International Conference on Robotics and Automation*, pp. 2552–2557. Roma.

Stamatatos, E. & Widmer, G. (2005). Automatic identification of music performers with learning ensembles. *Artificial Intelligence*, *165*, 37–56.

Thornburg, H., Leistikow, R. J., & Berger, J. (2007). Melody extraction and musical nset detection via probabilistic models of framewise STFT peak data. *Audio, Speech, and Language Processing, IEEE Transactions*, *15*(4), 1257–1272.

Timmers, R. & Desain, P. (2000). Vibrato: Questions And Answers From Musicians And Science. In *Proceedings of the sixth International Conference on Music Perception and Cognition*.

Timmers, R. & Honing, H. (2002). On music performance, theories, measurement and diversity. *M.A. Belardinelli (ed.). Cognitive Processing (International Quarterly of Cognitive Sciences)*, pp. 1–2,1–19.

Todd, N. (1992). A model of expressive timing in tonal music. *Music Perception*, *91*, 3540–3550.

Trajano, E., Dahia, M., Santana, H., & Ramalho, G. (2004). Automatic discovery of right hand fingering in guitar accompaniment. In *In Proceedings of the International Computer Music Conference (ICMC)*, pp. 722–725.

Traube, C. & Depalle, P. (2003). Extraction Of The Excitation Point Location On A String Using Weighted Least-Square Estimation Of A Comb Filter Delay. In *In Procs. of the 6th International Conference on Digital Audio Effects (DAFx-03*.

Tuohy, D. R. (2006). *Creating tablature and arranging music for guitar with genetic algorithms and artificial neural networks,*. Master's thesis, University of Georgia.

Tuohy D., R. & Potter W., D. (2005). A genetic algorithm for the automatic generation of playable guitar tablature. In *International Computer Music Conference*, pp. 499–502.

Verfaille, V., Guastavino, C., & Depalle, P. (2005). Perceptual Evaluation of Vibrato Models. *Conference on Interdisciplinary Musicology (CIM05)*, pp. 1–19.

Wen, X. & Sandler, M. (2008). Analysis and Synthesis of Audio Vibrato Using Harmonic Sinusoids. In *Audio Engineering Society Convention 124*.

Witten, I. H. & Frank, E. (2005). *Data mining: practical machine learning tools and techniques*. Waltham, USA: Morgan Kaufmann, 2nd edn.

Witten, I. H., Frank, E., Trigg, L., Hall, M., Holmes, G., & Cunningham, S. (1999). Weka: Practical Machine Learning Tools and Techniques with Java Implementations. In *CONIP/ANZIIS/ANNES'99 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems*, pp. 192–196.

Appendix A: music collection

	J.S. Bach	J.S. Bach	A.Barrios	Anonymous	F. Tarrega	S. Myers	F. Tarrega	A. Barrios	L.V. Beethoven	F. Sor		
	BWV 1007	BWV 999	Catedral, Prelude	Romance	Lagrima	Cavatina	Adelita	C Min. Prelude	Moonlight Sonat	Etude BMin.	Born	Death
Alexandre Pier Federici			2011-AndanteMusic								-	-
Ana Vidovic						2007-YouTube					1988	Alive
Andrea Gasperi			1996-ScreenStuio								1964	Alive
Andrei Krylov	2009 - AndreiKrylov							2007-AndreiKrylov Music			1959	Alive
Andres Segovia		1965 - Bravo! Records								1962-MCAClassics	1893	1987
Andrew Schulman		1989-CentaurRecords									1960	Alive
Angel Romero						1990-Telarc		2001-Delos	1976-AngelRecords		1946	Alive
Bob Fetherolf		2008-FullSailMusic									-	-
Cary Greisch							2001-BellaMusic				1958	Alive
Cesar Amaro					2009-YouTube						1948	2012
Chandra Rajagopal							2009-Youtube				1987	Alive
Christoper Parkening						2007-AngelRecords					1947	Alive
Craig Ogden						2010-X5MusicGroup					1988	Alive
Cristiano Porqueddu			2009-BrilliantClassics								1975	Alive
Dan Hopson				2011-DanHopson							1950	Alive

Figure 1: Music collection table 1/5

	J.S. Bach	J.S. Bach	A.Barrios	Anonymous	F. Tarrega	S. Myers	F. Tarrega	A. Barrios	L.V. Beethoven	F. Sor		
	BWV 1007	BWV 999	Catedral, Prelude	Romance	Lagrima	Cavatina	Adelita	C Min. Prelude	Moonlight Sonat	Etude BMin.	Born	Death
Daniele Magli										2010-YouTube	1960	Alive
Danny Masters			2010-DannyMasters								1976	Alive
Eduardo Fernandez		1989-Decca MusicGroup Limited									1952	Alive
Edward Trybek			2007-EdwardTrybek								1982	Alive
Eric Henderson								2009-YouTube			1958	Alive
Erling Moldrup				2011-Muzart							1943	Alive
Eros Roselli										2008-YouTube	1966	Alive
Filomena Moretti			2010-Transart								1973	Alive
Franciscus Terpstra								2001-Franciscus Terpstra			1949	Alive
Gareth Koch		1997-AustralianBroadcastingCorporatio									1962	Alive
Gerry Johnston				2011-GerryJohnston							1950	Alive
Goran Sollicher	1992-DeutscheGrammophonGmbH									1998-DeutscheGrammophonGmbH	1955	Alive
Irina Kulikova	2011-Naxos										1979	Alive
Jean Jacques Fimbel									2010-JeanJacque Fimbel		1955	Alive
Jen Chi Encin							x				1966	Alive
Jerome Ducharme							2005-YouTube				1978	Alive

Figure 2: Music collection table 2/5

	J.S. Bach	J.S. Bach	A.Barrios	Anonymous	F. Tarrega	S. Myers	F. Tarrega	A. Barrios	L.V. Beethoven	F. Sor		
	BWV 1007	BWV 999	Catedral, Prelude	Romance	Lagrima	Cavatina	Adelita	C Min. Prelude	Moonlight Sonat	Etude BMin.	Born	Death
Joe						2010-YouTube					1980	Alive
John Demans					x						-	-
John H. Clarke								2007-UrbanTribe Production			1982	Alive
John Q										2009-YouTube	-	-
John Williams	1958-IDIS						1995-Sony				1941	Alive
Jonas Lefvert						2011-YouTube					1981	Alive
Jonathan Adams		1998-SonicGrapefruit			1998-SonicGrapefruit				1998-SonicGrapefruit		1961	Alive
Joseph Sullinger									2008-EroicaClassical Recordings		1969	Alive
Juanillo De Alba				2011-CountdownMedia GmbH							1952	Alive
Julian Bream										2006-YouTube	1933	Alive
Kevin McCormick			2007-MiralisRecords	2004-MirablisRecords	2004-MirablisRecords				2004-MirablisRecords		1968	Alive
Lianto Tjahjoputro							2009-YouTube				1963	Alive
Liona Boyd					2004-MostonRecords				2004-MostonRecords		1962	Alive
Luigi Attademo		2009-BrilliantClassics									1972	Alive
Manuel Barrueco		2004-BigJoKeMusic	2005-EMI								1952	Alive
Marcelo Kayath									2009-MusicalConcepts		1964	Alive

Figure 3: Music collection table 3/5

	J.S. Bach	J.S. Bach	A.Barrios	Anonymous	F. Tarrega	S. Myers	F. Tarrega	A. Barrios	L.V. Beethoven	F. Sor		
	BWV 1007	BWV 999	Catedral, Prelude	Romance	Lagrima	Cavatina	Adelita	C Min. Prelude	Moonlight Sonat	Etude BMin.	Born	Death
Michael Lucarelli								2006-YouTube			1959	Alive
Michel Fiorelli							2010-Youtube				1961	Alive
Milos Karadaglic					2011-Deutsche Grammophon GmbH				2011-Deutsche Grammophon GmbH		1982	Alive
Narciso Yepes										2003-Deutsche Grammophon GmbH	1960	Alive
Nelson Amos				2008-NelsonAmas							1952	Alive
Pascal Beausseron							2012-YouTube				1970	Alive
Per Olov Kindgren										2007-YouTube	1956	Alive
Pere Salicru				2010-Edivox							1964	Alive
Pete Downes	2007-WiserProduction										1952	Alive
Raoulle Sansfaon							2010-YouTube				-	-
Rey De La Torre					1947-Classical Monuments						1917	1994
Ricardo Prieto	2009-RicardoPrieto										1973	Alive
Richard Ames	2009-YouTube										1965	Alive
Robert Westaway	2005-BluePebbleMusic	2005-BluePebbleMusic		2005-BluePebbleMusic	2005-BluePebbleMusic				2005-BluePebbleMusic		1985	Alive
Rodrigo Escoba								2011-TrackMusic			-	-
Rodrigo Lorente				2012-NatTeamMedia							-	-

Figure 4: Music collection table 4/5

	J.S. Bach	J.S. Bach	A.Barrios	Anonymous	F. Tarrega	S. Myers	F. Tarrega	A. Barrios	L.V. Beethoven	F. Sor		
	BWV 1007	BWV 999	Catedral, Prelude	Romance	Lagrima	Cavatina	Adelita	C Min. Prelude	Moonlight Sonat	Etude BMin.	Born	Death
Roger Lurel							2011-YouTube				1969	Alive
Sean Kelly	2007-OpeningDay Entertainment	2007-OpeningDay Entertainment			2007-OpeningDay Entertainment	2007-OpeningDay Entertainment					1958	Alive
Sharon Isbin		2002-WarnerClassics									1956	Alive
Simon Dinnigan						2009-Sony					1968	Alive
Soren Bodker Madsen										1998-Barbarossa	1956	Alive
Stanley Myers						2001-Milan Entertainment					1930	1993
Susan McDonald			1995-MayflyRecords								1965	Alive
Taylor Jones	2011-LionGroup Records										-	-
Terry Muska									2011-TalkingTaco Music		1946	Alive
Tim Hall								2008-YouTube			1965	Alive
Tom Tilley				2012-Custodian Records	2012-Custodian Records						1965	Alive
Tom Ward						2011-YouTube					1983	Alive
Vicente Covers			2010-Naxos								1982	Alive
YouTube Unknown Performer 1										2010-YouTube	-	-
YouTube Unknown Performer 2								x			-	-
YouTube Unknown Performer 3								x			-	-

Figure 5: Music collection table 5/5

Appendix B: publications by the author

ISI-indexed peer-reviewed journals

Serrà, J., Özaslan, T., & Arcos, J.L. (2013). Note onset deviations as musical piece signatures. PLoS ONE, vol.8(7) p. e69268.

Arcos, J.L., Guaus, E., & Özaslan, T., (2013). Analyzing musical expressivity with a soft computing approach, Fuzzy Sets and Systems, vol. 214: Elsevier, pp. 65-74.

Invited book chapters

Özaslan, T., Guaus, E., Palacios, E., & Arcos, J.L. (2011). Identifying attack articulations in classical guitar. Computer Music Modeling and Retrieval. Exploring Music Contents. Lecture Notes in Computer Science, vol. 6684: Spriger-Verlag, pp. 219-241.

Full-article contributions to peer-reviewed conferences

Özaslan, T., Serra, X., & Arcos, J.L. (2012). Characterization of embellishments in ney performances of makam music in Turkey. International society for Music Information Retrieval Conference (ISMIR), Porto, Portugal, pp. 13-18.

Guaus, E., Özaslan, T., Palacios, E., & Arcos, J.L. (2010).A left hand gesture caption system for guitar based on capacitive sensors. International Conference on New Interfaces for Musical Expression (NIME), pp. 238-243.

Özaslan, T., Guaus, E., Palacios, E., & Arcos, J.L. (2010). Attack based articulation analysis of nylon string guitar. International Symposium on Computer Music Modeling and Retrieval (CMMR), pp. 285-297.

Özaslan, T., & Arcos, J.L. (2010). Legato and glissando identification in classical guitar. Sound and Music Computing Conference (SMC), pp. 457-463.

Work Shop

Özaslan, T., Serra, X., & Arcos, J.L. (2012). Signal analysis of ney performances, CompMusic Workshop.

Technical Reports

Özaslan, T., & Arcos, J.L. (2011). Automatic vibrato detection in classical guitar. IIIA Technical Reports, TR-IIIA-2011-05.

Thesis

Özaslan, T. (2009). Expressive analysis of violin performers. Master's thesis, Universitat Pompeu Fabra, Barcelona, Spain.