Automatic recognition of Persian musical modes in audio musical signals

PhD Thesis

Peyman Heydarian
Music Technology Group
Faculty of Art, Architecture and Design
London Metropolitan University
Email: nivpey@gmail.com

June 2016

Abstract

This research proposes new approaches for computational identification of Persian musical modes. This involves constructing a database of audio musical files and developing computer algorithms to perform a musical analysis of the samples. Essential features, the spectral average, chroma, and pitch histograms, and the use of symbolic data, are discussed and compared. A tonic detection algorithm is developed to align the feature vectors and to make the mode recognition methods independent of changes in tonality. Subsequently, a geometric distance measure, such as the Manhattan distance, which is preferred, and cross correlation, or a machine learning method (the Gaussian Mixture Models), is used to gauge similarity between a signal and a set of templates that are constructed in the training phase, in which data-driven patterns are made for each *dastgàh* (Persian mode). The effects of the following parameters are considered and assessed: the amount of training data; the parts of the frequency range to be used for training; down sampling; tone resolution (12-TET, 24-TET, 48-TET and 53-TET); the effect of using overlapping or non-overlapping frames; and silence and high-energy suppression in pre-processing. The santur (hammered string instrument), which is extensively used in the musical database samples, is described and its physical properties are characterised; the pitch and harmonic deviations characteristic of it are measured; and the inharmonicity factor of the instrument is calculated for the first time.

The results are applicable to Persian music and to other closely related musical traditions of the Mediterranean and the Near East. This approach enables content-based analyses of, and content-based searches of, musical archives. Potential applications of this research include: music information retrieval, audio snippet (thumbnailing), music archiving and access to archival content, audio compression and coding, associating of images with audio content, music transcription, music synthesis, music editors, music instruction, automatic music accompaniment, and setting new standards and symbols for musical notation.

Acknowledgements

I am grateful for invaluable help and support from my family, friends, musicians, ethnomusicologists and scientists, and from the many others who have assisted me and collaborated with me during the course of my thesis.

I would particularly like to thank my supervisors, Lewis Jones and Allan Seago, for their support and helpful advice during the course of my PhD research. I started this PhD in part because while I was lecturing on music technology at London Metropolitan University Lewis Jones suggested to me to pursue my PhD studies there, and this research has benefitted from his wide-ranging multidisciplinary approach to musical inquiry.

I would like to thank my former BSc and MSc lecturers and supervisors in Shiraz and Tarbiat Modarres Universities: Professor Mohammad Ali Masnadi Shirazi for his support whenever I needed it, Professor Rahim Ghayour, Dr Mojtaba Lotfizad, and especially Professor Ehsanollah Kabir, who encouraged me to do my Master's thesis on music note recognition for santur and directed me to this field. I would also like to thank Dr Kambiz Badie, my supervisor at Iran Telecom Research Center who for the first time encouraged me to explore the possibility of Persian *dastgàh* recognition. Thanks to Dr Christ Harte, Professor Juan Pablo Bello, Professor Baris Bozkurt and Professor Emilia Gomez for kindly sharing their computer codes with me. Thanks also to my MPhil supervisors Professor Josh Reiss and Professor Mark Plumbley.

And thank you to my musician colleagues whose performances with me are used in my analysis and also for fruitful chats with them: Vassilis Chatzimakris, Avan Abdullah, Suna Alan, Aysegul Erdogan, Vasiliki Anastasiou, Cigdem Aslan, Dr Pejman Azarmina, Dr Mohammadali Merati, Olcay Bayir, Leonardo Cini, Aygul Erce, Parastoo Heydarian, Francesco Iannuzzelli, Mansour Izadpanah, Mohammad Jaberi, Ali Torshizi, Sarwat Koyi, Ewan Macdonald, Christelle Madani, Emad Rajabalipour, Rana Shieh, Forough Jokar, Shirley Smart, Rihab Azar and Elaha Soroor.

Thanks to my mother and sister, Nasrin Danesh and Parastoo Heydarian, for their love and support. This thesis is dedicated to my family and to the memory of my father Heshmatollah Heydarian.

Finally, thanks to Ed Emery, Afsaneh Rasaei, Nasser Danesh, Seddigh Tarif, and to Professor Owen Wright for the invaluable discussions and musical collaborations that we have had.

Table of Contents

List of Figures

List of Tables

1 Introduction

The aim of this PhD research is to enable automatic *dastgàh* (mode) recognition in audio signals conveying Persian music; and the methods described are also applicable to closely related musical traditions. To this end, a method for the automatic identification of the modal tonic and the mode in musical signals is presented and discussed, along with discussion of the effect of parameters and their optimised values.

1.1 Motivations for and the aims of this PhD research

The *dastgàh* is the underlying system of pitch organisation classical Persian music, which defines the scale notes and their relative importance, and sets the path for melodic progress [1]. It also accounts for the mood of a piece to some extent.

Although in the fields of music analysis and synthesis a tremendous amount of work has been done on Western music since the late twentieth century, relatively little work has been carried out on non-Western musical traditions, and this is notably the case for Iranian music, which is the principal focus of this study. This research reflects the author's equal, parallel engagement in the theory and practice of music and in signal processing. The author's background as a composer and performer of Iranian music motivated him to start working on a music transcription system for Iranian music during his MSc research (1998– 2000), and he was the first researcher to publish a work on Persian musical signal processing [2]. He continued working in this field during the course of his MPhil research at the Centre for Digital Music at Queen Mary, University of London (2004-08), where he further advanced transcription algorithms and proceeded to work on Empirical Mode Decomposition and its application to the extraction of long-term rhythmic structures, and also developed a mode recognition algorithm which was presented in his MPhil thesis [5]. Before embarking on the present PhD research he worked on ethnomusicology and music performance at SOAS, University of London (2009-10). While the work presented here builds upon and substantially advances upon the methods presented in the MPhil thesis (2008), extensive further research, undertaken since 2008, is presented and discussed in this thesis.

1.2 Applications of automated *dastgàh* recognition

Potential applications of automated *dastgàh* recognition include: music information retrieval, audio snippet (thumbnail), music archiving and access to the musical content, audio compression and coding, associating images with audio content, music transcription, music synthesis, automatic

music accompaniment, music editors, music instruction, setting new standards and symbols for musical notation, and copyrighting. A valuable application of this research is to provide access to the content of files and automatically to add a tag for the musical mode. If such a tag is generated automatically at the time of uploading the musical work, it can then be accessed via text queries and by introducing a new search criterion. *Dastgàh* recognition can help music students and the general public in becoming familiar with the mode of a piece that they hear. More generally, by increasing the knowledge of the public, it can potentially revolutionise the way people perceive and have access to music.

1.3 Persian musical scales and their structure

Persian music is based on a modal system of 7 main modes and their 5 derivative modes. Each mode is characterised by a modal tonic, a rest note (*ist*, see Section 2.1), a particular pattern of scalar intervals, and typical melodic motifs. They can be classified into 5 distinct scalar tunings, which are the subject of this investigation. They are described in detail in Chapter 2.

1.4 Research challenges presented by Persian scales

The *dastgàh*, an important feature of Persian music, is worthy of consideration in the metadata of audio musical files: it is the underlying system of Persian music and specifies the scale and the modal tonic; it conveys emotions which are usually associated with a piece; and the modal information of audio files potentially connects different styles and traditions which are historically connected. For instance, Iranian and non-Iranian pieces can be associated, based on their mode. Mode recognition can also be used for play-list generation, connecting people beyond modern political and language borders. It has applications in music education, and is a precious tool for ethnomusicologists who know the methodology but are not versed in a particular tradition such that they are able to distinguish aurally between different modes and to understand this aspect of a particular musical tradition to the extent that an indigenous musician would understand them. It is also possible that the computational methods accounted for here will enable us to extract new features, information and characteristics which are not normally perceivable. Furthermore, knowing the scale and mode of a piece improves the reliability of automatic music transcription systems.

1.5 Context, scope and constraints of the research

This interdisciplinary thesis is located in and between three disciplines: digital signal processing (DSP), musical practice, and ethnomusicology. While the first involves signal processing tools and methods for the analysis of audio musical files, the latter two define the orientation of the research and the musical parameters to be considered.

Until recently there was not ubiquitous public access to the audio content of musical files for the majority of online music. For example, people can search for songs on YouTube and social media via text tags, but not through the audio content. Although it is also now possible to search for files via an audio musical query, for instance by query-by-humming (i.e. sung melodies), by query-by-example, (i.e. playing a melody or theme), or by playing a track and searching for the match on websites that provide such a service (see section 4.9), it is not yet possible to search via the *maqàmic* musical mode, from which nearly a billion people within the relevant cultural sphere (that of Persian, Kurdish, Arabic, Turkish and Greek music) could potentially benefit.

This thesis addresses this problem and deals mainly with measurable physical aspects of Persian music, principally the notes and harmonic content of a given musical audio file. In doing so, social, cultural and psychological aspects of music are acknowledged as significant contextual elements. Although detailed knowledge of the social and cultural contexts of music and of the people who create and maintain its traditions are essential to a thorough understanding of a musical system, they are not accessible in the audio signal alone, and might to some extent be indicated by the use of metadata (e.g. tags and keywords).

The algorithms developed in this research operate on audio musical signals. Performing the analysis on symbolic data (e.g. MIDI files) would be a much easier task, as in that case the notes, their durations and other attributes are known, as would also be the scale, pitch histograms, and melody contours, but that line of analysis is not considered here.

Geographical span

Iran is situated at the crossroads of several different cultures and civilisations. Over the ages, Iranian music and culture have been in contact with European and neighbouring Turkish, Arabic, and Indian musics and cultures. (The terms Iranian and Persian are discussed in Chapter 2.) The *dastgàh*, the underlying system of Persian classical music, is a phenomenon which on the one hand is closely related in modal structure to the *maqàm* system of Iranian ethnic music and Arabic, Turkish and Greek musical traditions; and on the other, is related to Indian *raga* and the Western

concept of key. The main area of investigation in this research is Persian classical music, played principally on the santur. However, the sphere of influence of Persian music exceeds the borders of Iran, and the applicability of the algorithms considered here in relation to Persian music is investigated beyond modern-day Iranian borders, to other musical cultures within Iran, and, more generally, to *maqàmic* musical traditions which are closely related to Persian music. Additionally, Persian classical music performed on various instruments other than the santur, and other musical genres including contemporary compositions and performances, are considered.

1.6 Contributions of the thesis

Most existing MIR research on *maqàmic* music uses methods which were initially developed for Western music analysis, yet authors typically emphasise differences between Western and non-Western musics. While signal processing projects on *maqàmic* music (Turkish, Arabic, Indian, etc.; see Chapter 3) typically take some account of the subtleties of the particular musical systems addressed, the musical characteristics they acknowledge differ from those needed for the analysis of Persian music. Computational systems devised initially for Western music are here modified and adapted to work on Iranian musical signals. Whereas the existing projects are typically conceived from a primarily technical viewpoint, the technical work discussed in this thesis is founded upon and complemented by a detailed study of the Persian musical system. The approach adopted here is both data-driven and music theory-based.

It cannot be assumed that audio samples of Iranian music being analysed conform fully to prevailing Iranian musical theory; for instance, Persian quartertones are flexible and they are not consistent in size between different modes. This variability is comparable to what Gedik and Bozkurt referred to as a gap between the theory and practice in Turkish music [3, 4]. The algorithms presented here are thus designed and modified to take these into account, and new methods are implemented in ways that are explained in Chapters 6 and 7. Even in current Western music, scales cannot be expressed solely in terms of twelve equal semitones; in practice, intonation in vocal and flexible-pitched instrumental playing frequently deviates from simple theory.

The author's MPhil thesis [5] presented work on Empirical Mode Decomposition and its application to the extraction of long-term rhythmic structures; it compared different pitch-tracking algorithms; and finally, although confined to only three of the modes, it demonstrated that automatic identification of the musical mode based on the tuning is feasible. The research presented in this thesis, however, is substantially new in the following specific respects:

- A tonic detection stage (not a feature of the MPhil research) is developed and performed prior to scale identification, and different classifiers and similarity measures are applied to the feature vectors (Chapter 7). The tonic detection system aligns the feature vectors, making the mode recognition methods independent of changes in tonality (Chapter 7). The test results show that, in order to avoid errors in tonic detection stage passing onto the mode recognition stage, a shift-and-compare process can be used prior to mode recognition, instead of a tonic detection stage.

- The signal processing algorithms presented in the MPhil research, which were originally applied to 3 modes, are here generalised to the 5 main tuning classes (see Chapter 2). They are extended to encompass a wider range of pitch systems (scalar interval pattern and modal tonic), instrumentations, and musical traditions closely related to those of Iran, such as Arabic, Turkish, and Greek musics (Chapters 6 and 7).

- Existing signal processing techniques, used on Western music [6, 7, 8, 9, 10], are modified, customised and extended for the analysis of *maqàm*-based musical systems, including Iranian music and musics of closely related traditions (see Chapters 6 and 7).

- The author's original work in MPhil research did not include silence and high-energy frame suppression. In order to remove frames with low and high energies, silence and high-energy frame suppression, which significantly improves the results, is now included in the analysis algorithm (Chapter 5). This pre-processing reduces the effect of noise and transients.

- Pitch deviations and harmonic deviations from multiple overtones are newly measured. Based on this, the inharmonicity factor of the santur (an important characteristic of the instrument) is calculated for the first time (Chapter 5). This is significant for the research because it defines the actual position of each note, leading to the conclusion that the inharmonicity factor is narrower (a smaller musical interval) than the deviations from 24-TET routinely encountered in Persian music.

- Parts of the code in the MATLAB environment, which was taken from [5], [6] and [11], are customised and substantially extended.

- Parameter optimisation is performed: the effects of different parameters, such as frequency range, down sampling, the amount of training data, tone resolution (12-TET, 24-TET, 48-TET and 53-TET), and the effect of using overlapping or non-overlapping frames are reported and discussed (Chapter 7).

- The use of symbolic data, newly included here, is tested and compared with spectrum, chroma and pitch histograms (features previously introduced in the MPhil) (Chapter 7).
- Geometric classifiers (dot-product and bit-masking) are applied and the results are compared with Manhattan distance and Auto-correlation (classifiers previously introduced in the MPhil research) (Chapters 6 and 7).
- The Gaussian Mixture Model (GMM), a generative method, is introduced as an extra classification scheme (Chapters 6 and 7).
- A substantially enlarged and reformed audio database (144 pieces, totalling 10829 seconds of material) is used; further samples are included and analysed. The MPhil database, which included samples in three *dastgàhs* (*esfehàn, chàhàrgah* and *shur*) is augmented with new samples in two other *dastgàhs* (*màhur* and *segàh*), so that it includes all the five main scales in Persian music Samples of instruments other than the santur, and of musics of closely related to the Persian, are newly incorporated into the database to validate the algorithms on a more diverse set of test samples, including pentatonic samples (an example of a non-Iranian scale) and samples representing other *maqàmic* musical traditions, including noisy old gramophone recordings (Chapter 4).

  The database now includes:

  1. Performances on the santur alone in various modes (*Shur, esfehàn, chàhàrgah, segàh, màhur* and pentatonic scale) and tonalities (E and A for *shur*; A for *esfehàn* and *chàhàrgah*, Fs for *segàh*, G for *màhur* and E for pentatonic scale in the main part of the database; and more varieties for other parts of the database).
  2. Performances of Iranian music on other instruments, in various modes and tonalities.
  3. Performances of music of other cultures, also in various modes and tonalities.
  4. Samples recorded during rehearsals and live performances by the author and his ensembles at the School of Oriental and African Studies (SOAS), University of London.
  5. Samples solicited from other musicians, and commercial recordings including CD, DVD, and old vinyl recordings (see Chapter 4).

1.7 Outline of the thesis

The aims and contributions of the thesis are introduced and explained in Chapter 1.

Chapter 2 explains essential aspects of Persian music and the santur. Sections 2.2 - 2.4 elaborate on the Persian intervals, the *dastgàh* system, and the nature of Persian composition. Section 2.5 discusses the social and cultural context of music. Section 2.6 describes the santur, a popular Iranian instrument, on which most of the database samples are played. Section 2.7 discusses the fundamental frequency and the overtones of the strings of the santur.

Chapter 3 reviews the technical literature on the application of music information retrieval to Western and *maqàm*-based musical traditions. Works representing the state of the art on identification of *dastgàh*, *maqàm*, key and *raag* are reviewed.

Chapter 4 introduces the database constructed to gauge the algorithms that are developed throughout the thesis. A dataset of audio musical files is created in order to examine the algorithms that are presented in the subsequent chapters, and which establishes a benchmark for assessing their performance.

Chapter 5 describes the pre-processing stage used to prepare data for the evaluations explained in Chapter 7.

Chapter 6 presents the research methodology used in the subsequent chapters (Section 6.2), and introduces the features and classifiers used (Sections 6.3 and 6.4 respectively)

Chapter 7 is devoted to experiments undertaken on several approaches to computational mode recognition. Alternative features and classifiers are examined and compared. The results are evaluated against a ground truth by expert musicians (Afsaneh Rasaei, a premier Iranian singer, and the author).

Chapter 8 presents conclusions, a summary of the main research achievements, and suggestions for future work stemming from or prompted by this research.

2 Background to Persian music and the santur

2.1 Persian or Iranian

At the outset, the terms “Persia”, “Iran”, “Persian”, “Parsi” and “Farsi” need to be clarified for English-speaking readers. Iran means the land of Aryans and has been the native name, used by Persians and other ethnicities in Iran ever since. Persia is the name by which Iran was known in the West from ancient times up to the year 1935, when the Iranian government requested all countries to call her by the native name “Iran”. The main language spoken in Iran is Persian, which is also called “Parsi” or “Farsi” [12]. There are Persian speakers outside Iran, whose musical traditions are different, regardless of speaking in the same language; and there are people, living inside Iran who speak in other Iranian languages, or in other languages. When the term “Persian Classical Music” is used, the urban art music, which is also known as Iranian traditional music is, referred to. This is different from Persian folk music, which comes from Fars or Khorasan provinces in Iran or music from Afghanistan and Tajikistan.

2.2 Persian intervals

There are various divergent views on Persian intervals, differing significantly in the extent to which an equally-tempered scheme is imposed upon subtle and variable intervals [5, 13, 14]. From the signal processing perspective, the main point is that in addition to Western intervals (reference to Western intervals throughout this thesis is to 12-TET) there are *flexible* quartertones in Persian music that lie between two neighbouring semitones (‘quartertone’ here is a literal translation of the Persian musical term *rob-e pardeh* (a quarter of a tone), where the first term is Arabic in origin and the second is Persian). Colonel A. N. Vaziri defined the Persian quartertones for the first time, where *sori* (𝄲) denotes a half sharp and *koron* (𝄳) denotes a half-flat interval [9]. The letters “q” and “s” are used for *sori* and *koron* respectively in this thesis[1]. Although this system is widely used by Iranian musicians, it does not describe the intervals accurately, as the fundamental frequency of these quartertone notes and even other Persian notes are not generally fixed. The size of a quartertone interval depends on the mode, the piece, and also the performer's mood.

Farhat proposed a theory of flexible notes [14], where a Persian wholetone is slightly larger but a semitone is significantly smaller than their counterparts in a 12-TET. There are two flexible

[1] Some use ASCII-fied versions of the symbols "p" for *koron* and ">" for *sori*.

intervals, between a semitone and a wholetone in size (the smaller is around 135 cents and the larger is around 160 cents), which he terms 'neutral tones'; and he states that two consecutive neutral tones form a minor third. There are exceptional cases, such as the *gushé* (melodic figure) of *muyé* in *chàhàrgah* (the tenth row in **Figure 1**, the second row of *chàhàrgah* scale), where the interval between the fourth and the minor sixth is not a minor third: it is a flexible quartertone larger than a minor third. There is also an interval greater than a whole tone but smaller than 3 semitones (approximately 270 cents), which Farhat calls the 'plus tone'. The letters "n", "N" and "P" are used for the small and large neutral tones and the plus tone respectively in this thesis. The intervals can be summarised as follows:

- Semitone: significantly smaller than its 12-TET equivalent (90 cents)
- Small neutral tone (three-quartertone) between 125 and 145 cents (135 cents average)
- Large neutral tone (three-quartertone) between 150 and 170 cents (160 cents average)
- Persian wholetone: slightly larger than its 12-TET equivalent (204 cents)
- Plus-tone (five-quartertone): greater than a wholetone but smaller than three semitones; between 260 and 280 cents (270 cents average).

Table 1 lists average intervals extracted from audio samples of vocal and instrumental music [14]. Comparing the intervals in **Table 1** with equally-tempered quartertones (24-TET), the small and large neutral tones (n and N) can be quantised to 3 quartertones, and the plus tone (P) to 5 quartertones. Thus, a quartertone falls either between 41.6– 48.3 cents (q1, a constituent part of n), or 50– 56.7 cents (q2, a constituent part of N), or 52– 56 cents (q3, a constituent part of P). A *fuzzy* membership function, centred at 50 cents with a deviation of 8.4 cents, encompasses all these intervals. Although the Persian intervals are flexible, they are still named in accordance with their nearest tempered counterpart. For instance, the intervals in the range 41.6 – 56.7 cents are all considered Persian quartertones (a word-by-word translation from *rob-e pardeh*).

Table 1 Persian intervals in cents

Interval	Average value	Equally-tempered counterpart
Semitone	90	100
Small neutral tone (n)	135	150
Large neutral tone (N)	160	150
Whole tone	204	200
Plus tone (P)	270	250

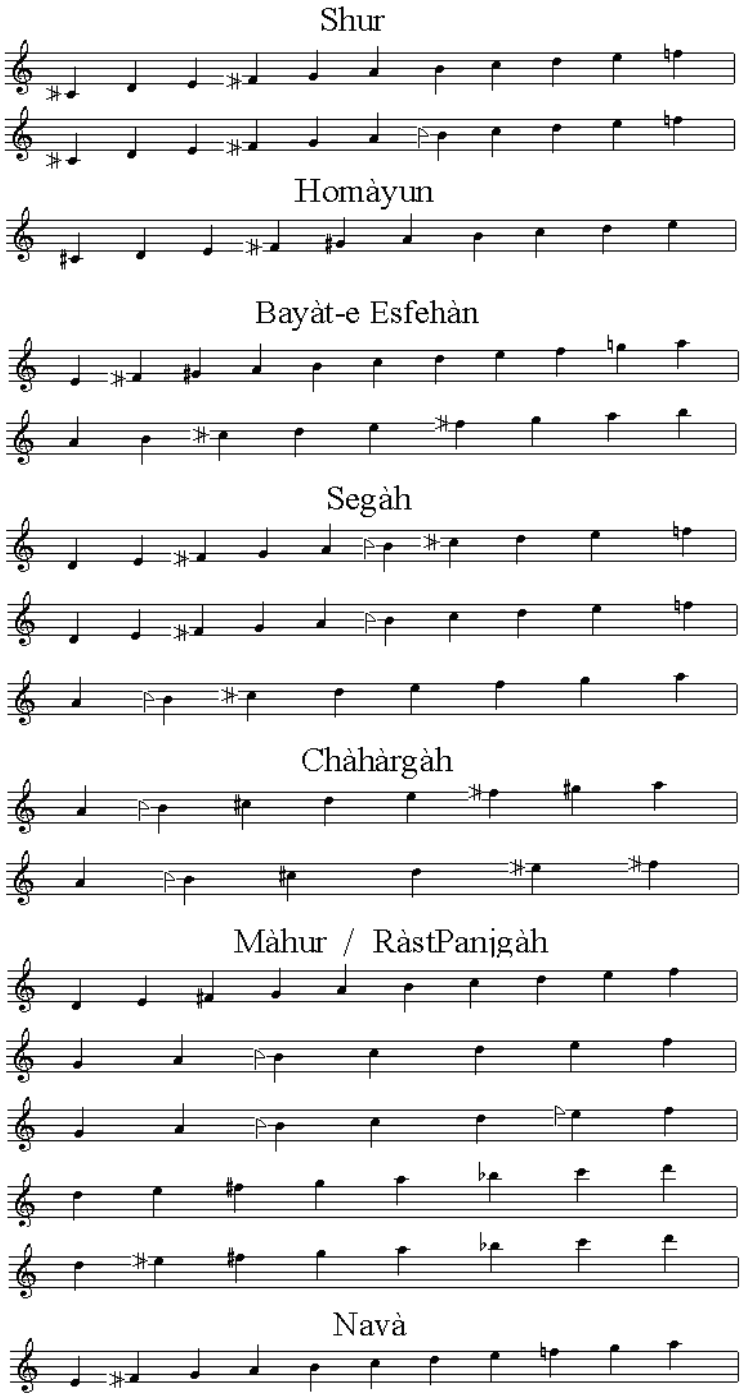


Figure 1 The scalar tuning system[2] of a 12-bridge santur

[2] Here the emphasis is on the intervallic tuning scheme; compass, about which more information can be found in [5, 14], is not represented. The scale notes from a fixed tonality (C) are shown in Table 3.

As discussed in section 2.2, in the standard Persian 24-quartertone system, as conceived of and understood by most Persian musicians, the "quartertones" are not, in practice, all of equal size; some notes are variable, being inflected up or down according to convention and individual players' preference. A total of thirteen principal notes are needed to play in all of the Persian modes: seven diatonic notes, three semitones, and three quartertones [5]. There are two types of tone inflections: fixed accidentals (comparable to the key signature of Western music), which prevail throughout a performance, and moving accidentals, which change during a performance. The thirteen principal Persian notes with which the Persian repertoire can be played on the santur are:

E - F - F𝄲 - F# - G - G# - A - B♭ - B - C - C𝄲 - C# - D

Two adjacent notes separated by a quartertone interval are used only very rarely, for trills and other ornaments. **Figure 2** shows all Persian scale intervals, expressed over a one-octave range, rationalised according to equal quartertone steps, where each vertical bar corresponds to a scale degree.

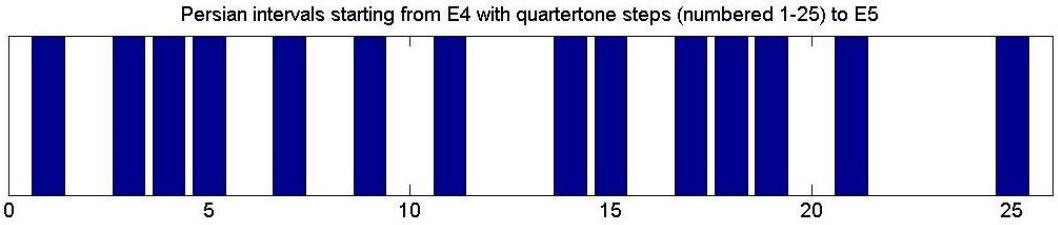


Figure 2 Persian scale intervals (E4 - E5)

The Persian reference pitch, which customarily depends on the comfortable range of a singer or accompanying instrument(s), is not necessarily standard concert pitch, although nowadays most musicians tune their instruments to A4 = 440 Hz using electronic tuners. It is noteworthy that although relative pitch is very important, the absolute pitch makes a difference in both performance practice and perception. For example, the limitations for voice and instrument(s) (especially the santur) bring different performance techniques and feelings when performing in *màhur* from F from when playing in *màhur* from C.[3] Arash Mohafez is of the opinion that transposed scales should be defined as different modes (*maqàms*). For instance, *àvàz-e delkash*, a transposed *shur* which is reached via *màhur*, has an ascending-descending melodic pattern, while *shur* itself, as a *dastgàh*, has only an ascending pattern. The *àvàz-e delkash* only modulates to two other modes,

[3] Jean During discusses the use in Kurdish practice in Iran of similar interval patterns at multiple pitches which give rise to different *dastgàh*s [15].

while the *dastgàh* of *shur* can modulate to five or more other modes. In general, an *àvàz* can create a maximum of three modal spaces (modulations), while a *dastgàh* can have six or more modal spaces [1].

2.3 Persian modes

Persian music is based on a modal system of seven main modes and their five derivatives that are collectively called the twelve *dastgàh*s [5, 14]. *Dastgàh* is the core of Persian music; usually the main modes are called *dastgàh*[4] and the derivative modes are called *àvàz*[5]. The precise classification of *dastgàh* is a matter of debate for Persian musicologists. There is a set of core pieces which individually are called *gushé*[6], which collectively constitute the *radif* (the core repertoire of Persian classical music). The *gushé*s are not precise pieces, but melodic models upon which improvisation is based [14]. As a result of a common cultural platform, a unique Perso-Turko-Arabian *maqàm* system evolved. However, with the passing of the centuries, the national traditions gradually diverged. In the nineteenth and twentieth centuries they were influenced in various ways by Western music and societies, and their cultures sought to redefine themselves in terms of a modern national identity. In the course of transitions over several centuries, the former twelve *maqàm* system in Iran gradually evolved into the twelve *dastgàh*s (seven *dastgàh* and five *àvàz*).

The relationship of the *dastgàh* and *maqàm* concepts in and around modern Iran is complex. Persian classical music (also referred to as Persian traditional music throughout the twentieth century) the Persian art music which was practised as the court music for several centuries, is based on the *dastgàh* system. The *maqàm*, although used for Persian classical music until about the seventeenth century, is characteristic of Persian, Kurdish, Azeri, and other ethnic musics from around Iran. Beyond Iran, *maqàm* now has a different role: it is used for both folk and classical musics[7]; and whereas the *maqàm* elsewhere is identified according to the mode, the Iraqi *maqàm*, which is close to the Persian *radif*, is identified as a repertoire of melodies.

In a *maqàm* performance, different pieces are played in a single mode, while the performance in a *dastgàh* comprises a certain sequence of modulations from an opening section in the main mode of a *dastgàh* (*daràmad*), through modulations to derivative modes (*àvàz*) and finally a return to the

[4] *Dastgàh* means device, organisation, or system in Persian.
[5] *Àvàz*, which means singing, refers to either non-metric singing or a derivative mode.
[6] There are two types of *gushé*: modal and melodic.
[7] There are different names and terms for different genres in Turkish and Arabic music.

starting mode. During the modulations, the modal tonic gradually moves upward. Modal tonic is also called the tonal centre, or centre of pitch gravity which are perceptual phenomena. Farhat calls it the *finalis*. The starting note is called *àqàz* and the ending note is called *ist* or *pàyàn* [14].

The seven main modes and their five derivatives are collectively called the twelve *dastgàh*s. They capture most of the structures in Persian music: *shur*, *abu'atà*, *bayàt-é tork*, *afshàri*, *dashti*, *homàyun*, *bayàt-é esfehàn*, *segàh*, *chàhàrgàh*, *màhur*, *ràst-panjgàh*, and *navà* [14, 16]. The *dastgàh* may be played in different tonalities on a santur.

Figure 2 shows all the intervals that are needed to play in the Persian modes on the santur. **Figure 32** (Chapter 5) shows the scale intervals for each *dastgàh*. For example, in *dastgàh-e shur* there are two fixed accidentals (Cs3 & Fs3) and one moving accidental (Bq4); *abu'atà*, *dashti*, *bayàt-e-tork* and *afshàri* are all derivatives of *shur*, and have the same tuning. Each *dastgàh* has a number of derivatives, called *àvàz*. Moving from a *dastgàh* to an *àvàz* is the usual way of modulation in Persian music. A primary modulation refers to a change in tuning during the modulation, while a secondary modulation occurs when the tuning remains the same. Most of the time, modulations involve a change in modal tonic and/or tuning. A *dastgàh* may be completely defined by all of the following: the tuning, the starting note (*àqàz*), the modal tonic (*shàhed*), and the rest (i.e. final) note (*ist*).

Some *gushés* are independent modes and play a different role when they are reached via a *dastgàh* as a modulation (see Section 2.2). For example, *àvàz-e delkash* of *màhur* is similar in scale to a *shur* set a fifth higher; and *mokhàlef-e chàhàrgah* is similar in scale to a *homàyun* set a sixth higher, but they differ in compass and their role in modulation. **Table 2** shows the relationship between the modal *gushé* and *dastgàh*.

2.4 Composition

Music in Iran is categorised, according to a scheme devised by Farhat, into urban, ethnic and pop [10]. Urban music, prevalently heard in the larger cities, includes both classical art music, stemming from court music, and pop music. Classical Persian music consists of free-rhythmic pieces (*àvàz*[8]) and rhythmic pieces, typically in 2/4, 4/4, or 6/8. Ethnic music, which is in an Iranian form of *maqàm*, is that of the various ethnic groups living in towns, villages, deserts, and in the mountains. In addition to pieces in free and simple rhythms (2/4, 4/4, or 6/8), irregular rhythms

[8] *Àvàz* does not mean a derivative mode here.

such as 5/8 and 7/8 are more often encountered in ethnic music. Classical Persian music uses more ornaments and free rhythms than ethnic music, which Farhat characterises as having simple melodies and direct rhythms [14]. The intervals and modes are similar, with the exception of the semitone scale divisions on older ethnic instruments, such as *tanbur* and *shemshàl*, which are equally tempered. Iranian pop music, which has dominated the music scene in Iran since the mid-twentieth century, as elsewhere in the world [10], draws on either or both of the classical and ethnic traditions; it tends to simplify them and to reflect influences from other cultures, notably Western pop music.

The structure of twentieth century Persian classical art music consists of three instrumental forms and one vocal form. The instrumental forms are *pishdaràmad*, *chàhàrmezrab*, and *reng* [14]:

- *Pishdaràmad*, a rhythmic piece played before *daràmad* (the opening section of a *dastgàh*, a prelude), invented by Darvish Khàn[9] (a master of the *tar*) was intended as a prelude to the *daràmad*. It may be in duple, triple, or quadruple time, and it draws its melody from some of the important *gushés* of the piece (**Table 2**). Another metric form with a similar function, *peshrev*, existed in the past, but it was obsolete by the time of Darvish Khàn.
- *Chàhàrmezrab* is a solo piece with a fast tempo, usually based on the melody immediately preceding it. It is usually played in 6/8 or 2/4.
- The third instrumental form is the *reng*, a simple dance piece that is usually played at the conclusion of the *dastgàh*.
- There are two vocal forms: *àvàz* and *tasnif*. *Àvàz* is a free-rhythmic vocal improvisation; and Tasnif, which is usually placed directly before a *reng*, is a composed song.

The process of creative performance, called *bedàhe navàzi* (tr. improvisation), which is at the heart of Persian music, is different from improvisation in Western music, inasmuch as it involves both composition and new ways of rendering classical pieces; thus, there is no distinction between the role of the performer and the composer [17]. It should be noted that musicians do not take a random sequence of *gushé*s in their performance; a performance is usually centred on a set of important *gushé*s,[10] whose order is conventionally accepted. The *daràmad* (opening section) comes at or near the beginning of the performance, and the following *gushé*s are organised according to a gradually ascending pitch scheme, until a *forud* (cadence) leads a return the original mode. Usually, metered *gushé*s are played between non-metered *gushé*s [18].

[9] Born in Tehran in 1872.

[10] Important *gushés* are called *Shahgushé*, while *tekké* and *naghmé* are less important *gushés* [18].

Table 2 Classification of Persian modes

Dastgàh		More *gushé*
7 main modes	5 derivative modes	
shur	*abu'atà*	
	bayàt-é tork	*shekasté*
	afshàri	
	dashti	
		kord bayàt
homàyun	*bayàt-é esfehàn*	*ushshaq (an abu'atà , when called through bayàt-é esfehàn), delkash (a shur, when called through bayàt-é esfehàn)*
		bidàd
		shushtari
		bakhtiàry
segàh		*mokhàlef-é segàh (similar to bayàt-é esfehàn, when called through segàh)*
chàhàrgàh		*zàbol*
		mokhàlef-é chàhàrgàh (a homàyun, when called through chàhàrgàh)
		muyé
màhur		*shekasté*
		delkash (a shur, when called through màhur)
		dàd
		eràq
		ràk
ràst-panjgàh		
navà		

Table 3 presents the scale tuning for Persian modes from the tonality of C. Symbols '#', 'f', 's' and 'q' denote *sharp*, *flat*, *sori* and *koron* respectively. 'w', 's', 'n', 'N' and 'p' denote a wholetone, a semitone, small neutral tone, large neutral tone and plus tone[11]. The intervals are flexible. For example the last two intervals in *bayàt-e esfehàn* which were (n, p) in the 1900s, due to influences from Western music, are gradually moving towards (s, ws). The same can be said for *homàyun* and some other *dastgàhs*.

Nowadays these forms and structures are not strictly followed. The texture of Persian ensemble music is heterophonic, meaning that the members of the ensemble play the melodic scheme simultaneously in different ways, characterised a high degree of improvisation and ornamentation.

[11] A neutral tone is around three quartertones, and a plus tone is around five quartertones.

The ornaments consist of *dorràb* (fast playing of a note for 3 times), *riz* (fast playing of a note several times as long as its duration), trills (fast playing of a note and a note above or below it) and *eshàré* (moving to a higher or lower note).

Table 3 The tuning system from the tonality of C and the relative intervals

Dastgàh	**Intervals in an octave**	**Relative intervals**
Shur	C-Dq-Ef-F-G-Af-Bf-C C-Dq-Ef-F-Gq-Af-Bf-C	n, N, w, w, s, w, w n, N, w, N, n, w, w
Homàyun	C-Ds-Eq-Fq-G-Dq-C	P, n, w, N, n, w, n
Bayàt-e esfehàn	C-D-Ef-F-G-Af-B-C C-D-Ef-F-G-Af-Bf-C	w, s, w, w, s, w, w w, s, w, w, s, n, p
Segàh	C-Dq-Eq-F-Gf-Af-Bf-C C-Dq-Eq-F-Gq-Af-Bf-C	n, w ,N , n, w, w, N n, w ,N , w, n, w, s
Chàhàrgàh	C-Dq-E-F-G-Aq-B#-C C-Dq-E-F-Gq-Aq	N, p, s, w, n, p, s N, p, s, n, w
Màhur / Ràst-panjgàh	C-D-E-F-G-A-B-C C-D-Eq-F-G-A-Bf-C C-D-Eq-F-G-Aq-Bf-C	w, w, s, w, w, s, w w, N, n, w, w, s, w w, N, n, w, N, n, w
Navà	C-D-Ef-F-G-Af-Bf-C	W, s, w, w, n, N, w

2.5 Social and cultural context of music

Music is not just a series of notes or a signal and has to be studied within its cultural context. The study of music in culture or music as culture is a matter of discussion in ethnomusicology [19]. This is especially the case in the analysis of non-Western musical systems where practice is liable to deviate from the theories [3]. The social context and the psychological aspect of music are important and especially, when dealing with emotions that a piece conveys, the same melody could be interpreted in different ways by people of different ethnicities and cultures. For example, the same Kurdish melody *Wara qorbàn* is considered to be a classical piece, while it is a less serious piece in Turkish and Greek music, where it is called *Gelemen ben* and *Ti ta thélis ta leftà* respectively.

Music and poetry

Many of the Persian pieces are related to lyrics or may even be composed to accompany certain poems. For instance *masnavi* is a metric musical form that is related to a lyrical structure. Thus poetry contributes to the meaning behind a performed music and its perception.

Effects of language and religion on music

Music is a way of expression and is a sign of a society's culture, traditions, and way of life. The nature of music does not necessarily depend on language, religion or even national borders. For instance, Kurdish music in northern Iraq is different from Kurdish music in Iran and Turkey; Persian musics in Iran, Afghanistan and Tajikistan are also different, even though people speak in the same language (Persian); likewise Azerbaijan and Armenia have different religions and languages, and they are not in a good political relationship, but their musics are very close. A similar statement could be made for Greek and Turkish music. However, language, religion, political and geographical borders influence music; the extent of this would be worth examination, but is beyond the scope of the present research.

Musics of the minorities and music in diaspora

Minority groups in a country make use of music to prove themselves and to influence the mass and their media. For instance, Jews, Armenians and recently the Kurds within Iran have had a substantial role in Iranian music, regardless of their small population, compared to the other ethnicities. On the other hand, migration to another country creates music in diaspora. Diasporic music sometimes adheres closely to the original traditions and sometimes fuses with the music of the host community, importing new elements and establishing new trends. These interactions give rise to new potentials for creativity, and can not only form a bridge between the original community and the host but may also, in due course, exert a reciprocal effect upon the indigenous tradition.

2.6 The santur and its physical properties

The santur is the most popular Iranian instrument and most of the samples of our database are played on this instrument. The *tar* (long-necked lute), *kamàncheh* (spike fiddle), *ney* (end-blown flute), *tonbak* and *daf* (frame drum) are the other popular instruments in Iran. Iranian music is also played on Western instruments, such as the piano and the violin.

The santur originated in Iran and is played in various countries. It is a direct ancestor of piano, and in English terminology it is referred to as a hammered dulcimer.

Mehdi Setayeshgar mentions the santur as an ancient instrument, and traces the instrument back to Assyrian and Babylonian inscriptions (669 BC) [17]. Although in some ancient books its invention was associated with Farabi (870-950 AD), Masoudi includes the santur in a list of instruments of the Sassanid era (224-651 AD), indicating that it existed before Farabi [21]. Setayeshgar is of the opinion that santur is an Aramaic word, and that Jews have also used the instrument [20]. The name santur occurs in the Torah (Daniel 3:5) [16]. The santur was taken to Europe in the middle ages, where it has been known as the dulcimer in English literature since the fifteenth century. The instrument appears in *The Oxford Companion to Music*, with the first known reference being in 1660 AD [23].

Characteristics of the santur

The santur (Figure 3-a), is a shallow trapezoidal box zither, a chordophone in the Hornbostel-Sachs classification of musical instruments [24], with two rows of bridges [18]. It is played with a pair of slender hammer sticks. The sound is produced by the vibration of strings that are held at both ends of the shallow box, and the resonance of the body shapes and amplifies it. The instrument has a sustained sound which can be heard for a few seconds after a note is played.

The hammer sticks (Figure 3-b) are held between the index and middle fingers, in order to hit the strings. When the strings are struck, a small deflection creates a loud sound. The striking ends of the sticks are usually coated by felt, and the repeated impact over time compresses the felt, making it thinner and harder.

(a)

(b)  (c) 

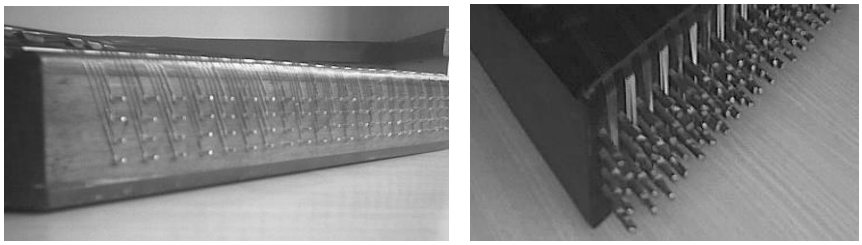
Figure 3 a) Santur; b) Sticks; (*mezràb*) c) Tuning key

Four closely adjacent parallel strings are sounded together for each note. The strings are supported on the left and right edges of the soundboard. They are stretched across the soundboard, between the string holders (hitch pins; Figure 4-a) and the tuning pegs (wrest pins; Figure 4-b, Figure 5-b), and they cross and bear upon on a bridge situated between these two ends (Figure 5-a). The tension of the strings is adjusted by rotating the tuning pegs using a tuning key (Figure 3-c). The key is also used as a hammer to hit the tuning pegs to adjust their seating. Unlike some Western kinds of hammered dulcimer, each of the santur's bridges is individually and independently movable, and thus the pitch can continuously be changed by several tones. The parallel sides of a 9-bridge[12] santur made by Salari[13], on which the measurements presented here were taken are 90.5cm and 34.8cm long[14]. The diagonal sides (left and right) are almost equal in length: 38.9cm and 39.0cm. The top (soundboard) and the back plates (upper and lower boards) are 6.4cm apart. Their thicknesses are 5.5mm and 8.0mm respectively. The lengths of the four strings of a note are not exactly the same; they are between 36.8 and 37.8cm for F5-F6, and between 84.8cm and 86.25cm for E3.

[12] Modern Iranian santurs have 9 bridges, and 11 and 12-bridge santurs are also in common use [5].

[13] *Daryoush Salari* is a prominent santur maker in Iran, noted for his adoption of innovative production techniques.

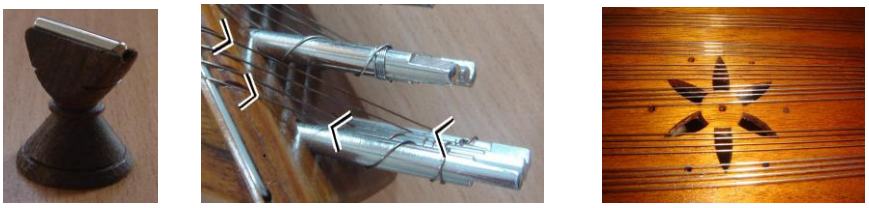
[14] These values can be compared to dimensions of an 11-bridge santur presented in [5].



(a) (b)

Figure 4 a) String holders; b) Tuning pegs

The diameters of the strings are between 0.35-0.36mm depending on the age and the tension of the strings. Each bridge is 2.3cm high, and is surmounted a horizontal metal rod of 2.5mm diameter (Figure 4-a). The tuning pegs are arranged in four rows, 2-6cm below the right edge of the top plate (Figure 4-b).



(a) (b) (c)

Figure 5 a) Bridge with metal rod as string-bearing surface;
b) Strings between top plate (soundboard) edge and tuning pins;
c) Soundhole

Four rails (bars) of wood, along with sound posts (rigid wooden props), support the top plate a fixed distance above the back plate. They bear the downward force exerted by the bridges, and the end-to-end compressing force exerted by the strings. The resonant body of the santur is hollow, and the sound posts keep the instrument from collapsing. Two sound holes (Fig. 4-c), in this case 5cm in diameter, influence the timbre and serve to enhance the sound quality.

2.7 The fundamental frequencies and their overtones

This section explains the fundamental frequencies of the notes of the santur; the pitch and harmonic deviations, and the inharmonicity factor are discussed in Chapter 6. The tone range of the 9-bridge santur is E3q (160.1 Hz)-F6 (1396.9 Hz), which is usually extended downwards by tuning the first (lowest-pitched) course of strings down to C3 (130.8 Hz) rather than E3q. The

santur has three tone areas (pitch registers): the strings of the lower (first octave), known as the 'yellow' notes (z*ard*[15]) because they are made of brass (the relatively high density of which results in strings of suitably high mass), extend from the right to the left of the body of the instrument, with their bridges situated to the right side of the soundboard, resulting in the longest available vibrating length. A second set of strings, known as 'white' notes (*sefid*[16]) because they are made of steel, serves the middle and upper registers (second and third octaves); they are divided by a row of bridges situated to the left of centre, essentially in the ratio 1:2, such that the left part of each string sounds essentially an octave higher than its counterpart to the right. In practice the 1:2 ratio is departed from to inflect pitches by a small interval – mainly a quartertone or semitone. The highest (treble) register (the third octave) is called *sefid-é-posht-é-kharak* (tr. behind-the-bridge white notes). In comparison with the 9-bridge santur, the range of the 11-bridge santur starts a third lower, from C3 (130.8 Hz) to F6 (1396.9 Hz), although the lowest string is usually tuned to C#3 (138.6 Hz) or A2 (110 Hz).

The fundamental frequency f_0 and K^{th} harmonic of a string are a function of its length l, the tension of the strings (pulling force) T, string material constant μ and harmonic[17] K, as given by Equation (1) [2] multiplied by K for the Kth harmonic:

$$f_0 = \frac{1}{2l}\sqrt{\frac{T}{m}} \qquad (1)$$

As discussed above, the string lengths of a typical 9-bridge santur the range from about 36.75cm to 86.25cm. For an 11-bridge santur, the string lengths are typically between 39.0cm and 90.2cm. In practice it is more convenient to play each mode on a particular type of santur in certain tonalities; for example, it is easier to play *dastgàh e shur* on an 11-bridge santur from E or A, rather than from F, whereas on a 9-bridge santur it is more convenient to play it from G or C.

A convenient tuning system for 11 or 12-bridge santurs is shown in **Figure 1**. As a quick procedure, the tuning can be altered by just moving the bridges laterally.

Supposing an equally tempered scale of 24 quartertones, f_0^2 the frequency of a note located at a distance of d quartertones from a note with a fundamental frequency of f_0^1 can be calculated:

[15] *Zard* (tr. Yellow), referring to the brass strings.
[16] *Sefid* (tr. White), referring to stainless steel strings.
[17] K=1 for fundamental frequency; K=2, 3, … for harmonics.

$$f_0^2 = f_0^1 \times 2^{d/24} \qquad (2)$$

Using this formula, **Table 4** shows the f_0 of santur notes, calculated with reference to A4 = 440 Hz concert pitch.

Pitch class and pitch height

Shepard (1964) states that human judgement of pitch depends on tonality and height [21], which are called pitch class and pitch height respectively [25]. The pitch class refers to a note's name, while pitch height refers to the absolute frequency. Pitch class and pitch height can be represented by putting the notes on a helix, where rotation in horizontal plane shows the change in pitch class and vertical distance corresponds to a change in pitch height, as seen in **Figure 6**, which is reproduced from Noland's PhD thesis [9, 25].

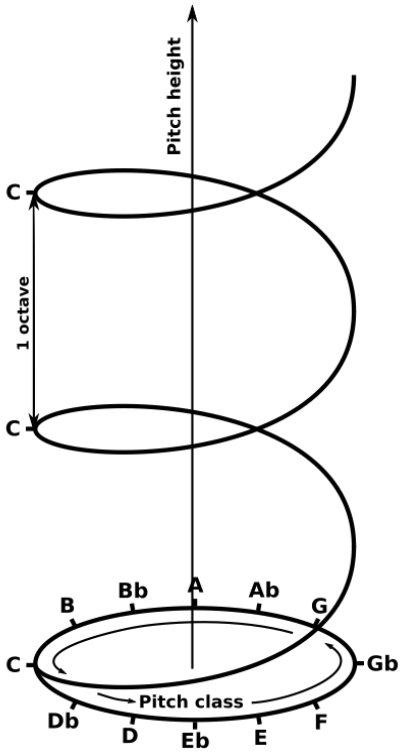


Figure 6 Shepard's helical pitch model [25] reproduced from Noland's PhD thesis [9]

Tone profile

A tone profile is a 12-element vector, where each element corresponds to one of the 12 pitch classes in the Western tempered scale, and its amplitude shows its importance in a given key. This

can be transposed and in general there will be one tone profile for the minor scale and one tone profile for the major scale, assuming 12-TET [9].

Table 4 Fundamental frequencies of a 12-bridge santur[18]

Note	Fundamental Frequency (Hz)	Note	Fundamental Frequency (Hz)
C#3	138.6	Bq4	479.8
D3	146.8	B4	493.9
E3	164.8	C5	523.2
Fs3	179.7	Cs5	538.6
G3	196	C#5	554.4
A3	220	D5	587.3
Bq3	239.9	E5	659.3
C4	261.6	F5	698.5
C#4	277.2	Fs5	718.9
D4	293.7	G5	784
E4	329.6	A5	880
F4	349.2	B5	987.8
Fs4	359.5	C6	1046.5
F#4	370	D6	1174.6
G4	392	E6	1318.5
G#4	415.3	F6	1396.9
A4	440		

Summary

This chapter started by clarifying the question of Iranian music in relation to Persian music, and continued by providing background material and an overview of Persian music, including intervals, modes, and the compositional framework. Subsequently, the santur (instrument) and its physical properties were described. Chapters 3 and 6 review the existing literature on audio signal processing and music information retrieval and the common features and classifiers in signal processing.

[18] Note that here *sori* and *koron* are taken exactly as a quartertone over or below the note, while in practice their positions depend on the mode, the piece and the performer's desire.

3 Literature review: the application of musical signal processing techniques to Western and non-Western musics

This chapter reviews works concerning the application of musical signal processing techniques to Western and non-Western musics, with an emphasis upon the musical traditions of the Mediterranean and the Near East.[19] This section starts with definitions of the fields of DSP and MIR. Section 3.2 reviews works on computational analysis of chord and key in Western music, and intervals and modes in *maqàm*-based musics. Sections 3.3 and 3.4 review works on world music and Persian music respectively. Section 3.5 discusses commercially available audio search engines.

3.1.1 Digital Signal Processing (DSP)

Digital Signal Processing (DSP), as distinct from analogue signal processing, refers to the manipulation of signals in time, frequency or another domain, where the variables are discrete. Any signal that is to be processed by a computer needs to be converted to digital form, using an analogue-to-digital converter. This branch of science deals in general with audio, image, and seismic data processing, and other forms of digital data. Here the focus is on musical signal processing, which, together with speech signal processing, is one of two branches of the field of audio signal processing.

3.1.2 Analogue to digital converters

Analogue to digital converters are devices which take samples of an analogue signal, usually at regular times (uniform sampling), and convert them to discrete numbers. It is also possible to have non-uniform sampling, where the samples are not taken at equally spaced times.

3.1.3 Music Information Retrieval (MIR)

Engineers and scientists have been working on automatic analysis of music for around half a century. Michael Kassler used the term Music Information Retrieval (MIR) for the first time in 1966, during the course of an extensive encoding project at Princeton University [26] in which he created a new programming language for music and named it MIR [27]. In the 1970s scientists started to develop pitch trackers. However, it was not until the turn of the millennium that, as a

[19] We are mainly concerned here with *maqàm*-based musical traditions.

result of the progress in the Electronic Engineering and Computer science fields and the emergence of cheap and powerful personal computers, MIR was consolidated as a new discipline. The International Society for Music Information Retrieval (ISMIR) is a forum for research on music-related data, and it runs the ISMIR Conference, which has been the most prominent annual conference in this field since 1999 [28].

MIR is a multi-disciplinary branch of science that involves audio content analysis and description, using Digital Signal Processing (DSP), artificial intelligence (AI) or other computational methods. MIR involves designing and making tools for understanding and organising music and searching large musical collections [26]. This field encompasses musicology, cognitive psychology, linguistics, and library and computer sciences. George Tzanetakis defines MIR as the interdisciplinary science of retrieving information from music, connected to various fields such as signal processing, psychology, machine learning, information science, and human-computer interaction [29]. MIR is also referred to as 'music informatics'. 'Music technology', a more general term, includes music analysis and fields such as music synthesis, musical interaction, performance modelling, and music cognition, which are beyond the scope of MIR.

Nowadays ethnomusicologists are able to use state-of-the-art signal processing tools, and computer-aided ethnomusicology[20] has become a common trend [26]. Before websites such as YouTube were established, music was not as accessible as it is now; people were not able to hear musics of other cultures to the present large extent; and music recommendation via keywords was not as easy as it is now. The next step is to allow access to the content of audio musical files, in order to enable versatile analysis and content-based music recommendation.

This thesis develops methodology whereby current MIR methods can be customised and adapted for Iranian music and non-Western music in general and argues that a true understanding of ethnic music is necessary in order to apply the necessary adjustments.

[20] Musicology, according to a traditional definition, concerns the analysis of scores and other Western music documents; ethnomusicology concerns the study of orally transmitted traditions and involves fieldwork and ethnography [29]. In a broader sense, musicology is not confined to Western and documented musics, and encompasses all musics, with sub-disciplines such as cultural, empirical, analytical, comparative, and ethnomusicological studies [23].

3.2 Chord, key and melody recognition for Western tonal music

3.2.1 Chord identification

Chris Harte presented an algorithm for automatic chord[21] identification, using a quantised chromagram[22] [6], calculated, based on the Q-transform [7]. In his approach, a 36-bin[23] chromagram is applied to locate the boundaries of semitones accurately[24]. Then every three adjacent bins are merged and a 12-bin semitone-quantised chromagram is produced. This is compared to a set of chord templates to construct a sequence of chord estimates. This approach deals with four types of triadic chords: major, minor, augmented and diminished. Thus there will be four times 12 semitones, that is 48 chord templates in total. The chromagram of the test sample is masked and multiplied by all these 48 templates and the one that yields the highest energy represents the closest match. His results are in the range of 53.9% to 77.9% for different sets of songs [6].

3.2.2 Chord segmentation and recognition

Sheh and Ellis propose an algorithm for chord segmentation and recognition, using the Hidden Markov Models (HMM). They use the Pitch Class Profiles (PCP) (chroma)[25] as the feature and estimate the chord sequence using the HMM. Their model is trained using the Expectation Maximisation (EM) algorithm. They compare PCP with cepstral coefficients and conclude that PCP far outperforms the cepstral coefficients. Their algorithm was tested on 20 samples of Beatles songs and achieved a frame-level accuracy of 75% [30].

3.2.3 Key recognition

Key recognition methods can be classified into three categories, based on theory, cognitive studies and statistics of real music. The first involves templates with ones at diatonic scale notes and zeros elsewhere. The second shows a weighted version of the first category templates, based on the importance of the notes. The third category shows a data driven weighted version of the diatonic interval.

Izmirli multiplies the data driven templates point-by-point by a diatonic scale, so that non-diatonic

[21] A chord is the simultaneous sounding of two or more notes [31].
[22] The chroma is explained in detail in Chapter 5.
[23] The bins merge the elements and make an average over the number of merged elements.
[24] The resolution is 1/3 of a semitone.
[25] There is a slight difference between the PCP and the chroma (HPCP), where the chroma includes the harmonic content as well, mapped and augmented to the pitch classes.

notes are totally removed from the templates [32].

Noland presents an algorithm for computational tonality estimation, where the predominant key is estimated, based on signal processing and Hidden Markov Models [9, 10]. For this purpose, she conceives of 'tonality' as a more general term than a 'key'[26], that may involve several keys being played at the same time.

She observes that scientists had used statistical signal processing to include the time progression and to get flexibility on prescribed harmonic relationships. Among these the Hidden Markov Model (HMM) helps to understand the meaning and effect of the parameters, as the different states in HMM can model the progress in time. She implements a 24-state HMM, where each state corresponds to one of the 12 Major and 12 minor keys. In her approach the Hidden Markov Models (HMM) express relationships between chords and keys, as the probability of emitting observable chords from a hidden sequence. The hidden states in HMM denote the keys, and the observations show the chords and their relationships to each key. Each observation corresponds to a chord or a pair of chords sequence [9].

Noland finds though different tests that the initialisation of the transition and observation probabilities in unsupervised HMM are very important – much more important than extensive training. Initialisation by a listener's judgment was proven to work better than binary or random initialisation [9]. She reports that down sampling[27] by a factor of 16 improves the results slightly, and concludes that high frequency components were unnecessary and harmful. This also reduces the calculation cost. When the system was tested on 110 songs by the Beatles, 91% of the samples were correctly classified [38]. She also reports that using a threshold for constant-Q transform reduces the calculation cost by 50% [9].

Noland compared the HMM method of key estimation with the tone profile correlation method, and concluded that the difference in performance was dependent on different datasets: HMM performed better with datasets with changing timbre and style. Thus HMM is better able to generalise music than the tone profile correlation method. Noland identifies the weaknesses of the HMM method, as follows:

1. One weak point of HMM is that long-term melodic structure cannot be modelled; according to

[26] The key of a piece usually refers to the tonic and chord. The key of a movement may change through a process, which is called modulation, but at the end, there is usually a return to the original key.
[27] Down sampling is the process of reducing the sampling rate of a signal.

the Markov assumption, every state is dependent only on the previous state, whereas more than one state may lead to a chord.

2. The likelihood of staying in one state decreases exponentially, which is not necessarily so for music and is not consistent with tonal progression in particular, for instance, the circumstance where a musician is playing tremolo for a while.

She concludes that there are other variants of HMM with more than one state which could be used to solve these issues, and that it would be possible to consider a hierarchical tonal model that represents different levels of tonal information.

Finding the key boundaries

Elaine Chew describes finding the key boundaries or the points of modulation as an extremely hard task [34]. She uses spiral arrays and music knowledge to find the points of modulation. Spiral array is a geometric model for tonality, a spatial arrangement of pitch classes, chords and keys. It is an arrangement of pitch classes on a lattice, where the pitch classes are positioned round a spiral. **Figure 7** shows a spiral array. On the vertical axis adjacent pitches are an octave and a 3rd apart. It is 4h, where h is an interval of a perfect 5th.

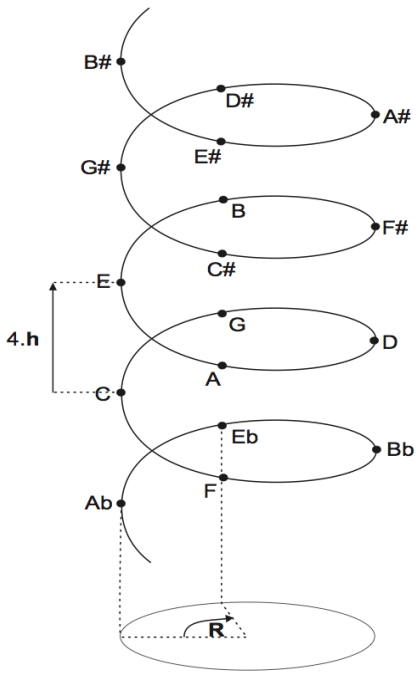


Figure 7 Chew's Spiral Array, where neighbouring pitches are separated by pitch height h and rotation R [34]. Figure reproduced from Noland's PhD thesis [9]

Chuan and Chew propose a key-finding method based on a fuzzy analysis spiral array centre of effect generator algorithm (FACEG) and three key determination policies: nearest neighbour (NN), relative distance (RD) and average distance (AD) [35]. They made three extensions to the basic system: modified Signal Array (mSA), fundamental frequency identification (F0) and post-weight balancing (PWB). They used audio synthesised by MIDI and acoustic piano as their excerpts. They report that the fuzzy analysis system with mSA outperforms the other extensions. In their test on a piece by Chopin, the pitch tracker operates better during the first 8 seconds, where few notes are being played. The system that employs mSA and includes audio frequency features in the weights works better after 8 seconds.

Papadopoulos and Tzanetakis propose a modelling for chord and key structure, using the Markov Logic Networks (MLN), a combination of Markov networks and first order logic. They use the chroma as the feature and obtain a global key in addition to achieving similar results to a state of the art HMM. Using the prior information about global key makes a small, but statistically significant improvement in chord estimation. The join chord and key exact estimation is 82.27%, compared to 75.35% for a chroma-based template matching method. They left local key estimation for future work [36].

3.2.4 A melodic similarity measure

Emilia Gomez describes the modal tonic as one of the important descriptors of an audio musical signal [37]. She states that pitch histograms are appropriate for monophonic music, while Harmonic Pitch Class Profiles (HPCP), also known as the chroma are appropriate for polyphony music. The HPCP vectors are ring-shifted to the position of the maximum value. The resulting vector is called the Transposed HPCP (THPCP) [38]. The templates used in her method are normalised to remove the influence of the dynamics and the instrument timbre that is reflected in the spectrum [39]. This method is not dependent on individual notes and operates on the overall tuning. It works on polyphonic music.

3.3 Mode recognition in world music

3.3.1 Work on Indian music: raag recognition

Parag Chordia and Alex Rae present a *raag* recognition system based on pitch class and pitch class dyad distributions. A *raag* is a collection of melodic gestures, and the way to develop them. A melodic gesture is a sequence of notes, often inflected with microtones, which can be used as the basic melody composition blocks. The scale, the most stressed note (*vadi*) and the second most stressed note (*samvadi*), which is usually the 4^{th} or 5^{th} scale degree, characterise a *raag*. Out of about 100 *raags*, of which some 50 are popular, Chordia and Rae included performances in 31 *raag*s in their analysis [40].

They used Pitch Class Distribution (PCD) and Pitch Class Distribution Dyad (PCDD) as their features. Pitch Class Distributions are simply pitch histograms of the framed signal, i.e. the number of pitch instances divided by the number of notes. Pitch Class Distribution Dyads, which are also called bi-grams or dyads are pitch histograms, made at onset times, considered in groups of two. They show the transition from one note to another. As only twelve 12-TET semitones are considered throughout their research, Indian intervals and inflexions are not well recognised. As they are 12 pitch classes, there will be 144 dyads. **Figure 8** shows PCD for two of the main modes in Indian music.

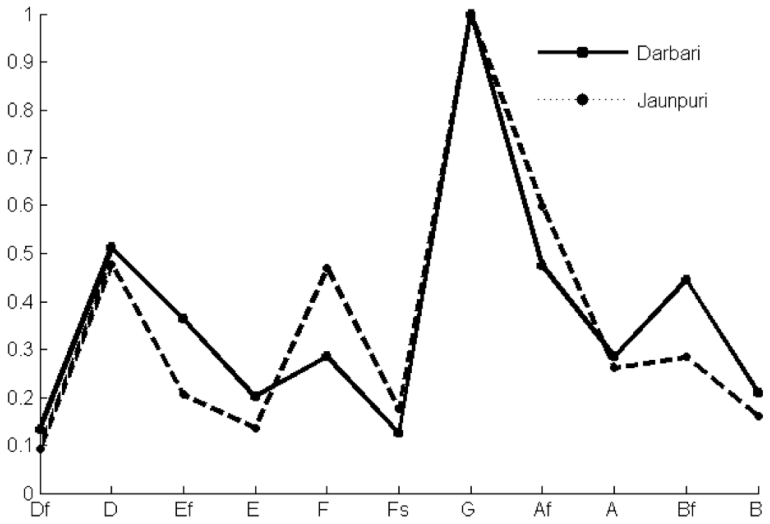


Figure 8 Pitch-class distribution for *darbari* and *jaunpuri* raags. Reproduced from [41]

Their classification methods were based on Support Vector Machines (SVM), Maximum a Posteriori (MAP) based on a Multivariate Likelihood Model (MVN), and Random Forests.

They achieve an accuracy of 99% with a ten-fold cross validation (CV), where the training data was used in tests as well, and an accuracy of 75% for the unseen case, where the test samples were different form the training samples.

3.3.2 Work on Turkish music: tonic detection and makam identification

Gedik and Bozkurt reviewed the use of pitch histograms for Western and non-Western music, and have presented a music information retrieval system for *makam* recognition in Turkish music [3, 4]. Their approach involves automatic detection of the tonic of the signal. Then the frequency content is shifted to match the templates in a relative pitch system basis, and pitch histograms are constructed. They consider reduction of the dimension of feature vector as a problem and suggest considering a higher dimension pitch space for Turkish music (compared to 12 semitones in Western music). They address the following problems:

1. The discrete 12 semitone scale is not applicable to Turkish music and a continuous scale is needed for Turkish music with its microtonal characteristics
2. As there is a gap between theory and practice in the Middle Eastern music, it is better to rely more on the data rather than on the music theory.

They used the YIN algorithm for pitch tracking [42] in order to construct the pitch histograms for the challenging tasks of *makam*[28] recognition and tonic detection, by implementing a high-dimensional pitch histogram of 53 commas per octave for Turkish music. First, the tonality is estimated with shifting and comparing the normalised pitch histograms with templates[29]. The mode is recognised subsequently, by comparing the pitch histograms with the templates of each *makam*. They compare different distance measures such as City Block (Manhattan, Minkowski's distance of order one), cross-correlation and Bhatacharyya and conclude that City Block and intersection work the best for Turkish music. They tested their method over 172 files in 9 classes and due to a lack of

[28] *Maqàm* is rendered as "*makam*" in Turkish.
[29] Maximum match occurs when the shifted elements are aligned with the tonic of the template.

enough samples, they did a leave one-out cross validation for both training and test phases.

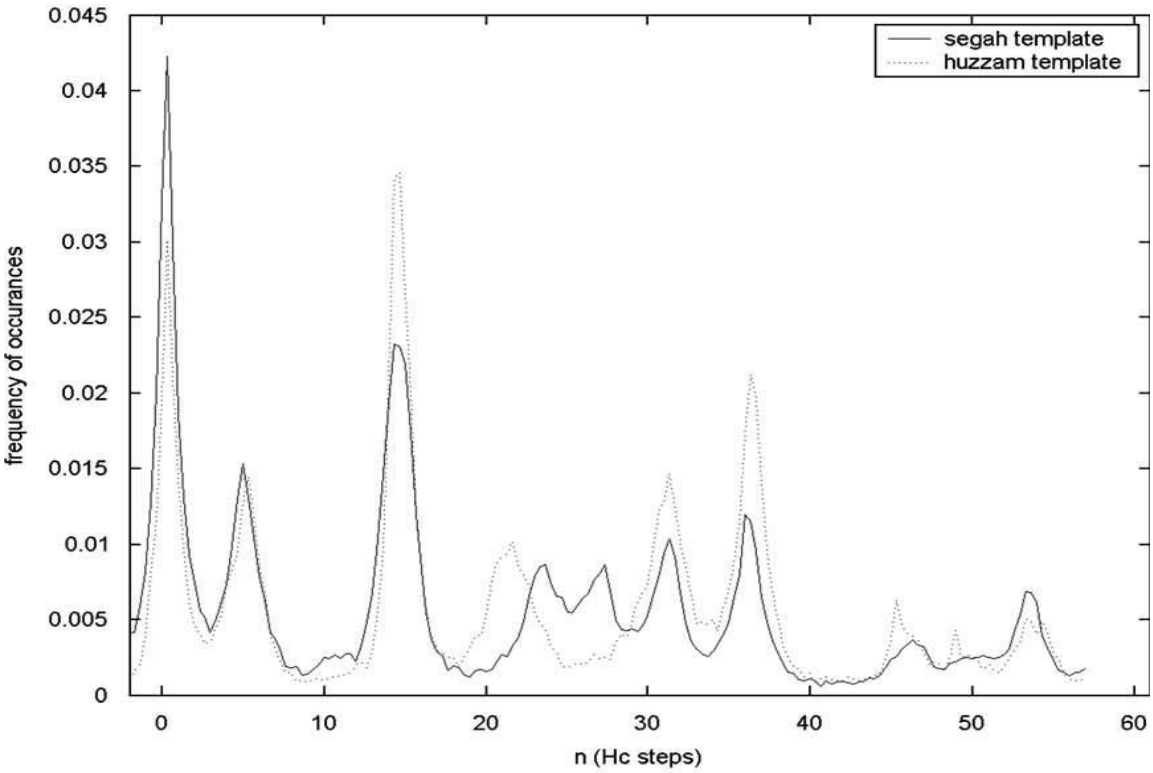


Figure 9 shows the templates for two modes, *segàh* and *huzzam*.

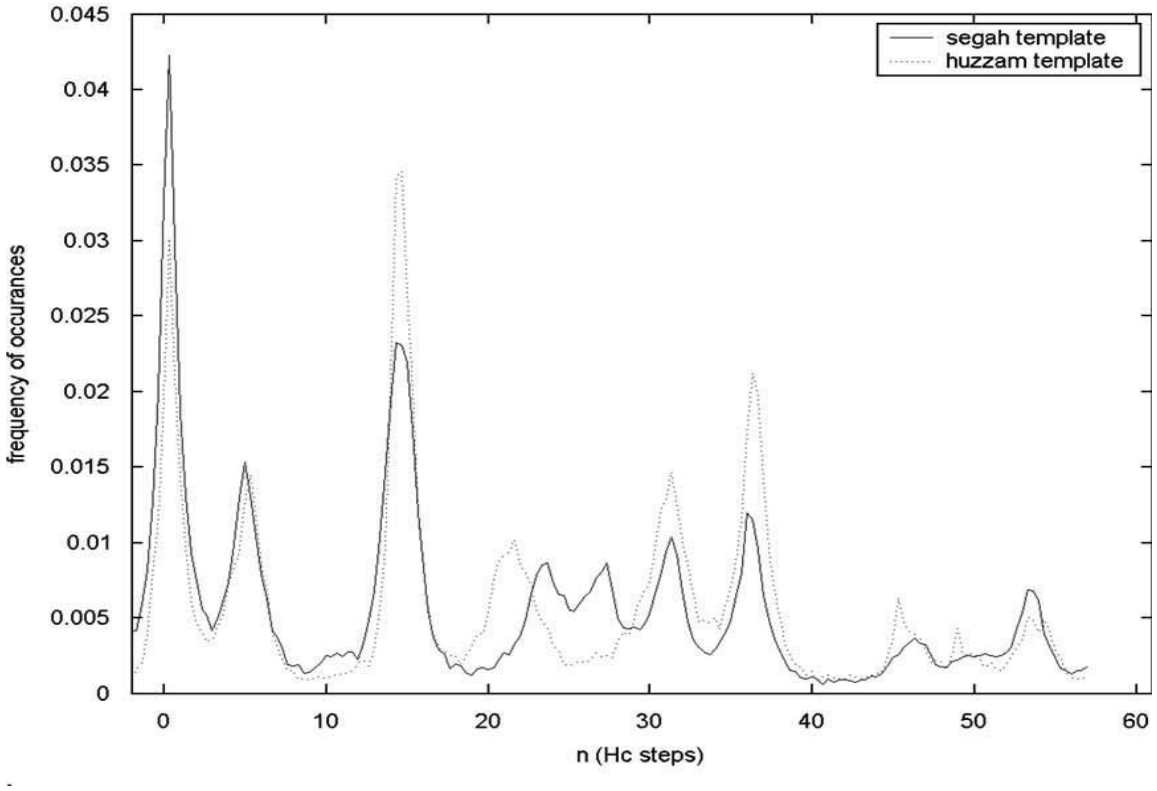


Figure 9 Pitch histogram templates of makam *segàh* and *huzzam.*[30] Reproduced from [3]

They describe their method as very discriminative for modes such as *saba* and *segàh*, but say that it does not recognise a few modes such as *kurdili hijacker* and *huseyni*. Fifty-three of the samples were recognised falsely and the performance rate that they achieved was 69.19%. They lose the temporal dimension and the musical context as a result of making pitch histograms, instead of using the time-varying F0[31]. Ascending-descending characteristics of the notes, modulations and also motives are also missed, because of using the histograms.

In another piece of research, Gedik and Bozkurt compared a discrete pitch system versus a continuous pitch for Turkish music for recognition of only two modes, *saba* and *hijaz*. The result showed very different F-measure results of 21% and 95%, showing that for Turkish music, a continuous pitch space should be chosen, rather than discrete intervals as in Western music [4].

[30] Hc refers to the Holdrian comma, a division of the scale to 53 equal parts (53-TET) [3].
[31] Pitch histograms record only tone occurrences, not the order of them and the timings.

3.3.3 Work on Arabic music: maqàm recognition, tonic detection

Ionnidis Leonidas developed a *makam (maqàm)* recognition system for Arabic and Turkish music in his Masters thesis [43, 44].

He defined a makam as a compositional device with a scale with an ascending or a descending development (*sayir*), and a tonic with modulations to certain grades of the scale. He considered a makam as a genre, as it is related to mood and considered approximately 30 *makams* which can be made by connecting two tetrachords or pentachords. Leonidas is inspired by the Turkish *makams* that Gedik and Bozkurt used in their studies.

His database consists of files extracted from commercial CDs, including mainly Turkish songs, but also songs from Greece, Egypt and Iran. He uses a frequency range of 100 Hz–5000 Hz in chroma calculation. As he was working on the basis of 53 Holdrian commas in an octave, the pitch resolution is 1200/53= 22.64 cents.

He compared two methods: one is based on chroma and template-matching techniques. His first approach is based on chroma averages. He tested several distance measures and found out that Euclidean distance was the best, and distance measures which were based on cross correlation provided the worst accuracy rates.

The second approach operates based on a trained classifier and he found that the Support Vector Machine (SVM) was the best classifier in his tests. He reports an F-measure of 0.69 and 0.73 for these methods respectively[32]. His detailed results for precision, recall and F-measure for the two methods are listed in **Table 5**. The first row shows the results, based on HPCP. The results in the second row are based on HPCP, after moving the components to the tonic. Based on error analysis, he concludes that some of the errors are musically coherent and the two methods can complement each other. However, some of his assumptions can be challenged: for instance, he assumes that there is no standard diapason, whereas nowadays most musicians tune to concert pitch; or he considers an equal temperament for Western music and a non-tempered scale for non-Western music, whereas a tempered scale can be assumed for MIR purposes, for Persian or for *maqàmic* music in general, as explained by Heydarian [5].

[32] See glossary of terms for definitions.

Table 5 Comparison of template matching and machine learning methods. Reproduced from [45]

	Template matching			Machine learning		
	P	R	F	P	R	F
HPCP	0.50	0.45	0.45	0.29	0.31	0.29
Tonic-Shifted	0.69	0.69	0.69	0.72	0.73	0.73

3.3.4 Work on Greek music: a pitch tracker with quartertone resolution

Aggelos Pikrakis presented a pitch tracker where a low-complexity method, based on the autocorrelation [46]. This is a simplified version of the multi-pitch tracker proposed by Tolonen et al. [47], simplified and customised for Greek music, including quartertone intervals and assuming that in a Greek ensemble the melodic scheme is dominant, over other sounds in a musical signal (e.g. harmony, drones, ornaments or any other note). Thus only the first (lowest) pitch is considered.

Tolonen's method is itself a simplified and computationally effective version of Meddis' and O'Mard's method [48]. In the beginning, a pre-whitening function is applied to the signal to remove the short-term correlations due to formants. The whitening flattens the signal spectrum. It may degrade the signal-to-noise ratio. Tolonen's tests have shown that it does not degrade the 2-channel pitch estimator. The signal is divided into two frequency bands over and below a cross frequency of, for instance, 1000 Hz. The cross-frequency is not a critical parameter; it can be set to a value between 800–2000 Hz, depending on the pitch range of interest. A generalised autocorrelation is performed over each frequency band and the results are summed up to make the summary autocorrelation function (SACF). Finally, post-processing is performed to remove the tone partials (harmonics). The SACF is rectified[33]; it is expanded in time by a factor of 2, 3 or more. This is subtracted from the rectified SACF and is rectified again. The process can be done with different time expansions to remove the 2nd, 3rd, and higher partials. The result is called the enhanced SACF (ESACF), which reveals the signal periodicities. Transform-domain computation of ACF is computationally efficient and can be used for real-time applications. The spectrum can also be compressed non-linearly (Equation 28). When the pitch levels are not equal, an iterative method can be used, i.e. finding the first pitch and filtering it out, then, capturing the 2nd pitch and filtering it out, etc. ESACF can also be used for sound source separation of harmonic sounds.

[33] Negative components are inverted, that is they are multiplied by (-1).

Pikrakis reports that only 23% of the signal frames of his study[34] had the maximum amplitude at f_0. In this method, a moving window FFT with hop-size h is applied to the signal. The frequency content is split into two (or more) frequency bands. Then the following MATLAB command[35] is used to calculate the ACF as the sum of ACF of different frequency bands:

$$SACF = real\left(IFFT(abs(FFT(x_l)^p) + IFFT(abs(FFT(x_h)^p)\right) \quad (3)$$

where x_l and x_h are the low-pass-filtered and high-pass-filtered versions of a signal x of length N and p is a predefined decimal norm (it is taken 0.17 here). This, which is called the summary auto correlation function (SACF), is obtained through inverse Fourier transform of the magnitude of the frequency domain signal. Thus all the complications due to unknown phase relations are overcome, since the phase is forced to zero. On the other hand, if there is a distinct formant in the signal, it is maintained in the ACF. Thus one is not safe from the well-known errors of recognising a higher harmonic or a sub-harmonic of f_0 which leads to the octave error [7]. Furthermore, ACF is not suitable for the analysis of polyphonic musical signals [49].

For each window, the first K (e.g. K=5) prominent peaks are considered. Among them, the one with a lower frequency will be selected as the f_0, provided that its amplitude is greater than a threshold T_p. If the highest Fourier peak is less than a threshold T_h, then the frame will be considered as non-periodic and the output frequency will be set to "1". 2.5% of the studied signals were reported non-periodic. Pitch doubling, where the f_0 is absent was also reported [46]. Some errors were modified by averaging the pitch of two neighbouring frames, while the error would propagate if the octave error happens in sequential frames. Finally, the pitches are quantised to quartertone intervals.

3.4 Works on Iranian music: pitch tracking and mode recognition

There are three works in this field, which have been performed separately. One of them was undertaken by the author during his MSc and MPhil research at Tarbiat Modarres University,

[34] Pikrakis analyses Greek traditional music performance where the clarinet dominates. However, the figure might be different for other genres of music or for other instruments.

[35] The real part is taken because of the numerical precision of MATLAB, which produces a small imaginary part.

Tehran, 2000 [2] and Queen Mary, University of London, 2008 [5]; another work was published by Darabi et al. at Sharif University of Technology, Tehran, 2006 [50]; and finally a recent paper by Abdoli was published in the Proceedings of the ISMIR Conference, Miami, 2011 [51].

3.4.1 State of the art techniques for Persian pitch and mode recognition

- Heydarian provided an algorithm for pitch tracking of Persian intervals [2, 5, 41, 52] and also algorithms for classification of audio musical files, based on their scale tuning [5]. A method was devised based on comparison of (1) signal spectrum, (2) chroma and (3) pitch histograms by Minkowski's distance measure of different orders and cross correlation [5].

  In that research, the existing methods for music transcription that work on Western music, such as those reviewed earlier in this chapter, were customised and applied to Persian music on a database of santur samples in three different *dastgàhs*. A pitch tracker for monophonic Persian musical was also presented that involved quartertone intervals [2, 41, 52]. The different pitch trackers were also compared in the MPhil thesis [5]. Bozkurt had suggested a continuous scale for Turkish intervals, including 53 commas; however, based on verification tests, Heydarian concluded that a 24-TET is sufficient for Persian music.
  Three features were applied to a database of audio samples in three *dastgàhs* and the following conclusions were drawn: spectrogram yielded a high recognition rate of 92.48% for samples of one instrument, played from certain tonalities (frame size: 32768); pitch histograms recognised 70.37% of the samples. There is a calculation cost during the pitch tracking method and also the errors in pitch tracking are passed to the mode recognition stage; chroma feature recognised 83.3% of the samples. All are dependent on the tonality.

  One important aspect of the templates is that for *shur* mode, for instance, both F natural and F *sori* (half sharp) components exist next to each other, as the lower F is *sori* and the higher F is natural, where all notes are folded into one octave. An alternative way would be to consider more than 7 notes for each *dastgàh*; but, as it is usually possible that music is played an octave higher or lower, Heydarian decided, on balance, to limit the templates to a folded octave. These methods work well with a high recognition rate and operate on samples with more than one note in a frame. The tests were performed upon a database of one hour of Persian music in three *dastgàhs*; they are limited in scope in that they use samples of just one instrument (the santur), played in one tonality.

- A group of Iranian engineering students and musicians, led by Nima Darabi applied a melody-based approach to *dastgàh* recognition in their final BSc project. The results were published in Darabi's BSc thesis in 2004 and in a conference paper in 2006 [50]. In this method, scale intervals (note occurrences) and melody patterns (made of a sequence of the dominant pitch in each frame) of a piece, along with the tonic and *shàhed* (most frequently occurring note) are used as the feature vectors. Their classifier is a neural network with hidden layers with backward propagation algorithm, where a network is trained per *dastgàh*.

  The team included engineers and people who were familiar with Western music and their musical assumptions and the analysis does not describe the subtleties of the Persian music. For instance they use the term, *maqàm* to refer to derivative modes, whereas *àvàz* is the right term. Their analytical results were based on a small database and the tests were focused on samples of one *dastgàh* only. They have 30 short melodies for all the *maqàm*s. Seventy pieces in *dastgàh e homàyun* were included in their tests, 64 of them were compared to theoretical templates for this mode and were correctly recognised to belong to 11 different derivative modes of *homàyun*. Subsequently, a neural network is trained and tested with leaving one out technique, where it is trained by 55 samples and tested on the reminding 9 samples. They report that their system recognises the derivative modes in the second stage of tests with an average error rate of 28.2% over 5 different tests. Their method is not tested upon samples in the other modes. It is a 0–1 test, whether a piece is in *homàyun* mode and its derivatives or whether it is not.

- Abdoli devised a *maqàm* recognition system based on a fuzzy classifier type two [51]. He presented a 'fuzzy' classifier for *dastgàh* recognition. He used a pitch tracker and a fuzzy logic type two classifier, where each note is a fuzzy set and fuzzy logic takes into account the flexibility of Persian intervals. The *dastgàh* was recognised based on the similarity between these fuzzy sets and theoretical data. Abdoli tested the system on a data set of 210 tracks, including mainly solo vocals and some monophonic instrumentals and stated a recognition rate of 85% [51].

3.5 Commercially available audio search engines

Shazam [53] and SoundHound [54] are two online music recognition systems [55, 56].

Shazam is a music matching website that uses an algorithm to find commercially available tracks by recording a short snippet of music via mobiles in noisy environments. It makes a fingerprint of the query and finds a match for it in the bank of fingerprints of millions of music tracks. Each track is first analysed to find the prominent onsets, which are preserved even in noisy environments. Each pair of onsets is parameterised with frequency and the time duration between them. The values are quantised to make a set of 20–50 distinct landmark hashes per second of music. There is a bank of millions of these landmarks, with which the recorded audio snippets are compared. The fingerprint of each track is represented by hundreds of landmarks, and their timings, the information being held in an inverted index. Several landmarks of a piece may appear in other versions of the same piece or in other pieces that show similarities with the query. The robustness of Shazam is in its ability to find a match with a few landmarks that can be achieved even in noisy environment with a distorted recording.

Shazam is fast and computationally efficient. It can spot a file among over a million tracks in a few milliseconds. However, it does not have a melody recognition system thus does not recognise a melody if it is played with a different instrumentation. The developer states that Shazam is not generalised to live music, although it is claimed that it was successfully used in a few such cases, where the performance was in a timing closely similar to the bank, or the performers were just lip synching. Conversely, the algorithm is very sensitive to particular versions of songs; even if a piece is performed by the same person several times, with differences so slight as to be indistinguishable to the ear, the right version can be spotted by Shazam [55]. Ellis has implemented the algorithm in MATLAB environment, and it is publicly available in [57].

SoundHound is a mobile application that finds a piece via query-by-example. It matches a combination of melody, rhythm and lyrics, independently of key and tempo [54, 56]. In comparison with Shazam it recommends audio files which differ more from the query.

4 The database

4.1 Introduction

This chapter explains the characteristics of the database that was constructed and used throughout this research. A dataset of audio musical files is needed in order to pursue a quantitative evaluation of the algorithms that are developed and presented in the subsequent chapters. It is essential to make a suitable database because the validity of the analytical methods depends on the samples on which they are tested. Although most of the samples were specially recorded on a santur, the database also includes solo samples of other instruments, group performances and commercial music tracks, in order to provide a better perspective on how the algorithms operate on different types of audio musical samples.

4.2 Structure of the database

The different parts of the database, explained in the following subsections, are referred to as db1, db2, db3, db4 and db5 respectively. The majority of the samples are Iranian pieces, played on a santur; a few recordings on other Iranian instruments, the *kamàncheh* and *ney*, ensemble playing and samples of non-Iranian music on the santur are included in db2-db5, as explained below. The samples were obtained during rehearsals, live performances, and studio recordings, and a few are from commercial CDs. **Table 6** lists the different parts of the database, number of files, and their durations.

4.2.1 Santur solo recordings made in a studio (db1 and its extensions)

Db1, which includes the great majority of the samples, consists of scale notes, opening sections (*daràmad*), songs, metric and non-metric pieces, improvisations, and random sequences of notes, all played on the santur by the author; it comprises 5706 seconds (91 pieces) of music played on 11-bridge and 12-bridge santurs[36], in all five of the main Persian scales: *esfehàn, chàhàrgàh, shur, màhur, segàh*[37]. This main database is extended to include a separable 484 seconds (13 pieces) of scale notes and improvisations in a pentatonic scale. The first 6 samples in each *dastgàh* are contained in the DVD that accompanies this thesis.

[36] The non-struck santur strings were not artificially dampened.

[37] A few samples in pentatonic scale (E-G-A-B-D) are also included.

4.2.2 Small ensemble recoded in public performance (db2)

For additional tonic detection tests, the database was augmented with 3806 seconds (16 pieces) of live performances of Persian, Kurdish and Greek pieces, by a small ensemble including santur, voice, lyra (Greek fiddle) and daf (Kurdish frame drum).

4.2.3 Old Greek gramophone recordings (db3)

To assess the applicability of the algorithms developed to a body of relatively noisy old recordings, an extreme case for our tests, db3 consists 1075 seconds (6 pieces) of Greek pieces in rebetiko style digitised from gramophone recordings. Rebetiko, a fusion of oriental Greek music, the majority of whose scales are similar to Persian scales, and music from mainland Greece, was created in the early 20th century as a result of the forced migration of Anatoly Greeks to Athens.

4.2.4 Kamàncheh solo recordings made in a studio (db4)

To assess the applicability of the algorithms developed to an instrument characterised by less stable pitch than the santur, a body of 1948 seconds (11 pieces) of *kamàncheh* (Persian spike fiddle)[38] solo was recorded in a studio.

4.2.5 Piano solo recordings made in a studio (db5)

To assess the applicability of the algorithms developed to another struck chordophone, one limited to 12-TET, 629 seconds (7 pieces) of piano solo, including Iranian pieces in *màhur*, *esfehàn* and *chàhàrgah*, were recorded in a studio. These modes were played in a Westernised scale, the quartertones rounded to the nearest semitones in a melodically and socially acceptable manner, but as no quartertones could be played on the piano (unless specially retuned to a Persian scale), there is no sample in *segàh* mode.

4.2.6 Isolated note samples recorded on the santur in a studio (db6)

Ten samples of the following notes were played and recorded solely for the purpose of calculation of the inharmonicity factor[39]: F3, Aq3, C4, F4, Aq4, C5, F5, Ab5, C6, Eb6. These isolated samples were played with a variety of sticks and with different dynamics.

[38] Unlike those of the santur, the notes stopped on the strings of the *kamàncheh* are not fixed in pitch.

Table 6 Different parts of the database

Database part	Instrumentation	Recording	No of files	Duration (s)
db1	santur solo	Studio	91	5706
db1 extension	santur solo	Studio	13	484
db2	santur-lyra-voice-daf	Live concert	16	3806
db3	Greek ensemble	Gramophone	6	1075
db4	kamancheh solo	Studio	11	1948
db5	piano solo	Studio	7	629
db6	santur solo	Studio	100	100

Tables 7–11 provide detailed information about the main part of the recordings (db1). The files are mono, 16-bit, sampled at 44.1 kHz. The musical signal recorded comprises series of notes, arpeggios, passages, introductions, improvisations, metric pieces (songs and instrumental), and random sequences of notes. **Table 7** describes the *esfehàn* and *homàyun* modes. *Esfehàn* is a derivative (*àvàz*) of *homàyun*. *Esfehàn* A is a relative mode for *homàyun* Fs, and both are performed in a similar tuning; the tunings differ slightly, between the fixed and moving accidentals[40], such as C#3 in the case of *homàyun* and G4 natural for *esfehàn*.

Table 8 contains information about the samples in *chàhàrgàh* and its two derivatives, *zàbol* and *mokhàlef*. **Table 9** shows information on samples in *shur* and its derivative *dashti*. **Table 10** describes the samples in *segàh* and its derivative, *mokhàlef*.

And finally, **Table 11** provides the information about the samples in *màhur* and two of its derivatives, *shekasteh* and *delkash*.

[39] The inharmonicity of a string depends on its length, stiffness and tension. The inharmonicity factor is a constant that can be calculated either via these attributes, or by measuring the pitch and harmonic deviations of the strings of an instrument. It is explained in detail in Chapter 6.

[40] A fixed accidental is fixed throughout a modal performance, whereas a moving accidental changes during a performance.

Table 7 The tuning of, and information about the samples in *esfehàn* and *homàyun*

	esfehàn	*homàyun*	*dastgàh*
Tonality	A	Fs	Fs
Fixed accidentals	-	Fs, G#	-
Moving accidentals	Fs, G#	-	Fs, G#, Cs
Scale	2	0	2
Passage	2	0	2
Intro	2	0	2
Àvàz[41]	4	1	5
Metric pieces	4	1	4
Random	1	0	1
Total no. of files	15	2	17
Total time (s)	835	340	1175

Table 8 The tuning of, and information about the samples in *chàhàrgàh*, *zàbol* and *mokhàlef*

	chàhàrgàh	*zàbol*	*mokhàlef*	*dastgàh*
Tonality	A	C#	Fs	A
Fixed accidentals	G#, Bq, C#	G#, Bq, C#	G#, Bq, C#	G#, Bq, C#
Moving accidentals	-	-	-	D#
Scale	1	0	0	1
Passage	0	0	0	0
Intro	4	0	0	4
Àvàz	3	3	1	7
Metric pieces[42]	2	0	1	2
Random	1	0	0	1
Total no. of files	11	3	2	15
Total time (s)	489-x	146	57+x	688

[41] *Àvàz* here is a derivative mode.

[42] One of the songs has a modulation of duration x seconds from *chàhàrgàh* to *mokhàlef*. Therefore x seconds are subtracted from the duration of *chàhàrgàh*, and the same duration is added to *mokhàlef.*

Table 9 The tuning of, and information about the samples in *shur*[43]

	shur	*dashti*	*dastgàh*
Tonality	E, A	B	E, A
Fixed accidentals	Fs	-	-
Moving accidentals	Bq	Fs, Bq	Fs, Bq
Scale	2, 2	0	2, 2
Passage	0	0	0
Intro	2, 2	0	2, 2
Àvàz	2, 1	1	3, 1
Metric pieces	4, 3	4	8, 3
Random	1	0	1
Total no. of files	11, 8	5	16, 8
Total time (s)	480, 214	330	810, 214

Table 10 The tuning of, and information about the samples in *segàh*[44]

	segàh	*mokhàlef*	*dastgàh*
Tonality	Fs	D	Fs
Fixed accidentals	Fs, Bq	Bq, Cs	Fs, Bq
Moving accidentals	Cs	-	F
Scale[45]	2, 3	1	3
Passage	0	0	0
Intro	2	1	3
Àvàz	1, 2	2	3
Metric pieces	3	1	4
Random	0, 2	2	2
Total no. of files	12	7	16
Total time (s)	828	245	1073

[43] In *sure*, lower F is *sori* (Fs), while the octave F is natural (F) in the derivative mode, *dashti*
[44] In *segàh*, lower F is *sori* (Fs), while the octave F is natural (F) in the derivative mode, *mokhàlef*
[45] One of the scale files, an *àvàz* and 2 random files include both *segàh* and *mokhàlef*

Table 11 The tuning of, and information about the samples in *màhur*

	màhur	*shekasteh / delkash*	*dastgàh*
Tonality	G	G	G
Fixed accidentals	-	-	-
Moving accidentals	F#	Bq	F#, Bq
Scale[46]	4	1	4
Passage / arpeggio	2		
Intro / *Àvàz*	2	1	3
Metric pieces	15	0	15
Random	1	0	1
Total no. of files	24	2	25
Total time (s)	1680	66	1746

4.3 The spectrum of the signals

Figure 10 shows the spectrum of an isolated A4 sample. It can be seen that the sound is rich in harmonics and that the higher harmonics decay faster through time.

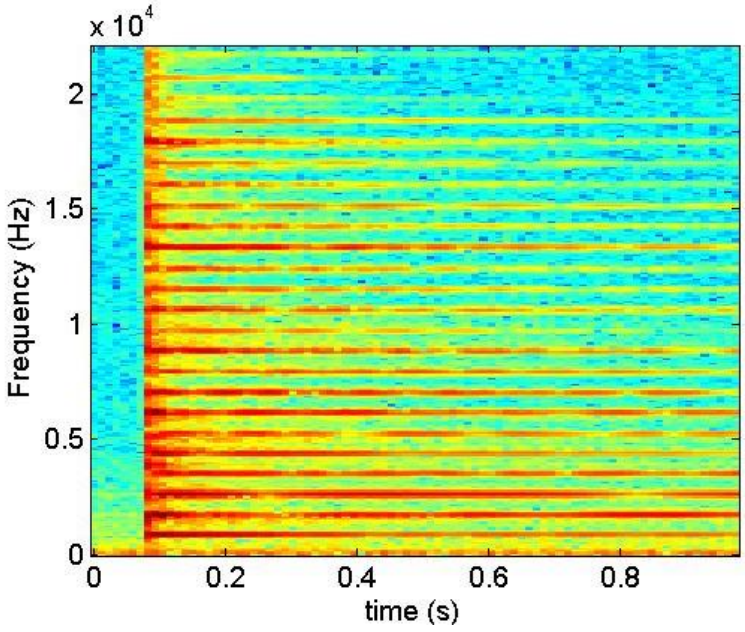


Figure 10 Spectrum of an A4 sample

[46] One scale file in *màhur* and *shekasteh*.

At the onset, in addition to the fundamental frequency and its harmonics, several other transients are present. This is partly due to string stiffness and the way the strings lie on the bridges. Furthermore, almost all strings on the santur are excited when any one note is struck by a stick.

Figure 11 represents the frequency of ten samples of the same note (G4), played with different sticks and with different dynamics. It can be seen that the magnitudes of overtones of the different samples are different. However, their positions on the horizontal axese are he same.

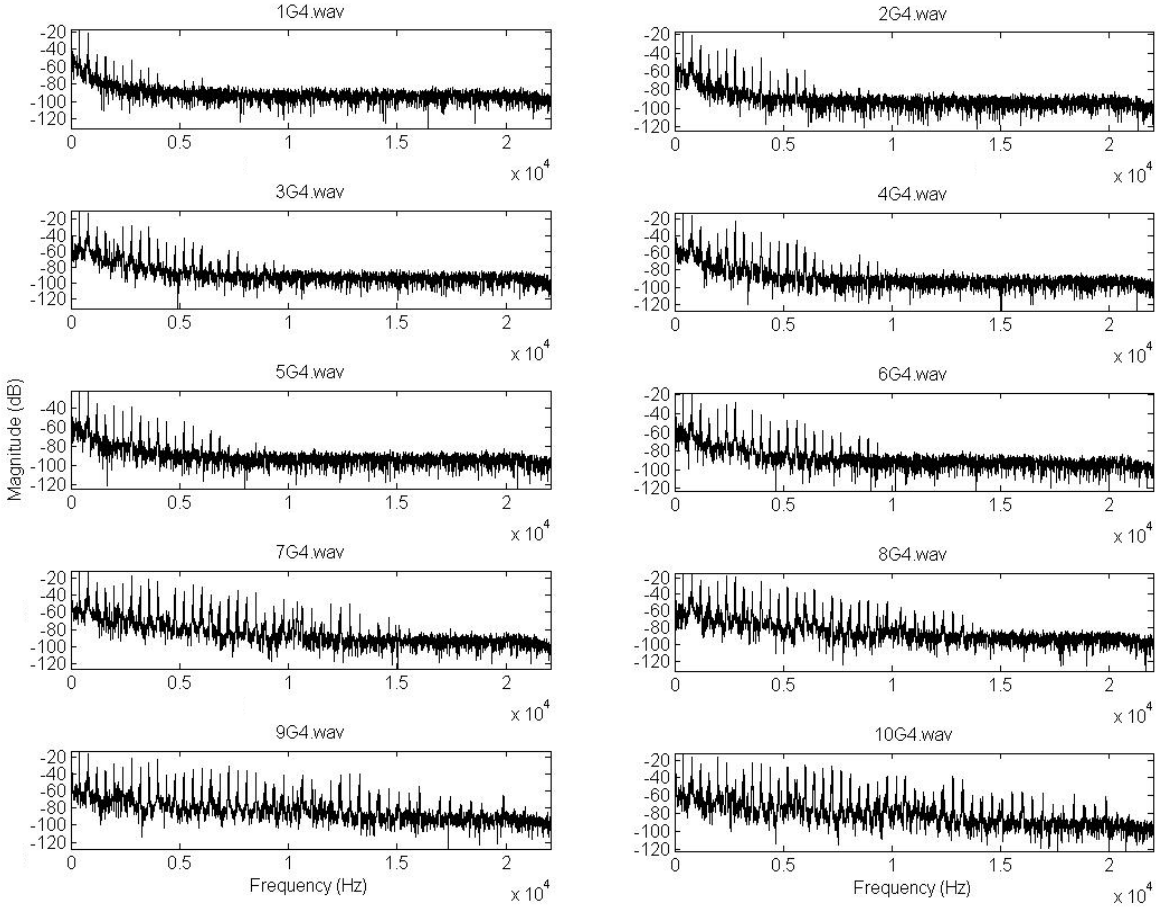


Figure 11 Samples of G4 played with different *mezràb* and different dynamics

Figure 12 shows the average of the ten graphs presented in **Figure 11**. The harmonics, now strengthened, can be seen more clearly. As santur has a sustained sound, the averaging function increases the amplitude of the harmonics, which appear in several frames, and reduces the effects of noise and unwanted transients.

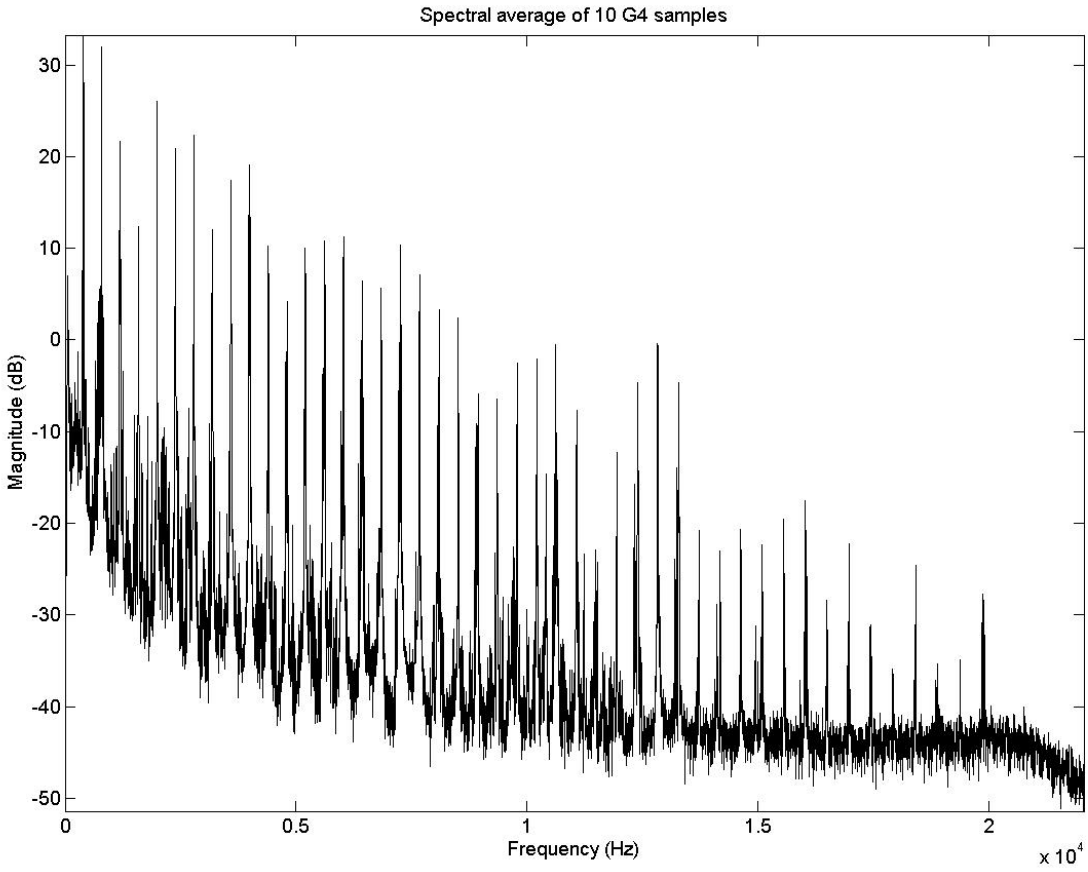


Figure 12 Spectral average of G4 samples in **Figure 11**

The database consists mainly single melody lines. However, there are instances, where more than one note is played simultaneously or when the traces of several sustained notes overlap, where the samples can be considered as polyphonic[47]. The pieces are assumed to have reasonable signal to noise ratio, because otherwise noise or other transients would be interpreted as scale intervals, and this would leading to additional errors. However, for additional tests, a few noisy samples of gramophone recordings were also included. Non-pitched percussive signals or pitched percussion, where the pitches are not in the scales' intervals, are generally avoided. However, some of the samples include percussion. A drone, which is common in Iranian music (for instance in a *chàhàrmezràb*), increases the number of occurrences of one or more notes, and may lead to false recognition results, and thus was avoided.

[47] Polyphony here refers to the cases where either two or more notes are played at a time where a note sustains for several frames, until the subsequent note is played.

5 Pre-processing: signal processing to prepare data

5.1 Signal processing to prepare data for evaluation

In this chapter, different pre-processing measures investigate in Chapter 7 are introduced and explained, irrespective of whether or not they benefit the results obtained.

5.1.1 Onset detection

To avoid transients during the attack time, the analysis window should start at a proper distance from after the onset. Tests have to be undertaken to find the optimised distance of the analysis frames from the onsets. Onset detection prior to pitch analysis bypasses the fast changes in pitch and reduces the amount of calculations. Furthermore, short-duration notes and less likely intervals such as consecutive quartertones and other ornaments can be avoided. Supposing that the duration of an ornament is a hemidemisemiquaver[48], with a tempo of 60 and a sampling rate of 44100, it is 62.5 ms or 2756 samples. The pitch trackers can start well after the transients (around 2756 samples in this case).

When the sampling window starts at a proper point (the optimal point to be found through tests), after the onset, most of the transients will be bypassed. The result can be quantised (held constant) between the onsets and the transients, so that the fast changes in pitch are avoided.

Different features can be used for onset detection [11]: the temporal methods such as those based on the energy of the signal or its derivatives are better for strongly percussive signals, while the spectral domain methods work well with strongly pitched signals. Complex-domain spectral differences can be used for both cases, but at a higher calculation cost. The wavelet helps in precisely localising the events, but needs post-processing to remove the spurious peaks which increase the computational load. A combination of the wavelet and other methods could be a better option. Finally, the statistical methods are the best, but are heavy in terms of the computations involved. In any method the signal is transformed to an intermediate and simplified version, where the onsets are accentuated [11].

[48] Sixty-fourth note.

The energy is calculated, using the following summation [11]:

$$E(n) = \frac{1}{N} \sum_{m=-N/2}^{N/2-1} [x(n+m)]^2 w(m) \quad (4)$$

Where x is the signal, N is the frame size, and $w(m)$ is an N-point Hann window or the smoothing kernel. It is then passed through a low-pass filter and a peak picker with adaptive threshold finds the onsets.

The HFC (High Frequency Content) is calculated using the short-time Fourier transform of signal [11]:

$$X_k(n) = \sum_{m=-N/2}^{N/2-1} x(nh+m) w(m) e^{-2j\pi mk/N} \quad (5)$$

Where h is the hop size[49]. The spectrum is weighted preferentially before the Energy calculation so that the high frequency contents are accentuated [11]:

$$E(n) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} W_k \left| X_k(n) \right|^2 \quad (6)$$

Thus, the HFC produces sharp peaks at attack transients; it works well in finding the onsets of pitched-percussive sounds and thus is particularly suitable when the santur is the sound source.

5.1.2 Enframing

After the onset detection, the signal is framed using a Hann window of appropriate length unless otherwise stated

5.1.3 Silence and high-energy suppression

Another important preprocessing task is to omit frames whose energy is less than or more than certain thresholds. Here frames with energies less than 10% of the average frame energy are removed. Similarly, a threshold for the maximum permitted energy can be set. These will help to avoid the transients and spurious peaks, during the onset, and silent frames. The minimum and maximum thresholds depend on the maximum, mean and minimum energy of the frames as well.

[49] Hop size is the distance between start of a window and the start of the next window. Overlap is the distance between the start of a window and the end of the previous window. Hop size + Overlap = window length.

5.1.4 Normalisation

The frequency domain frames are normalised to reduce the influence of note dynamics. Even if the frames have been normalised before silence and high-energy frame suppression, they need to be normalised again, afterwards.

5.2 Determination of pitch and harmonic deviations, and the inharmonicity factor

This section presents an investigation which, for the first time, measures and calculates the pitch and harmonic deviations, and consequently the inharmonicity factor, of the santur. It was anticipated that this would be an essential preliminary step to testing whether mapping the harmonics to the fundamental frequency in chroma calculations is affected by the inherent inharmonicity of the santur.

5.2.1 Inharmonicity equations

Due to inharmonicities, harmonics of a fundamental frequency are not located exactly at multiples of f_0 and frequencies of the overtones are expected to move slightly upwards. The positions of the overtones for a stiff string can be calculated using Equation (7) [58, 59]:

$$f_h = h f_0 \sqrt{1 + \beta h^2} \tag{7}$$

Where f_0 is fundamental frequency, f_h is the position of an overtone, h is harmonic index, and β is inharmonicity factor.

Therefore, the fundamental frequency is slightly deviated:

$$f_1 = f_0 \sqrt{1 + \beta} \tag{8}$$

And Equation 7 can be rearranged as:

$$f_h = h f_1 \frac{\sqrt{1 + \beta h^2}}{\sqrt{1 + \beta}} \tag{9}$$

Where f_1 is the measured fundamental frequency and deviates slightly from f_0.

Different factors such as thickness, length and tension force of a string contribute to inharmonicity. Increasing the thickness and decreasing the length of a string or the tension force result in a higher

inharmonicity factor. The following equation describes the inharmonicity factor of a string in terms of its length, l, its diameter, d and string tension T [58, 59]:

$$\beta = \frac{E\pi^3 d^4}{64 l^2 T} \tag{10}$$

where E is Young's modulus[50].

The inharmonicity factor for different notes on a 9-bridge and an 11-bridge santur is calculated below.

5.2.2 Measuring the inharmonicity factor

The dataset consists of 10 isolated note samples, which are played with different dynamics and using felts with different thicknesses, and in one case with a bare pair of sticks. The frame size is 32768 samples and with a sampling rate of 44.1 kHz, frequency resolution becomes 44100/32768= 1.35 Hz. Positions of the fundamental frequency and its overtones can be used to calculate the inharmonicity factor.

The pitch deviation is measured for the following notes: F3, Aq3, C4, F4, Aq4, C5, F5, Ab5, C6, Eb6. The measurements show that the first and second octaves are compressed by 11 and 28 cents respectively, while the third octave is stretched by 20 cents. Apart from an F4 sample which has a lower pitch deviation and is interpreted as mistuned, the bass and middle pitches tend to be less than the tempered values (24-TET) as they are moving towards higher notes, whereas the treble pitches tend to be more. Thus, the treble pitches on a santur are stretched similarly to those of the piano [60], while the bass and middle pitches are compressed in contrast.

The harmonic deviation of the first 8 overtones of f_0^1 is calculated, using Equation 7. The inharmonicity factor[51] β can be calculated in terms of f_0^1 and overtone positions f_h:

$$\beta = \frac{(f_h / h f_0^{\,1})^2 - 1}{(h^2 - (f_h / h f_0^{\,1})^2)} \tag{11}$$

[50] Young's modulus is the constant of elasticity of a substance. It represents the ratio of stress to strain for a string or a bar of the given substance. Young's modulus is the force per unit cross section of a material divided by the increase in its length resulting from the force.

[51] The impedance between the bridges and the sound board is ignored here.

Or, in terms of the h^{th} and m^{th} overtones, f_h and f_m:

$$b = \frac{\left(\frac{m.f_h}{h.f_m}\right)^2 - 1}{h^2 - m^2.\left(\frac{m.f_h}{h.f_m}\right)^2} \qquad (12)$$

The inharmonicity factors of different notes are not the same. As one moves towards higher notes of each tone area (bass, middle and treble), the inharmonicity factor increases; and the values calculated from different harmonics also vary. An average value can be calculated across several notes and several of their harmonics.

It should be noted that variations of inharmonicity factor using the first few harmonics are high. Thus, it is generally more accurate to use higher harmonics in the calculations [58]. This improves the accuracy of calculations due to frequency resolution as well. As just the first eight overtones are taken into account here, calculation of the inharmonicity factor based on h8 and h4 or h5 is a good choice to avoid using adjacent or proximal partials. **Figure 13** shows the inharmonicity factor calculated based on the seven combinations of overtones 1 to 7 with overtone 8 (1 and 8, 2 and 8, et seq.). The curve which exhibit considerable changes at C4, Aq4, Eb5 and Ab5, which corresponds to the measurement using the neighboring h8 and h7 harmonics, is excluded from the calculations. The average value of the inharmonicity factor obtained for a 9-bridge Salari santur is 0.00031. This can be compared with the inharmonicity factor of a piano, which is around 0.0004 [58].

In this section, the fundamental frequencies of 12 notes, and their lowest 7 overtones, were used to calculate the inharmonicity factor of a Salari santur. Future work could involve determining a more accurate value of this parameter by considering the remaining 15 notes of the santur; and by taking higher overtones into consideration. It would also be helpful to obtain the attack and decay curves for the partials of different notes (due to nonlinearities, the positions of the frequency components change during the attack and decay); and to compare analyses of the sound of several santurs. The thicknesses of the santur strings are nearly uniform, and the 12 notes that were taken into account in this research represent different areas of the santur where strings have different lengths and

tensions. Thus it is expected that factoring in the additional recommended tests will result in a more accurate value for the inharmonicity close to the value, which was calculated here.

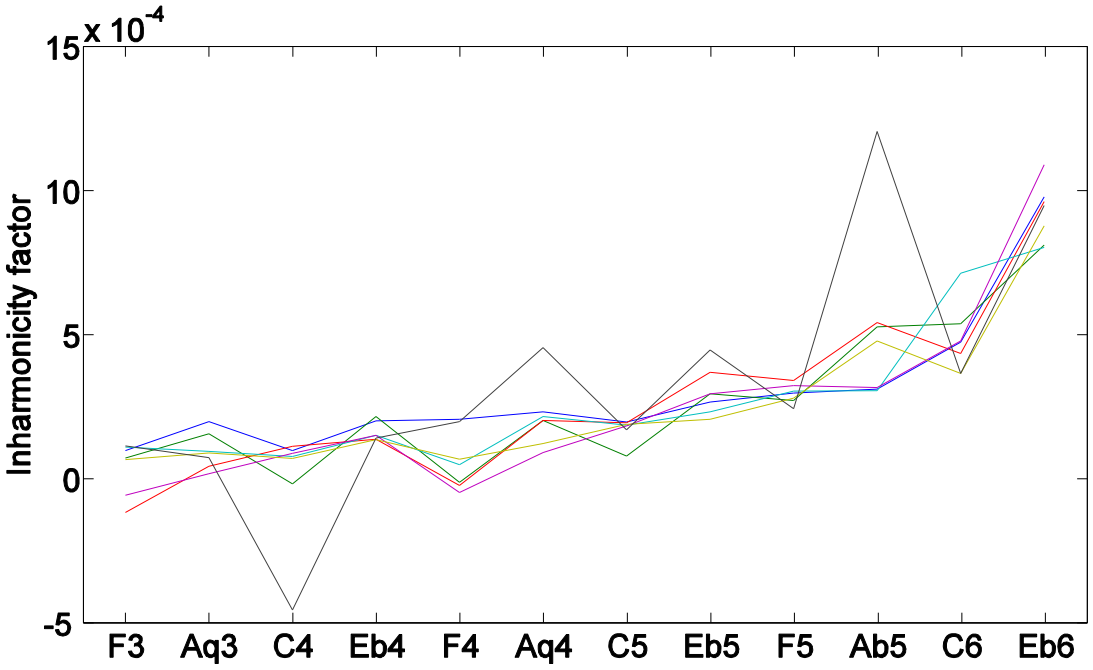


Figure 13 Inharmonicity factor based on the 8th overtone vs. the 1st to 7th overtones.[52]

5.2.3 Investigating the inharmonicity effect on chroma calculation

In order to investigate the effect of the inharmonicity factor on a folded Q-transform and the resulting chroma, the deviation of the hth harmonic that is folded to the first octave is calculated, based on Equation 9:

$$folded\_dev(h) = \frac{f_h}{hf_1} = \frac{\sqrt{1+h^2 b}}{\sqrt{1+b}} \qquad (13)$$

Equation 14 maps the deviation calculated in Equation 13 into quartertone intervals (24-TET):

$$folded\_dev_q(h) = 24\log_2(\frac{f_h}{hf_1}) \qquad (14)$$

[52] The colours represent the inharmonicity factors based on the seven combinations of overtones 1 and 8, 2 and 8, et seq.

Chroma in this research is calculated for a frequency range of $f_{min} = 130$ Hz to $f_{max} = 1400$ Hz, where signal is down sampled by a factor of 8 from 44100 Hz to 5512 Hz; thus for the different notes in this frequency range, the 3rd to 42nd overtones are considered in chroma calculations. **Table 12** shows the deviations of the first eight overtones when these components are folded over frequency components in one octave (similar to fundamental frequencies), assuming an inharmonicity factor of 0.00031s. It can be seen that higher overtones cause more deviations. However, their amplitudes are less and thus they do not affect the resulting components considerably. The effect of the deviation of folded higher harmonics from the fundamental frequency imposes a limit on finer tone resolutions. However, the extent of the effect of inharmonicity factor is limited, as most of the energy of the signal lies in the fundamental frequency and its first few harmonics, unless measures are taken in order to amplify the higher overtones.

Table 12 Deviation of the folded overtones 1–8 from the fundamental frequency

Overtone number	1	2	3	4	5	6	7	8
Deviation (quartertone)	0.0161	0.0429	0.0803	0.1283	0.1868	0.2556	0.3347	0.4240

5.3 Tonic detection

A tonic detection stage in pre-processing enables the analysis of test samples, irrespective of tonality.

Figure 14 shows a flowchart for *dastgàh* recognition, using chroma as the feature with a tonic detection and chroma alignment stage.

Here are two possible approaches for tonic detection:

1. Finding the tonic in the pre-processing step, aligning the features, and then identifying the scale, using either theoretical or data-driven templates.
2. Comparing the signal with templates of each *dastgàh* and their 23 shifted versions, and finding the tonic and scale at the same time, in manner similar to the chord recognition algorithm of Harte [6].

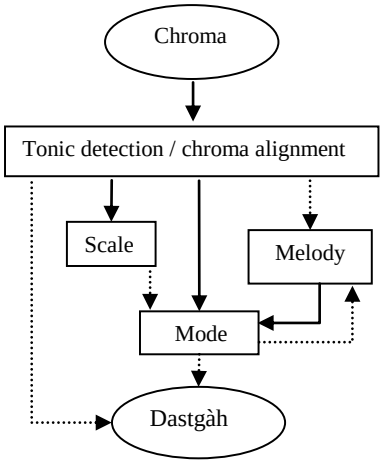


Figure 14 *Dastgàh* recognition with tonic detection flowchart

This section presents a tonic detection stage, based on dot-product prior to mode recognition; as a heuristic way to making a general template, all theoretical scales are transposed to the same tonic (first element) and summed up to make a general template which includes all possible intervals. This leads to a weight of 5 for the sum of tonics of 5 scales for example, and includes non-zero components at possible intervals of each scale (**Figure 15**).

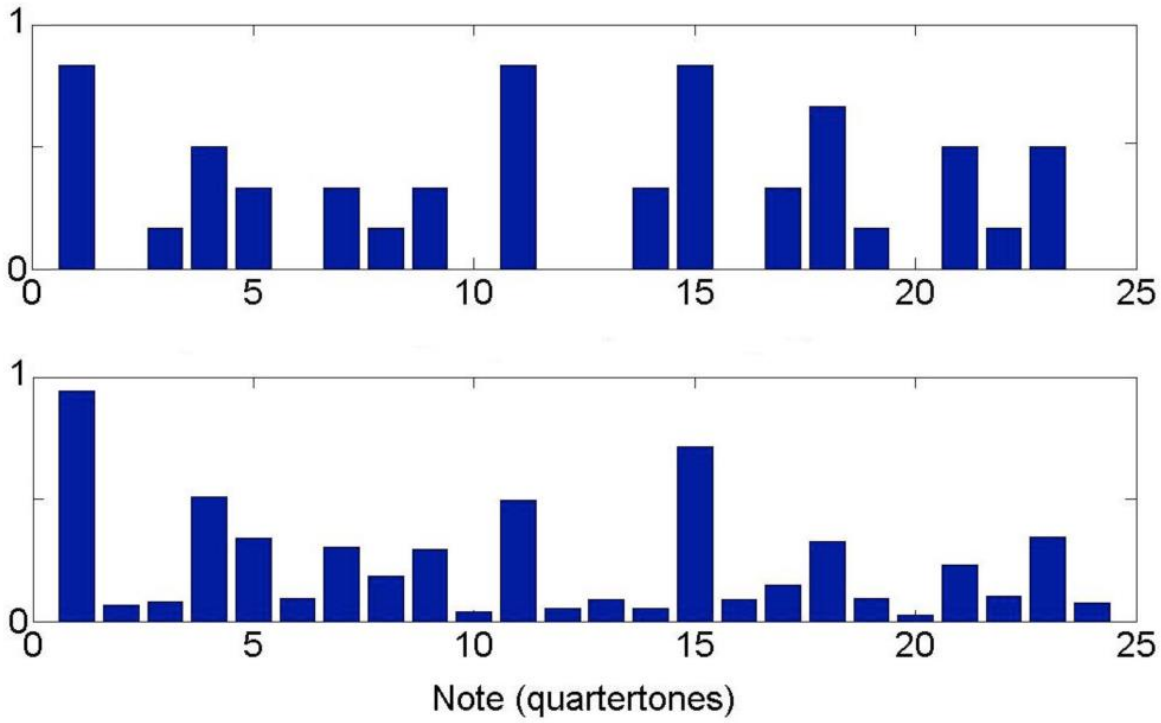


Figure 15 Sum of a) theoretical scales (**Figure 32**); b) training templates (**Figure 35**)

All intervals in all scales are included, magnifying the effect of the common notes of different scales, especially the most important notes: the tonic, the subdominant and the dominant. The chroma of the samples is shifted 23 times, and each time it is multiplied by the theoretical or data-driven templates (top and bottom plots in **Figure 15**). The sum of the resulting components reveals the tonic, based on which the test samples are aligned. Subsequently the scale is identified using a classification scheme, as explained in the previous sections.

6 Method

6.1 Introduction

Persian music is based on a modal system of 7 main modes and their 5 derivatives. Each mode is characterised by a particular pattern of scalar intervals, a modal tonic, a rest note, and melodic motifs. A mode generally falls into one of five tuning classes.[53] This chapter presents algorithms for mode recognition by means of identification of these tuning classes in an audio musical signal. Methods are implemented and explained on the basis of spectral averages, pitch histograms and chroma features. Subsequently, a classification scheme (geometric distance measures, Manhattan distance, dot-product, bit-mask, or cross correlation) or a generative method (GMM) is applied to the extracted features. The term 'mode sequence' refers to modulations, and applies to a full *dastgàh* performance, in a broader sense. A listener recognises a *dastgàh* in one or more of the following three ways:

- Perceptually: based on the emotions that a piece conveys or, more accurately, on the emotions associated with the mode. Emotions associated with a piece are usually culture-specific; for instance, a melody perceived as melancholic in one culture may arouse cheerful emotions in another; and lyrics too may have a bearing on this. However, there are common elements in music perception: for example, *dastgàh e màhur* in Iranian music and the major scale in Western music arouse similar general "happy" feelings, while *àvàz e esfehàn* and minor scale are "serious" or "sad", compared to *màhur* / major scale. The 'mood' of a piece is a multidimensional function of mode and rhythm, both generally-accepted and culturally-specific; and, in the case of vocal music, also of the lyrics.

- Through melody recognition: i.e. by comparing a piece with known patterns. If the listener knows that a melodic segment is in a mode, and recognises it in a track heard, it shows that the track is in the same mode. As the Persian *dastgàh* system is melody based, melody is the most accurate way of recognising a Persian mode. Both music experts and ordinary listeners can identify melodies.

- By identifying the notes: based on scale intervals, frequency of occurrence of notes, and their order. Only experts in Iranian music are able to do this.

[53] The tuning classes are: *homàyun* and its derivative *bayàt-e esfehàn*, *segàh*, *chàhàrgàh*, *shur* along with its derivatives (*abu'atà*, *bayàt-e tork*, *afshàri*, *dashti*) and *navà*, *mahur* and *ràst-panjgàh*.

A computational system is able to carry out the classification task, in the last two of the above. It is not usually possible to draw upon symbolic data (scores) when analysing orally transmitted ethnic music; otherwise the task of mode recognition would become much easier. In this research, musical knowledge, signal processing and machine learning techniques are used to estimate the mode of an audio musical file.

To have knowledge of the tuning and the *dastgàh* helps in improving transcription systems as well, because by knowing the *dastgàh* and scale intervals, the smallest interval of interest is a semitone rather than a quartertone[54]; and the size of quartertones will be established to some extent by knowing these.[55] Furthermore, it shows the notes that are not likely to occur in a *dastgàh*. The position of a quartertone note is required in audio compression and coding in order to compress or code a Persian audio signal; and it is an important factor in music synthesis, automatic music accompaniment, defining default tunings in music editors, and in music instruction. The mode itself can be directly used for music labelling, as it embodies many of the essential features of the *dastgàh*. **Figure 16** shows the five scales (tuning arrangements) in Persian music.

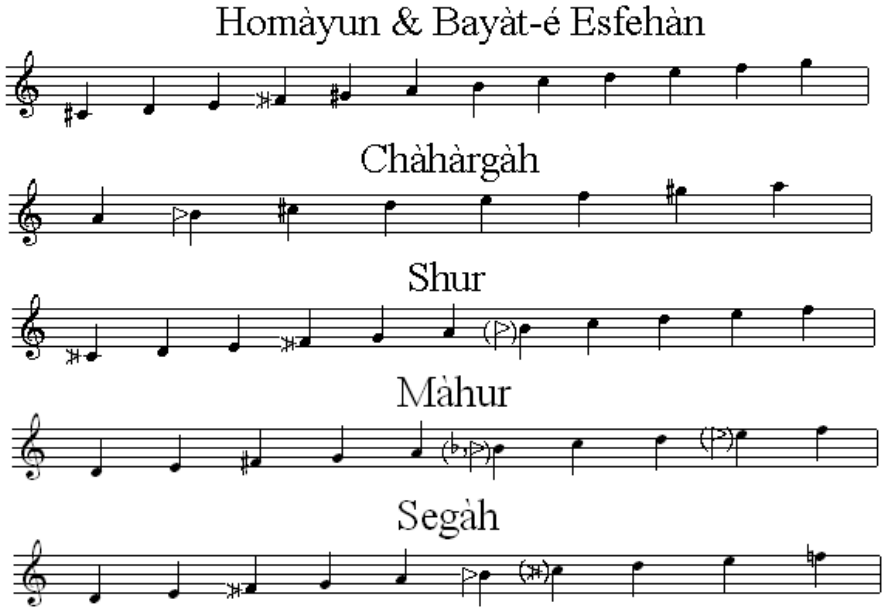


Figure 16 The five scales[56] on a 12-bridge santur

[54] The scale intervals consist of wholetones, three-quartertones and semitones (**Figure 16**).

[55] The interval size of a quartertone and the consequent position of the quartertone note depend on the mode, the melody and the performer's mood.

[56] The moving accidentals are shown in parentheses.

Algorithms developed to identify the mode through the tuning of a piece can completely analyse the pitch aspects of a *dastgàh* performance only in combination with other information, including the starting note, the order of notes, the rest note, knowledge of the notes not normally occurring in a mode, and melodic information.

6.2 Methodology

Generally, the *dastgàh* (main mode) and *àvàz* (derivative mode) of a piece can be identified perceptually by the listener via the culturally-specific emotions they carry, by the melody, or by one or more of the following observable features:

- The scale intervals, determined by the intervals used in the piece, although it must be recognised that there are different modes which share similar scales (Figure 1);
- The frequency of occurrence of the modal tonic and other scalar notes;
- The succession of notes;
- The rest note;
- Tone range (compass or ambitus).

It is important in *dastgàh* identification to focus on the beginning and the end of a performance, while taking into account modulations occurring between.[57] Problems that arise include the following:

- Some of the *dastgàhs* (which are principally distinguished based on melody rather than scale and modal tonic per se) cannot be determined directly by physical characteristics such as the scale intervals. In such cases, the mood of a piece and the melodies or lyrics serve to clarify it. For instance, some pairs of modes, which share the same scale and modal tonic, are distinguished one from another only by a characteristic melodic motif.

- Insufficiency of samples, perhaps occasioned by paucity of scale notes or brevity. If, for example, the notes A, D, E, and F, which are common to several modes, are played (Figure 16), it is not possible to estimate the mode.

In summary, in determining a *dastgàh*, changes in both scale and melody need to be followed, as shown in the following flowchart (Figure 17). Pitch as a feature can be used to obtain scale intervals and tone occurrences; alternatively, chroma provides a similar feature vector, inclusive of the effect of the harmonics; spectral average can also be used to identify the scale, the mode and the

[57] Note order is not considered here.

melody; finally, the melodic and modal sequence (taking account of modulations) represent the *dastgàh*. Although a melody's effect is indirectly reflected in the chroma averages, as the frequency of occurrence of notes, melody tracking is beyond the scope of this thesis, whose focus is on scale.

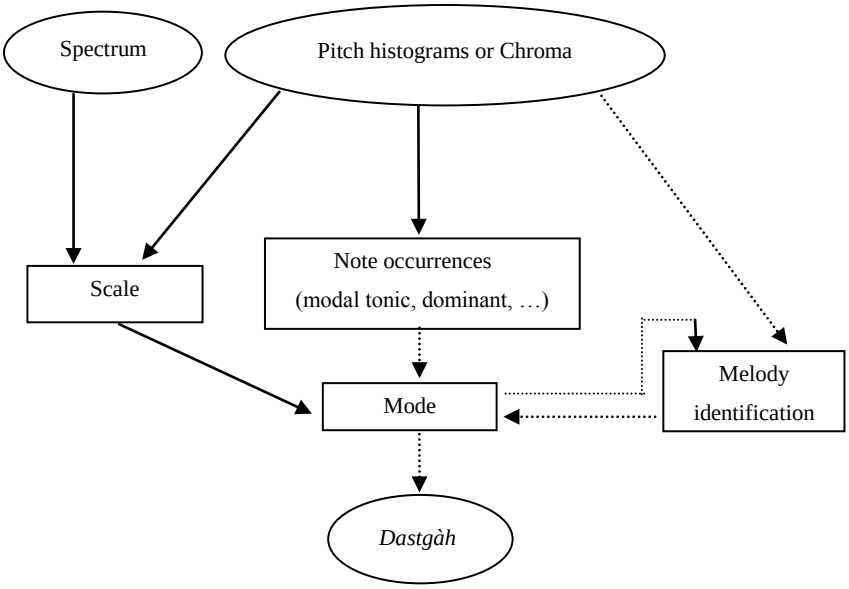


Figure 17 *Dastgàh* identification flowchart

The bidirectional arrow in Figure 17 shows that knowledge of the mode (along with pitch sequence) can be used to identify the melody, and melodic identification helps in identifying the mode. Here the spectrum, its simplified version (i.e. chroma), and pitch histograms are used to find the tuning and musical mode. Other possible ways, such as melodic similarity or knowledge of the notes that do not occur, are left for future work.

The task of classification involves two stages:

1. Extraction of one or more set(s) of features from audio musical files

2. Applying a classifier to those features

The choice of appropriate features is important as it affects both the computational cost and performance rate. Spectrograms, chroma and pitch histograms are popular features for mode classification.

Commonly accepted features and classifiers are reviewed in sections 6.3 and 6.4. They are described in dedicated sections, and will be revisited several times in the following chapters of the thesis, notably in works reviewed in Chapter 4 and in Chapter 7.

6.3 Features in Musical Signal Processing

Audio musical features can be classified into three categories, for the analysis of rhythm, pitch, and harmony. Low-level descriptors (time and frequency spectrum) and a mid-level feature (chroma; see Section 6.3.3) can be used to obtain high-level features, such as chords, keys, and mode.

6.3.1 Spectrogram and spectral average

Spectrogram and spectral average can be calculated through the following steps:

1. Calculate FFT and the power spectrum [61]

$$X(k) = FFT(x(n)w(n)) = \sum_{n=0}^{N-1} x(n)w(n)e^{-\frac{j2\pi kn}{N}}, \text{ k=0,…, N-1, n=0,…, N-1} \quad (15)$$

$$P_x(k) = \frac{X(k).X^*(k)}{N} \quad (16)$$

   Where n is the sample number in time domain, k is the sample number in frequency domain, N is the frame size, $x(n)$ is the framed time domain signal, $w(n)$ is an N-point Hamming window, $X(k)$ is the signal's spectrum, $X^*(k)$ is complex conjugate of $X(k)$ and $P_x(k)$ is the normalised power spectrum. The rows show the feature component and the columns show the observations.
2. As the signal is real, to avoid redundant calculations, the samples in the range of [1, Nf/2+1] are kept
3. The spectra can be averaged over all frames[58]:

$$Spec_{mean} = \sum_{i=1}^{m} P_x(k,i)/m \quad (17)$$

where m is the number of observations. It will also be divided over its mean value:

$$pattern = \frac{spec_{mean}}{mean(spec_{mean})} \quad (18)$$

[58] The covariance also needs to be calculated if Mahalanobis distance is to be used.

Figure 18 shows a short melodic motif played in *àvàz esfehàn* in staff notation, time domain signal, and spectrum. **Figure 19** shows the spectral average for the same melodic motif.

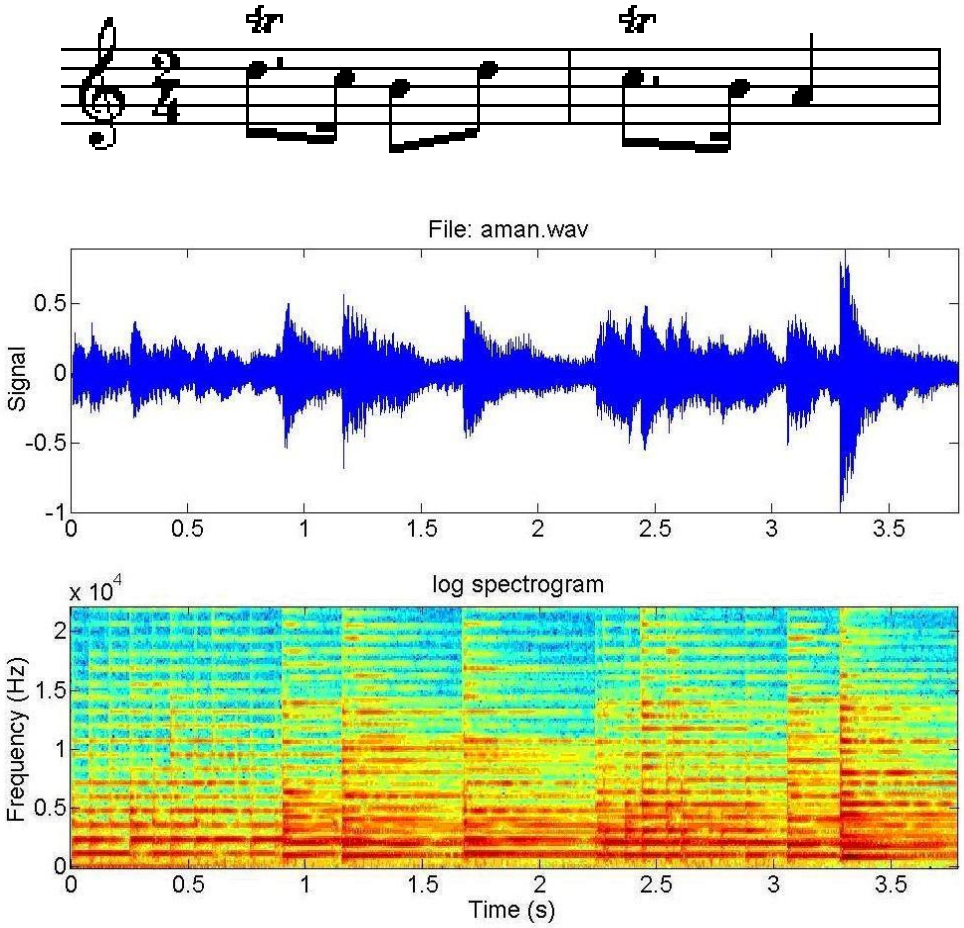


Figure 18 Scores, time domain signal and log spectrum of a motif in *esfehàn*

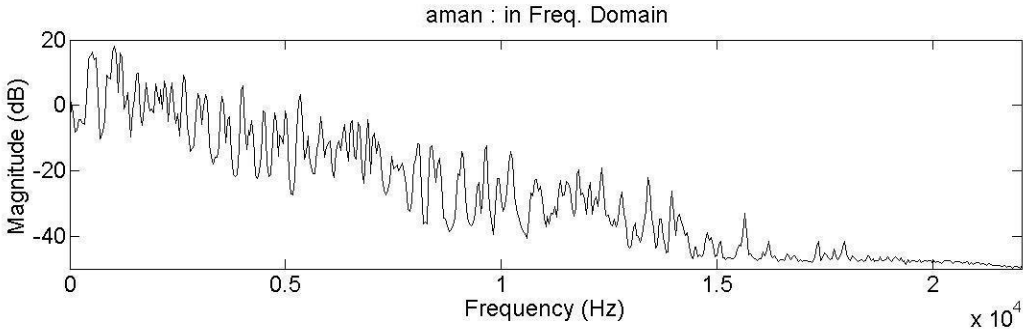


Figure 19 Spectral average of the motif in Figure 18

6.3.2 Pitch histograms

Pitch is an important perceptual property, corresponding to fundamental frequencies of notes. Pitch trackers are classified into three broad categories: time domain, frequency domain, and perceptual methods [66]. Different pitch trackers are compared in [5, 49, 65, and 66]. Pitch histograms (PH) are created by summing up the frequency of occurrence of notes.

Figure 20 shows the pitch contour of the motif in **Figure 18**, using a pitch tracker that is described in [5]. The pitch tracker is applied to the signal in **Figure 18** after a pre-processing onset detection step (described in Section 5.1.1); two alternative onset detection methods, energy and HFC, are compared. With this pre-processing, the pitch contour becomes smoother: most transients and the rapid trills are removed. In the case of the pitched-percussive sound of the santur, the HFC-based onset detection yields better results than the energy method.

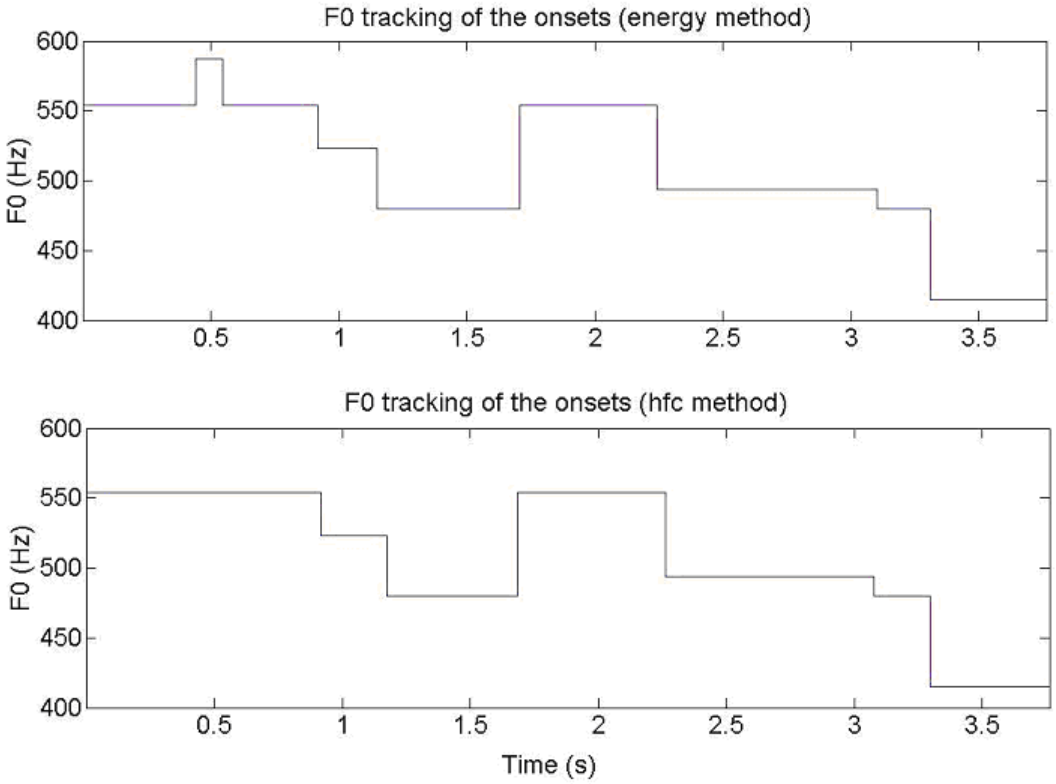


Figure 20 Pitch plot of signal in **Figure 18**, after onset detection (Energy and HFC)

6.3.3 Chroma

Chroma is a mid-level representation of signal, between spectrum and pitch: in comparison with the spectrum, it is less dependent on instrumentation[59], and it is robust to detuning, dynamics, and timbre [62].

The first step in creation of chroma is the constant Q-transform.

Constant-Q transform

Constant-Q [7] is a spectral analysis where the frequency domain components are logarithmically spaced. It is an alternative to Fourier Transform, where quality factor, Q of the constant-Q transform is kept the same[60].

$$Q = \frac{Centre\_frequency}{bandwith} \tag{19}$$

It is similar to ways of human perception and musical scales. Constant Q transform $X(k)$ of a signal $x(n)$ is defined as:

$$X(k) = \sum_{n=0}^{N(k)-1} W(n,k)x(n)e^{-j2\pi f_k n} \tag{20}$$

where length of the window $W(n,k)$ depends on the bin's position. The centre frequency f_k in the k^{th} bin is defined as:

$$f_k = 2^{k/\beta} \cdot f_{\min} \tag{21}$$

where $f_{\min}$ is the lowest frequency of interest and β is the number of bins per octave, representing the tone resolution. The Harmonic Pitch Class Profiles (HPCP), which are also called the chroma, are subsequently calculated by mapping the constant Q to one octave.

The effect of inharmonicity is masked by the bins of increasing width that are used in Q transform. **Figure 21** shows the log-frequency chromagram (top) and the chroma average or average pitch class of the melody in **Figure 18** (down). The log spectrum and chromagram show the logarithm of the intensity. The colour red represents high intensity, and blue represents low intensity. Pitch and mode progression over time are disregarded in chroma average.

---

[59] Instrumentation here means the instruments which are played in the audio recording analysed. Different instruments have different timbres; e.g. two santurs, and even different strings of a particular santur, have different timbres.

[60] The width of a frequency bin in constant-Q transform is proportionate to the centre frequency. The sampling rate does not affect this parameter. However, the higher the sampling rate is, the higher will the frequency resolution be: Fs/N, where Fs is the sapling rate and N is the frame size.

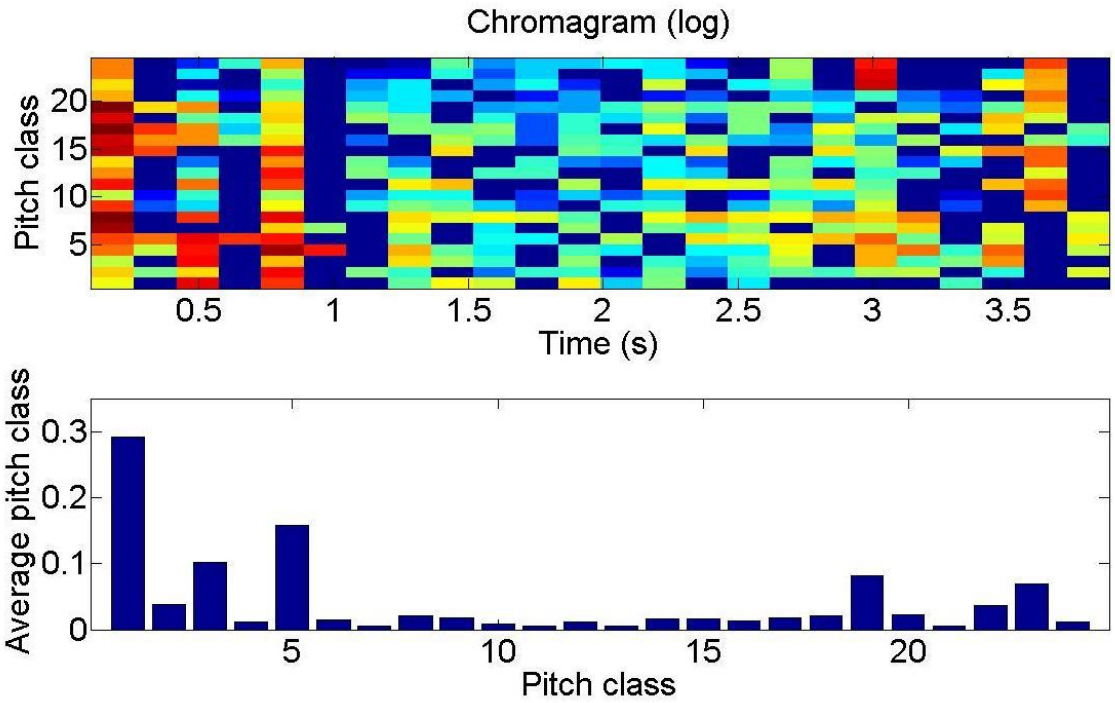


Figure 21 Chromagram logarithm and average chromagram of the motif in **Figure 18**

Alternatives to Q-transform

Chuan and Chew [63] present a variant of Q-transform, where a high frequency FFT is performed in the beginning and in the binning process elements close to the semitones are kept and merged, while other elements are removed. Also, extra weighting is applied to high frequency elements. Finally, octaves are summed up to produce the chromagram.

Peeters [64] presents another variant to Q-transform, in which a high-frequency FFT is performed and then a logarithmic merge is performed, with 3 bins per note. Only the centre bin of the three is kept, the others being deleted to reduce noise.

The Peeters and Chuan, and Chew variants are not appropriate for the non-Western musics (e.g. Iranian music) considered here, as the pitch system is not fixed, and precious information might be found in the removed elements, which are assumed to be redundant for the purpose of analysing Western music. Furthermore, in old recordings, such as tape or gramophone recordings (included in, for example db3 here), the pitches might not be very consistent, and there might be a spectral leakage.

Paus [45] performs a cubic interpolation of the spectrum in order to find a more accurate measure of the logarithmically spaced components of the chromagram than would be achieved by a simple summation.

Tuning the chromagram

After the chroma is derived through a 72-bin Q-transform, a peak-picking algorithm, based on quadratic interpolation, can be applied to find the exact peaks and their amplitudes. Finally, a tuning algorithm can use the peaks to find the tuning pitch of the audio samples [6]. For our purposes, chroma has these advantages over spectrum:

- That the order of the feature space is reduced from a frame size such as 32768 to 24, reducing the subsequent calculation costs[61].
- Chroma does not depend on octaves as the components are logarithmically spaced and the pitches are folded into one octave; whereas in spectral average, octave change affects the results, components of different octaves are treated separately and there is no one-to-one relation between them.
- Chroma is less sensitive to timbre; as the bins merge several frequency components, the effects of instrumentation, octave errors, harmonic content,[62] and inharmonicity factor[63] are reduced.

6.4 Classifiers

The classification schemes used here can be divided into two broad categories: the first including deterministic classifiers, Manhattan distance and dot-product operators; and the second, a machine learning method, the Gaussian Mixture Model (GMM). They are explained in the following two subsections and are applied to audio musical data in Chapter 4.

[61] Assuming a frame size of 32768 for the spectrum and a 24 temperament for the chroma.
[62] As all components are folded into one octave, the difference in the amplitudes of the harmonics is evened.
[63] Verification tests show that as three neighbouring bins are merged after the tuning step, the effect of inharmonicity is negligible.

6.4.1 Different distance measures

1. Minkowski's distance of different orders over an N-dimensional space is constructed by altering the power of the differences, m, and summing over the feature space:

$$dist_{Mink} = \left((x_1 - y_1)^m + (x_2 - y_2)^m + ... + (x_N - y_N)^m\right)^{1/m} \quad (22)$$

Minkowski's distance of order 1 is called the Manhattan or Taxicab distance:

$$dist_{Man} = |x_1 - y_1| + |x_2 - y_2| + ... + |x_N - y_N| \quad (23)$$

Minkowski's distance of order 2, is called Euclidean distance:

$$dist_{Euc} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_N - y_N)^2} \quad (24)$$

where N is the frame size and $x_1, \ldots, x_N$ and $y_1, \ldots, y_N$ are samples of the power spectrums.

2. Cross-correlation of the signal and the patterns can also be used for classification:

$$cc(m) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)y(n+m) \quad \text{m=0, 1,…, N-1} \quad (25)$$

where N is the frame size, n is the sample number, m is the lag, $x(n)$ is the time domain signal and $y(n)$ is the reference pattern. The value of the highest peak in the cross-correlation is compared to find the closest pattern to the signal.

3. Another option would be to multiply, point-by-point, the signal by theoretical templates (dot-product or bit-masking), as Izmirli used in his research [32].

6.4.2 A Generative method: The Gaussian Mixture Model (GMM)

A Mixture is a probabilistic model that describes a sub-population. The GMM is applied to classification tasks such as image recognition, text-independent speaker verification and Music Information Retrieval [33, 67, and 68].

Assuming that we have a set of feature vectors of order d,[64] the mixture density for mode, m is a weighted linear combination of k Gaussian densities:

$$p(c|\lambda_m) = \sum_{i=1}^{k} w_i^m \, p_i^m(c) \tag{26}$$

where the sum of the mixture weights is $\Sigma_{i=1}^{k} w_i^m = 1$, c is the d-dimensional chroma feature vector, and the probability density function (pdf) $p_i^m(c)$ is:

$$p(c) = \frac{1}{(2\pi)^{d/2}|\Sigma|^2} e^{\left[\frac{-1}{2}(c-\mu)^T \Sigma^{-1}(c-\mu)\right]} \tag{27}$$

where μ is the mean chroma vector, Σ is the d-by-d diagonal covariance matrix

The GMM parameters are estimated using the Expectation Maximisation (EM) algorithm:

$$\lambda_m = \{w_i^m, \mu_i^m, \Sigma_i^m\} \tag{28}$$

Assuming having the observation vector c_t, the likelihood that a set of N observations are in mode m is:

$$p(c|\mathrm{mod}\, e) = \prod_{t=1}^{N} p(c_t|\mathrm{mod}\, e) \tag{29}$$

where $P(c_t \,|\, \mathrm{mod}\, e)$ is a mixture of k multivariate Gaussians:

$$p(c_t \,|\, \mathrm{mod}\, e) = \sum_{n=1}^{k} p(n \,|\, \mathrm{mod}\, e).p(c_t \,|\, n, \mathrm{mod}\, e) \tag{30}$$

and $P(n \,|\, \mathrm{mod}\, e)$ is the prior probability of mixture n given the mode.

Applications of the features and classifiers explained in this chapter, as they appear in the works of other authors, are reviewed and discussed in Chapter 3; in Chapter 7, these features and classifiers are applied to Persian musical signals in order to classify audio samples, based on scale and mode.

[64] d=24 for a 24-TET.

Supervised and unsupervised learning

A supervised learning algorithm produces a classifier, based on labelled training data, where each model is fitted into one class (here, a musical mode) and the test samples are applied to the model and subsequently classified; an unsupervised learning algorithm operates on unlabelled training data. In the latter case, the number of classes would still need to be specified. Supervised and unsupervised learning are explained by Duda and Hart in [8]. GMM is normally an unsupervised classifier, where the number of classes is specified and all samples are classified automatically; here, however, as labelled training samples are already available, a supervised GMM is implemented. This reduces the calculation cost and possible errors in grouping the samples.

7 Evaluations

This chapter explains the tests undertaken and the optimised parameters arrived at. The following sections explain the samples and feature vectors used, the pre-processing steps performed, and the experiments undertaken. Results are presented, analysed and discussed. The accompanying DVD presents samples of the codes, in the MATLAB environment, for the methods used in this chapter (see Appendix III); the chroma feature vectors for db1; and the first five audio files of db1.

7.1 Samples

As explained in Chapter 5, the samples comprise mainly 5706 seconds (91 pieces) of music played on the santur[65], including patterns, melodies and improvisations in five Persian modes: *esfehàn*, *chàhàrgàh*, *shur*, *màhur*, *segàh*, and their derivatives (referred to as db1). Additional samples include 3806 seconds (16 pieces) of live performances by a small ensemble (db2); 1075 seconds (6 pieces) of old Greek pieces digitised from gramophone recordings (db3); 1948 seconds (11 pieces) of *kamàncheh* solo (db4); and 629 seconds (7 pieces) of piano solo studio recordings (db5).

7.2 Pre-processing

This section concerns pre-processing steps, silence and high-energy frame suppression, and normalisation, as explained in Chapter 5. The features, numerical tests, and results are then presented and discussed. As, apart from modulations to derivative modes of a *dastgàh*, change of tonality rarely occurs during a classical Persian performance, in sections 7.2.1 to 7.2.4 it is assumed, unless otherwise stated, that there is one modal tonic throughout a piece.

7.2.1 Silence suppression

Any frame with energy less than 10% of the mean energy of the frames (which are mostly noisy frames) is eliminated in the silence suppression stage. As the santur has a relatively sustained decaying sound (a note can be heard for a few seconds after it is struck), there are not many frames with energies under 10% of the mean energy within the database, and silence suppression with such a threshold (explained in Sections 7.3.1 and 7.3.3) does not greatly affect the results. This

[65] The non-struck santur strings were not dampened.

parameter could affect the results if the samples were to include silent frames or frames with energies below 10% or a higher alternative threshold.

7.2.2 High-energy frame suppression

Any frame with energies over a certain threshold, such as 90% of the mean energy, is opted out. In this way, transients at the onset are removed. For each database, tests have to be carried out to find optimum values of the silence and high-energy suppression thresholds.

7.2.3 Normalisation

The frequency domain frames are normalised to reduce the influence of note dynamics. Even if the frames have been normalised before silence and high-energy frame suppression, they need to be normalised again afterwards.

7.3 Experiments and results

Training and test processes

Each of the recognition methods discussed in this chapter involves a training process in which, using part of db1, the template or models are trained. During the subsequent test process, samples are gauged versus the trained patterns or models. The test samples may or may not include the training samples, as explained in the following sections.

Training and test samples

To increase variety, the files 1–5 intentionally included scale notes, arpeggios and simple melodies for training, and in several experiments, either the rest of the files or all of the files are used. In another case, the mean of all the files of each class of db1 is used for training; however, in order to investigate the effect of the amount of training data, three separate experiments are undertaken using gradually increasing sub-parts of the files 1–6, 6–10, and 8–15 for training in (**Figure 26**). (The training samples were not used in these experiments.) The mean of these three experiments shows the effect of using different samples for training. Finally, theoretical templates, based on the scale intervals and symbolic data, which are independent of the training samples, are used in some further experiments.

Features

Spectrum, chroma, and pitch histograms, which were used and compared as features in the MPhil thesis, are used in a similar way in this research, where the parameters are optimised anew. The dimensionality of spectrum, the most straightforward of these features, is high, and it is dependent on instrumentation and timbre. If a perfect pitch tracker could be obtained, pitch, which corresponds to musical notes, would be the ideal feature; however, in practice, errors which arise in pitch tracking are passed on to the mode recognition stage, and this is particularly notable when more than one note is played simultaneously. Chroma (see Section 6.3.3), a simplified version of the spectrum, which has pitch classes as its elements (in which the components are folded into one octave), is more appropriate for polyphonic music [37].

Classifiers

Two approaches are implemented here:

1. Geometric distance measures, such as Manhattan distance and cross-correlation, classify the samples (based on the distance between the average of the features and a set of training templates that are created for each class), as was demonstrated in the MPhil research; and in this new research, a dot-product operator is also applied; comprehensive tests are performed using the geometric distance measures, and their different parameters are optimised.
2. A new approach based on machine learning is applied. Machine learning models are particularly useful classifiers when the data is multidimensional or boundaries of the classes are not known. The Gaussian Mixture Model (GMM)[66], is usually used in text-independent speaker verification, which is comparable to musical mode identification. GMM models are able to adapt the chroma to fit to boundaries of each mode, thus enabling first-order melodic progression through time.

Geometric models, with average chroma as the feature, consider the templates as points in a 24-dimensional space. Machine learning methods are able to learn the model from the relationships between the data. As every frame, rather than an average, is included, progress over time is taken into account.

[66] GMM is a Bayesian model, a one-state HMM.

7.3.1 Spectral average tests

The spectral average recognition system is described in the following steps:

- Mono, 16-bit samples; sampling rate Fs=44100 Hz
- Windowing: Hamming window of variable lengths
- Spectral average is calculated (as explained in Chapter 6)
- Reference patterns are calculated for each mode class during the training process
- A similarity measure compares the spectrum and a set of reference patterns.

Figure 22 shows the spectral average of a random sequence of notes in *dastgàh-e esfehàn*. Although spectral averages of the other two *dastgàhs* look very similar to this to the eye, the numerical calculations reveal their differences.

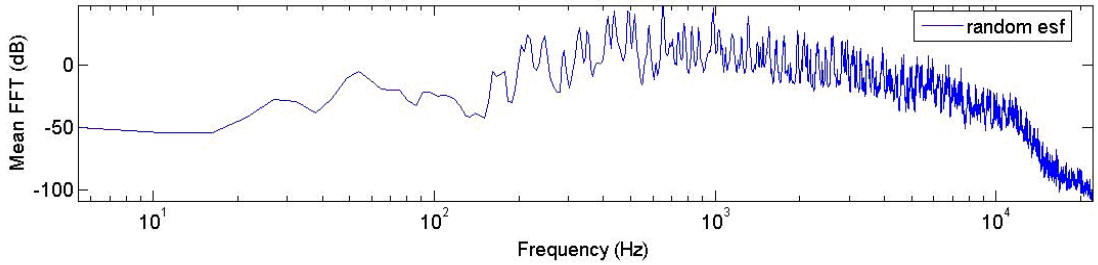


Figure 22 The spectral average for *dastgàh e esfehàn*, using a random sequence of notes

Five samples per *dastgàh* (**Tables 7–11**, Chapter 4) are used for training, and the average spectrum of these samples sets a pattern for each class. Test samples are gauged in comparison with reference patterns of each *dastgàh*. Manhattan distance (Equation 23) is used to classify the samples into five classes. In [5] it was shown that Minkowski's distance of order 1 outperforms higher orders of the distance measure and cross-correlation; the tests in this thesis are performed both with and without using the training samples. With a frame size of Nf= 32768 (743ms) and with the training samples included in the tests, recognition rate is 87.91%; but if the training samples are not included, the recognition rate drops to 83.33%. If dot-product is used instead of Manhattan distance, the recognition rates are 72.53% when training samples are included and 68.18% when they are excluded.

Detailed results are shown in confusion matrices[67] in **Table 13** and **Table 14.** The errors in both cases are the same. The numbers in each row show how many samples of each *dastgàh* belong to either of the classes in each column. The row headings are the actual modes, and the column headings are the modes as identified by the system. For example, in **Table 13**, 15 out of 17 samples in *esfehàn* are recognised truly as being in *esfehàn*, one is misclassified as a *shur*, and one as a *segàh*. *Esfehàn* and *shur* differ in just one fixed accidental: G# for *esfehàn* and a moving accidental, Bq for *shur* (see **Tables 7** and **11**). The confusion between *shur* and *esfehàn* is attributable to the recurrence of predominant shared notes: notes common to the two *dastgàhs* were played more in the misclassified samples, or the differentiating notes could have been played more quietly, and thus have been covered by other tones. Here, out of 17 and 12 *esfehàn* samples in **Tables 13** and **14** respectively, one (a passage) is classified as a *shur*, and the other (the song *Àmàn*) is recognised as a *segàh*. As *esfehàn* and *segàh* are very different (they differ in a moving accidental, G# for *esfehàn*, and fixed and moving accidentals Bq and Cs for *segàh*), the latter cannot be explained in terms of similarity of scales. One reason for this misclassification is that a note common to the two scales (D in *Àmàn*[68]) is played several times as a tremolo, and the last note (A) is likewise common to the two scales. This misclassification does not reveal a hidden perceptual similarity between the two modes. The scale of *chàhàrgah* differs distinctively from other modes, and it is noteworthy that that all *chàhàrgah* samples are correctly recognised.

Three *shur* samples are recognised as *esfehàn*; and two *segàh* samples as *chàhàrgàh*. Although *segàh* and *chàhàrgàh* differ significantly, they are perceptually similar, in such a way that most *segàh* pieces can be played in *chàhàrgàh* and vice versa. This is interpreted as a hidden similarity between these two modes. Although the scales differ, the frequency of occurrence of the notes they share is similar.

That three *segàh* samples are recognised as *shur* is due to the similarity of their scales; a fixed accidental for *segàh* (Bq) is a moving accidental for *shur*, and they share a moving accidental Cs[69]. Also significantly shared between *shur*[70] and *segàh*[71] are the Fs and F. Two of these three samples consist of randomly played notes; the other is a metric instrumental improvisation.

---

[67] A confusion matrix has rows representing the different classes and columns corresponding to the classification results, showing the number of samples which were recognised correctly or were confused with another class.

[68] Only a short, 5-note melodic motif from the beginning of the song is used in the test (**Figure 18**).

[69] In the Westernised version of *shur* or *segàh* (in which quartertones are rounded to the nearest 12-TET semitone), Cs is played as C#.

[70] The lower F is *sori* (Fs) and the higher F is natural.

Table 13 Confusion matrix for spectral average, first experiment: Classification using spectral average, including training samples

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	15	0	1	1	0
Chà (A)	0	15	0	0	0
Sur (E, A)	3	0	19	0	0
Sga (Fs)	0	2	3	10	0
Mhr (G)	0	0	0	2	20

Table 14 Confusion matrix for spectral average, second experiment: Classification using spectral average, excluding training samples

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	10	0	1	1	0
Chà (A)	0	10	0	0	0
Sur (E, A)	3	0	14	0	0
Sga (Fs)	0	2	3	5	0
Mhr (G)	0	0	0	2	15

Table 15 Confusion matrix for spectral average, third experiment: Classification using spectral average, including all samples for training

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	15	0	1	1	0
Chà (A)	0	15	0	0	0
Sur (E, A)	0	0	22	0	0
Sga (Fs)	0	0	0	15	0
Mhr (G)	0	0	0	0	22

Two *màhur* samples are recognised as *segàh*, which cannot be justified in terms of scalar similarity, other than in that a fixed accidental Bq for *segàh* is a moving accidental in *màhur* (*shekasteh*

[71] The lower F is *sori* (Fs) and the higher F played in *mokhàlef* mode, is natural.

derivative); this confusion refers either to a hidden similarity between the two modes, which the machine has captured; or it is possible that their common notes, such as the 2nd scale degree of *segàh* which is the tonic of *màhur* (G), are played so many times (as in *esfehàn-segàh* confusion) that they have overwhelmed the differentiating notes. It is interpreted as a common source of error.

In a third experiment, in which the means of all the samples in each *dastgàh* are used to build a pattern for that *dastgàh*, the recognition rate becomes 97.80% (**Table 15**). In this case the only confusions are between two *esfehàn* samples, which are recognised as a *shur* and a *segàh*. From the third experiment, where all test samples are used in training, it can be learnt that it is distinctly worthwhile, in addition to scales and basic melodies (which are included in the first few training samples), to use longer files including a variety of songs and improvisations, which improve the pattern of each *dastgàh*.

Silence suppression of frames with energies lower than 10% of the mean does not affect the performance rate, as most of the frames in the case of the santur signals used include higher energies, however, if frames with energies over 90% of the mean are removed, the recognition rates for the cases with and without using training samples become 90.11% and 86.36% respectively. The results remain the same when frames with energies over 95% of the mean are removed. Detailed results are shown in **Tables 14** and **15**.

Comparing **Tables 13** and **16** (i.e. without and with silence and high-energy suppression), it can be seen that the nature of errors is changed by the introduction of silence and high-energy suppression: one error for *esfehàn* is moved from *shur* to *chàhàrgah*; three confusions of *shur* with *esfehàn* are now disappeared; two *segàh* samples misrecognised as *chàhàrgàh* are now misrecognised as *shur;* and the *màhur* results remain the same.

Therefore it can be concluded that the removal of high-energy transients improves the recognition results, changing the nature of errors which arise: it decreases the chance of confusion between *shur* and *esfehàn*, while, conversely, there are elements in the removed frames (and transients) that would help in distinguishing between *chàhàrgah* and *esfehàn*. On balance, the removal of high-energy transients is recommended, as the overall performance is improved.

If the overall means of all test samples are used for training, the performance becomes 97.80%, where the only confusions are an *esfehàn* sample recognised a *segàh* and a *segàh* recognised as a *màhur* (**Table 18).**

Table 16 Classification results using spectral average with SS & HES, including training samples

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	15	1	0	1	0
Chà (A)	0	15	0	0	0
Sur (E, A)	0	0	22	0	0
Sga (Fs)	0	0	5	10	0
Mhr (G)	0	0	0	2	20

Table 17 Classification results using spectral average with SS & HES, excluding training samples

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	10	1	0	1	0
Chà (A)	0	10	0	0	0
Sur (E, A)	0	0	17	0	0
Sga (Fs)	0	0	5	5	0
Mhr (G)	0	0	0	2	15

Table 18 Classification results using reference spectral averages, including the training samples

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	16	0	0	1	0
Chà (A)	0	15	0	0	0
Sur (E, A)	0	0	22	0	0
Sga (Fs)	0	0	0	14	1
Mhr (G)	0	0	0	0	22

As explained above, the scales of *segàh* and *màhur* are very different. The performance rate is equal to the similar case in **Table 15**, where silence suppression and high-energy suppression were not performed. However, here the removal of high-energy frames that included transients has changed the nature of on of the errors: a *segàh* to *màhur* confusion in **Table 18** replaces an *esfehàn-shur* confusion in **Table 15**. One reason for this could be that at the onsets, almost all of the strings

of the santur are vibrating through the vibrations of the resonance body; and another could be the lack of enough samples of the differentiating pitch classes.

It is noteworthy that apart from two confusions in **Table 13**, there is no confusion between *segàh* and *chàhàrgàh*, which are perceptually the closest *dastgàhs* among these five. That musicians can play most *segàh* pieces in *chàhàrgàh*, with a scale change, where the pieces are still acceptable, shows how machine and human errors can differ in nature.

In the experiments of the next stage, the parameters are varied to see their effects on the performance rate. The effects of using different distance measures, different frame sizes, and variable amount of training data are investigated. The performance versus frame size is calculated for N_f =512, 1024, …, 32768 samples (11.6 ms – 743 ms). Manhattan distance and cross-correlation are used as the classifiers (Section 6.4.1).

Manhattan distance yields a maximum recognition rate of 90.11% at a frame size of 32768, when the first 5 files are used for training; their lengths are 197.6 s, 158.1 s, 140.0 s, 145.6 s and 107.8s for the 5 *dastgàhs* respectively. The training data is used in tests, and silence and high-energy frame suppression are performed. The frequency resolution is F_s / Nf = 1.35 Hz. The smallest interval of interest for these five *dastgàh*s is F3-Fs3, assuming a 24-quartertone scale: (179.73-174.61)/2=2.56 Hz, which supports the choice of Nf =32768 (743 ms).

The effect of frame size can be seen in **Figure 23** and **Figure 24**. A Manhattan distance and a cross-correlation are used, in two cases: with and without silence suppression and high-energy suppression. In these figures 'HS' indicates both high-energy and silence suppression. **Figure 23** shows the case where no part of the test data is used in training, **Figure 24** shows the case where part of the training data is used in tests (re-substitution). Solid lines show where Manhattan distance is used and dotted lines show where a cross-correlation classifier is used.

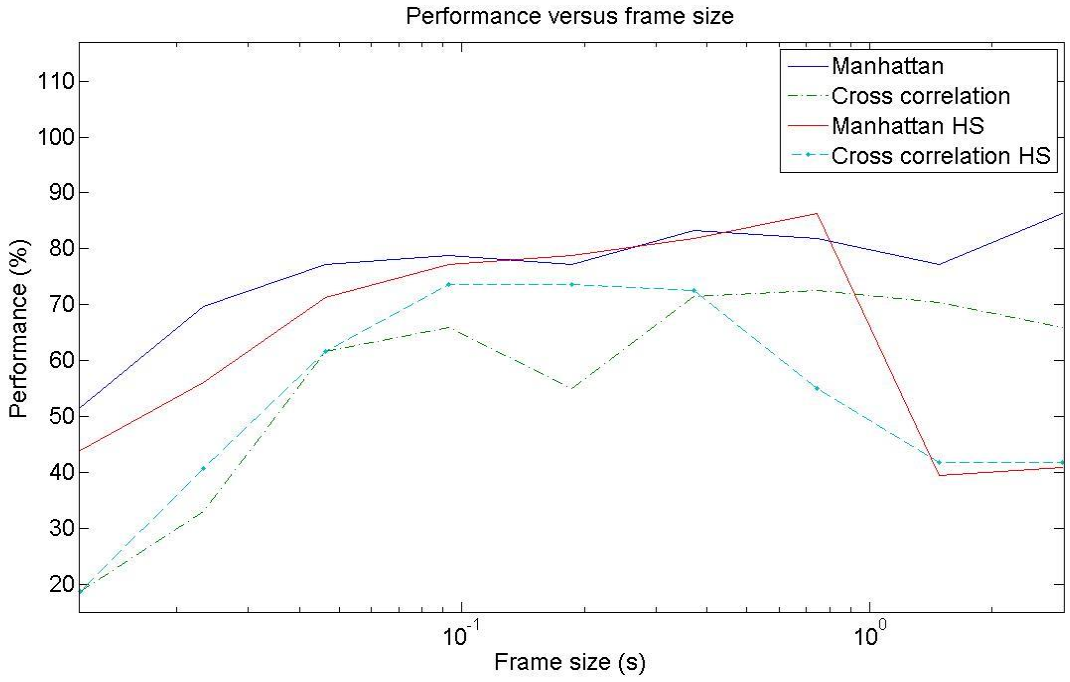


Figure 23 Performance vs. frame size with and without silence and high-energy suppression (HS), training data excluded

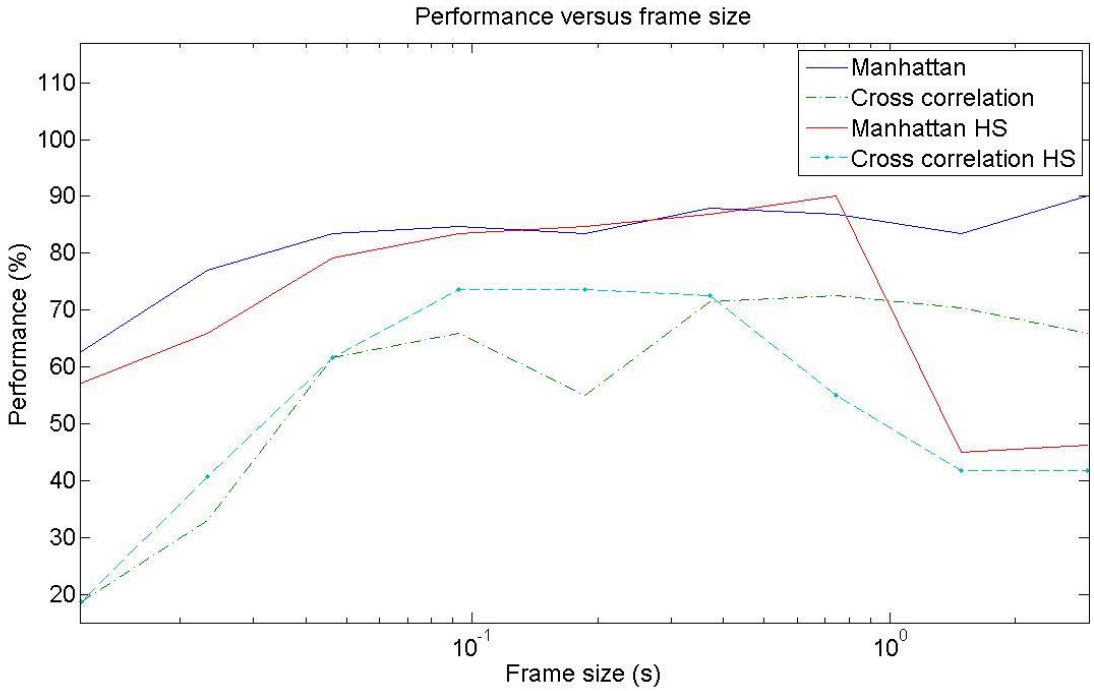


Figure 24 Performance vs. frame size with and without silence and high-energy suppression (HS), training data included

Table 19 shows that silence and high-energy suppression, the choice of classifier and also whether part of the training data is used in tests, all affect the optimum frame size. The maximum performance in each row is rendered in bold type. The maximum recognition rate when the training data is not used in tests is 86.36%, which happens either at a frame size of 131072 samples or, if silence and high-energy suppression are performed, at a frame size of 32768 samples (at a lower frequency resolution). With silence and high-energy suppression, increasing the frame size further reduces the recognition rate, whereas when the training files are not used (top row), the performance improves by increasing the frame size.

In the next stage of the experiments, performance versus the amount of training data is calculated based on 0 to 354.1 s of the first 8 files of the database (15617733 samples) at a frame size of 32768 samples (**Figure 25**). The training files for each *dastgàh,* are put together in series, one after another. In the case of this database (db1), it can be seen that at least 81.5 s of data is needed to achieve a recognition rate of around 86.8%. With more than 262 s of training data, no further change in performance occurs and it stays at 89.0%. It should be noted that the required amount of training data also depends on the tempo, as fast tempo music contains more data per unit of time.

The performance curve changes by using different sets of training data. For example, **Figure 26** compares the performance versus amount of training data when using 0–151 s of files 1–6, 6–10, 8–15 and the average of these three cases, which is a smoother curve. The average at the bottom shows that over 65 s of training data is needed to produce a recognition rate of around 83.2%. The maximum recognition rate over the average curve of the three cases is 89.4%. It can be seen that the performance depends on the training samples.

Figure 27 shows performance versus the amount of training data, with silence and high-energy suppression. Over 45.3 s of training data would be needed to result in a recognition rate of around 81.0%. The maximum recognition rate over the average curve is 83.9%.

Most of the notes were played in the middle octave of santur (around 277.2 Hz – 698.5 Hz), and most of the energy is concentrated in frequencies below 5 kHz. A test was performed to determine the effect of frequency range on performance; and the signal could be high-pass filtered, to amplify the effect of the low-amplitude high-frequency components, and low-pass filtered to avoid the effect of unnecessary low frequency components.

Table 19 Performance versus frame size (%), using spectral average

Nf (samples)	512	1024	2048	4096	8192	16384	32768	65536	131072
Training files not used Manhattan	50.00	71.21	77.27	78.79	77.27	83.33	81.82	77.27	**86.36**
Training files not used Cross-correlation	18.18	28.79	56.06	57.58	50.00	**65.15**	**65.15**	60.61	59.09
Training files not used SS, HS[72], Manhattan	37.88	48.48	59.09	74.24	78.79	81.82	**86.36**	39.39	40.91
Training files not used SS, HS, Cross-correlation	18.68	39.39	56.06	**65.15**	**65.15**	63.64	50.00	34.85	28.79
5 training files used SS, HS, Manhattan	47.25	60.44	70.33	81.32	84.62	86.81	**90.11**	45.05	46.15
5 training files used SS, HS, Cross-correlation	18.68	40.66	61.53	**73.63**	**73.63**	72.53	54.95	41.76	31.87
5 training files used Manhattan	61.54	78.02	83.52	84.62	83.52	87.91	86.81	83.52	**90.11**
5 training files used Cross-correlation	18.68	32.97	61.54	65.93	54.95	71.43	**72.54**	70.33	65.93
Means of all training, Manhattan	81.32	92.31	93.41	94.51	94.51	97.80	97.80	**98.90**	96.70
Means of all training used, Cross-correlation	24.18	34.06	48.35	52.75	49.45	54.95	56.04	**62.64**	58.24
Means of all training used, SS, HS, Manhattan	84.62	89.01	93.41	96.70	97.80	97.80	97.80	**98.90**	96.70
Means of all training used, SS, HS, Cross-correlation	25.27	45.05	49.45	54.95	60.44	**72.53**	74.73	62.64	58.24

[72] SS and HS (HES) are abbreviations of silence and high-energy suppression respectively.

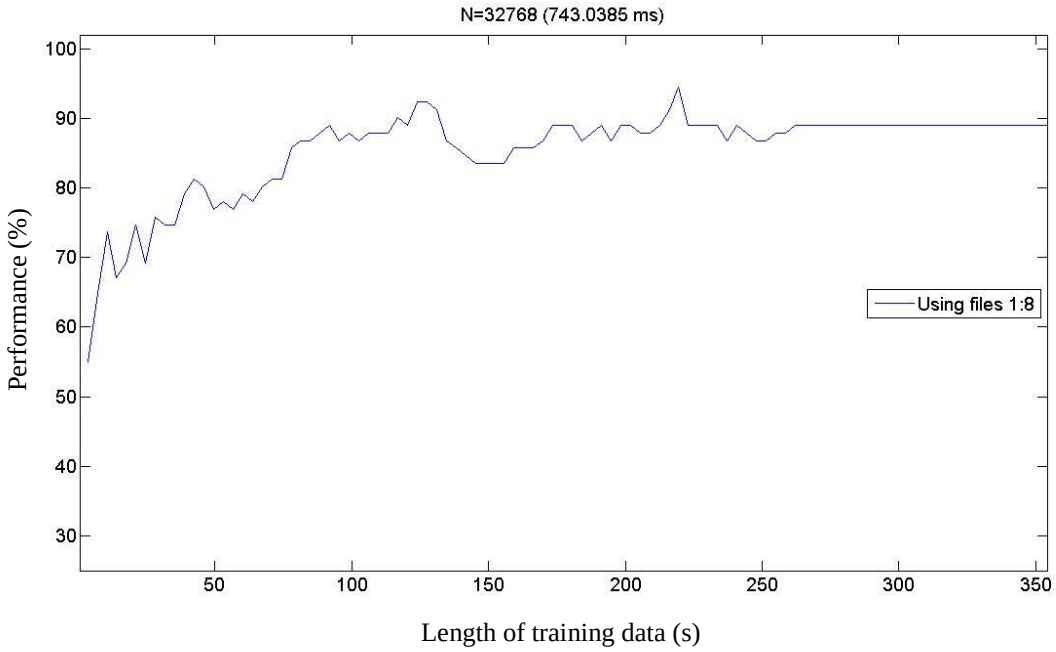


Figure 25 Performance versus the amount of training data (from 0–354.1s of the first 8 files of db1)

Figure 28 shows the recognition rate versus frequency range, where the training samples are excluded from the tests. The lowest frequency starts from the first element (i.e. 1.3 Hz) and the upper bound is variable.[73] For instance, keeping 1292 Hz of the spectrum results in a recognition rate of 78.8%. A significantly higher recognition rate of around 86.4% is achieved if at least 3794 Hz of the frequency range is kept. Thus several harmonics are needed to get to the maximum recognition rate. Interestingly, the very low frequency section of the data (under 40.4 Hz) should have captured some characteristics of the scale, so that the minimum recognition rate is around 30.0%, that is, slightly higher than the equal chance of 20% that would be expected for each of the five classes. The reason could be due to superposition of other components appearing in the low-frequency region. In particular, quartertone intervals of less than 10 Hz contribute to improved performance, even when just using very low frequency components.

[73] Steps (hop-size) are 10 samples, or (10/32768)*44100=13.6 s.

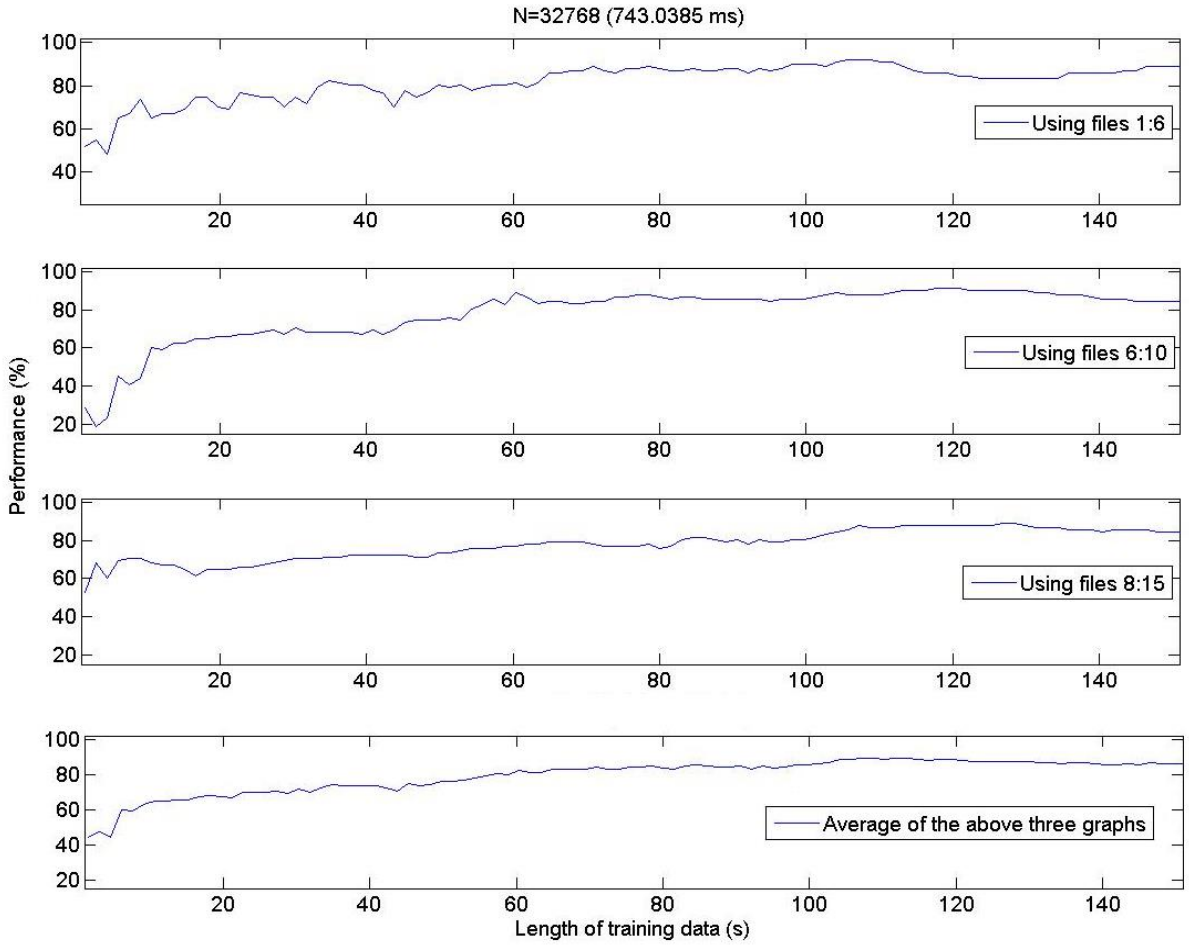


Figure 26 Performance versus amount of training data (from 0–151 seconds of files 1–6, 6–10, 8–15 of db1, and average of these three graphs)

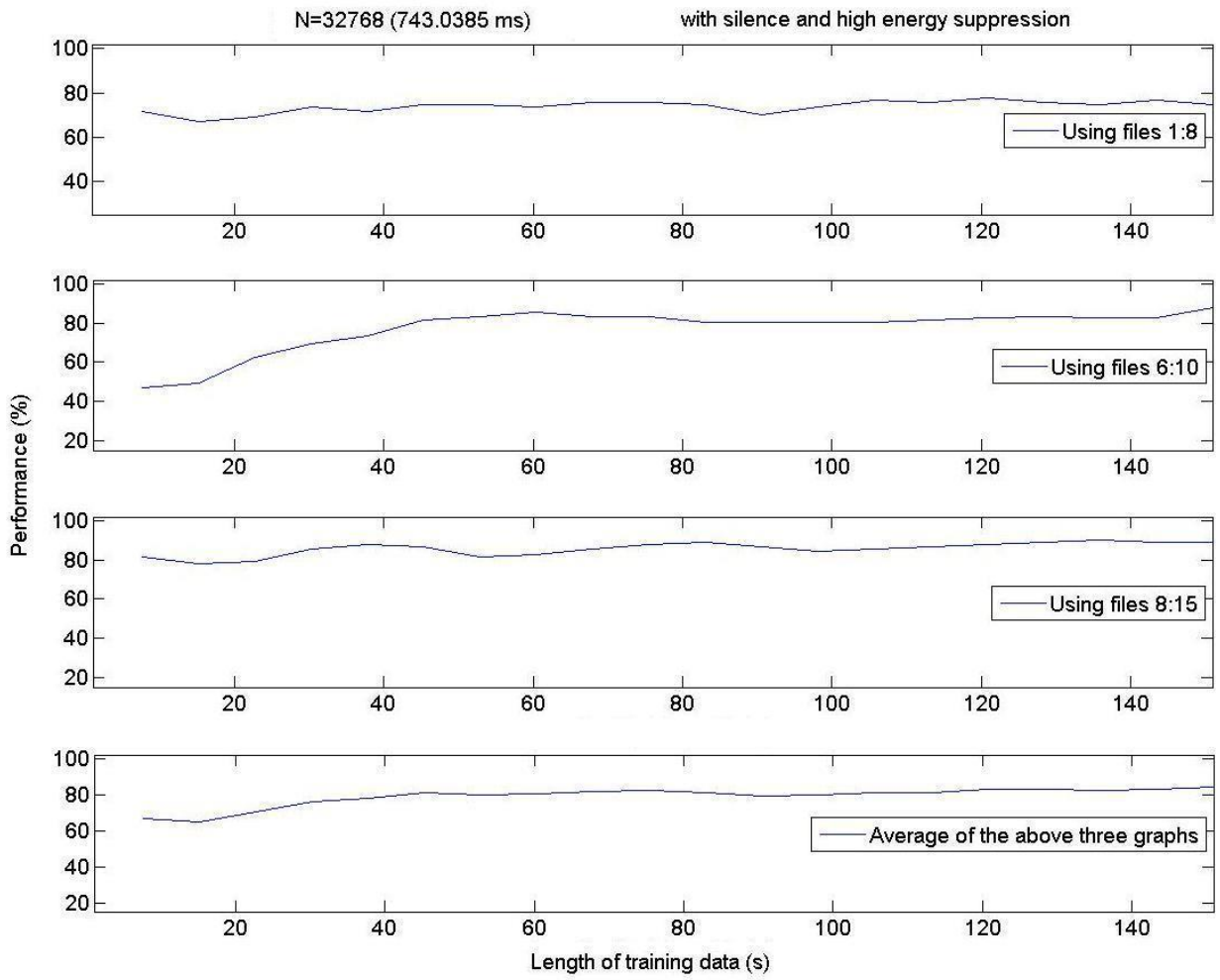


Figure 27 Performance versus amount of training data (from 0–151 seconds of files 1–6, 6–10, 8–15 of db1, and average of these three graphs), with silence and high-energy suppression

In another test, the first 1000 samples are removed and the lower frequency bound starts from 1345.8 Hz[74] (**Figure 29**). It can be seen that an upper frequency bound of 3015 Hz or more is required to achieve a recognition rate of 68.2%, but with more data, this drops, up to 5250 Hz that a recognition rate of 77.3% is achieved. An upper frequency bound of over 11600 Hz of the spectrum results in a stable recognition rate of 71.2%. The recognition rate starts decreasing, for this dataset, from an upper frequency of 6085 Hz.

In the third experiment on the frequency range, the upper frequency bound is held constant at 6085 Hz and the lower bound is varied (**Figure 30**). At a lower frequency bound of 458.9 Hz, a

[74] (1000/32768)*44100=1345.8 Hz.

maximum performance of 89.4% is achieved, which is even higher than was the case with a full frequency range. From this point upwards, the performance decreases.

It can be concluded that there are unwanted elements in very low and very high frequency contents of our dataset. Thus, to reduce the amount of calculation, the frequency range can be limited to 458.9 Hz–6085 Hz to include the necessary harmonics, which were investigated in this thesis. The values are obtained for the main part of the database of santur files in 5 *dastgàh* (db1). To have a better understanding of the frequency range, more tests should be performed on a more diverse database.

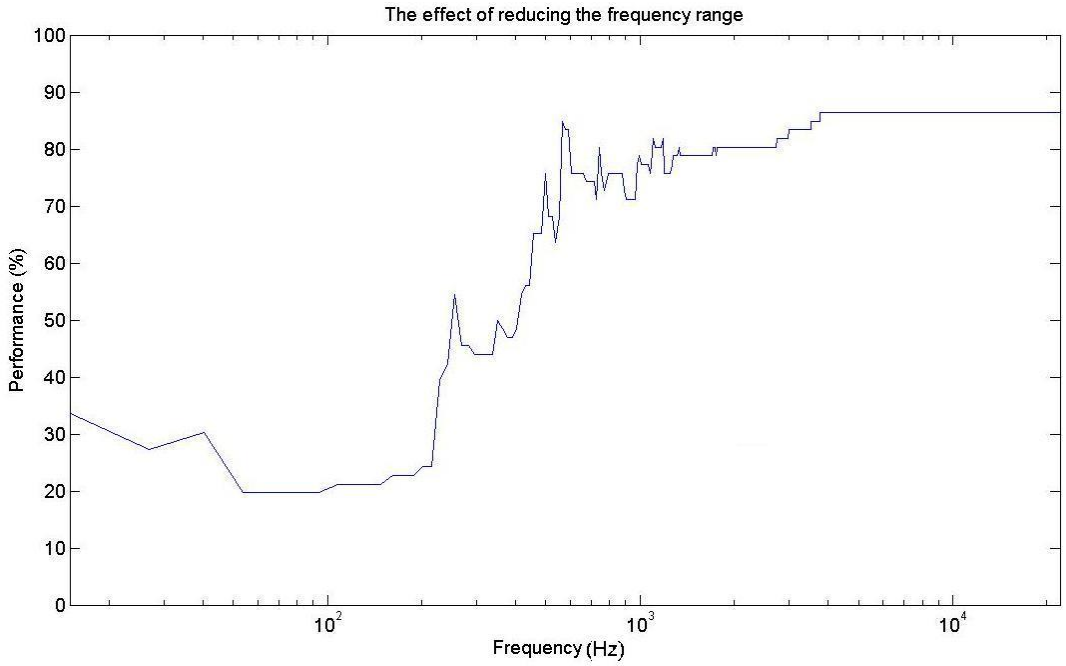


Figure 28 Effect of reducing the frequency range (upper frequency: 1.3 Hz–22050 Hz), excluding the training data, with silence and high-energy suppression

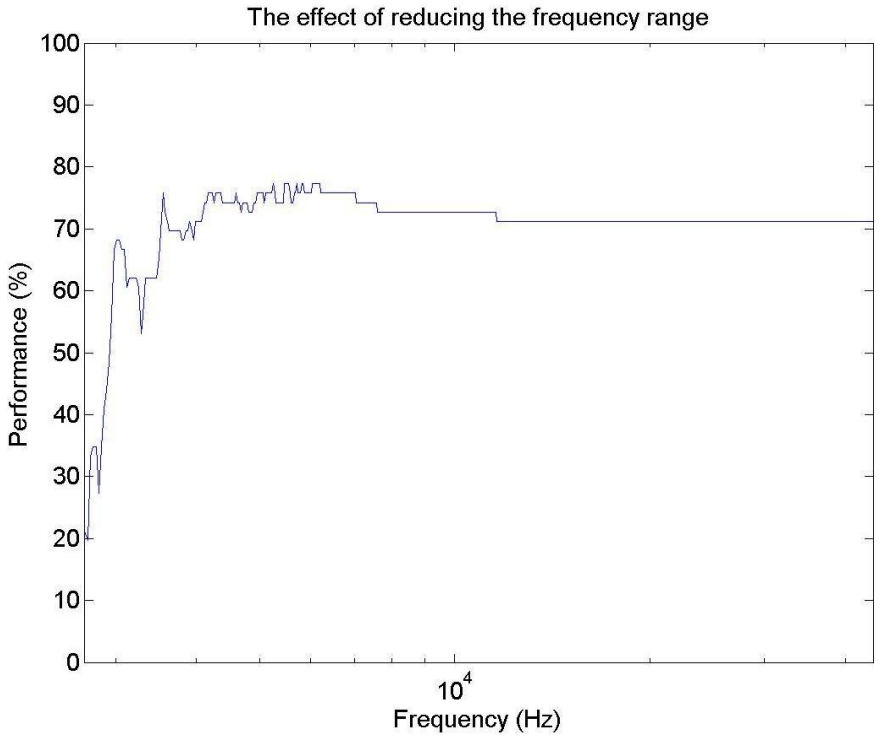


Figure 29 Effect of reducing the frequency range (lower frequency: 1345.8 Hz–22050 Hz), excluding the training data, with silence and high-energy suppression

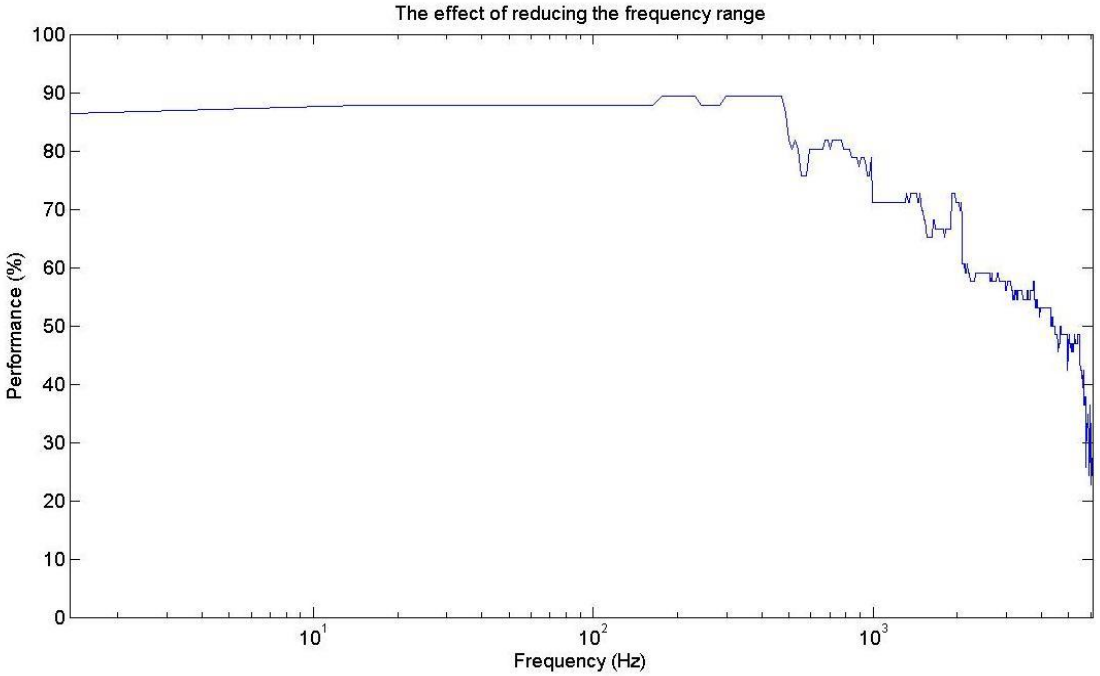


Figure 30 Effect of reducing the frequency range (lower frequency: 1.3 Hz–6085 Hz), excluding the training data, with silence and high-energy suppression

Figure 31 shows the maximum, mean and minimum frame energies for each of the files in the five *dastgàh*s. The minimum curve (in red) gives an estimate of the energy of a silent frame. A threshold can be set for silence suppression at a proper point over this value. The maximum curve (in blue) shows the maximum frame energy of the files; and the mean curve (green line) shows the average frame energy for each file. A threshold can be set under the maximum and over or below the mean curve to reduce the amount of transients entering the analysis, while retaining the important data. The frame energies could be normalised over the frame size and number of frames to give an estimate of each element's energy. In the case of streaming audio analysis, which is proposed as a good idea for further work, the thresholds should be assigned dynamically.

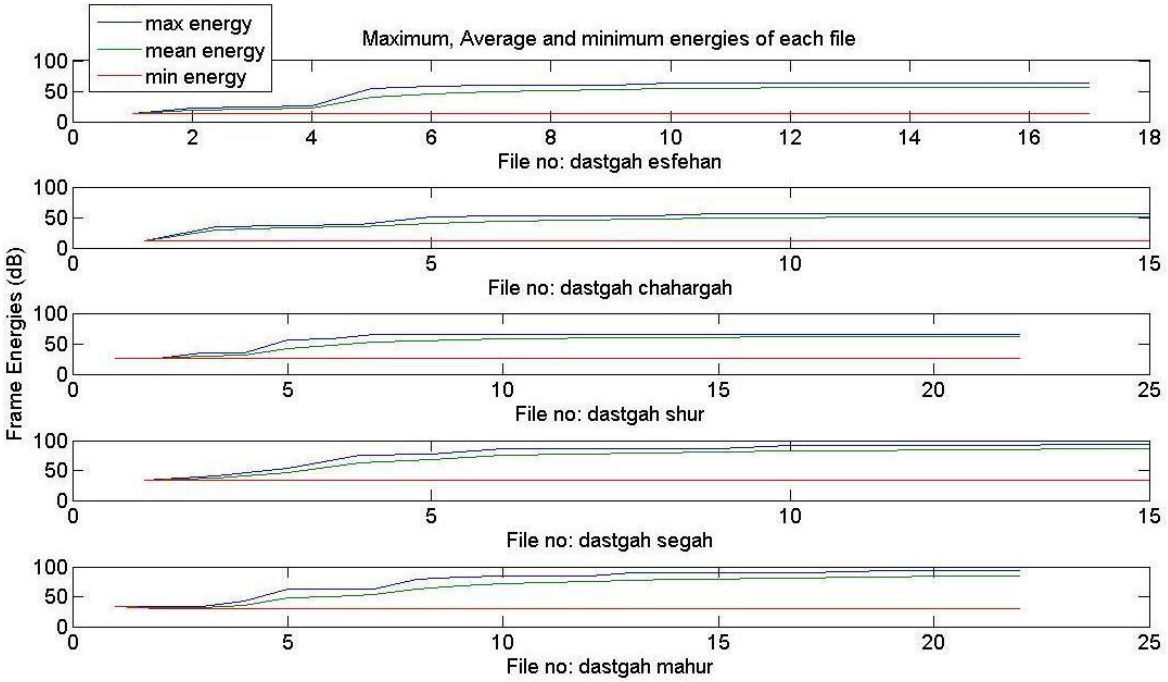


Figure 31 Maximum, mean and minimum frame energies for the files in the five *dastgàh*s

A demonstration is also provided which offers the options to either open a file or to acquire audio through a microphone and soundcard of a computer; subsequently the spectral average of the test sample is calculated, and compared with the templates of all modes of interest. The *dastgàh* is recognised and a membership function shows how close the query is to each *dastgàh*. The distances between query and the 5 patterns, $dist$ are divided by the smallest distance. Then a

number close to 1 (here $dist_0 = 0.98$) is subtracted from them:

$$dist = \frac{dist}{\min(dist)} - dist_0 \tag{31}$$

And finally they were divided by their sum, so that the result summed up to 1. It could be multiplied by 100 to show the membership as a percentage:

$$\mu = \frac{dist}{sum(dist)} \times 100 \tag{32}$$

Finally, the algorithms with optimised parameters are tested on additional samples, where the first five samples of each *dastgàh* are used for training. In a test on santur performances of a Russian song (one file, 110 s) and a *klezmer* tune (2 performances of the same song, two files, 53 s and 60 s long respectively), the three files were accurately classified as *esfehàn*. They were played in harmonic A minor (a scale similar to *esfehàn*). The method was also tested on two *esfehàn* samples, one *bakhtiàry* sample (a derivative of *homàyun*), four *shur* samples (one in E4, one in A4, and two in G4)[75]. All the samples in *esfehàn* and *shur* were recognised successfully, even though some of them contain percussion and noise-like sounds.

In another test, the algorithm was applied to samples of Kurdish music played on the piano, accompanied by *kamàncheh* and voice. Only one of the eight files in *chàhàrgàh* was recognised successfully (8 files, 10:20 min). Seven files were recognised as *shur*, which is not even a close scale to *chàhàrgàh*. The algorithm was also tested on 11 *kamàncheh* samples in different tonalities in the five *dastgàh*s, all of which were recognised as a *shur*, when the first five santur files in each mode were used for training. This shows that the method based on spectral average critically sensitive to instrumentation.

The system works well as long as the modal tonic, tone-range, frequency of note occurrence, and instrumentation are similar to those in the representing patterns (training templates) of each *dastgàh*. It notably works well with db1, whose samples consist of different songs, in different tonalities and heterophonic music, played in an improvisatory manner, including several instances of two notes played simultaneously. Where fast sequences of notes are played, notes appear in several frames as they continue to sound simultaneously, resulting in polyphony.

[75] The G4 *shur* samples consist of the *daràmad* (opening section) and *salmak* (a *gushé* of *shur*). They are available at: http://www.duke.edu/~azomorod/dastgah.html

Nevertheless, this method is a blind classifier. It has little understanding of the data and is blind to whether the mode is classified, and to genre (considering each *dastgàh* as a different genre), melody and timbre. However, the amplitudes of the fundamental frequencies and the corresponding first few harmonics are strong enough that the main target (the mode recognition), is met, even if the method assumes the *dastgàh*s to be different genres or melodies. An ideal *dastgàh* classification system should capture the scale and melodic features of the sample and should not be dependent upon instrumentation, rhythmic structure, or harmonic content.

In the following sections, additions to the current system are presented in which the pitch-histograms and chroma are constructed from the harmonic structure of spectra. They will help in reducing the effect of noise and spurious peaks and decreasing the dimensionality of the feature space.

7.3.2 Pitch histogram tests

In this section pitch histograms for *esfehàn, chàhàrgàh, shur, segàh and màhur* are constructed and applied to the task of mode identification. Pitch histograms are constructed by counting the frequency of note occurrence. They provide note statistics for a piece and resemble a radically simplified version of the spectrum and chroma (Section 7.3.3), where only the fundamental frequency is retained. The frequency of occurrence of notes and knowledge of notes that do not occur in a performance in a particular mode can also be used to find the mode. **Figure 32** shows the scale of *esfehàn*, *chàhàrgàh*, *shur*, *segàh* and *màhur* based on the intervals, which are defined in **Figure 16**, assuming equal weights and 24-TET. They can be used directly as the templates of each *dastgàh* for mode recognition. They are used in Section 7.4 for tonic detection.

In practice, some notes are more likely to occur than others. For example, the tonic usually predominates, or at least it is among the three most frequently recurring notes, as are the subdominant and dominant in order of prominence after the tonic[76].

To account for the number of tone occurrences, pitch histograms in **Figure 33** are constructed based on the *radif* (*dastgàh* repertoire) for the five modes; the opening section (*daràmad*) of each *dastgàh* is taken from *The radif of Mirzà Abdollàh* [70], the standard reference book of *dastgàh*.

[76] The terms tonic, subdominant and dominant refer here to the modal tonic and its fourth and fifth respectively in Persian music.

For *segàh* and *màhur modes*, in which the *daràmad* is not long enough, the pieces *naqmeh*[77] and *kereshmeh* respectively are included. A common tonic of A (element 11 on the horizontal axis) is assumed for all the *dastgàhs*. It can be seen that tonic is the most frequently occurring note for the first three modes, and it is among the three most frequent notes for the other two modes. Ornaments such as *esharé* and *tekié* (see Chapter 2) in which, for example, an adjacent tone is sounded once, which typically increase the occurrence of the tonic and essential scale notes in relation to other tones, are included in the symbolic note histograms (templates) based on *radif*; however, those ornaments such as *riz* (tremolo), the trill (rapid alternation of two adjacent tones), and *dorràb* (sounding of the same pitch twice, rapidly, in anticipation of the stressed main note), in which one note is sounded several times, are counted in the symbolic note histograms as just one note.

[77] *Naqmeh* is also rendered as *naghmeh*.

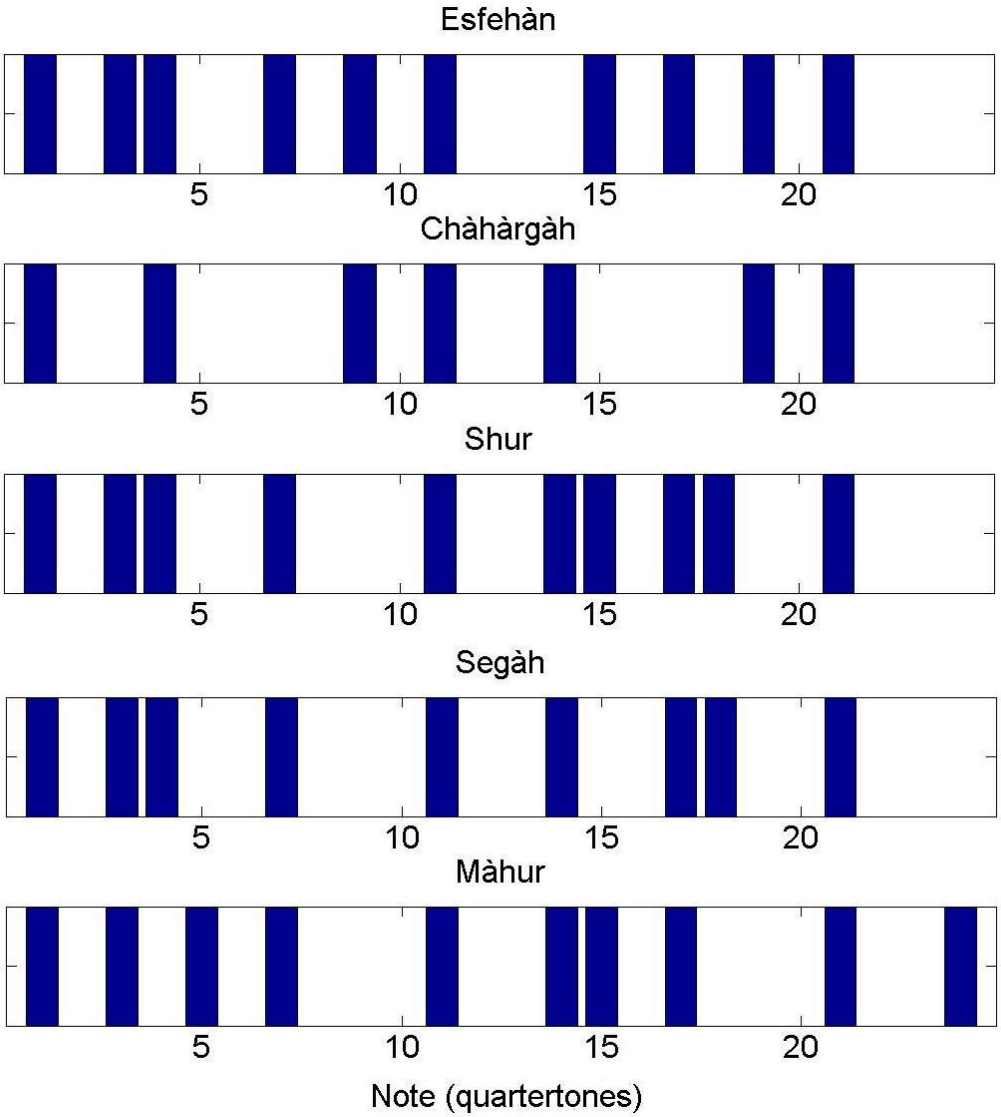


Figure 32 Scale intervals for five *dastgàh*s, based on **Figure 16**

An autocorrelation-based pitch tracker for polyphony music

Here a pitch tracker developed by Aggelos Pikrakis is applied (Chapter 3). The performances of the pitch tracker are listed in **Table 20**, when using the signal of either of the audio channels (CH1 or CH2) or mean of them[78]. A Hann window is used; the frequencies are limited to a range from 130 Hz to 1400 Hz; and the frame size is Nf/Fs=4096/44100=92.9 ms and frames are non-overlapped. The maximum performance for the mean of two channels is 76.4% using a Hann

[78] All the tests in this chapter are done upon the same database of isolated santur notes (db6).

window, which reduces to 59.3% for overlapping frames of hop-size (step) 10 ms. It is 75.0% using a Hamming window and non-overlapped frames.

Table 20 Pitch detection performance, frame size = 92.9 ms (4096 samples)

Audio channel	Window	Step (ms)	Performance (%)	Time[79] (s)
CH1	Hann	92.9	66.4	1.3
CH2	Hann	92.9	65.0	1.4
(CH1+CH2)/2	Hann	92.9	76.4	1.4
(CH1+CH2)/2	Hamming	92.9	75.0	1.7
(CH1+CH2)/2	Hann	10	59.3	11.7

Assuming a 24-TET scale, pitch histograms are made for both training and test phases, as shown in **Figure 34**. Manhattan distance is used as the similarity measure. Using the pitch instead of spectral averages, the order of feature vector reduces substantially. The treble and middle notes are mapped to the bass region, to reduce it further to just 24, corresponding to 24 quartertones in an octave (the order is now the same as chroma, Section 7.3.3). The resulting pitch histogram is called a folded pitch histogram (FPH) in comparison with the unfolded pitch histogram (UPH) where the whole range is considered [69]. Equation 33 shows how the different octaves are mapped:

$$P_{fph}(i) = \frac{1}{N}\sum_{n=1}^{N} P_{uph}(n,i)\,,\ \text{i=1, …, 24} \qquad (33)$$

where N is the number of octaves of interest, n is the octave index, i is the note index, $P_{uph}(n,i)$ shows the pitch histogram for note i at octave n and $P_{fph}(i)$ is the unfolded pitch histogram.

Tempered quartertones (24-TET) are used in this section.[80] **Figure 34** shows the folded pitch histograms for the five modes, calculated by the method in Section 3.3.4. Although polyphonic music transcription is not still a solved problem, the mode identification results through pitch histograms are quite satisfactory for polyphony music, if a few candidates are considered for the

[79] Calculation time per file of 1 second duration.
[80] An equally tempered quartertone is half a tempered semitone (50 cents).

pitch at each frame, rather than attempting to find the dominant pitch. With a frame size of 8192 (185.8ms), which provides a frequency resolution of 5.4 Hz, and using 140 s of training samples[81] and the Manhattan distance, 46.2% of the samples are recognised successfully, when the training samples were used in the tests. This is significantly lower than the performances of the other methods. The confusion matrix in

[81] The first 5 files in each *dastgàh* are used for training. The length of the shortest training set (first 5 files in *shur*) is 140 s.

Table 21 shows the errors in this test: five *esfehàn* samples are recognised as *shur*, where the two scales share fixed accidentals, F and Fs and differ in fixed and moving accidentals, G and Bq respectively; a *esfehàn* sample is confused with a *segàh* sample, which can be due to the pitch tracker's neighbouring quartertone errors of B-Bq or semitone error of G-G#, and the moving accidental Cs for *segàh*, returns to C which is in common with *esfehàn* scale; one *chàhàrgah* sample is confused with *shur* and one is confused with *màhur*, which again can be attributed to errors in pitch tracking; 5 *chàhàrgah* samples are confused with *segàh* where the two modes are perceptually related; 5 *shur* samples are confused with *esfehàn* which differ in fixed accidental G# for *esfehàn* and moving accidental Bq for *shur*; 4 *shur* samples are confused with *segàh* which differ in fixed accidental B for *shur*; 4 *segàh* samples are confused with *esfehàn*, and 4 with *shur*; a *màhur* sample is confused with *esfehàn*, where they differ in fixed accidentals F# and G# for the two modes respectively; two *màhur* samples are confused with *shur*, where they differ in fixed accidental F# and moving accidental Eq for *màhur*; and finally 4 màhur samples are confused with *segàh*, where they differ in fixed accidental F# for *màhur* and moving accidental Cs for *segàh*, while having common note such as Bq which is a fixed accidental for *segàh* and a moving accidental for *màhur*. In comparison with the case of chroma average (**Table 24**) similar confusions occurred using the pitch histograms, where the number of confusions is significantly increased.

If training samples are excluded from the tests, the recognition rate drops to 37.9%. The nature of errors in this case (**Table 20**) is similar to that when the training samples are included (

Table 21), but with a higher value in each confusion element, except one *chàhàrgah – shur* confusion, which is conversely removed.

Table 21 Confusion matrix: PH and Manhattan distance; training samples included in tests

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	11	0	5	1	0
Chà (A)	0	7	1	5	2
Sur (E, A)	5	0	13	4	0
Sga (Fs)	4	0	4	7	0
Mhr (G)	1	0	2	15	4

Table 22 Confusion matrix: PH and Manhattan distance; training samples not included

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	7	0	4	1	0
Chà (A)	0	4	0	4	2
Sur (E, A)	4	0	9	4	0
Sga (Fs)	4	0	2	4	0
Mhr (G)	1	0	2	13	1

Table 23 shows performance versus different numbers of *dastgàhs* in cases when the training files are and are not used in tests, using the Manhattan and cross-correlation distance measures. The sixth scale included is the pentatonic scale, to explore the possibility of augmenting a different (foreign) scale to the classes. It can be seen that although there is a significant drop in performance when two more Iranian scales are added, the effect of adding the pentatonic scale is negligible. The reason could be that pentatonic scale is very different from Iranian scales.

The constructed pitch histograms work well on samples of different instruments and reduce the dimensionality of the feature space. The main drawback is due to the errors at the pitch tracking stage, which are passed on to the scale recognition stage. To solve the problem, a number of pitch candidates can be chosen instead of just one pitch; they can then contribute in the pitch histograms with a membership function. In this way, polyphonic signals could also be analysed. (This is left for future work.)

Table 23 Performance rates (%) using the pitch histograms

	No of *dastgàh*s = 3	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Training files, not used Manhattan	66.7	37.9	37.5
Training files, used Manhattan	70.4	46.2	45.1
5 training files, not used cross-correlation	56.4	33.3	31.9
5 training files, used cross-correlation	61.1	38.5	36.3

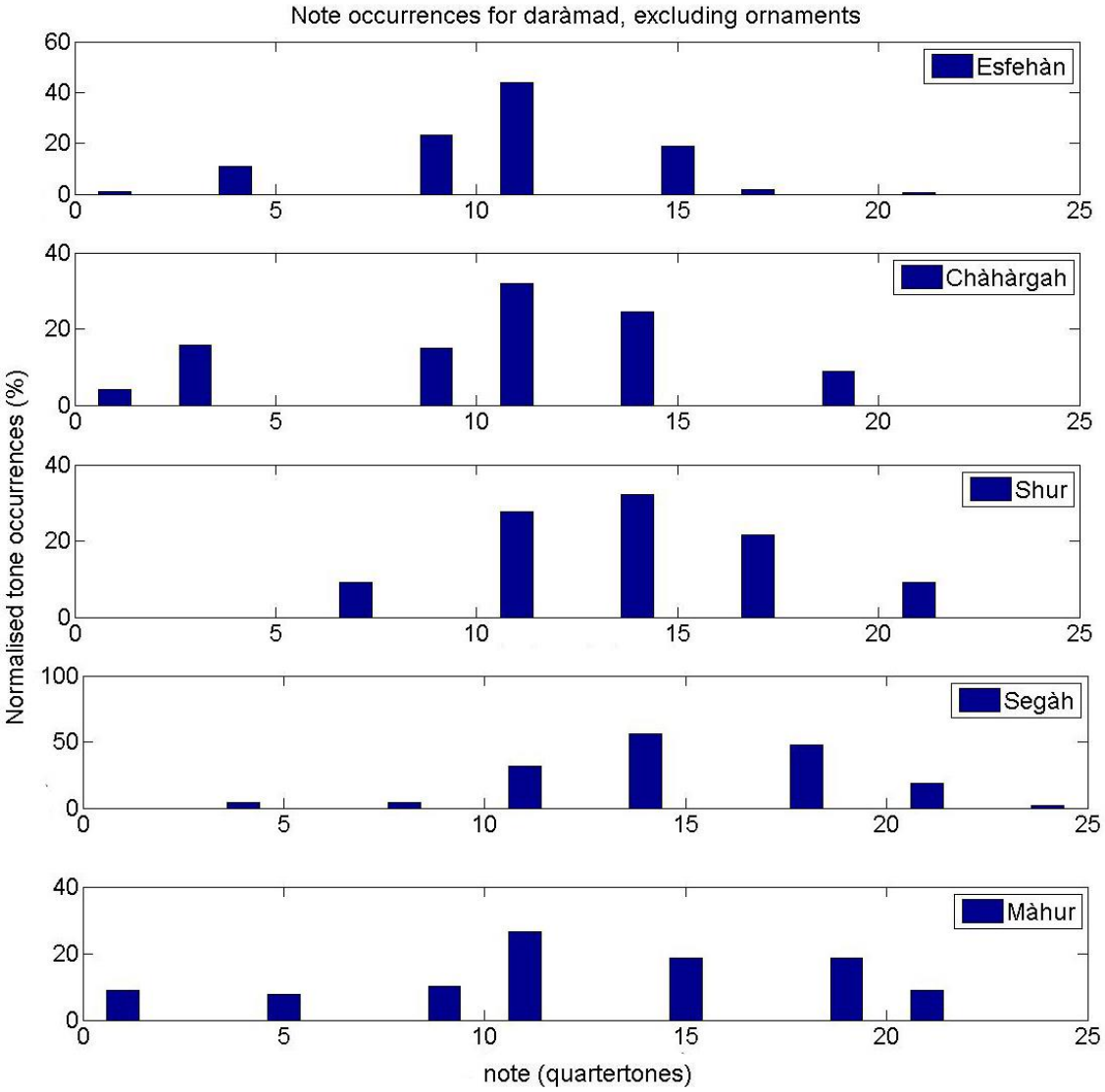


Figure 33 Note histograms based on *radif* [70]

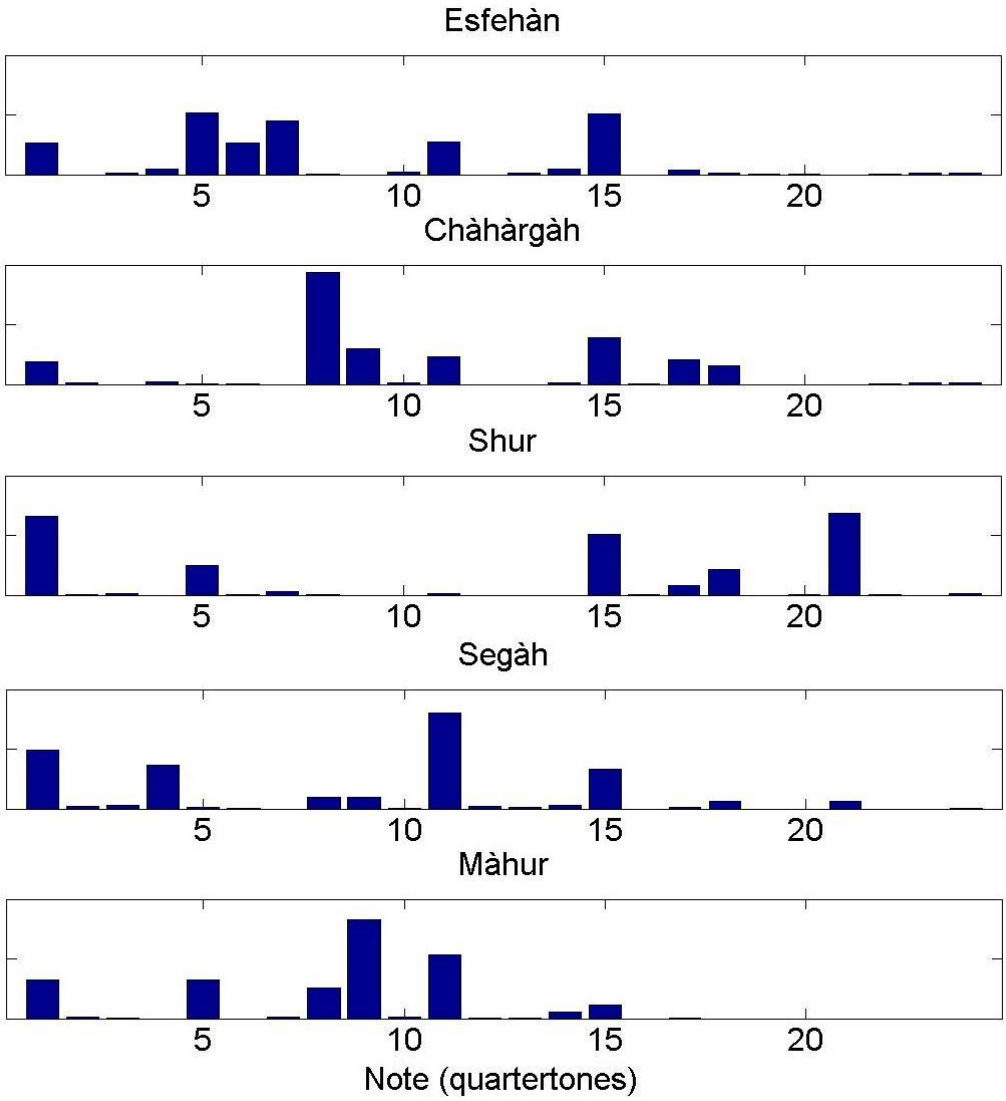


Figure 34 Pitch histograms for *esfehàn, chàhàrgàh, shur, segàh and màhur*

7.3.3 Chroma average tests

As described in Section 6.3.3., chroma is a simplified version of signal spectrum, which is close to human perception and is less timbre-sensitive because:

1. The neighbouring frequency components are summed up[82] in frequency bins. This masks the effect of slightly misplaced fundamental frequencies and harmonics because of inharmonicity factor or another reason. Furthermore, a deviation from exact frequencies is permitted, which

[82] Misplaced frequency components for inharmonicity are merged with the main components.

suits flexible Persian intervals, as slight mistunings are allowed.

2. Finally, fundamental frequencies and their harmonics are folded up to one octave, and the effect of different harmonic weights is evened.

Figure 35 shows the chroma averages for the five modes with a 24-TET. As will be shown in this section, chroma yields a lower recognition rate compared to the spectral average. However, the dimensionality of the feature space is, depending on the frame size, significantly reduced from, for instance 32768 to 24 for a quartertone resolution. Furthermore, it works on samples of different instruments and the effects of octave change and timbral differences are minimised by mapping the Q transform into a single octave.

Constant Q transform [6] is the first step in chroma generation. It is a spectral analysis where the frequency domain components are logarithmically spaced. It is similar to the modalities of human perception and musical scales. Constant Q transform $X(k)$ of a signal $x(n)$ is defined as:

$$X(k) = \sum_{n=0}^{N(k)-1} W(n,k)x(n)e^{-j2\pi f_k n} \tag{34}$$

where length of the window $W(n,k)$ depends on the bin's position.

The centre frequency f_k in the k^{th} bin is defined as: $f_k = 2^{k/\beta} \cdot f_{\min}$ (35)

where $f_{\min}$ is the lowest frequency of interest and β is the number of bins per octave, representing the resolution.

Chroma is computed by mapping the constant Q spectrum to a single octave:

$$Chroma(k) = \sum_{n=1}^{N} X(k + n\beta) \tag{36}$$

where N is number of octaves, and β is the number of bins per octave. Here chroma features are mostly made using 72 bins per octave[83] and three bins are merged to create each quartertone. A histogram with three bars is created to find out which of the three bins has more energy over all pitch classes. Subsequently, every three bins are merged and it is reduced to 24 quartertone

[83] The effect of reducing the number of bins to 36 (12-TET) or increasing it to 144 (48-TET) is also tested.

components, corresponding to a 24-TET[84]. Chroma is a simplified version of the spectrogram, which captures the harmonic content and contains less noise and fewer spurious peaks. The chroma representation codes developed by Harte and Sandler [6] are used throughout this research. However, the codes are customised and the intervals are adapted to a quartertone resolution.

Table 24 lists the database parameters, such as the number of modes of interest, smallest interval of interest, and genre. **Table 25** lists the chroma parameters. Most parameters, including the kernel threshold (sparse threshold), are taken, based on Harte's recommendations [6]. However, the resolution is refined from a semitone to a quartertone and as the major interest lies in long-term information, not in the temporal details, a larger frame size is taken.

The signal is first down sampled by a factor of 8 from 44100 Hz to 5512 Hz and the constant Q transform is calculated with $f_{\min} = 130$ Hz, $f_{\max} = 1400$ Hz, $\beta = 72$ bins and window length of $N = 8192$. $\beta = 72$ bins are considered per octave to provide enough resolution to distinguish neighbouring quartertones regardless of the tuning. Having 72 frequency bins makes the smallest interval step of $2^{1/72} - 1 = 0.0097$. Thus the required frequency resolution is 0.0097*130=1.25 Hz and the required frame size, corresponding to this frequency resolution would be N=5512 / 1.25 = 4410. Here the frame size is taken as $N = 8192$. The resulting frequency resolution would be 5512/8192= 0.67 Hz, which is more accurate than the necessary 2.56 Hz resolution (F3-Fs3) and 1.25 Hz. Every three bins are merged subsequently and the required quartertone frequency resolution is $2^{1/24} - 1 = 0.0293$ times 130 Hz, that is 3.81 Hz.

Table 24 Database parameters

Music genre	Iranian classical
Number of *dastgàhs* of interest	5
Smallest interval of interest	quartertone

Tone resolution

In the analysis of samples of Western music [6, 9, 30, 34, 35, 36, 68], researchers use a 12-TET system, where the tone resolution is a semitone. Gedik and Bozkurt [3] use a 53-TET (1/53th of an octave, the Holdrian comma, which is 22.64 cents) as the tone resolution that suits the analysis of Turkish music. Leonidas [43, 44], who was inspired by them, uses the same tone resolution.

[84] In this way a higher frequency resolution is achieved.

Pikrakis uses a 24-TET resolution [52]. In this thesis, a 24-TET is primarily used, and tests are pursued on the effect of increasing the resolution to 48-TET and 53-TET, and of decreasing it to 12-TET and 6-TET. It is concluded that 24-TET is the optimum tone resolution (Section 7.2.2).

Using the chroma, with optimised parameters, where the first 5 files are used in training[85], with and without 1/8 overlap, the recognition rate is 73.62% and 75.82% respectively, using the Manhattan distance. Cross-correlation similarity measure yields a recognition rate of 39.56% both with and without overlap. Although overlap increases the time resolution and might ordinarily be expected to be beneficial, this is not the case in this experiment with the santur, which produces sustained notes, each of which sounds for as long as a few frames. These results show that overlapping frames improve the results slightly in the case of a Manhattan distance, and do not affect the results in the case of the cross-correlation classifier.

Table 25 Chroma parameters (24 temperament)

Number of bins per octave	72
Smallest interval of interest	quartertone
Signal: down sampled from 44100 Hz	5512 Hz
Down sampling factor	8
Frame size	8192
Frequency Resolution	0.67 Hz
Overlap	1/8 frame overlap or no overlap
Hop size	non-overlapped (1) or (1/8) overlapped
Constant-q minimum frequency	130 Hz
Minimum frequency MIDI note	C3
Constant-q maximum frequency	1400
Maximum frequency MIDI note	F6
Sparse threshold	0.0054

[85] Durations of the training files for the 5 *dastgàh*s are 197.6 s, 158.1 s, 140.0 s, 145.6 s and 107.8 s respectively.

Note occurrence thresholding and creating a sparse chroma average

A tone-sparse threshold is set to remove noise, and the effects of inharmonicity and less frequent notes, such as ornaments and notes played in error. There are thirteen principal notes in Persian music, not all of which are used in a *dastgàh* performance. As usually less than 13 notes per octave are expected, the least 11 chroma components are set to zero and just the 13 highest components are kept. They are then normalised over the number of frames (when averaging) and energy (total component value). Also, a threshold can be set on maximum component value (0.04), to compensate for occurrences beyond a certain limit, for instance to avoid the drones. This way a sparse[86] template is made with limits on minimum and maximum numbers of tone occurrences. The majority of omitted components are not part of the scale; they are either noise, or effects of instrumentation and higher partials, or rare ornaments.

Here the sparse chroma is made for each file individually. An alternative would be to do the sparse chroma process for all training data at once. The advantage would be that the elements, corresponding to rare moving accidentals which happen in more than one file, could sum up to strengthen these essential but weak components by making their amplitude high-enough so that they would not be removed. However, by making the sparse chroma for individual files and then summing and averaging them, the effect of rare notes that were strong enough to appear in at least one sparse chroma, is retained.

After removal of the least 11 elements in chroma average, followed by normalising them again, as described above, the performance using a Manhattan distance becomes 74.73% and 80.22% for the cases with 1/8 overlap and no overlap frames respectively. **Table 20** shows the confusion matrix of the latter case. Four *esfehàn* samples are recognised as *shur*, where they differ in one fixed accidental (G# for *esfehàn*) and one moving accidental (Bq for *shur*); one is recognised as a *segàh*, where they differ in one accidental (G# and Bq) and a moving accidental (Cs for *segàh*, supposing the lower C was the Westernised version, C#, not Cs); and one is recognised as a *màhur*, where they differ in fixed accidentals (G# for *esfehàn* and F# for *màhur*) and two moving accidentals (Bq and Eq for *màhur*). All *chàhàrgàh* sample are recognised successfully. One *shur* sample is recognised as a *esfehàn*, and one is recognised as a *chàhàrgàh*; 5 *segàh* samples are recognised as *chàhàrgàh*, and one as *màhur*; and finally 4 *màhur* samples are recognised as *segàh*. The number

[86] The sparse signal in the following tests is created only by removal of the elements, which are below the minimum threshold, unless otherwise stated. This is different from sparse threshold in **Table 25**.

of confusions between *chàhàrgah* and *segàh*; *màhur* and *segàh*; and *esfehàn* and *shur* is remarkable, as their scales are not similar. Either they reveal hidden similarities, for example in the case of *segàh* - *chàhàrgah* confusions; or they are due to the fundamental frequencies and harmonic content, which are folded into one octave.

The performance using cross correlation and sparse chroma with no overlap and 1/8 overlapped frames becomes 65.93% and 69.23% respectively, a substantial improvement.

Table 26 Confusion matrix for chroma average and Manhattan distance; training samples are included in tests

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	11	0	4	1	1
Chà (A)	0	15	0	0	0
Sur (E, A)	1	1	20	0	0
Sga (Fs)	0	5	0	9	1
Mhr (G)	0	0	0	4	18

The results show that using overlapping frames the performance rate is decreased in the case of sparse chroma with Manhattan distance, contrary to the previous case with all pitch class elements, where the results were slightly improved. Overlapping frames slightly improve the results in the case of a cross-correlation classifier. The Manhattan distance produces better results respect to cross correlation in all cases.

In another test, samples[87] in a new scale (pentatonic scale) were taken into account along with the existing dataset to see if the algorithm is applicable to other scales. The recognition rate in the case of 6 scales becomes 82.35% and 77.45% for the cases with no overlap and with 1/8 overlap respectively.

Figure 36 shows chroma averages of the five modes in the case of a lower tone resolution: 12-TET. Comparing 12-TET with 24-TET, the recognition rate drops from 73.6% (24-TET) to 64.8% (12-TET). **Figure 37** shows that the chroma averages for the five modes with a 48-TET

[87] Frame size is 4096 and sampling rate is 22050 Hz for pentatonic samples.

frequency resolution, which yields a recognition rate of 71.4%, again lower than with 24-TET. If the tone resolution is increased further, to 53-TET (commonly used in Turkish music), the recognition rate is 72.53%. One reason for the decline in performance when using a higher resolution (48-TET and 53-TET) is that the variable positions of the quartertones cause unwanted confusions. However, it is shown that for the santur samples (db1) in this test, 53-TET is a better tone resolution than 48-TET. In the case of 24-TET, the averaged neighbouring microtones proved to provide a better recognition rate, and thus this is the preferred tone resolution.

Although 12-TET is insufficient for the analysis of Persian modes, the recognition rate is still remarkably high, given that only twelve semitones are taken into account. In the cases of 12-TET, 48-TET and 53-TET, the numbers of bins per octave would be 36, 144 and 159 respectively, as three bins per note are used. With 144 frequency bins (48-TET), and 159 frequency bins (53-TET), the required frequency resolutions are 0.004825*130=0.627 Hz and 0.004369*130=0.568 Hz respectively, which are slightly less than the actual frequency resolution of 0.67 Hz; adding more bins beyond a certain limit will not improve the tone resolution for low frequency notes. **Table 21** shows the performance versus the number of notes in equally tempered scales of 12, 24 and 48 notes, where the frames are not overlapped. The first row presents the results using non-sparse chroma averages; the second row shows the results using a sparse chroma, where the least 5, 11 and 22 components are removed for the three temperaments respectively. It can be seen that 24-TET is the optimum resolution. However, the recognition rates of 12-TET and 48-TET are still remarkably high.

Table 28 shows the results for the same conditions when the analysis frames are overlapped as much as 1/8 of the frame size. The results are slightly improved in the case of a sparse chroma average in the second row and in this case the results become very close. All results in **Table 28** are improved or are close to respective values, except the cases of 24-TET with sparse chroma, where the performance drops substantially. Thus, overlap and sparse chroma affect the results, but not necessarily in the same direction. The effect of hop size, as the above results show, is not significant and it is not always desirable.

Table 27 Performance vs. tone resolution, non-overlapping frames

	12-TET Performance (%)	24-TET Performance (%)	48-TET Performance (%)	53-H Performance (%)
Chroma average (non-sparse)	64.83	73.62	71.43	69.23
Sparse chroma average	70.33	80.22	72.53	72.53

Table 28 Performance vs. tone resolution, 1/8 overlapped frames

	12-TET Performance (%)	24-TET Performance (%)	48-TET Performance (%)	53-H Performance (%)
Chroma average (non-sparse)	68.13	73.63	71.42	73.63
Sparse chroma average	74.72	74.73	73.63	73.63

The effect of down sampling

Up to this point, a down sampling of 8 was performed, that is, from every 8 samples, one would be taken into consideration. **Table 29** shows the performance versus down sampling rate in 1/8 overlapped frames and non-overlapped frame cases. It can be seen that in the case of 1/8 overlapped frames, the performances are very close, while in the case of non-overlapped frames, a down sampling of 8 yields a higher performance. This indicates that removing part of the information in down sampling has been useful in the case of non-overlapped frames. Non-overlapped frames worked better with both down sampling rates. It was not possible to produce chroma with a down sampling rate of 2 or with no down sampling on my computer, due to memory capacity limitations.

Table 29 Performance vs. down sampling rate

Down sapling rate	Performance (%) Non-overlapped frames	Performance (%) 1/8 overlapped frames
8	80.22	73.62
4	76.92	74.73

The effect of onset detection

An onset detection function, based on high frequency components was performed and it was observed that if the analysis data is simply taken starting from the third onset, the performance rates become 81.32% and 71.42% for the non-overlapped and 1/8 overlapped frame cases respectively. When full signal was taken into consideration, the performance rates would be 80.22% and 74.73% for the non-overlap and 1/8 overlapped frame cases respectively. Thus, onset detection affects the results.

Tests on samples of different instruments

Although data-driven templates represent our database the best, one may argue that the templates, which were constructed based on santur samples, may not work well with samples of other instruments. To examine this, the chroma-based system is applied to extra samples that were explained in Section 4.2. In three tests on santur, piano-*kamàncheh*, and *kamàncheh* performances, the recognition rate becomes 33.3%, 26.7% and 40%, using a Manhattan distance, or 33.3%, 26.7% and 20% using the cross-correlation. Thus templates independent of instrumentation and tonality need to be constructed, where possible.

Dot product and bit-mask operators

Dot product is an alternative to Manhattan distance measure. In dot product, every element of a frame is multiplied by the corresponding element of the pattern, and they are summed up over all frames of the test sample. If the pattern contains elements which are either 0 or 1, this would be called bit-masking. Bit-masking is computationally more efficient than dot product or Manhattan distance, but it misses the note occurrences and the importance of the scale notes as, in this case, every non-zero component of the training templates is changed into one (the component values are either 0 or 1). **Table 30** compares the recognition rates of Manhattan distance, dot product and bit-mask. Manhattan yields the highest rate.

Table 30 Performance with different distance measures

Distance measure	**Performance (%) 1/8 overlap**	**Performance (%) No overlap**
Manhattan	80.22	75.82
Dot product	72.53	74.73
Bit mask	50.55	48.35

Symbolic versus data-driven templates

So far, data-driven templates were made, based on real audio data. Data-driven templates are dependent on the instrumentation. The elements would have influences of overtones, such as harmonics of a tone and inharmonicity factor, as well as being dependent on tone occurrences.

An alternative way would be to build symbolic templates based on theory, which would be independent of instrumentation and the training samples. However, symbolic templates fail to include all characteristics of the data, for example deviations from the theory that exist in real audio signals.

In general, a template based on real samples of music learns the harmonic characteristics of the ensemble, and makes a better template for scales, as tone occurrences are being included. These are advantages over templates that are based on the scale degrees. While learning the harmonic characteristics of a typical ensemble may increase the average recognition rate, it can be said that learning the harmonic characteristics of the ensemble may turn into a disadvantage, as the aim is to create a system that works regardless of the instrumentation. However, an audio musical sample shows the important notes in a scale, while a scale template with equal weights does not simply represent the importance of the notes.

Although it is ideal to have the pitches for creation of the pitch histogram templates, this feature is considered unnecessarily high-level, as the errors in pitch tracking pass on to the mode recognition phase and the final recognition rate become lower than the case of spectrum and chroma features.

The drawback of using chroma-based templates is that the harmonic content affects the results; for example, the 3rd harmonic of a note could be perceived as a standalone note. One solution would be to construct symbolic templates, based on theory, where the elements are weighted according to the importance of the notes in a typical performance an alternative solution would be to mask the data-driven templates with symbolic templates, with ones at the positions of the scale notes and zeros elsewhere.

The effect of masking with equally weighted data-driven templates is also investigated here. As an alternative classification scheme, the pattern of each *dastgàh* is set based on the scale notes, with ones at the scale notes and zeros elsewhere (**Figure 32**). With 1/8 overlap frames, this yields a recognition rate of 74.73%. In this way, the system becomes less dependent on the timbral and harmonic content, and the instrumentation. However, in this case, some information is missed, as

the importance of notes is not reflected in these templates. There are elements which are not predicted in theory but which exist in data-driven chroma templates; also, it would be beneficial to have the number of tone occurrences reflected in the weighting of the elements. **Figure 35** shows the templates that are constructed based on real santur audio files for each *dastgàh*. From the santur templates (**Figure 35**), it can be seen that usually, the tonic, dominant and certain other scale degrees in each mode have a higher weight than the other elements. **Table 31** represents the four most frequently occurring notes for each *dastgàh*. It can be seen that:

- Element 18 is the tonic for *esfehàn*, and 2nd and 5th scale degrees are the most frequently occurring notes after the tonic, being closely weighted;
- Element 18 is the tonic for *chàhàrgàh* and the 5th scale degree is the next most frequently occurring note, after which the 4th and the 6th occur similarly to one another;
- Element 8 is the tonic for *shur*, the 4th scale degree is the second most important note, followed by the 2nd and 5th and then the 3rd and 7th;
- Element 12, while is the tonic for *segàh*, is only the 3rd most occurring note; the 3rd scale degrees is the most frequent note, followed by the 2nd then the 6th (the fourth frequently occurring note);
- Element 5 is the tonic for *màhur* and has the most frequent note, then the 3rd and 5th scale degrees are the most frequently occurring notes.

Table 31 Tone occurrences (data-driven)

	The most frequently occurring note	Second important interval(s)	Third important interval(s)	Fourth important interval(s)
Esfehàn	Tonic: element 18	2	5	3, 4, 6, 7
Chàhàrgàh	Tonic: element 18	5	4, 6	3, 7
Shur	Tonic: element 8	4	2, 5	3, 7
Segàh	Third interval: element 12	2	tonic	6
Màhur	Tonic: element 19	3	5	6, 4, 7

Thus in all the modes except *segàh* the most frequent note is the tonic. In *segàh*, the tonic is the third most occurring note and its occurrence is close to the fourth most occurring note. It should be noted that these templates are constructed, based on averages over five samples for each *dastgàh*. Longer or shorter samples could lead to different results.

Table 32 Performance rates (%): non-sparse chroma & non-overlapped frames vs. no of *dastgàhs*

	No of *dastgàh*s = 3	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Training files, not used Manhattan	71.79	66.67	69.44
5 training files, used Manhattan	77.78	73.62	76.47
Training files, not used cross-correlation	69.23	40.91	37.50
5 training files, used cross-correlation	66.67	39.56	35.29

Generating data-driven training patterns has the advantage of customising them to other aspects of the samples, such as the style. For example, the importance of each note in a scale is reflected in the weight of the pattern's components. The equally weighted patterns that were constructed can be modified based on the scale, according to weights of the patterns yielded through training samples. However, a constriction should be made for the number of occurrences of a note. For example, in a *chàhàrmezràb*, the modal tonic is played as a drone hundreds of times and this should not make the weight of the tonic in our patterns very much higher than the other components. Also, this would be dependent on the training samples.

Noland [9] builds on the work of Gomez [39], using a model that takes into account upper partials (harmonics), where the partials would decay exponentially. The results improved compared with when equal weights were used as template elements [9].

Table 33 Performance rates (%): sparse chroma, non-overlapped frames vs. no. of *dastgàhs*

	No of *dastgàh*s = 3	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Training files, not used Manhattan	84.62	74.24	76.39
5 training files, used Manhattan	88.89	80.22	82.35
Training files, not used cross-correlation	71.79	62.12	65.28
5 training files, used cross-correlation	70.37	65.93	68.63

Table 34 Performance rates (%): non-sparse chroma, 1/8-overlapped frames vs. no. of *dastgàhs*

	No of *dastgàh*s = 3	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Training files, not used Manhattan	69.23	66.67	69.44
5 training files, used Manhattan	75.92	73.62	76.47
Training files, not used cross-correlation	66.67	39.39	36.11
5 training files, used cross-correlation	66.67	39.56	35.29

Table 35 Performance rates (%): sparse chroma, 1/8-overlapped frames vs. no. of *dastgàhs*

	No of *dastgàh*s = 3	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Training files, not used Manhattan	74.36	68.18	70.83
5 training files, used Manhattan	79.63	74.73	77.45
Training files, not used cross-correlation	66.67	65.15	51.39
5 training files, used cross-correlation	66.67	69.23	59.80

Comparing **Table 32-33** it can be observed that with sparse chroma,[88] all results are improved, compared to non-sparse results, apart from one exception – the case with 3 *dastgàhs*, using the cross-correlation, where the results remain unchanged. Thus it can be concluded that sparse chroma average is preferable. Also non-overlapped frames yield better results over 1/8 overlapped frames in all cases, apart from the case of 5 *dastgàh*s with a cross-correlation classifier.

Furthermore, it can be seen that by increasing the number of classes of interest from 3 to 5, the performance of Manhattan distance drops, while it is improved by adding the 6th class.
This can be interpreted as that by increasing the number of modes from 3 to 5, the choices become more. While this is also the case for the pentatonic scale (6th class), as it is very different from the other scales it can be distinguished from the others more easily and the overall performance is increased in a test that involves the 6 classes. In the case of cross correlation, by increasing the number of classes from 3 to 5 to 6, the performance drops.

In another round of tests, with non-overlapped frames and sparse chroma, involving 5 *dastgàhs* where, one of the files 1 to 10 is used for training each time and that file is excluded from tests (leave-one-out technique); the resulting performances are 60.47%, 70.93%, 73.26%, 59.30%, 72.09%, 79.07%, 58.14%, 80.23%, 74.42% and 75.58% respectively, i.e. 70.35% in average. If 1/8 overlapped frames are used, the performances become 70.93%, 79.07%, 65.12%, 60.47%, 77.91%, 81.40%, 67.44%, 83.72%, 73.26% and 76.74% respectively, i.e. 73.60% in average. The first 5 files are intentionally played to provide the scale notes. They also include short melodies and they are shorter than the other performances. The 6th file is longer than all of the first 5 files and is an actual *dastgàh* performance, which produces the highest recognition rate in this round of experiments. This shows that providing solely the scale notes is not sufficient, and the weights of the elements of the template of each *dastgàh* have to be adjusted using standard performances in each *dastgàh*.

[88] The sparse chroma was obtained by removing 5, 11 and 22 least components of the chroma average.

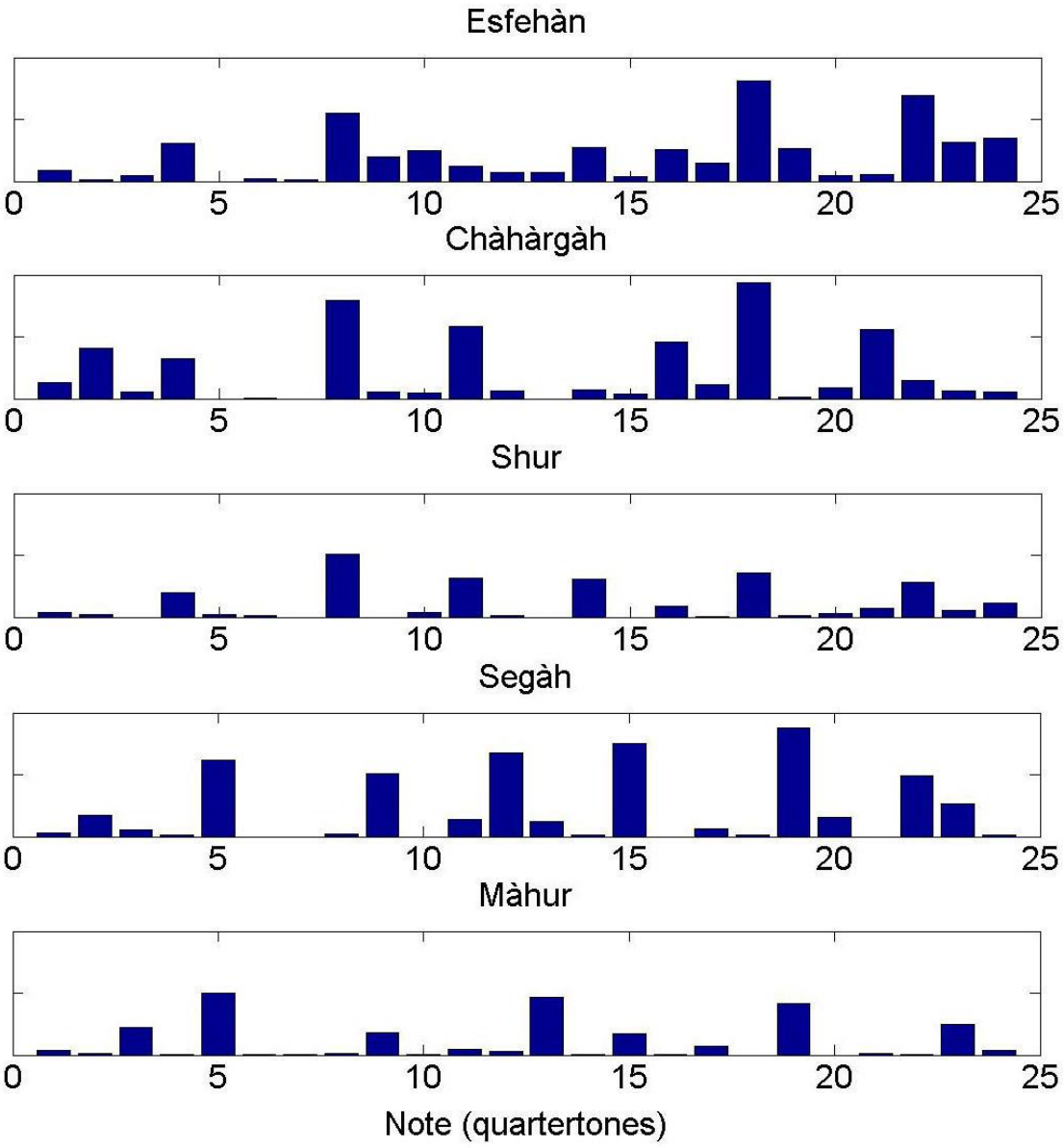


Figure 35 Chroma averages: *esfehàn, chàhàrgàh, shur, segàh and màhur* (24-TET)

The two average (70.35% and 73.60%) performance rates can be compared with the results of previous experiments, where the first five files were used in training and the training files were used in tests (80.22% and 74.3% for non-overlapped and 1/8 overlapped frames respectively); or with experiments with similar conditions, where training samples were not used in tests (74.24% and 68.18% for non-overlapped and 1/8 overlapped frames respectively). Comparing the case of leave-one-out and the case where the first five files were used in training, but not in the tests, it can be seen that when the amount of training data is little, overlapped frames perform better.

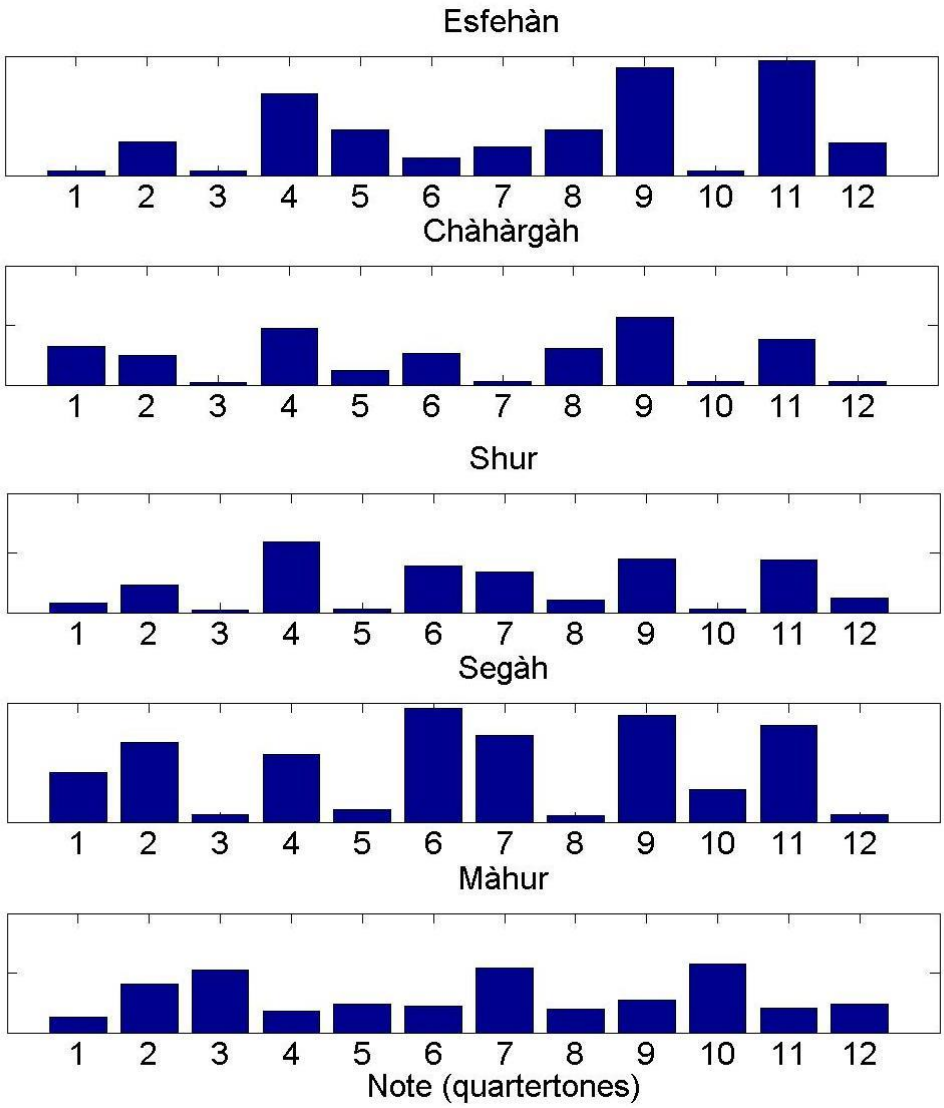


Figure 36 Chroma averages: *esfehàn, chàhàrgàh, shur, segàh and màhur* (12-TET)

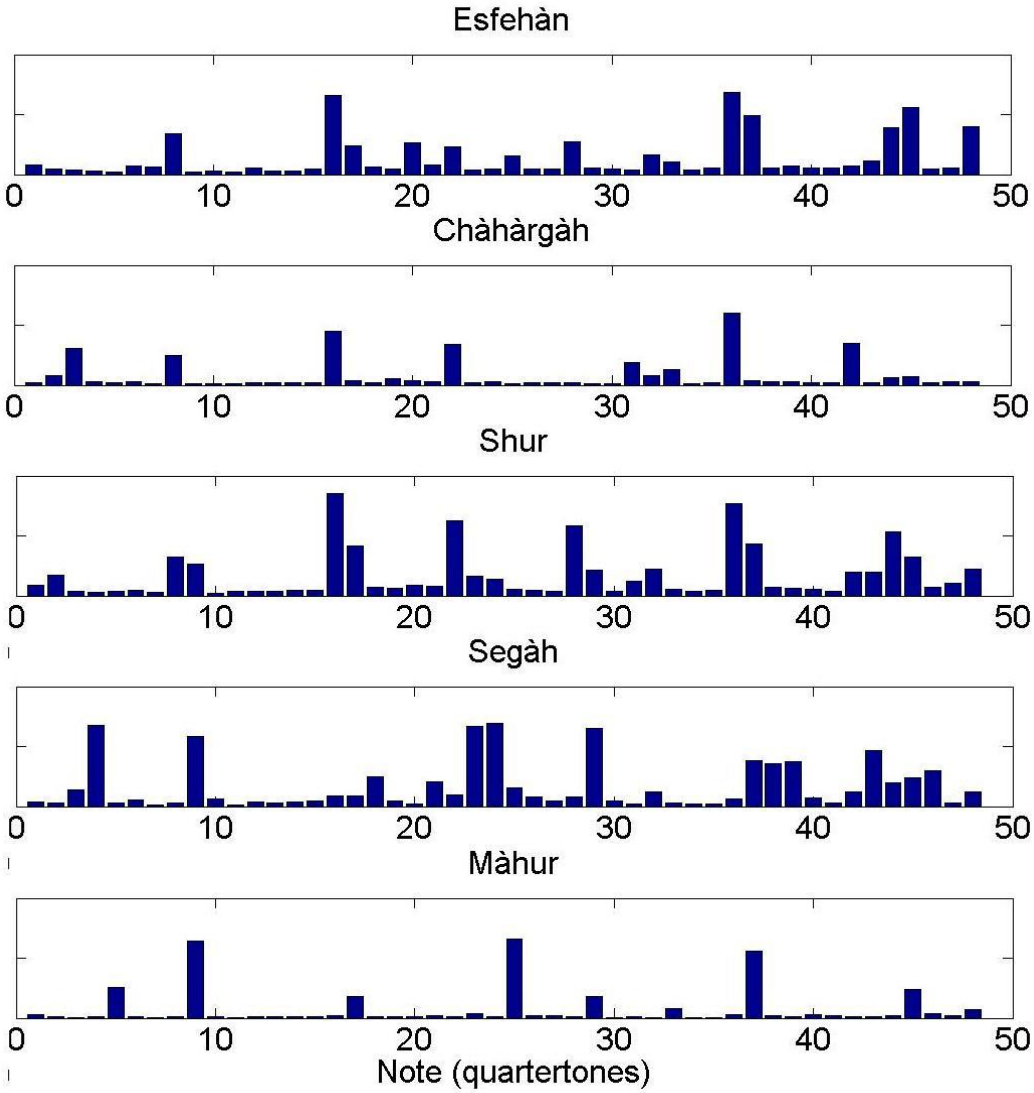


Figure 37 Chroma averages: *esfehàn, chàhàrgàh, shur, segàh and màhur* (48-TET)

7.3.4 Implementing a generative method, the GMM

In this section a GMM model is applied to audio musical data, with chroma as the feature. As every frame is included, rather than an average, the progress over time is taken into account. HMM / GMM models are able to adapt the chroma to fit to the boundaries of each mode. This enables first order melodic progression through time. It is worth noting that some scale notes are more important than others, due to their role in phrases, and their relationship to other notes, in addition to their number of occurrences. This complexity is automatically included here, as the system learns by the training data.

Initial state

It is necessary to find the initial state probabilities for Persian modes. During the past two centuries *dastgàh* e *shur* has been the most popular Persian mode. *Àvàz-e esfehàn* is also a popular mode. **Table 36** lists the number of pages of each mode contained in *The Radif of Mirzà Abdollàh* [70]. This frequency of occurrence could be used as an estimate of importance of each mode in the core repertoire of Persian music. The third column displays the ratio of number of pages of each *dastgàh* to all, as a rough initial probability of each class. However, by assuming unequal initial states, even if the initial states are obtained correctly, calculations are bound to the style(s) that the assumptions are based on. In a modern performance, any mode is expected to be played with equal probability. Thus all the initial states can be assumed to be equal to 1/5.

Table 36 Number of pages of the scores in *The radif of Mirza Abdollah* [70]

Dastgàh	Number of pages	Initial probability
shur, bayàt-e kord, dashti, abu'atà, afshàri, bayàt-e tork and navà	74+11+12+22+14+25+41= 198	0.41
homàyun and bayàt-e esfehàn	49+14= 63	0.13
segàh	37	0.08
màhur and ràst-panjgàh	74+42= 116	0.24
chàhàrgah	65	0.14

GMM experiments and results

In the following series of experiments GMM method is used for musical mode estimation, where the distribution of chroma of the training[89] samples of each mode is modelled by a Gaussian mixture density. The training dataset is used to initialise the parameters of a Gaussian Mixture Model: k-means[90] algorithm finds the centre of each class, the prior probabilities are calculated, based on the proportion of samples that belong to each class, covariance matrices are calculated as

[89] The training and test processes are explained in Chapter 6.

[90] K-means clustering is a method of partitioning n observations (samples) of dimension d (24 in these experiments), into k clusters, where each observation belongs to the cluster with the nearest mean.

sample covariances of the points associated with the centres. This is just the initialisation for Expectation Maximisation (EM), upon which the mixtures are trained subsequently.

The same database (db1) is used as in previous sections, including scale notes, opening sections, melodies and random sequences of notes, played on a santur. In some cases two or more notes are played at the same time. Fast sequences of notes are also played, where traces of notes sustain into the time the subsequent notes are played. Samples are mono, 16 bit wave files at a sampling rate of Fs=44100 Hz.

The signal is down sampled to 11025 Hz and constant Q transform is calculated with $f_{\min} = 130$ Hz, $f_{\max} = 1400$ Hz, $\beta = 72$ bins and a window length of $N = 8192$. Seventy-two bins per octave provide enough resolution to distinguish between neighbouring quartertones regardless of the tuning.

In a test using a GMM model with 5 mixtures and 1/8 overlapped frames, where the parameters are estimated using EM, 83.5% and 77.3% of the samples were recognised successfully for the cases where the training files were used and were not used in the tests, respectively.

Error analysis

Looking at the confusion matrix (**Table 37**), it is observed that 11 out of 17 *esfehàn* samples, 14 out of 15 *chàhàrgàh* samples, 19 out of 22 *shur* samples, 11 out of 15 *segàh* samples and 17 out of 22 *màhur* samples are recognised accurately.

- Two *esfehàn* samples are recognised as *chàhàrgàh*: they share a fixed accidental, G#, also C#, which is a moving accidental for *esfehàn* (*homàyun*), is a fixed accidental for *chàhàrgah*
- One *esfehàn* sample is confused with *màhur*: they differ in fixed accidentals #G and F# for the two modes respectively
- All *chàhàrgàh* samples are recognised successfully
- 3 *shur* samples are recognised as *esfehàn*, where they differ in fixed accidental G# for *esfehàn* and moving accidental Bq for *shur*
- 3 *segàh* samples are recognised as *chàhàrgàh*, where they are perceptually related
- One *segàh* is confused with *shur*, where they share fixed accidental Fs, although Fs for *segàh* is slightly sharper than Fs for *shur*, they share a moving accidental Cs too, where Cs in *shur* is

significantly sharper and is close to C#, fixed accidental Bq for *segàh* is a moving accidental for *shur*

- 2 *màhur* samples are recognised as *esfehàn* and 4 *màhur* samples are recognised as *segàh*.

These errors, specifically the last, are not due to similar scales: they are either due to folding the fundamental frequencies and harmonics to one octave, or they capture other similarities in temporal performance.

Table 37 Confusion matrix for GMM tests: 1/8 overlapped frames, training samples included

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	15	1	0	0	1
Chà (A)	0	15	0	0	0
Sur (E, A)	3	0	19	0	0
Sga (Fs)	0	3	1	11	0
Mhr (G)	2	0	0	4	16

Table 38 shows the performance versus number of *dastgàh* for the following four cases: with sparse and non-sparse frames; each with no overlap and with 1/8 overlapped frames. In all four cases the training samples were used in tests.

1. Non-sparse case: it can be seen that overlapped frames work better in the case of 5 modes, while the non-overlapped frames work better when there are samples in 6 modes. Also, moving from 5 modes to 6 improves the estimation in the case of non-sparse-non-overlapped signal, while it decreases the performance in the case of non-sparse signal with 1/8 overlapped frames.

2. Sparse case: non-overlapped frames yield a better estimation; and there is a further improvement moving from 5 to 6 modes.

It can be seen that there is no linear relation between these parameters, as there was in the case of deterministic methods such as chroma average feature with Manhattan distance.

Table 39 shows the mode estimation versus number of mixtures 1-10.

Table 38 Estimation (%): GMM method; number of mixtures=5

	No of *dastgàh*s = 5	No of *dastgàh*s = 6
Non-sparse, Non-overlap	62.64	84.31
Non-sparse, 1/8 overlap	83.52	63.73
Sparse, Non-overlap	74.73	75.49
Sparse, 1/8 overlap	70.33	73.54

Table 39 Estimation (%): GMM method; 1/8 overlapped frames; number of mixtures 1-10

No of mixtures	1	2	3	4	5	6	7	8	9	10
Estimation (%)	69.2	74.7	83.5	78.0	83.5	**85.7**	79.1	72.5	80.2	80.2

7.4 Performing tonic detection prior to mode recognition

The recognition rates with a tonic detection stage in pre-processing, using Manhattan distance and dot-product for three datasets (santur solo samples in 5 *dastgàh* (db1), small ensemble (db2), and rebetiko music (db3)) are 45.05% (Manhattan distance) and 52.75% (dot-product); 39.18% (Manhattan distance) and 51.55% (dot-product); and 39.82% (Manhattan distance) and 46.02% (dot-product). Modifying the weights of the elements changes the results: for example, if in the case of santur solo samples (db1) and dot-product operator, non-zero components of the sum of theoretical scales are set to 1 (eliminating the importance of the individual notes which is reflected in their weights), the recognition rate drops from 52.75% to 45.05%. In all the above experiments, a tonic detection, based on dot-product was performed prior to mode identification.

Although the use of theoretical scale templates with dot-product and bit-masking misses the frequency of occurrence of notes, as the component values are derived based on scales (0-1) and do

not take into account the frequency of occurrence of notes, bit-masking is computationally more efficient than Manhattan distance and it produces higher recognition rates.

In the data-driven template method, there is a training phase in the beginning, where a template is generated for each mode. In the test phase, the chroma vector of a given audio file is constructed and aligned, and a classification scheme (Manhattan distance or dot-product operator) is used to gauge the similarity between the query and the templates.

Using sum of theoretical scales for tonic detection stage and the theoretical scales for *dastgàh* identification stage, dot product as the classifier, 49.45% and 44.80% of the samples are identified correctly for db1 and db3 respectively. In another experiment on db1, using data-driven templates for both tonic detection and *dastgàh* identification stages, 70.33% and 50.55% of the samples are correctly classified for the case where the training samples where used, and the case where the training samples were not used in the tests. The results become 61.60% and 47.20 if the last experiment is applied to db3.

Keeping all conditions the same, using Manhattan distance in tonic detection only or both in tonic detection and classification, the recognition rate decreases to 49.60 and 38.40% respectively.

Table 40 shows the recognition rates of various tests on different parts of the database.

Case 1: the tonic detection stage assumes that the scale is known and uses data-driven templates;
Case 2: the tonic detection stage assumes that the scale is known and uses theoretical scale templates;
Case 3: theoretical scale templates are used in both tonic and mode recognition, using data-driven templates for mode recognition;
Case 4: data-driven templates are used both in tonic detection and for mode recognition.

In case 1, in which it is assumed that the mode is given, the tonic is detected using shift and multiplying data-driven templates (based on the first 5 files of db1) with the chroma feature of the samples. The samples are subsequently aligned by moving the tonic to element 1, and the *dastgàh* is identified by multiplying them (dot-product) to templates of each *dastgàh*. In an experiment on db1 (see the first row of case 1 in Table 40) 98.90% of the samples were recognised successfully. The confusion matrix in **Table 41** shows that the only error was a *màhur* sample that was

recognised as an *esfehàn* (the two modes differ in that *esfehàn* has a fixed accidental G# and *màhur* a fixed accidental F#, which is only a quartertone higher than the fixed accidental Fs in *esfehàn*). The latter could be due to bin frequency error between Fs and F#. The other rows of case 1, show the identification rates, using additional test samples of db2, db3, db4 and db5. In case 2 it is assumed that the mode is given and theoretical templates are used for tonic detection. The recognition rates are presented.

High recognition rates in Cases 1 and 2, respect to Case 3 and 4 show the effect of a perfect tonic detection before scale recognition. Comparing Case 1 versus Case 2 it can be seen that using data-driven training samples slightly decreases the performance respect to the case of theoretical templates, most notably in rows 4 and 5, that include db4 and db5 with *kamàncheh* and piano samples (most samples in db1, db2 and db3 included santur instrument either in solo or accompanied by other instruments), which is due to the fact that the training templates were made of santur samples. Comparing Cases 3 and 4, it can be seen that using data-driven samples increases the performance. This implies that data-driven templates have a higher tonic detection rate and improve the final results.

Table 40 Performance (%) with tonic detection, using dot-product classifier

Test samples	Case 1	Case 2	Case 3	Case 4
db1	98.90	97.80	49.45	70.33
db1, db2	98.97	94.85	47.42	67.01
db1, db2, db3	94.39	92.52	45.79	64.49
db1, db2, db3, db4	93.86	88.98	44.74	63.16
db1, db2, db3, db4, db5	93.60	87.20	44.80	61.60

Table 41 Confusion matrix for chroma with tonic detection and Manhattan distance; training samples are included in tests

	Esf (A)	*Chà* (A)	*Sur* (E, A)	*Sga (Fs)*	*Mhr* (G)
Esf (A)	17	0	0	0	0
Chà (A)	0	15	0	0	0
Sur (E, A)	0	0	22	0	0
Sga (Fs)	0	0	0	15	0
Mhr (G)	1	0	0	0	21

Components of the templates can be modified in order to improve the system. If in the case of dot-product classifier, the components of the data-driven training template for tonic detection which are less than 0.25 and 0.20 of the maximum peak are set to zero, the performance drops from 61.60% to 60.80% and 57.60% respectively. Alternatively, if non-zero components of theoretical scales are set to 1, the recognition rate drops from 44.80% to 37.60%. Although these two heuristic modifications did not enhance the recognition rate, they show the effect of the modifications of the template components upon the final results.

Dot-product and bit-masking are computationally more efficient than Manhattan distance. Dot-product yields better recognition results. In summary, in this section, the chroma is used as the feature set and dot-product and Manhattan distance as the classifiers. In the beginning, the chroma of the test samples is aliened (tonic is shifted to the first element), using a shift and bit-masking process. This is followed by a mode identification stage.

In the future, intuitive templates could be made in order to improve the tonic detection stage and to reach scale recognition rates as high as those achieved with known tonics. The tonic detection system could potentially be enhanced by modifying the weights of theoretical intervals, based on the frequency of note occurrence or another combination of theoretical and data-driven templates.

If, instead of finding the tonic separately, the chroma of the test samples are shifted 23 times and are compared with the training templates each time, the recognition rates for Manhattan distance and data-driven samples (db1), dot-product and data-driven samples (db1), and theoretical scale intervals and bit-masking become 86.81%, 72.53% and 50.55% respectively, where the training files were also used in the tests. If files 1-5 of db1, which were used in training, are excluded, the recognition rates change to 83.33%, 72.73% and 51.52% respectively. In bit-masking with scale intervals, of course, no audio sample was used for training.

7.5 Summary of performance rates of various methods

The recognition rates of different methods are represented in **Table 42.** In columns 2 to 4, a Manhattan distance measure is used to compare training and test feature vectors, based on spectral average, chroma average, and pitch histograms, where each feature has an optimal frame size of its own. The optimum recognition rates for these features using a Manhattan distance are 90.1%, 80.2% and 46.2% respectively, when the training data is used in tests; and 86.4%, 74.2%, and

37.9% respectively, when the training data is not used. In the final column, chroma is used as the feature and a machine learning method (GMM) is applied to estimate the mode. The recognition rate is 85.7% when the training data is used in tests, and 80.3% when it is not used. If the training templates are shifted 23 times, to reduce the effect of different tonalities between the training and test templates, the recognition rate using chroma and Manhattan distance is 86.8%.

Table 42 Performance rates of various methods (%)

	Spectral Average	Chroma Average	Pitch Histograms	GMM
Training files not used, Manhattan	86.8	74.7	37.9	80.3
Training files used (5 files), Manhattan	90.1	80.2	46.2	85.7

The spectral average feature with Manhattan distance yields the highest recognition rate, but it is constrained by limitations of the instrumentation and harmonic content. The dimensionality of the feature space is also high, and playing the same melody in different octaves (which a listener would recognise as essentially the *same* melody) does *not* lead to similar results, as they will be seen as differences between the feature vectors of the samples and the data-driven templates.

The recognition rate of the pitch histograms is low, as the pitch tracking errors pass into the mode identification stage.

The chroma average feature with Manhattan distance, and the chroma with a GMM classifier yield the two highest recognition rates. The first involves less computational cost, but it is noteworthy that in the spectral and chroma average and pitch histogram methods, the melody progression over time was lost, as everything was averaged. The summing-up process reduces the effects of instrumentation and timbre, and also reduces the noise level, while the fundamental frequencies and their harmonics are intensified. However, the temporal information, such as the note sequence is missed. The signal should be segmented to find the points of modulation prior to averaging. GMM takes the melody progression and other temporal details into account, on a frame-by-frame basis.

GMM, a machine learning approach, learns the model from the relationships between the data, not based on the information that is given. Although a supervised GMM was implemented here, as GMM is generally an unsupervised classifier, all samples would be classified automatically if just the number of classes was given.

It is recommended for future work that a combination of both methods should be included, as the errors differ in nature. The main disadvantage of the chromagram is that the pitch height[91] is ignored, despite the fact that it contains information. However, absolute pitch is not necessary for Persian scale identification, as it is common to tune the instruments in different pitches, especially to suit the voices of the vocalists.

Different parts of a signal carry different information that affects the results. For instance, at the onsets, further to the fundamental frequency of the note and its harmonics, several other strings vibrate too, and there are several pitches, harmonics and transients in the spectrum of the signal. Thus the analysis frame should start at a short time distance after the onsets.

Here, in most experiments, the whole bandwidth of the spectrum was considered in the feature vector, which contains unnecessary information and noise. It was observed, when effects of silence and high-energy removal were tested, that silence suppression (10% threshold) did not affect the mode recognition results, while high-energy suppression did affect the results. Further tests would need to be undertaken to find the desired portion of the spectrum, and also to set the optimum threshold for silence and high-energy suppression.

[91] Pitch height refers to absolute frequency, in contradistinction to pitch class, which corresponds to the notes and ratio of their frequencies.

8 Discussion and Conclusions

This thesis is the result of multi-disciplinary research in the fields of DSP, musical practice, and ethnomusicology. The first of these involves signal processing tools and methods for the analysis of audio musical files, while the latter two define the orientation of the research and the musical parameters considered. Several methods are devised and implemented for automatic recognition of the Persian musical modes.

8.1 Summary

In this thesis the underlying structure of Persian music is explained and discussed and algorithms are developed to identify Persian modes in an audio musical file according to five general scales. As a result of comparison of several alternative equal divisions of the octave, 24-TET is preferred throughout for the classification of Persian intervals. A database, initially consisting of 5706 seconds of santur pieces (91 pieces), including scalar patterns, existing melodies, and improvisations in all five of the Persian scales (*esfehàn, chàhàrgàh, shur, màhur, segàh*) and their derivatives, was used in the main evaluations and parameter optimisation. This database was expanded to include a much more diverse set of samples (see Chapter 4) to test the wider applicability of the methods developed.

The algorithms developed and presented here are based on spectral average, chromagram and chroma average, and pitch histogram features. A tonic detection step is devised and the inharmonicity factor of the santur is calculated. A similarity scheme, based on geometric methods such as the Manhattan distance, cross correlation, dot-product, bit-mask, and a machine learning method (GMM), is applied to the features to identify the scale and the mode. The results of this scheme are compared with data manually labelled by the author (applying his knowledge as an expert musician).

The optimum recognition rates for spectral average, chroma average, and pitch histogram features, using a Manhattan distance, are 90.1%, 80.2% and 46.2% respectively (the training data is used in the tests and frames are non-overlapped). The estimation, using a machine learning method (GMM), is 85.7%. If the training templates are shifted 23 times, to reduce the effect of different tonalities between training and test templates, the recognition rate using chroma and Manhattan

distance increases to 86.8%. This recognition rate was achieved using santur samples as the data-driven training templates. If the scale is used as the theoretical template, with 1s at the positions of scalar notes and 0s elsewhere, the recognition rate becomes 50.6% using bit-mask. Using symbolic templates based on *radif*, the recognition rate becomes 74.73%. The significant drop in recognition rate, from 86.8% when data-driven templates (chroma averages) were used to 50.6% and 74.73% shows the importance of the frequency of occurrence of scale notes.

8.1.1 Comparison of alternative features

Spectrum, chroma, and pitch histograms are compared here as the features. The highest recognition rate is achieved by the use of the spectrum, where the system depends on the harmonic content, instrumentation, octave register, and tonality of the training samples, and the dimensionality of the feature space is high.

Pitch histograms appear at first sight to be ideal for scale representation: they are independent of instrumentation and harmonic content; and the calculation cost of the comparison section is low, as with chroma. However, the pitch-tracking stage has a calculation cost and pitch-tracking errors pass on to the mode recognition phase, with the result that the final recognition rate is lower than when using the spectrum or chroma. Although pitch histograms are dependent on the tonality, this can be overcome by a tonic detection stage or a shift and compare process in pre-processing.

Chroma, a substantially smaller version of the spectrum inasmuch as frequency bins have merged the components, is less dependent on instrumentation than spectrum, and the calculation cost is much less than with spectrum and pitch histograms, although the results are still affected by harmonic content, and are still dependent on tonality. However, with a tonic detection stage or a shift-and-compare process, the tonic can be shifted to the desired tonality.

All three features can either be used on a frame-by-frame basis or in average form. The summing-up process (averaging of all frames) has two desirable consequences: (1) reducing the effect of instrumentation and timbre in general[92]; and (2) reducing the effects of noise as the fundamental frequencies and their harmonics are intensified. However, all three features, when averaged, lose temporal information, including the note sequence. If a sample contains modulations to different

[92] When more than one instrument is played in a piece, summing up reduces the effect of instrumentation and timbre. In the case of chroma, folding into one octave reduces the effect of instrumentation further.

modes or modal tonics, the points of modulation need to be found prior to averaging, where each segment is treated separately.

8.1.2 Parameter optimisation

The parameters of the mode recognition algorithms developed in this research have been optimized, based on the training samples used. The optimized parameters depend on aspects of the samples and the processes, including the onsets, silence suppression and high-energy suppression, tone resolution, frequency range, and the amount of training data. Parameter optimisations in this section are made by straight comparison, with no shift and compare in template comparison.

8.1.2.1 Effects of onset detection and silence and high-energy frame suppression

Different parts of a signal carry different information that can affect the analysis results. Firstly, at the onset (when a note is struck), in addition to the fundamental frequency of the note and its partials, several other strings are also excited to vibrate, so several extra fundamental frequencies, their partials, and transients are briefly present in the spectrum of the signal. Secondly, silence suppression and high-energy frame suppression affect the recognition rate and change the nature of errors (silence suppression perhaps only marginally, as the santur has a relatively sustained sound). In Chapter 7 it is reported that silence suppression does not affect mode recognition results, whereas high-energy suppression does affect the results.

8.1.2.2 *Frame size, the amount of training data, and tone range*

Table 19 shows the performance versus frame size when spectral average is used as the feature: the optimised frame size diminishes from Nf=131072 samples (with no suppression) to Nf=32768 when silence suppression and high-energy frame suppression are performed. In both these cases (without and with suppression), the maximum recognition rate is 86.36% when the training samples are not used in tests, or 90.11% if the training samples are used. When two alternative types of frames, Hamming and Hann, were compared in the pitch tracking task, it was found that Hamming works better.

The effect of the amount of training data on recognition rates was also investigated (see **Figure 25**): as the duration of training data increases, the recognition rate increases to 89.0% with 262 seconds of training data. **Figure 26 and Figure 27** show that over 65 s (using files 1-6) or 45.3 s (using files 6-10) of training samples are needed to produce a recognition rate of around 83.2% without silence suppression and high-energy suppression and 81.0% with silence suppression and high-energy suppression. The maximum recognition rates over the average curve of the three experiments are 89.4% without silence suppression and high-energy suppression and 83.9% with silence suppression and high-energy suppression. Thus performance is dependent on the duration of the training samples used and on whether silence suppression and high-energy suppression are implemented.

Frequency range is another parameter that affects the recognition rate. There are unwanted elements in very low and very high frequency contents of the dataset (db1), and the signal could be high-pass filtered to amplify the effect of the low-amplitude, high-frequency components, and low-pass filtered to avoid the effect of unnecessary low frequency components. Most of the notes in db1 were played in the middle register of the santur (277.2 Hz – 698.5 Hz) and most of the energy in db1 as a whole is concentrated below 5 kHz.

The effect of frequency range using spectral averages was examined for db1 (**Figure 28-30**): at a lower frequency bound of 458.9 Hz, a maximum performance of 89.4% is achieved, which is higher than was the case with the full frequency range. To reduce the amount of calculations and include only the necessary harmonics, it would be advantageous to limit the frequency range to 458.9 Hz–6085 Hz. Silence suppression does not greatly affect the recognition rate results, while high-energy frame suppression improves the results.

8.1.2.3 Pitch histogram parameters

Although it would be ideal to use the frequencies of note occurrence as pitch histograms, this feature is considered unnecessarily high-level, as errors in pitch tracking pass onwards to the mode recognition phase, and the final recognition rate becomes lower than in the case of the features spectrum and chroma. The recognition rate is 46.2% when training samples are used, and is 37.9% when training samples are not used.

8.1.2.4 Chroma parameters

Different frame sizes were used for feature chroma, and 8192 proved the best; a sparse function was performed on the chroma to remove the least 11 elements in 24-TET, retaining a maximum of 13 principal notes (see 7.2.3).

The effect of changing tone resolution was also investigated. When 12-TET, 24-TET, 48-TET and 53-TET are compared (using sparse chroma average, when 5, 11, 22 and 22 components respectively were removed), the recognition rates are 70.33%, 80.22%, 72.53% and 72.53% respectively.

In the case of the Gaussian Mixture Models (GMM) method, the number of mixtures was varied between 1 and 10 to determine the optimum value. Further to the parameters considered above for chroma, and the highest estimation (85.7%) occurs when 1/8-overlapped frames with non-sparse chroma and 6 mixtures are used (see **Table 39**).

The effect of onset detection

Onset detection function affects the recognition rate. By starting the analysis frame an optimum distance from the onset, notes of short duration and rapid ornaments (e.g. consecutive quartertones and other ornaments involving intervals not usually allowed in a particular mode) can be excluded. Furthermore, some onsets can be skipped: for instance, if the analysis data is taken from the third onset instead of from the start of the file, the recognition rate slightly increases, from 80.22% to 83.3% with no-overlap and slightly increases from 74.3% to 74.5% with 1/8-overlapped frames.

Comparing similarity measures

Table 30 compares Manhattan distance, dot-product, and bit-mask. The recognition rates are 80.22%, 72.53% and 50.55% respectively. Whereas in comparing Manhattan distance and dot-product, data-driven templates were used, scale intervals were used in the bit-mask comparison experiment. Manhattan distance yields the highest recognition rate.

Symbolic versus data-driven templates

Data-driven templates, which are customised to the existing training samples, are dependent on overtones, the inharmonicity factor (the characteristics which vary between different instruments), and tone occurrence; consequently, a wide variety of training samples should be included in the

database to generalise the data-driven templates. Because of these shortcomings of data-driven templates, symbolic templates based on theory were made as an alternative (see 7.2.3). Unlike data-driven templates, symbolic templates are independent of the instrumentation and harmonic content of the training samples. However, they do not include all characteristics of the data; for example, tone occurrences and deviations from simple theoretical models in pitch or harmonic content (e.g. rationalisation of various sizes of quartertone to 24-TET) characteristic of real audio signals are not included. In general, a data-driven template based on audio samples of recorded music, in contradistinction to theoretically based templates, has these advantages: unlike 0-1 scalar templates it includes tone occurrences; and unlike symbolic-based templates, it learns the harmonic characteristics of the ensemble. Although learning the harmonic characteristics of an instrument or ensemble may increase the average recognition rate, it can also be conceived of as a disadvantage, as the overriding aim is to create a system that works regardless of instrumentation.

Data-driven and symbolic-based templates show the relative importance of scale notes, while scalar templates with equal weights do not reflect the relative importance of notes.

The effect of making equally-weighted data-driven templates was investigated: as an alternative classification scheme, the pattern of each *dastgàh* was set, based on the scale intervals, with ones at the scale intervals and zeros elsewhere (**Figure 32**). This yields a recognition rate of 74.73% with 1/8 overlap frames. While in this way the system becomes less dependent on timbral and harmonic content (reflecting instrumentation), some information is missed, as the relative importance of notes is not reflected in such templates. There are elements in data-driven chroma templates which are not predicted in theoretical scales; and the weights of the elements are ignored. To account for the theoretical note occurrences, histograms based on the *radif* (*dastgàh* repertoire) for the five modes (**Figure 33**) were constructed.

8.1.3 Tonic detection

Investigating the chroma of the training template generated using santur samples (db1), it was observed that tonic is the most frequently occurring note for *esfehàn*, *shur* and *chàhàrgah* modes; and that it is among the three most frequent notes for the other modes (*màhur* and *segàh*).

As the tonic is not known for musical samples in general, a mechanism for matching the tonic of a sample with the tonic of the templates is a necessary stage prior to mode and scale recognition.

The samples are bit-masked with scale templates (or dot-producted with theoretical or data-driven templates) and summed up. As a heuristic solution, the tonic was detected by comparing the distance between a sample's chroma and the scale templates for all modes, and the results (assuming that all templates have the same tonic) were summed up. The templates were then shifted by a quartertone and the same process was pursued for 23 steps. After performing the shift-and-sum process 23 times, the tonic is spotted via the minimum distance. The distance between a sample's chroma average and the shifted version of the sum of all templates is calculated, and the position of the tonic is then indicated by the component shift that corresponds to the shortest distance (it is assumed that the tonic does not change throughout a performance and that all the templates are in the same tonality). Subsequently, the mode is identified, as explained in Section 7.2.2.

It is important to know in which particular tonality a templates is. While for theory-based templates this is known, for data-driven templates, based on audio musical files, if the tonic is identified by expert knowledge or automatically, the templates can be shifted and aligned accordingly. Alternatively, the feature vector of a test sample and the training templates can be compared directly (with no tonic detection in pre-processing) by a 23 shift-and-compare process.

8.2 Applications of the modal analysis

Although the main focus of this research is Persian music, the results are applicable to other traditions and cultures, especially those across the Near East and Mediterranean regions, and further adjustments could be extended to other scales, including the pentatonic scale and the Major and minor scales of Western music. A publicly available system can be implemented to provide access to the archived musical content. For instance, a system that is able to browse files of different online archives which are in the same mode, through an audio example (query-by-example).

There are numerous potential applications for this research, including music information retrieval; audio snippet (thumbnail); music archiving and access to archived musical content; audio compression and coding; associating images with audio content; music transcription; music synthesis; automatic music accompaniment; music editing; music instruction; setting new standards and symbols for musical notation; and copyrighting. With the recent advances in the production of versatile personal technologies, such as iPhones, application programmes can be developed, based

on the theories developed and presented in this thesis, extending to large groups of people the possibility of benefiting from the algorithms outlined here.

8.3 Limitations of the approach

The method assumes a monophonic or polyphonic signal with a strong melodic line, with reasonable signal to noise ratio, because otherwise noise or other transients would be interpreted as scale intervals, leading to additional errors. Non-pitched percussive signals or a pitched percussion where the pitches do not correspond to scale intervals, are undesirable. A drone, which is common in Iranian music (for instance in a *chàhàrmezràb*), increases the frequency of occurrence of one or more notes, and may lead to false recognition results; to address this, an optimum saturation level for notes should be established (see sparse chroma in section 7.2.3), so that the number of occurrences of a note and its energy are bounded by a certain threshold. A one-octave scale is assumed in chroma and pitch histogram features, whereas in some cases a two-octave scale exists in practice (see section 8.4).

The santur has a clear and a more or less stable, fixed-pitched sound. On other Persian instruments, such as *kamàncheh* (spike fiddle) and *ney* (a vertical flute), where fingering or airflow determine the notes, instability of pitch can affect the results; and intervals and pitch inflection are even more flexible in vocal music.

8.4 Future research directions

For future work, a more diverse database needs to be constructed including samples in additional derivative modes and more vocal and multi-instrumental samples. Musical knowledge (e.g. recognising intervals alien to a certain mode, which are not expected to occur); and the order of the notes can also be reflected in the methods. And a mechanism should be devised to encompass additional modes, and modern compositions and improvisations which may not necessarily follow the traditional scale and modal system at all times.
In this research the octaves were folded into one; however, in practice some Persian modes span more than one octave and have different tunings in different octaves. For instance, for a *shur* from a modal tonic of E, the lower F is *sori* (Fs), while the higher F is natural (F). As any piece can be played from various tonalities and in different octaves, the chroma and pitch histograms in this research were limited to one octave. A modified version of chroma and pitch histograms can be made to include more than one octave in calculations.

In order to make the templates independent of timbre and instrument, it is proposed that general templates, based on theoretical intervals that are independent of the training data, should be developed. It would be possible to weight scale degrees according to their anticipated prominence in a performance in a particular mode; a more detailed assessment is needed to establish the relative importance of intervals in *dastgàh* recognition.

In order to eliminate unnecessary notes and noise, a sparse chroma, considering only the 13 highest peaks, was created. Because the harmonics of a note are folded to the ground octave, summed up with the fundamental frequency of the note, and added to the elements of the sample or template, as many non-harmonic elements as possible should be eliminated, because non-harmonic components, related to inharmonicity and other factors are detrimental to templates. In a future investigation, to assess the effect of lowering the resolution to Western semitones, or of increasing it, the tests could be repeated in 12-TET and other resolutions.

The test results show that the choice of features and classifiers affects the type of errors arising. Thus alternative features and classifiers could be considered, and the results can be combined in order to increase accuracy.

It is noteworthy that although computers have limitations in comparison with human perception, there are tasks in which machines excel; there is a strong possibility that machines could capture

new features and attributes from music that human beings would not be able to perceive, or which it would take a long time for a person to perceive. This can be used in creating new tags and perhaps also in the formation of new patterns and fashions of listening.

The tonic detection system could also be enhanced. The weights of theoretical or data-driven templates might be adjusted, based on tone sequences and the importance of scale notes, in order to improve tonic detection. Furthermore, the effect of constructing a combination of theoretical and data-driven templates could be examined to make templates which are less dependent on the data.

Symbolic versus data-driven templates

Most templates in this research were data-driven, made on the basis of real audio data. Such templates are dependent on instrumentation, and their elements are influenced by overtones: harmonics of the tones, the inharmonicity factor, and tone occurrences. In the case of GMM, the order of the notes is also significant.

An alternative is to build symbolic templates based on theory, which are independent of instrumentation and training samples. However, symbolic templates do not include the deviations from the theory which exist in real musical signals.

One solution is to construct symbolic templates, based on Persian music theory, where the elements are weighted according to importance of the notes in a typical performance. Another way is to build note histograms (tone profiles) based on notated music (as in Chapter 7). For instance, **Figure 33** shows tone profiles constructed based on a few introductory pieces in *radif* [70]. The tone profiles can be combined with more pieces to provide a more general representation of each *dastgàh*. The effect of using such templates in comparison with data-driven templates and equally-weighted templates should be investigated and compared further.

Noise and transients

Noise and transients are another source of error in making the templates. It would be preferable to remove non-pitched sounds such as percussion strokes, especially percussive sounds with slight pitch changes during the transients, from the signal. It should be noted that when the four clustered strings of a santur are struck all the others vibrate to some extent, due to the vibrations of the resonating body. Although the averaging process reduces the effect of noise and transients to a large extent, these latter affect the GMM results significantly, as all frames are used in this method. Transient and non-pitched voice removal could be added in pre-processing.

Tracking modal modulation

The analysis method presented here assumes a single mode for a whole piece. To capture modal progression, a frame-by-frame analysis method or another segmentation stage has to be applied. A straightforward way of doing this would be to follow the energy envelope and to divide the signal into different sections, notably when modulation corresponds to a pause or when the energy falls below a certain level. However, it should be noted that the mode may also change without playing being interrupted.

The minimum length of segments for modulation analysis is dependent on the style and tempo of the music; protocols for minimum sample lengths would need to be developed. In Persian classical performance, mode progression usually takes several minutes. Because it is unlikely that a mode change would occur in less than a certain amount of time (less than 30 seconds is exceptional in Persian music), the feature average could be made over such a time period, and the modulations be tracked correspondingly. However, in modern compositions and improvisations, mode change can occur more frequently than this, and at any time.

Dastgàh recognition is more complicated than simple chord and key recognition. In a typical Persian performance it may take a long time to recognise the scale, since certain notes, interval patterns and, in some modes, particular melodic features need to be heard and recognised to distinguish between modes. Examining the start and end of a piece would be particularly helpful because, generally speaking, performances start in one mode, move to other mode(s), and return to the starting mode to end.[93] Modern compositions and improvisations, however, do not necessarily return to the initial mode.

Sufficiency of information to make tone profiles and pitch histograms

The diagnostic notes which differentiate significantly between modes might not necessarily occur in the portion of a file available for analysis. It should be noted that in addition to the frequency of their occurrence, some scale tones, owing to their role in phrases and their relationship to other notes, are more important than the others. Krumhansl [71], Gomez [37] and Pauws [45] view the disregarding of tone order as a weakness of tone profiles.

[93] Another option is to estimate the mode in the beginning, at the end, and across the whole file, and to compare them to establish their relationship, and then make logical conclusions accordingly. When a modulation occurs, the mode in which the piece remains principally, as opposed to the other modes, can be taken as the main mode of the piece.

Summary of proposed further research

Current research on mode estimation can be extended in various ways:

- To evaluate the analysis methods in relation to a more diverse database, more samples, including vocal and polyphonic samples of different instruments in different tonalities, could be recorded.
- Regarding the analysis of the santur alone, further work is needed to find a more precise value for the inharmonicity factor, involving more notes and higher harmonics in the inharmonicity calculations. The inharmonicity factor of different santurs could also be calculated and compared.
- Other parameters of the santur, such as the resonance frequencies of the instrument's body, should be measured, to calculate the transfer function of the instrument in order better to understand the instrument. The resonance frequencies of the instrument's body were not considered here, as they just affect the amplitudes of the harmonics, not their positions; thus they do not affect the chroma feature calculations and mode identification that is a focus of this thesis.
- Knowledge of the notes that occur and that usually do not occur in a mode should be assessed and incorporated: in order to adjust the weight of the symbolic templates, research should be pursued on the relative importance of each scale degree.
- Work should be pursued on segmenting the audio files in order to find the boundaries of each mode and thereby to track modulations.
- As the scales of some of the modes exceed one octave, the possibility of extending the scale analysis to encompass two or more octaves should be investigated. For instance, in a *shur* performance from E (tonic), the first-octave F is *sori* (half sharp), whereas the second-octave F is natural.
- To capture the course of a complete *dastgàh* performance, it would be necessary to track modulations comprehensively.
- As the Persian *dastgàh* system is essentially ordered melodically rather than modally, melodic similarity methods could be applied; melodic patterns would be recognised and used in comparing audio samples. In particular, there are different modes with identical scales, such as *shur* and *navà*, which can be distinguished only by considering both their tonics and interval patterns, and there are different modes with similar scales and modal tonics, such as *afshàri* and *abu'atà*, which can be distinguished by melodic motifs.

- Knowing the mode and having recourse to melodic resources, standard melodic patterns typical of the classical musical repertory could be applied in music synthesis and automatic music accompaniment.
- Finally, the ultimate goal would be to perform a modal analysis on streaming audio in real time.

9 REFERENCES

[1] Gómez, E., Haro, M. and Herrera, P., “Music and geography: content description of musical audio from different parts of the world”, *Proceedings of the 10th International Conference on Music Information Retrieval* (ISMIR), Kobe, Japan, 2009.

[2] Heydarian, P., “Music note recognition for Santoor”, MSc thesis, Tarbiat Modarres University, Tehran, 2000 (supervised by E. Kabir).

[3] Gedik A., Bozkurt B., “Pitch-frequency histogram based music information retrieval for Turkish music”, *Signal Processing*, vol. 90, issue 4, April 2010, pp. 1049–63.

[4] Gedik, A. C., Bozkurt, B., “Evaluation of the makam scale theory of Arel for Music Information Retrieval on traditional Turkish art music”, *Journal of New Music Research*, vol. 38(2), 2009, pp. 103–116.

[5] Heydarian, P., “Applying signal processing techniques to Persian music and santur Instrument”, MPhil thesis, Queen Mary, University of London, 2008 (supervised by Josh Reiss and Mark Plumbley).

[6] Harte, C. A., Sandler, M. B., “Automatic chord identification using a quantised chromagram”, *Proceedings of the 118th Convention of the Audio Engineering Society*, Barcelona, May 2005.

[7] Brown, J. C., "Calculation of a constant Q spectral transform", *Journal of the Acoustical Society of America*, vol. 89, 1991.

[8] Duda, R. O., Hart, P. E., and Stork, D. G., *Pattern classification*, second edition, Wiley, Oxford, 2001.

[9] Noland, K., “Computational tonality estimation: signal processing and Hidden Markov Models”, PhD thesis, Queen Mary, University of London, 2009.

[10] Noland, K. and Sandler, M., “Key estimation using a Hidden Markov Model”. *Proceedings of the 7th International Conference on Music Information Retrieval* (ISMIR), Victoria, Canada, 2006.

[11] Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M. and Sandler, M. B. “A tutorial on onset detection in music signals”, *IEEE Transactions on Speech and Audio Processing*, September 2005.

[12] Yarshater, E., “Persia or Iran, Persian or Farsi”, *Iranian Studies*, vol. 22, no.1, 1989.

[13] Vaziri, A. N., *Dastur-e tàr*, Tehran, 1913.

[14] Farhat, H., *The dastgàh concept in Persian music*, Cambridge University Press, Cambridge, 1990.

[15] During, J., *Musique et mystique, dans les traditions de l'Iran*, Porsesh Publications, Tehran, 1999.

[16] Miller, L. C., *Music and song in Persia: The art of àvàz*, University of Utah Press, Salt Lake City, 1999.

[17] Nooshin, L., "Improvisation as 'other': creativity, knowledge and power – The case of Iranian classical music", *Journal of the Royal Musical Association*, vol. 128, 2003.

[18] Wright, O., *Touraj Kiaras and Persian music: An analytical perspective*, SOAS Musicology Series, Ashgate, Farnham, 2009.

[19] Nettl, B., *The study of ethnomusicology: Twenty-nine issues and concepts*, Illinois Books editions, University of Illinois Press, Urbana & Chicago, 1983.

[20] Setayeshgar, M., *Lexicon of Iranian music*, Ettela'at Publications, Tehran, 1986.

[21] Masudi, A., *The meadows of gold: The Abbasids*, Translated and edited by Paul Lunde and Caroline Stone, Keegan Paul International, London, 1989.

[22] Mashhoon, H., *History of Iranian music*, Rokh Publications, Tehran, 1984.

[23] Arnold, D., *The New Oxford Companion to Music*, Oxford University Press, Oxford, 1984.

[24] Hornbostel, E. M. von, and Sachs, C., "Systematik der Musikinstrumente. Ein Versuch", *Zeitschrift für Ethnologie*, vol. 46 (1914), pp. 553–590; translated by Anthony Baines and Klaus Wachsmann as "Classification of Musical Instruments", *Galpin Society Journal*, vol. 14, pp. 3–29, 1961.

[25] Shepard, R., "Circularity in judgements of relative pitch". *Journal of the Acoustical Society of America*, vol. 36, 1964, pp. 2346–2353.

[26] Tzanetakis, G., Kapur, A., Schloss, W. A. and Wright, M., "Computational ethnomusicology", *Journal of Interdisciplinary Music Studies*, vol. 1, issue 2, fall 2007, art. 071201, pp. 1–24.

[27] Kassler, M., "Toward musical information retrieval", *Perspectives of New Music*, 1966.

[28] Downie, J. S., Byrd, D. and Crawford, T., "Ten years of ISMIR: reflections on challenges and opportunities", in *Proceedings of the 10th International Conference on Music Information Retrieval* (ISMIR), Kobe, Japan, 2009.

[29] Tzanetakis, G., talk at IEEE Signal Processing Society, 2011 (online, accessed March 2013): http://www.brainshark.com/brainshark/brainshark.net/portal/title.aspx?pid=zCZz12TSh9z0z0

[30] Sheh, A., and Ellis, D. P. W., "Chord segmentation and recognition using EM-trained Hidden Markov Models", *Proceedings of the 4th International Conference on Music Information Retrieval* (ISMIR), Baltimore, 2003.

[31] *The New Grove Dictionary of Music and Musicians*, second edition, edited by Stanley Sadie, vol. 5, 2001.

[32] Izmirli, O., "Audio key finding using low-dimensional spaces", in *Proceedings of the 7th International Conference on Music Information Retrieval* (ISMIR), Victoria, Canada, 2006.

[33] Marolt, M., "Gaussian mixture models for extraction of melodic lines from audio recordings". *Proceedings of the 5th International Conference on Music Information Retrieval* (ISMIR), Barcelona, 2004.

[34] Chew, E., "The spiral array: an algorithm for determining key boundaries", *Music and Artificial Intelligence, Lecture Notes in Computer Science*, vol. 2445, 2002, pp. 18–31.

[35] Chuan, C., and Chew, E., "Audio key finding: considerations in system design and case studies on Chopin's 24 preludes", *EURASIP Journal on Advances in Signal Processing*, 2007.

[36] Papadopoulos, H. and Tzanetakis, G., "Modelling chord and key structure with Markov Logic", *Proceedings of the 13th International Conference on Music Information Retrieval* (ISMIR), USA, 2012.

[37] Gómez E., "Tonal description of music audio signals", PhD thesis, Universitat Pompeu Fabra, 2006.

[38] Gómez, E., Haro, M. and Herrera, P., "Music and geography: content description of musical audio from different parts of the world", in *Proceedings of the 10th International Conference on Music Information Retrieval* (ISMIR), Kobe, Japan, 2009.

[39] Gómez, E., Herrera, P., "The song remains the same: identifying versions of the same piece using tonal descriptors", *Proceedings of the 7th International Conference on Music Information Retrieval* (ISMIR), Victoria, Canada, 2006.

[40] Chordia, P., and Rae, A., "Raag recognition using pitch-class and pitch-class dyad distributions", *Proceedings of the 8th International Conference on Music Information Retrieval* (ISMIR), Vienna, 2007.

[41] Heydarian, P., Reiss, J. D., "The Persian music and the santur instrument", *Proceedings of the 6th International Conference on Music Information Retrieval* (ISMIR), London, 2005.

[42] de Cheveigne, A., Kawahara, H., "Yin: a fundamental frequency estimator for speech and music", *Journal of the Acoustic Society of America*, vol. 111, 2002, pp. 1917–1930.

[43] Leonidas, I., "Estimating the makam of polyphonic music signals: template-matching vs. class-modelling", MSc thesis, Universitat Pompeu Fabra, Barcelona, 2010.

[44] Leonidas, I., Gomez, E., Herrera, P. "Tonal-based retrieval of Arabic and middle-east music by automatic makam description", *9th International Workshop on Content-based Multimedia Indexing*, Madrid, 2011.

[45] Pauws, S., *Keyex: Audio key extraction*. MIREX Audio Key Finding entry (online, accessed December 2012), 2005.
http://www.music-ir.org/evaluation/mirex-results/articles/key_audio/pauws.pdf

[46] Pikrakis, A., Theodoris, S., Kamaratos, D., "Classification of musical patterns using variable duration Hidden Markov Models", *IEEE Transactions on speech and audio processing*, vol. 11, no. 2, May 2003, pp. 1795– 1807.

[47] Tolonen, T.; Karjalainen, M., "A computationally efficient multipitch analysis model", *IEEE Trans. on Speech and Audio Processing*, vol. 8, issue 6, November 2000, pp. 708 – 716.

[48] Meddis, R. and O'Mard, L. "A unitary model for pitch perception," *Journal of Acoustical Society of America*, vol. 102, September 1997, pp. 1811–1820.

[49] Klapuri, A., "Automatic Transcription of Music", MSc thesis, TUT University, Finland, 1998.

[50] Darabi, N., Azimi, N. H., Nojumi, H. "Recognition of *dastgàh* and *maqàm* for Persian music with detecting skeletal melodic models", *Proceedings of The second annual IEEE BENELUX/DSP Valley's Annual Research & Technology Symposium (DARTS)*, Belgium, 2006.

[51] Abdoli, S., "Iranian traditional music *dastgàh* classification", *Proceedings of the 12th International Conference on Music Information Retrieval* (ISMIR), Miami, 2011.

[52] Heydarian, P., Kabir, E., Lotfizad, M., "Music note recognition for santur", *7th Annual Conference of the Computer Society of Iran*, 2001.

[53] www.shazam.com (online, accessed May 2015).

[54] www.soundhound.com (online, accessed May 2015).

[55] Chandrasekhar, V., Sharifi, M. and Ross, D. A., "Survey and evaluation of audio fingerprinting schemes for mobile query-by-example applications", *Proceedings of the 12th International Conference on Music Information Retrieval* (ISMIR), Miami, 2011.

[56] Wang, A. L., "An industrial-strength audio search algorithm", *Proceedings of the 4th International Conference on Music Information Retrieval* (ISMIR), Baltimore, 2003.

[57] Ellis, D., "Robust landmark-based audio fingerprinting", web resource, available: http://labrosa.ee.columbia.edu/MATLAB/fingerprint/, 2009 (online, accessed 28 April 2015).

[58] Fletcher, N.F., Rossing, T. D., *The physics of musical instruments*, 2nd edition, Springer-Verlag, New York, 1998.

[59] Ortiz-Berenguer, L., Casajús-Quirós, F., Torres-Guijarro, M., Beracoechea, J. A., “Piano transcription using pattern recognition: aspects of parameter extraction”, *Proceedings of DAFx04 conference*, Naples, 2004.

[60] Martin, D. W., Ward, W. D., “Subjective evaluation of musical scale temperament in pianos”, *Journal of the Acoustical Society of America*, vol. 94, no. 1, 1993, pp. 46–53.

[61] Oppenheim, A. V., Schafer, R. W., Buck, J. R., *Discrete-Time Signal Processing*, Prentice Hall, 1999.

[62] Gómez, E., Haro, M. and Herrera, P., “Music and geography: content description of musical audio from different parts of the world”, *Proceedings of the 10th International Conference on Music Information Retrieval* (ISMIR), Kobe, Japan, 2009.

[63] Chuan, C. H. and Chew, E., “Fuzzy analysis in pitch class determination for polyphonic audio key finding”. *Proceedings of the 6th International Conference on Music Information Retrieval* (ISMIR), London, 2005.

[64] Peeters, G., “Musical key estimation of audio signal based on hidden Markov modelling of chroma vectors”. *Proceedings of the 9th International Conference on Digital Audio Effects* (DAFX06), Montreal, 2006.

[65] Monti, G., “A Multipitch Prediction-driven Approach to Polyphonic Music Transcription”, PhD thesis, Queen Mary, University of London, 2005.

[66] Shepard, R., *Pitch perception and measurement*. In Perry Cook, editor, *Music, Cognition, and Computerized Sound: An Introduction to Psychoacoustics*, MIT Press, Cambridge, MA, 1999, pp. 149–165.

[67] Reynolds, D. A. "Speaker identification and verification using Gaussian mixture speaker models”, *Speech Communication*, vol. 17, 1995, pp. 91–108.

[68] Heittola, T. and Klapuri, A., “Locating segments with drums in music signals”, in *Proceedings of the 3rd International Conference on Music Information Retrieval* (ISMIR), Paris, 2002.

[69] Tzanetakis, G., “Manipulation, analysis and retrieval systems for audio signals”, PhD thesis, Princeton University, 2002.

[70] Talai, D., *The radif of Mirzà Abdollàh*, Mahoor Institute of Culture and Art, Tehran, 1998.

[71] Krumhansl, C. L., *Cognitive foundations of musical pitch*, Oxford University Press, New York, 1990.

[72] Powers, D. M. W., "Evaluation: from precision, recall and f-measure to ROC, informedness, markedness & correlation", *Journal of Machine Learning Technologies*, 2011.

[73] *The Harvard Concise Dictionary of Music and Musicians*, The Belknap Press of Harvard University Press, Cambridge, MA, 1999.

[74] Nettl, B., Encyclopedia Britannica (online, accessed 15 May 2015): http://www.britannica.com/EBchecked/topic/363629/maqam

Appendix I: List of abbreviations

ACF Auto Correlation Function

CV Cross-validation

DSP Digital Signal Processing

FFT Fast Fourier Transform

FPH Folded Pitch Histogram

GMM Gaussian Mixture Model

HES High Energy Suppression

HMM Hidden Markov Model

HPCP Harmonic Pitch Class Profile

MAP Maximum a Posteriori

MIDI Musical Instruments Digital Interface

MIR Music Information Retrieval

MLN Markov Logic Networks

MVN Multivariate Likelihood Model

PCP Pitch Class Profile

PH Pitch Histogram

SS Silence Suppression

SVM Support Vector Machine

TET Tone equal temperament (e.g. 12-TET; 24-TET)

UPH Unfolded Pitch Histogram

Appendix II: Glossary of terms

Cent: unit of scientific method for measuring musical intervals that was introduced by A. J. Ellis (1804– 90). One cent is a one-hundredth of a semitone [23].

Chord: the simultaneous sounding of two or more notes [31].

Chroma: a Q-transform, folded into one octave [6]. See also pitch class.

Dastgàh: a mode in Persian classical music. Persian music is based on a modal system of seven main modes and their five derivatives that are collectively called the twelve *dastgàh*s [2, 10].

Decimation: the process of low-pass filtering followed by down sampling [61].

Down sampling: the process of reducing the sampling rate of a signal [61].

F-measure: a measure of the performance of a system. F measure is: 2/(1/r+1/p), where p is precision and r is recall [72].

Key: usually relates to the tonic of a piece. The key of a movement may change through a process, called modulation, but at the end, there is usually a return to the original key. In tonal music, key is a pitch relationship that sets a single pitch class as a tonal centre or tonic (or key note), with respect to which the remaining pitches have subordinate functions [73].

Maqàm: a set of pitches and characteristic melodic patterns or motifs in music of the Middle East and parts of North Africa and the traditional pattern of using them [74, 18]; or a mode in Iranian folk music [5].

Microtones: intervals that are less than a semitone. However, we do not need to use this term here as the smallest interval of interest is a three-quartertone (which in Persian music is not necessarily precisely a quarter of a tone).

Mode: a concept referring to melody type or scale type. Mode has been used to show classes of melodies; since the 20^{th} century, it has been used as a framework for composition and improvisation [40]. A mode in this thesis refers to a *maqàm* or *dastgàh*, not the mode in European medieval music or other traditions, unless otherwise stated.

Music Information Retrieval: interdisciplinary science of retrieving information from music [26].

Pitch class: is a set of all pitches in different octaves that refer to a specific note. For instance, pitch class A refers to A0, A1, A2, A3, … [6] See also ‘chroma’.

Precision: in pattern recognition, precision or positive predictive value is the ratio of retrieved to relevant samples: p=true positive/(true positive+false positive) [72].

Q-transform: is a simplified version of a signal’s spectrum, close to human perception, obtained by applying increasing width bins to merge the spectrum components [7].

Quartertone: literally, half a semitone; however, the quartertone in Persian music is flexible in size and can be less or more than half a semitone, depending on the mode, the piece, the performer’s mood, and also on the geographical region [5, 14]. For example, the quartertone sharpening in Turkish music is a greater interval than the quartertone sharpening in Persian music.

Recall: in pattern recognition, recall is the ratio of recognised samples to relevant samples. r=true positive/(true positive+false negative) [72].

Appendix III: Samples of codes

Samples of the following codes are included in the enclosed DVD.

Parameters common to all methods:

"nod" is Number of *dastgàh*s and can be 5 for the various Persian scales or 6 to be inclusive of a pentatonic scale as well.

"ss": silence suppression (0 or 1)

"hs": high-energy suppression (0 or 1)

"Nf": frame size, assigned through i_n

Files common to all methods (paths should be modified according to database folders):

"music_files5.m" file includes the vectors that contain labelled names of audio files of db1;

"paths.m" sets the main path of the database files and the subfolders.

1. Spectral average – Manhattan distance and cross-correlation

Files:

- "ph_sig_to_specmean_27_1_santur.m" constructs and stores the spectral average feature vectors for both training and test samples through functions "spec_calc_52_train(Nf(i_n),Nl(i_n),Nh(i_n),P2,dn,dir1_train,ss,hs)" and "spec_calc_52_test(Nf(i_n),Nl(i_n),Nh(i_n),P2,dn,dir1_test,ss,hs)" respectively; by setting up ss=1 and hs=1, silence suppression and/or high-energy suppression is performed; otherwise they should be set to 0
- "test_13_select_measure_2_1.m" runs the tests, using various classifiers marked by "label": "test_13_minkowski_2.m", "test_13_dotproduct_2.m", "test_13_cc.m".

2. Pitch histograms – Manhattan distance and cross-correlation

Files:

"aggelos_PH_03_training" calculates the pitches and constructs the pitch histograms

"test_F0_10_Manhattan.m" runs the tests with various classifiers (Manhattan distance or cross-correlation); alternatively, "test_F0_10_bitmask.m" identifies the dastgàh by masking the signal's feature vector with scale intervals.

3. Chroma – Manhattan distance and cross-correlation

Parameters:

info.fs = 0; % Sampling freq of audio file

info.fsdownsampled = 0; % fs after downsampling

info.downsample = 8 %4 % downsampling factor

info.preprocessing = 'downsample'; % type of preprocessing

info.numberframes = 0; % number of frames calculated

info.windowlength = 0; % window length in samples -- This parameter will be set automatically (in getparameters)

info.overlap = 1/8; %1 for no overlap % default 1/8 % window overlap factor

info.hopsize = 0; % window overlap in samples -- This parameter will be set automatically (in framefft) based on overlap

info.framespersecond = 0; % effective frames per second

% Constant Q parameters:

info.binsperoctave = 72; % bins per octave for const q

info.fmax = 1400; % top frequency for const q

info.fmin = 130; % bottom frequency for const q

info.numcqbins = 0; % number of bins in a Constant Q vector

info.sparkernelfile = ''; % mat file containing sparse spectral kernels matrix

info.sparsethreshold = 0.0054; % sparsekernel threshold

Setting the tone resolution:

% to change the resolution to a quartertone or (other tone resolutions), the parameters in the

the following files are changed to change the tone resolution to 12-TET or else:

% hpcp.m: 36==>72 & 1:35 ==> 1:71

% hpcppeak: 12==>24 & 36 ==>72

% newtune: 1:12 ==> 1:24 & tuningvectors = zeros(24,2);

% tracks: lines 8, 26, 27 & 36

% semiwindow: line 17

Files:

- "chromaexample_01.m" constructs the chroma average templates and saves them;
- "radif_hist_04_1.m" has tone histograms, based on radif;
- "radif_hist_04_2_w_ornaments.m" has tone histograms, based on radif, also including *esharé* and *tekié* ornaments
- "test_chroma_05_04_2_3_man_dot_bit_tonic_W_shift.m" constructs the training templates and runs the tests, using Manhattan distance and dot-product, with 23-shift
- "test_chroma_03_cc.m" runs the tests, using cross-correlation

4. Chroma – GMM

The chroma parameters are the same as above.

Number of GMM mixtures.

Files:

- "chromaexample_03_polyphony.m" constructs chroma feature vectors for all samples;
- "music_gmm1_train.m" trains the GMM models with

- Then “music_gmm1_test.m” calculates the probabilities respect to the existing GMM model;
- “music_gmm1_train_and_test2_1.m” includes both training and test, using which the optimised values of parameters such as the number of mixtures can be determined.

Appendix IV: Author's Publications

Part of the work that is presented in this thesis has roots in former publications of the author which are augmented by new material. Here is a list of the author's publications:

1. Heydarian, P., Jones, L., "Tonic and scale recognition in Persian audio musical signals", *The12th International conference on Signal Processing* (ICSP2014), Hangzhou, China, October 2014.
2. Heydarian, P., "Automatic identification of the Persian musical modes in audio musical signals", *1st International Digital Libraries for Musicology workshop* (DLfM), London, September 2014.
3. Heydarian, P., Jones, L. and Seago, A. "Automatic mode estimation of Persian musical signals", *Proceedings of the 133rd Convention of the Audio Engineering Society*, San Francisco, October 2012.
4. Heydarian, P. and Jones, L. "Measurement and calculation of the parameters of Santur", *Proceedings of the Annual Conference of the Canadian Acoustical Association* (CAA), Vancouver, Canada, October 2008.
5. Heydarian, P., Jones, L. and Seago, A. "Methods for identification of the tuning system in audio musical signals", *Proceedings of the 125th Convention of the Audio Engineering Society*, San Francisco, October 2008.
6. Heydarian, P., "Applying signal processing techniques to Persian music and santur instrument", MPhil thesis, Queen Mary, University of London, 2008 (supervised by Josh Reiss and Mark Plumbley).
7. Heydarian, P., Jones, L. and Seago, A. "The analysis and determination of the tuning system in audio musical signals", *Proceedings of the 123rd Convention of the Audio Engineering Society*, New York, October 2007.
8. Heydarian, P., Reiss, J.D. "Extraction of long-term rhythmic structures using the Empirical Mode Decomposition", *122nd Convention of the Audio Engineering Society*, Vienna, May 2007.
9. Heydarian, P., Reiss, J.D., "Extraction of long-term structures in musical signals using the Empirical Mode Decomposition", *Proceedings of the 8th International Conference on Digital Audio Effects* (DAFX05), Madrid, 2005.

10. Heydarian, P., Reiss, J.D., "The Persian music and the santur instrument", *Proceedings of the 6th International Conference on Music Information Retrieval* (ISMIR), London, 2005.

11. Heydarian, P., Reiss, J. D., “A database for Persian music”, Digital Music Research Network (*DMRN) Summer Conference*, Glasgow, 2005.

12. Heydarian, P., Reiss, J. D., "Pitch determination of santoor signals", *Prep2005*, Lancaster, 2005.

13. Moin, M.S., Barzegar, N., Maghooli, K., Naderi, S., Heydarian, P., "Biometrics: future of person authentication in information and communication technology", *Second Workshop on Information Technology and Its Disciplines* (WITID), Kish Island, Iran, February 2004.

14. Heydarian, P., Kabir, E., Lotfizad, M., “Music Note Recognition for santoor”, *7th Annual Conference of the Computer Society of Iran*, February 2001.

15. Heydarian, P., Kabir, E., Lotfizad, M., “Music Note Recognition for Santoor”, MSc thesis, Tarbiat Modarres University, Tehran, 2000.