Convolutional Methods for Music Analysis

Velarde, Gissel

DOI (link to publication from Publisher):
10.5278/vbn.phd.tech.00005

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

CONVOLUTIONAL METHODS FOR MUSIC ANALYSIS

BY
GISSEL VELARDE

DISSERTATION SUBMITTED 2017

AALBORG UNIVERSITY
DENMARK

Convolutional Methods for Music Analysis

Ph.D. Dissertation by

Gissel Velarde

Department of Architecture, Design and Media Technology
Aalborg University, Denmark

January, 2017

Dissertation submitted: January, 2017

PhD supervisor: Associate Professor David Meredith
Aalborg University

Assistant PhD supervisor: Senior Lecturer Tillman Weyde
City, University of London

PhD committee: Associate Professor Cumhur Erkut (chairman)
Aalborg University

Professor José Manuel Iñesta Quereda
University of Alicante

Associate Professor Emilia Gómez
Universitat Pompeu Fabra

PhD Series: Technical Faculty of IT and Design, Aalborg University

ISSN (online): 2446-1628
ISBN (online): 978-87-7112-887-1

Published by:
Aalborg University Press
Skjernvej 4A, 2nd floor
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Rosendahls, 2017

Dedicado a Sol Samantha y a Kira Isabel

Author's Curriculum Vitae

Gissel Velarde holds a *Licenciatura* degree in Systems Engineering from the Bolivian Catholic University and a Master of Science in Electronic Systems and Engineering Management from the South Westphalia University of Applied Sciences, where she was a DAAD scholarship holder between 2006 and 2008. Between 2008 and 2012, she worked in industry as a data analyst at Miebach GmbH in Germany. In 2010, she received a Best Paper Award nomination at the Industrial Conference on Data Mining in Berlin. In October 2012, she began her PhD study under the supervision of David Meredith and Tillman Weyde, supported by a PhD fellowship from the Department of Architecture, Design and Media Technology, Aalborg University. Between 2015 and 2016, Velarde was a research member of the European project, "Learning to Create" (Lrn2Cre8), a collaborative project within the Future and Emerging Technologies (FET) programme of the Seventh Framework Programme for Research of the European Commission.

Between 1990 and 2002, Velarde studied piano at the *Conservatorio Plurinacional de Música* in La Paz, Bolivia. She won first and second prizes at the National Piano Competition in Bolivia (in 1994 and 1997 respectively). She also released two music albums.

Velarde has published research papers dealing with computational methods for music analysis, music information retrieval, machine learning and data analysis. During her PhD study at Aalborg University, she has supervised student projects on the Bachelor program in Medialogy and was a teaching assistant on the Master of Science program in Sound and Music Computing.

Abstract

This dissertation presents novel convolution-based methods for music analysis. The aim of this research project was two-fold: first, to design, implement and evaluate a convolution-based automated framework for the analysis of music with applications to music segmentation, pattern discovery, and classification; and second, to study convolution in relation to music-analytical and perceptual properties. In this framework, we systematically studied and evaluated the effect of filtering and other processing techniques for representation and segmentation. Moreover, we studied and optimised the parameters of filters (Haar, Morlet, Gaussian and learnt filters), and machine learning algorithms (k-nearest neighbours, single linkage, support vector machines, convolutional neural networks) in pattern discovery and classification applications. This framework was designed, implemented and evaluated in the one-dimensional (1-D) space and the two-dimensional (2-D) space. The proposed methods were intended to be as general as possible, avoiding the use of domain-specific knowledge features. We found that filtering can improve recognition compared to non-filtered representations by improving robustness to musical variations. Moreover, local processing and processing at a large scale prove to be important in music classification, and a combination of large-scale and small-scale feature extraction strategies can be complementary for ensembling. In 1-D, our convolution-based segmentation method is comparable to a state-of-the-art Gestalt-based segmentation approach in classification experiments. In the last three Music Information Retrieval Evaluation eXchange campaigns, our proposed method for the discovery of repeated themes and sections applied to monophonic symbolic music has been shown to be a competitive approach over all measures in that evaluation. In 2-D, our convolution-based ensemble of classifiers reaches the state-of-the-art on composer recognition and achieves similar performance on genre classification. Moreover, our classifiers perform equally well on symbolic and audio music data. Finally, observation of filters automatically learnt by a convolutional neural network provides musicological insight on composer style. Future work might include the evaluation of the framework on larger datasets and on tasks not related to music.

Resumé

Denne afhandling præsenterer nye foldning-baserede metoder til musikanalyse. Formålet med dette forskningsprojekt var dobbeltsidigt: For det første at designe, gennemføre og evaluere en foldning-baseret, automatiseret ramme for analysen af musik med henblik på musiksegmentering, mønsteropdagelse og klassifikation; og for det andet at studere foldning (convolution) i relation til musikanalytiske og perceptuelle egenskaber. Inden for disse rammer har vi systematisk undersøgt og vurderet effekten af filtrering og andre behandlingsteknikker til repræsentation og segmentering. Desuden har vi undersøgt og optimeret parametrene for filtrene (Haar, Morlet, Gauss og lærte filtre) og maskinlæringsalgoritmer (*k*-nærmeste naboer, single linkage, support vektormaskine, convolutional neurale netværk) i mønsteropdagelse og klassificeringsapplikationer. Denne ramme blev designet, implementeret og evalueret i endimensionale (1-D) rum og i todimensionale (2-D) rum. De foreslåede metoder havde til hensigt at være så generelle som mulige, og undgå brug af domænespecifikke vidensfunktioner. Vi fandt, at filtrering kan forbedre genkendelse i forhold til ikke-filtrerede repræsentationer ved at forbedre robusthed af musikalske variationer. Desuden viser lokal forarbejdning og behandling på en stor skala sig at være vigtig i klassificering af musik, og en kombination af funktionsekstraktionsstrategier i storskala og lille skala kan være et supplement til ensembling. I 1-D kan vores foldning-baserede segmenteringsmetode sammenlignes med en state-of-the-art gestalt-baserede segmenteringstilgang i klassificeringsforsøg. I de sidste tre Music Information Retrieval Evaluation eXchange-kampagner har vores foreslåede metode til opdagelsen af gentagne temaer og sektioner anvendt på monofonisk symbolsk musik vist sig at være en konkurrencedygtig tilgang til samtlige målinger i denne vurdering. I 2-D når vores foldning-baserede ensemble af klassificeringer state-of-the-art på komponistanerkendelse og opnår lignende resultater på genreklassificering. Desuden præsterer vores klassificeringer lige så godt i forbindelse med symbolske og audiomusik-data. Endelig giver observation af filtre automatisk lært af et convolutional neuralt netværk musikvidenskabelig indsigt i forhold til komponiststil. Det fremtidige arbejde kan omfatte evaluering af rammerne i forbindelse med større datasæt og opgaver uden tilknytning til musik.

Thesis details

Thesis Title:	Convolutional Methods for Music Analysis
PhD Student:	Gissel Velarde
PhD Supervisor:	Associate Professor David Meredith, Aalborg University
PhD Co-Supervisor:	Senior Lecturer Tillman Weyde, City, University of London
Submission date:	January, 2017.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published collection of papers listed below. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty.

- Wavelet-filtering of symbolic music representations for folk tune segmentation and classification. Velarde, Gissel; Weyde, Tillman; Meredith, David. Proceedings of the Third International Workshop on Folk Music Analysis (FMA2013). Meertens Institute; Department of Information and Computing Sciences; Utrecht University, 2013. p. 56-62.

  Publication: Article in proceedings

- An approach to melodic segmentation and classification based on filtering with the Haar-wavelet. Velarde, Gissel; Weyde, Tillman; Meredith, David. In: Journal of New Music Research, Vol. 42, No. 4, 2013, p. 325-345.

  Publication: Journal article

- A wavelet-based approach to the discovery of themes and sections in monophonic melodies. Velarde, Gissel; Meredith, David. 2014. Music Information Retrieval Evaluation eXchange, Taipei, Taiwan, Province of China.

  Publication: Conference abstract of the algorithms submitted for evaluation at the Music Information Retrieval Evaluation eXchange (see `http://www.music-ir.org/mirex/`).

- A Wavelet-Based Approach to Pattern Discovery in Melodies. Velarde, Gissel; Meredith, David; Weyde, Tillman. Computational Music Analysis. ed. / David Meredith. Cham, Switzerland : Springer, 2016. p. 303-333.

  Publication: Book chapter

- Composer Recognition based on 2D-Filtered Piano-Rolls. Velarde, Gissel; Weyde, Tillman; Cancino Chacón, Carlos; Meredith, David; Grachten, Maarten. Proceedings of the 17th International Society for Music Information Retrieval Conference. 17. ed. International Society for Music Information Retrieval, 2016. p. 115-121.

  Publication: Article in proceedings

- Convolution-based Classification of Audio and Symbolic Representations of Music. Velarde, Gissel; Cancino Chacón, Carlos; Meredith, David; Weyde, Tillman; Maarten Grachten

  Submitted: In review.

  Publication: Journal article

Preface

The idea of using convolution for music analysis arose while I was reading a paper by Darrell Conklin about *viewpoints* to represent melodies. In that paper, a short melody of about four bars was presented in score notation, followed by different viewpoints, such as e.g., pitch and duration represented as integer numbers, pitch interval distance as the number of semitones between two successive notes, pitch contour direction of one note with respect to the next (ascending, descending, same), etc. Suddenly, wavelets came to my mind. I thought that it was possible to represent melodies and capture their musical properties in one single process! By convolution, a function would then compute a coefficient at each position of the melody over the whole time dimension, the process would repeat with a dilated function at another time-scale, and again at another time-scale and so on. I encoded melodies analyzed by Stein (1979), and compared visually his analysis to the ones produced by the continuous wavelet transform, reporting those findings in a late-breaking demo at the International Society for Music Information Retrieval (ISMIR) 2010 conference. Possibly, those first ideas towards the analysis of music via wavelets, made my supervisors bet on me as a sorcerer's apprentice. In these last four years, I have devoted my life to exploring these ideas in depth, guided by my supervisors David Meredith and Tillman Weyde, to whom I am deeply thankful. Wavelets and machine learning were introduced to me by Roberto Carranza Estivariz (1939-2005) when I was studying systems engineering in Bolivia. Carranza inspired me to pursue research and advised me to apply for a scholarship abroad. And so, this journey that began with the power of a small wave took me across an ocean.

Gissel Velarde

Aalborg University, October, 2016

Acknowledgments

It is a tremendous privilege to have been a PhD student of David Meredith and Tillman Weyde, and I am deeply thankful to them for giving me the opportunity to work under their supervision. Additionally, I would like to thank David Meredith for engaging me as a research member of the EU project "Learning to Create". I would like to thank the researchers of the "Learning to Create" project for the valuable meetings we had; especially, many thanks to Carlos Cancino Chacón and Maarten Grachten for their collaboration on our joint publications. I thank the partial financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, FET grant number 610859.

I would like to thank the Department of Architecture, Design and Media Technology, Aalborg University, for the financial support, and my colleagues in the department for the nice time during my doctoral study. Thanks to Thomas B. Moeslund for his openness to my initiatives, thanks to Gitte Sørensen for her efficient work on the administrative matters, thanks to Rikke Gade for a discussion on image processing topics. Thanks to Mads Boye for his technical support on IT during the last months of my research, thanks to Ernest Nlandu Kamavuako for giving me some advice on wavelets for pattern matching, thanks to Morten Grud Rasmussen and Robert Dahl Jacobsen for reviewing with me the theory and code of an implementation of the Morlet wavelet.

I am very honoured to have Cumhur Erkut, José Manuel Iñesta Quereda and Emilia Gómez as members of the PhD Committee, and Martin Kraus as moderator.

Finally, I would like to thank my family, even if they are far away, for their love and for believing in my abilities. Especially, I would like to thank Christian for joining me on this adventure, my mother for her unconditional support, and Sol and Kira for making me see the world from various dimensions.

Contents

List of Tables

Part I

Introduction

1

Introduction

1 Introduction

Good representations are of interest to the information retrieval community, because the performance of machine learning methods strongly depends on the representations used; the assumption is that some representations work better than others at modelling discriminatory features in the variability of the data (Bengio et al., 2013). This dissertation explores the effect of convolution as a filtering mechanism for music representation, segmentation, pattern discovery and classification, in combination with machine learning techniques. Parameters and processing techniques were systematically studied in order to find the best practices. The aim was to understand how convolution relates to music analysis and perception.

Convolution is a mathematical tool that transforms a signal into a set of coefficients using an analysing function (filter). Computational methods can use convolution to build robust and redundant representations of signals. Robustness is desired to deal with variations of the objects to be recognised. In the visual system, for example, neurons are stimulus-selective and can detect visual objects irrespective of their precise position, size or contrast (Schnupp, 2006). A similar phenomenon occurs in the recognition of auditory objects, e.g., musical patterns which are recognised by the listener even if transposed or stated at different instants in time (Deutsch, 2013). Neural redundancy is largely related to feature selectivity, and it is gradually reduced at higher processing stations in the brain (Chechik et al., 2006).

The neurology and physiology of sensorial perception have been extensively modelled via convolution (filtering). For example, cortical receptive-field profiles of simple cells, neurons that detect oriented lines or edges within their receptive field (sensory space region) (Hubel & Wiesel, 1962), have been modelled with Gabor filters (Daugman, 1980; Jones & Palmer, 1987; Marčelja, 1980). Thanks to the technological advances in neurology, researchers have been able to characterise the relationship between natural images and brain activity based on Gabor receptive-field models (Kay et al., 2008). Auditory perception, from the cochlea

to higher centres in the auditory pathway, has been modelled with bandpass filters (Daubechies & Maes, 1996; Karmakar et al., 2011; Sinaga et al., 2003). In image processing, vision is modelled by convolution or filtering and machine learning algorithms for object and texture recognition (Bengio et al., 2013; LeCun et al., 2010; Nixon & Aguado, 2012; Tuia et al., 2014). In audio music classification, Gabor filters have been successfully used in genre classification (Wu et al., 2011). The success of these models and methods on modelling perception and the similitude of the convolution process to neurological and physiological mechanisms, was the motivation to investigate its usefulness for modelling music perception and analysis.

2 Background

Convolution as a mathematical tool possibly appeared for the first time in its general form in a study by Sylvestre François Lacroix in 1754, and then it was extensively used by Jean Baptiste Joseph Fourier, Siméon Denis Poisson, Augustin Louis Cauchy and others (Dominguez, 2015). However, it was first in 1934 that the term *convolution* (in English) was used by Aurel Friedrich Wintner in a paper about the convolution of Bernoulli Distributions (Dominguez, 2015).

In the second half of the twentieth century and soon after the first programmable computers were developed (Lavington, 2012), technical reports about data discrimination and recognition were written (Fix & Hodges, 1951; Rosenblatt, 1957). Today's widely used support vector machine classifier was developed years later by Cortes & Vapnik (1995), although the origins of this classifier might be found in publications by Vapnik, contemporary to those of Rosenblatt (1957) (Gammerman & Vovk, 2015). In 1906, Santiago Ramón y Cajal and Camillo Golgi won a Nobel prize for their work on the nervous system, describing that neurons and their interconnections are fundamental to the brain's functions (Nilsson, 2010). Fifty years later, the perceptron (Rosenblatt, 1957), an artificial neural network, preceded the discovery of Hubel & Wiesel (1962) of simple, complex and hypercomplex cells of receptive fields in cat's visual cortex. In 1979, convolution was proposed as a model underlying perception and memory by Murdock Jr. (1979) and a year later, two works (Daugman, 1980; Marčelja, 1980) applied 2-D Gabor filters (Gabor, 1946) to model the responses of cortical simple cells. At the same time, the wavelet theory was being developed by Morlet, Goupillaud and Grossmann, who studied seismic waves (Farge, 1992). The Gabor filter is an approximation of the Morlet wavelet (Antoine et al., 1993). However, since the work by Marčelja (1980), Daugman (1980) and Morlet (1981), Gabor filters have been associated with image processing and Morlet wavelets with the wavelet community.

Two studies close in time and aim, considered the effect of filtering representations for robust classification of audio (Daubechies & Maes, 1996) and images (Lecun et al., 1998). Daubechies & Maes (1996) proposed the use of the wavelet transform to model cochlear processing for speech recognition. Convolu-

tional neural networks (CNNs) were proposed to build representations robust to the handwritten digit's shape variability within class (Lecun et al., 1998). Today, CNNs are state-of-the-art in image processing applications.

Since the nineteenth century, Hugo Riemann and other musicologists developed their theories based on concepts of structural units, hierarchies, relations, similarity, repetition, variation, and style (Lerdahl & Jackendoff, 1983; Monelle, 1992; Nattiez, 1975; Ruwet, 1966; Schenker, 1935; Schoenberg, 1984). Psychological studies have concentrated on mechanisms of musical perception and memory. Lamont & Dibben (2001) found that similarity ratings of music were primarily based on surface features like contour and texture rather than on motivic or harmonic relations. Müllensiefen & Wiggins (2011b) studied melodic structures empirically and found that exposure refines memory representations. Deutsch (1999) adapted visual Gestalt principles to auditory experiments in order to understand how musical patterns were perceived, and found evidence that similar principles operate on visual and auditory objects. Similarly, neurological studies have suggested a direct interaction in mechanisms of vision and audition and common neural substrates in the brain, to establish robust representations of the world (Ernst & Bülthoff, 2004; Hidaka et al., 2011; Schön & Besson, 2005).

Although in the late twentieth century, there have been some studies for automatic or semi-automatic analysis of music, the beginning of a new century correlated with an increased number of publications in the field of music information retrieval (MIR). Indeed, the first conference of the International Society for Music Information Retrieval took place in 2000. Since then, several computational music analysis methods have been proposed. These systems have usually been designed and evaluated either to deal with audio or symbolic representations of music. Audio refers to the digital representation of a recording of a specific musical performance in terms of a sampled waveform, while symbolic refers to the encoding of music in terms of notes and their properties, and thus relates more closely to a notated score. In the symbolic context, popular representations are strings and multidimensional feature vectors (e.g., Conklin, 2006; Ponce de León & Iñesta, 2004; van Kranenburg et al., 2013). In audio, the input to most of the methods is based on some transformation of the audio data such as Fourier or mel-frequency cepstral coefficients (MFCCs) (Müller, 2015). MIR applications include among others: segmentation (e.g., Cambouropoulos, 2001; Paulus et al., 2010), classification (see Corrêa & Rodrigues, 2016; Fu et al., 2011) and pattern discovery (see Collins, 2016; Janssen et al., 2013).

3 The evolution of the project

Initially, this research project aimed to study wavelets and the wavelet transform for music analysis. Wavelets can be seen as filters with specific mathematical properties, and the Continuous Wavelet Trans-

form (CWT) as the convolution of a signal and a wavelet family. The discrete version of the wavelet transform (DWT) was not within the scope of the project, as it is more suitable for compression and reconstruction purposes, applications not considered in this thesis.

There have been several studies using the CWT and the DWT on audio applications such as feature extraction for genre classification (e.g., Wu et al., 2011), pitch contour extraction and melodic indexing (e.g., Jeon & Ma, 2011), rhythmic content analysis (e.g., Smith & Honing, 2008), denoising (Berger et al., 1994), and audio compression (e.g., Srinivasan & Jamieson, 1998). In contrast, the use of wavelets on symbolic representations of music has been scarce: Pinto (2009) used the DWT for music indexing. Just recently, Shafer (2016) presented substantial work on the use of wavelets for symbolic music classification and pattern extraction.

Initial experiments on the use of wavelets for music analysis on symbolic representations were presented before this project started (Velarde, 2010; Velarde & Weyde, 2011, 2012a,b,c,d). The following publications reporting the use of wavelets for music analysis were applied in the one-dimensional (1-D) space, studying music representation, segmentation, classification and pattern discovery (see Papers I to IV). In these four publications, we studied several parameters, techniques and algorithms, but we only used the Haar wavelet. In the two-dimensional (2-D) space, we tested the effect of using two different filters for classification and found no significant difference in the classification results when filtering with a Gaussian filter or a Morlet wavelet (see Paper V). In recent years, deep learning methods such as CNNs have been shown to outperform previous approaches in image classification tasks. In these methods, the filters are automatically learned by the systems, in contrast with wavelet filters, which must obey some mathematical conditions. We noticed that the approach of the project was very related to CNNs in the sense that convolution is the underlying process in a hierarchical fashion. We therefore performed experiments using a CNN, obtaining filters which we directly related to musical features (see Paper VI). Therefore, there was no further motivation to restrict our study to wavelet filters, but instead, we recognised the general relevance of convolution for music analysis.

Next, considerations relating to the development of a framework for computational music analysis are presented.

4 Computational music analysis

Music analysis is a discipline that applies music theory with the aim of understanding how music is designed, structured and elaborated (Horton, 2014). In this discipline, there is no definitive analysis of a piece, but several different possible valid analyses (Marsden, 2016). On the other hand, computational music analysis methods, despite taking different approaches, are usually expected to produce an output

that matches a "ground truth" (Marsden, 2016). In some tasks, the ground truth is unquestionable (e.g., composer classification, when the composers of the analysed work are known with certainty), while in others, the "ground truth" represents only the subjective judgement or perception of specific individuals (e.g., discovery of repeated themes and sections). Despite these considerations, analyses produced by computers are of interest because they are applicable to e.g., recommendation systems, music education, music creation and music indexing. They can also serve to support musicology.

5 Research questions

This dissertation addresses the following research questions:

- To what degree can convolution support the analysis of music in relation to segmentation, pattern discovery and classification?
- What are the best techniques and parameter settings for carrying out such tasks?
- How can we understand a convolution-based representation of music from a music-theoretical and perceptual perspective?

6 General and Specific Objectives

The general objectives of this dissertation are: 1. to develop and evaluate an automated framework for music analysis based on convolution, with applications to segmentation, classification and pattern discovery; and 2. to study filters in relation to music-analytical and perceptual properties.

The specific objectives of the work reported in this dissertation are:

- design, implement and evaluate a framework applying convolution as the basis for the structural analysis of music applied to segmentation, pattern discovery and classification, and compare it with other algorithms;
- evaluate 1-D and 2-D filters for structural analysis in relation to music analysis and perception;
- evaluate and test systematically the use of filters, features of coefficients, time scales and similarity measures in classification tasks (e.g., composer, genre, tune-family classification) and pattern discovery (e.g., motivic analysis) in order to find the best techniques and parameter settings; and
- relate filters to musical properties.

Figure 1.1: Diagram of the method for music segmentation

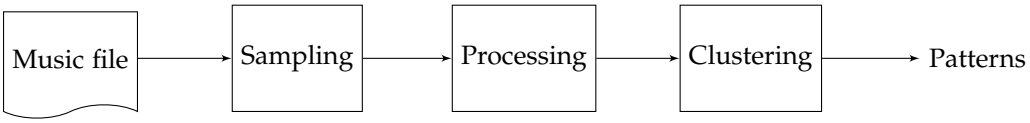


Figure 1.2: Diagram of the method for pattern discovery

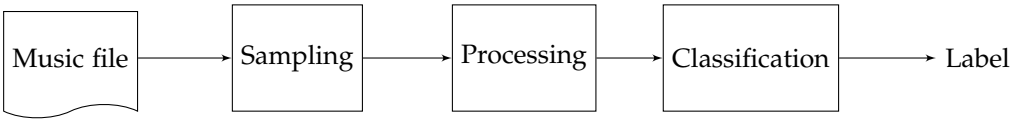


Figure 1.3: Diagram of the method for music classification

7 Methodology

The methodology is to study convolution as a mechanism for extracting relevant features of music for representation and segmentation, and then to apply machine learning algorithms to these features for classification and clustering. A summary of the techniques, algorithms and methods studied is presented in section 9.

The framework for computational music analysis has been applied to three applications: music segmentation, pattern discovery and classification (see Fig. 1.1 to 1.3, respectively). The main difference between these applications is the output. For music segmentation, the output corresponds to segments (or local boundaries) in a piece of music (Fig. 1.1), while for music classification the output is its label (Fig 1.3). Finally, for pattern discovery, the output of the system corresponds to the occurrences of patterns found in the piece (Fig. 1.2). In the *sampling* phase, music files are converted to 1-D or 2-D signals, and the following processing steps are therefore carried out in the 1-D or 2-D space (see sections 10 and 11). This is followed by a *processing* phase in which the signals are transformed in various ways to find a suitable representation for each application. For music classification and pattern discovery we also tested the effect of segmenting the input signals before processing.

8 Applications and evaluation

The framework presented in this dissertation has been applied to music segmentation, pattern discovery and classification, and it has been evaluated against state-of-the-art approaches.

8.1 Music segmentation

Figure 1.1 shows a flowchart of the method for music segmentation. The input to the system is a music file, and the system outputs the found segments (or the local boundaries) of the piece. We evaluated the effect of segmentation for classification, and tested different segmentation approaches (constant-duration, zero-crossings and local maxima) and compared them to the Local Boundary Detection Model (LBDM) (Cambouropoulos, 1997, 2001) in classification experiments. The wavelet-based segmentation and the LBDM measure local changes in melodic contour, such that the LBDM measures the degree of change of successive values and the wavelet segmentation detects locally maximal changes of average pitch in melodic contour at a defined time-scale (see Paper I). The classification accuracies obtained when segmenting at local maxima or zero-crossings are comparable to those obtained when segmenting with the LBDM (see Paper I, Tables 1 and 2 for comparison between local maxima segmentation and the LBDM; see Paper II, Figures 10 and 11 for comparison between zero-crossings and the LBDM).

8.2 Pattern discovery in music

Figure 1.2 shows the general form of the method used for pattern discovery. The input to the system is a music file and the system outputs clusters of pattern occurrences ranked according to their salience. This method has been submitted for evaluation to the Music Information Retrieval Evaluation eXchange (MIREX), on the task of discovering repeated themes and sections in symbolic, monophonic music. In the last three MIREX events, our proposed method for pattern discovery has proved to be a competitive approach over the different measures. In the MIREX 2016, our VM1 submission significantly outperformed all other submissions on establishment recall per pattern and occurrence recall per pattern, and it was suggested as a candidate for visual analysis as it outputs fewer patterns than other algorithms (see the evaluation at `http://www.music-ir.org/mirex/wiki/2016:Discovery_of_Repeated_Themes_%26_Sections_Results`). See Paper IV, sec. 12.3.1.2, for an evaluation on our submissions in 2014. Finally, see Paper IV, sec. 12.3.1.3, as an example of the produced output of the method and its visualisation.

8.3 Music classification

Figure 1.3 presents a diagram of the method for music classification. The input to the system is a music file, and the system outputs its computed label. In 1-D, we evaluated our method on the task of classifying folk tunes into tune families and compared its classification accuracy to a state-of-the-art approach based on string alignment by van Kranenburg et al. (2013). The accuracies of our proposed method are close to those reported by van Kranenburg et al. (2013) using a local approach with interval sequence features (see Paper II, Table 3 and van Kranenburg et al. (2013), Table 4). However, the accuracies reported by van Kranenburg

et al. (2013) using expert-annotated phrase boundaries are far more accurate than those obtained with our method, which does not use this information.

Moreover, we performed experiments on parent work recognition. We tested empirically musicological claims of motivic coherence in Bach's Two-Part Inventions (Dreyfus, 1996), see Experiment 1 in Paper II, sec. 4.1, and Experiment 2 in Paper IV, sec. 12.3.2.

In 2-D, our convolution-based method was evaluated on the difficult task of discriminating between Haydn and Mozart string quartet movements. Our method proved robust to encoding, transposition and amount of information (see Paper V, sec. 4). Moreover, we compared the classification accuracies of our method to those obtained by van Kranenburg & Backer (2004), reaching the state-of-the-art, while avoiding the use of hand-crafted features and voice parsing (see Paper V, sec. 4, Table 3.). Our convolution-based ensemble of classifiers was evaluated on composer and genre recognition on audio and symbolic representations of music. The ensemble of classifiers reached the state-of-the-art on composer recognition on two datasets of Haydn and Mozart string quartet movements (see Paper VI, Tables 3 and 4), and proved to be versatile, achieving similar performance on genre classification, using *The Well-Tempered Clavier* by J.S. Bach (see Paper VI, Tables 2 and 6). Moreover, our classifiers obtained similar performance on audio and symbolic representations (see Paper VI, Experiments 2 and 3). Finally, we are able to provide insight into style through the observation of filters automatically learnt by a CNN (see Paper VI, sec. 3.3).

9 Summary of the techniques, algorithms and methods studied

Tables 1.1 and 1.2 present a summary of the techniques, algorithms and methods studied in the 1-D and the 2-D space related to publication.

Table 1.1: Summary of the techniques, algorithms and methods studied in the 1-D space related to each publication

	Paper I: FMA 2013 (Velarde, Weyde, & Meredith, 2013b)	Paper II: JNMR 2013 (Velarde, Weyde, & Meredith, 2013a)	Paper III: MIREX 2014 (Velarde & Meredith, 2014)	Paper IV: CMA 2016 (Velarde, Meredith, & Weyde, 2016)
Representations: pitch signal (MNN), normalized pitch signal and wavelet coefficients	✓	✓	✓	✓
Representation: absolute wavelet coefficients				✓
Segmentation: constant-duration (or constant-length) and wavelet zero-crossings		✓	✓	✓
Segmentation: local maxima	✓	✓		
Segmentation: absolute maxima			✓	✓
Filter: Haar wavelet at a single scale	✓	✓	✓	✓
Segment length normalization by zero padding	✓	✓	✓	✓
Segment length normalization by interpolation		✓		
Geometric transformations: inversion, retrograde and retrograde inversion		✓		
Classifier: *k*-NN	✓	✓		✓
Distance measures: Euclidean and cityblock	✓	✓	✓	✓
Distance measure: Dynamic Time Warping				✓
Diagonal concatenation of segments			✓	✓
Horizontal concatenation of segments				✓
Clustering: single linkage (nearest-neighbour)			✓	✓
Ranking: compression ratio			✓	✓
Parameter tuning	✓	✓	✓	✓
Application: Folk tune classification. Dataset: The Dutch Folk Tunes (Grijp, 2008)	✓	✓		
Application: Parent work recognition. Dataset: *The Two-Part inventions* by J.S Bach (Bach, 1790)		✓		✓
Application: Discovery of repeated themes and sections. Dataset: The JKU PDD (Johannes Kepler University Patterns Development Database, 2014)			✓	✓
Comparison of segmentation based on wavelets (Velarde et al., 2013a,b) and the LBDM (Cambouropoulos, 2001)	✓	✓		
Comparison of the wavelet-based segmentation and classification method (Velarde et al., 2013a) and alignment methods (van Kranenburg et al., 2013)	✓	✓		
Comparison of our method for pattern discovery VM1 (Velarde & Meredith, 2014) and NF1(Nieto & Farbood, 2013), OL1(Lartillot, 2014), DM10 (Meredith, 2013)			✓	✓

Table 1.2: Summary of the techniques, algorithms and methods studied in the 2-D space related to each publication

	Paper V: ISMIR 2016 (Velarde, Weyde, Cancino Chacón, Meredith, & Grachten, 2016)	Paper VI: (Manuscript) (Velarde, Cancino Chacón, Meredith, Weyde, & Grachten, in review)
Representations: piano-roll p_{70qn}, p_{400n}, MNN and morphetic (diatonic) pitch and their filtered versions	✓	✓
Representation: p_{fl}, MNN and morphetic (diatonic) pitch and their filtered versions		✓
Representations: STFT and VQT spectrogram sp_{400n}, sp_{400n} and their filtered versions		✓
Pitch range centering C_b	✓	✓
Center of mass centering C_m	✓	
LDA	✓	✓
Segmentation: constant-length		✓
Filter: Morlet wavelet at a single scale and orientation	✓	
Filter: Gaussian filter	✓	✓
Classifier: SVM with linear kernel	✓	✓
Classifier: k-NN with Euclidean distance		✓
Classifier: CNN		✓
Parameter tuning	✓	✓
Application: Composer classification. The Haydn and Mozart string quartets. Dataset introduced by van Kranenburg & Backer (2004)	✓	✓
Application: Composer classification. The Haydn and Mozart string quartets. Dataset introduced by Hillewaere et al. (2010)		✓
Application: Genre classification. *The Well-Tempered Clavier* by J. S. Bach. Datasets (Bach, 1722, 1742)		✓
Comparison of methods (Velarde et al., in review; Velarde, Weyde, et al., 2016) and the string-based method by Hillewaere et al. (2010) and the style-makers and kNN based-method by van Kranenburg & Backer (2004)	✓	✓

10 Music analysis in the one-dimensional space

The computational analysis of music in the 1-D space is based on representing music as 1-D signals aiming to model melodic contour, similarity and variation. Previous approaches have used the following representations: strings (e.g., Hillewaere et al., 2010; van Kranenburg et al., 2013), contour vectors (Juhász, 2009), contours (Huron, 1996), polynomial functions (Müllensiefen & Wiggins, 2011a; Urbano et al., 2010), and Fourier coefficients (Schmuckler, 1999). We introduced (normalized) pitch signals and wavelet coefficients (Papers I and II).

As mentioned previously in this introduction, representation is fundamental for computational methods as machine learning algorithms strongly depend on them. In this work we aimed to study the effect of convolution (i.e., filtering with the Haar wavelet) and other processing techniques or transformations (e.g., transposition, geometric transformations) in 1-D signals for music applications. These representations are briefly introduced in the next section.

10.1 One-dimensional representations of music

In Papers I to IV, we introduced four 1-D representations (see Paper IV, Fig. 12.2 for illustration):

- *pitch signal representation*, described in Paper II, sec. 3.1.1. Illustrations between the correspondence of score notation and its pitch signal can be seen in Paper II, Fig. 1 (a), (b), and Paper IV, Fig. 12.10.
- *normalized pitch signal representation*, described in Paper II, sec. 3.1.1.
- *wavelet representation* (or *wavelet coefficient representation*), described in Paper II, sec. 3.1.2.
- *absolute wavelet coefficient representation*, described in Paper IV, sec. 12.2.1, para. 7.

Melodies are sampled to pitch signals first; the other three representations (normalized pitch signal, wavelet coefficient and absolute wavelet coefficient) are transformations of pitch signals. In the case of polyphonic works, a pitch signal can be sampled for each part, if each part was encoded separately (for example, see Paper II, sec. 4.1). Polyphonic works without information on part encoding could still be encoded as pitch signals by generating a melodic line by applying a skyline approach (Uitdenbogerd & Zobel, 1998) (as in Collins, 2014) (see Paper II, sec. 12.3.1). However, we have not studied the effect of the skyline algorithm on our 1-D approach. It is possible that for certain styles of music the performance of our 1-D algorithms would be affected by using this technique.

In the 1-D space, we apply the CWT (Antoine, 1999) with the Haar wavelet and select a single scale, thus effectively applying a filter on the signal (see Paper II, sec 3.1.2). The Haar wavelet (Haar, 1910) is illustrated in Paper II, Fig. 2. We discuss the problem of selecting a filter in Paper II, section 3.2. Filtering

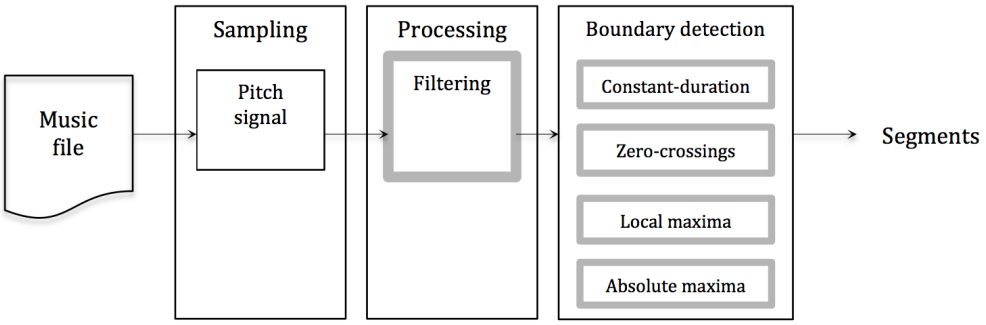


Figure 1.4: Diagram of the proposed method for music segmentation in the 1-D space. The system receives a piece of music and outputs its computed segments (or local boundaries). Thick grey boxes are optional processing steps. The modules in the boundary detection phase are exclusive, such that only one segmentation approach takes place. Filtering does not precede constant-duration segmentation to find local boundaries, as the local boundaries are set with constant-duration.

with the Haar wavelet at a single scale emphasises the signal's information on that specific time-scale (discussion on the selection of scale is given in Paper II, section 3.4).

Additionally, we tested the following geometric transformations: inversion, retrograde and retrograde inversion as seen in Paper II, Fig. 8.

10.2 Segmentation

Segmentation is a very important task for music analysis and perception (Lerdahl & Jackendoff, 1983) (see Paper II, sec. 2.2.1, for a deeper argumentation). In the 1-D space, we also evaluated the effect of filtering for segmentation, as well as the use of a base-line segmentation based on chunking signals at a constant-length (or constant-duration). The evaluated wavelet-based segmentation approaches are:

- *wavelet zero-crossings*, illustrated in Paper II, Fig. 4, where the scale for representation and segmentation is the same, therefore it is possible to observe visually the coincidence of zero-crossing and segmentation points in the wavelet coefficient representation;
- *wavelet local maxima*, illustrated in Paper II, Fig. 5: the scale for representation and segmentation is the same;
- *wavelet absolute maxima*, illustrated in Paper IV, Fig. 12.4, bottom graph: the scale for representation is different from the scale for segmentation and, therefore, the segmentation points do not necessarily coincide with the local maxima points in the represented signal.

Music segmentation was evaluated in classification experiments (see section 8.1). Figure 1.4 shows a diagram of the proposed segmentation method. A music file is first sampled to a 1-D pitch signal. The grey boxes are optional. The modules in the segmentation phase are exclusive, such that only one

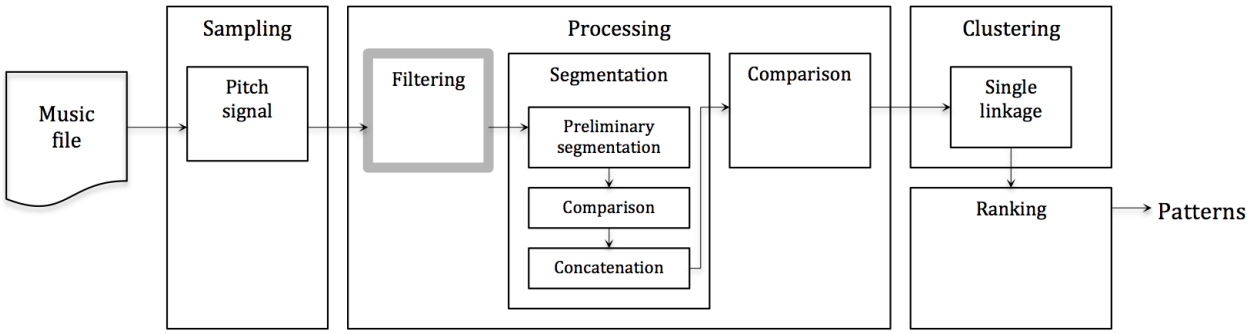


Figure 1.5: Diagram of the proposed method for pattern discovery in the 1-D space. The system receives a piece of music and outputs its computed clusters of pattern occurrences. Thick grey boxes are optional processing steps.

segmentation approach takes place. Filtering takes place for zero-crossings, local maxima and absolute maxima segmentation.

10.3 Pattern discovery

Our proposed method for pattern discovery (Papers III and IV), resembles the method of paradigmatic analysis developed by Ruwet (1966) and Nattiez (1975) (for an introduction see Paper IV, sec. 12.1). Figure 1.5 shows a diagram of the proposed method for pattern discovery from the input to the output (presented in more detail than in Paper IV, Fig. 12.1). A music file is first sampled to a 1-D pitch signal, and then undergoes several processing steps. Filtering is an optional step, as signals can be sent to the segmentation phase as pitch signals or as filtered signals. After segmentation, there are comparison, clustering and ranking phases. The output of the method corresponds to the computed patterns organised into ranked clusters. A detailed description of the method is given in Paper IV, sec. 12.2.

Experiments reporting the use of the method for pattern discovery in music are presented in Paper IV, sec. 12.3. Moreover, the method was evaluated against other approaches in three editions of the MIREX task on discovery of repeated themes and sections (monophonic symbolic version) (see section 8.2).

10.4 Music classification

Figure 1.6 presents a diagram of the proposed method for music classification in 1-D. A piece of music is first sampled to a 1-D pitch signal, and then undergoes several processing steps before classification with a *k*-Nearest Neighbour (*k*-NN) algorithm. Thick grey boxes are optional processing steps. We tested the effect of filtering on representation and segmentation. After segmentation, only signals represented as *pitch signals* are normalized in pitch, being transformed to *normalized pitch signals*. Filtered signals (i.e., *wavelet coefficient* and *absolute wavelet coefficient* representations) do not follow the processing module

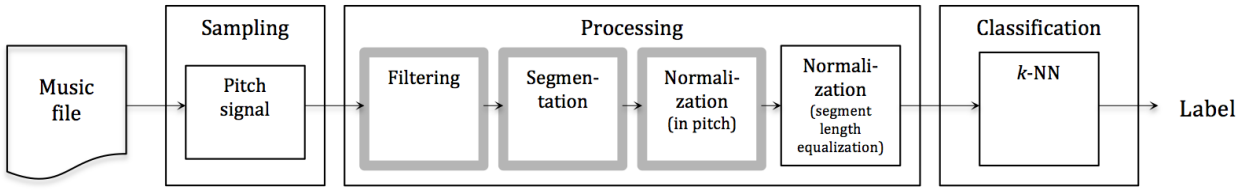


Figure 1.6: Diagram of the proposed method for music classification in the 1-D space. The system receives a piece of music and outputs its computed class label. Thick grey boxes are optional processing steps. If filtering takes place, the normalization in pitch does not take place.

normalization in pitch as wavelet filtering acts as a normalization process; wavelet representations in 1-D are transposition invariant. In order to measure similarity between segments, we used two normalization techniques (zero-padding or interpolation) to equalise the length of segments (see the effect of these two normalization techniques in Paper II, Fig. 6). Finally, we use a k-NN classifier. The final computed class label corresponds to the most frequently predicted class of the segments. We evaluated the method on folk tune classification and on parent work recognition (see section 8.3).

11 Music analysis in the two-dimensional space

The computational analysis of music in the 2-D space is based on representing music as 2-D signals aiming to model musical texture. The motivation for representing music in the 2-D space as pitch–time images is presented in Paper V, section 2.1. We aimed to sample both audio and symbolic music representations to 2-D pitch-time images in order to propose a general method for music analysis that does not depend on the type of input representation. The method processes and analyses music in a similar form regardless of it being an audio or a symbolic file. For symbolic representations we use piano-rolls and for audio we use spectrograms. Spectrogram images present spectral information over time and are used in audio music classification (e.g., Costa et al., 2012; Velarde et al., in review; Wu et al., 2011). Piano-roll images present notes over time and were successfully introduced for music classification by Velarde, Weyde, et al. (2016) (Paper V).

Similarly as in the 1-D space, we aimed to study the effect of convolution (i.e., filtering with Morlet, Gaussian and learnt filters) and other transformations (e.g., transposition, dimensionality reduction) in music applications. In the 2-D space, we applied the method to music classification.

11.1 Two-dimensional representations of music

Today, the most commonly used forms of visual representation for music are notated scores and piano-rolls (as commonly used in, for example, sequencer software). Audio data is often represented visually,

e.g., in the form of spectrograms.

We sample piano-roll images from symbolic representations of music (e.g., MIDI files), and sample spectrograms from audio files (see Papers V and VI). Examples of pitch–time representations of music can be seen in Paper VI, Fig. 3. The following pitch–time representations are used in the 2-D space:

- *Piano-rolls* of full-length pieces are described in Paper VI section 2.1.1. Piano-roll excerpts of the first 70 quarter notes and piano-roll excerpts of the first 400 notes are described in Paper V sections 3.1.3 and 3.1.4.
- *Spectrograms* are computed using the Short Time Fourier Transform (STFT) or Variable Q-Transfrom (VQT), see Paper VI, section 2.1.2.

We used two filters for our experiments: a Morlet wavelet and a Gaussian filter, for details on both filters see Paper V section 3.2.4. Additionally, we automatically learn filters by a CNN, see Paper VI, section 2.3.1.

Apart from studying the effect of filtering, we also study the effect of applying different processing techniques or transformations. Certain musical variations do not affect human recognition of music, e.g., recognising a melody and its transposed version, but could dramatically affect the performance of a system. In Paper V section 3.2, we introduced transformations aiming to find a better representation space (e.g., Linear Discriminant Analysis, center of mass centering, filtering) and to test the robustness of the method to transformations that could be considered as perceptually irrelevant for recognition (e.g., pitch range centering):

- *pitch range centering*, modelling transposition (see Paper V sec. 3.2.1)
- *center of mass centring*, (see Paper V sec. 3.2.2)
- *Linear Discriminant Analysis (LDA)*, (see Paper V sec. 3.2.3)
- *filtering*, (see Paper V sec. 3.2.4)

11.2 Music classification

In Paper V, we move from 1-D to 2-D representations, and introduce a method for music classification which does not need domain knowledge features such as contrapuntal features or any other handcrafted features which are dataset dependent. Moreover, the method does not depend on the encoding or parsing of separate voices, which makes it more general than both our own approach in 1-D (Velarde et al., 2013b), and previous methods that have been applied to the same task (Herlands et al., 2014; Hillewaere et al., 2010; Hontanilla et al., 2013; van Kranenburg & Backer, 2004).

The approach in Paper V exploits musical texture of large structures of music (excerpts of about 70 quarter notes in length or containing 400 notes) for its predictions. However, we hypothesised that the recognition of style might also need local processing of small patterns occurring translated in time and/or transposed in pitch. It was previously shown that local processing is very important for music classification (van Kranenburg et al., 2013; Velarde et al., 2013a). Therefore, in Paper VI we introduced a segmentation phase, extracting local features at small time scales. We experimented with ensembling classifiers based on feature extraction at small time scales of about 1 or 2 quarter notes, combined with feature extraction at the large scale. These approaches were evaluated on two style recognition tasks (composer and genre classification) using both symbolic and audio representations of music (see section 8.3, para. 3).

12 Discussion

The aims of this dissertation were, first, to evaluate the effectiveness of convolution for music analysis in applications such as segmentation, classification, and pattern discovery; and second, to study filters in relation to music-theoretical and perceptual properties. In this section, general findings and considerations within the scope of this framework are presented, as well as considerations outside its original scope.

Convolution is indeed a crucial component of the computational music analysis methods presented in this dissertation, and has been shown to significantly improve recognition over non-filtered representations. However, it is not the only process that helped us to classify music or to find musical patterns. Its relevance should be attributed to providing a robust and discriminative representation of music.

In 1-D (Papers I to IV), we have shown that appropriately tuned filters deliver a representation robust to melodic variation, which might emphasize relevant parts of the melodic contour, having a beneficial effect on the similarity measure; possibly due to the transposition invariance of the wavelet representation and the use of an appropriate time-scale (Papers I and II). Additionally, we found another representation, the normalized pitch signal, which proved to be powerful for musical material which is restated with less degree of variation (Paper IV).

In 2-D (Papers V and VI), we have shown that appropriately tuned filters deliver a representation robust to textural variation, which might highlight contour, having a beneficial effect on the similarity measure, and significantly improving recognition over non-filtered representations.

Moreover, we found that local processing is crucial in music categorization and pattern discovery, but processing at large scale also proves relevant (Papers II and VI), so that a combination of feature extraction at small and large time scales makes the system robust across datasets (see Paper VI).

Filtering as a segmentation mechanism has been studied in the 1-D space only. The method for segmentation was evaluated in classification experiments, rather than by using a ground truth containing

information on perceived local boundaries. Segmentation was also used as a subprocess in pattern discovery. In classification and pattern discovery experiments, absolute maxima segmentation worked better than zero-crossing segmentation, helping to improve recognition and being twice as fast as zero-crossing segmentation (Paper IV). Moreover, constant-length segmentation at a small scale of 1 qn, proved to work well for recognition when the musical material is restated with less degree of variation (see Paper IV). In 2-D, constant-length segmentation was also successfully used in our classification experiments (Paper VI). It seems possible that this straightforward approach combined with other mechanisms is useful for musical recognition, given that small segments might be "atomic" structural musical units, and do not need any segment length normalization for the similarity measure. Conklin (2006) also found high classification accuracies using pitch class segments of beat-length.

In music classification experiments, we have modelled musical texture in the 2-D space based on pitch–time representations. The success of this visual representation of music, exploits the relation between the visual and auditory processing for music perception (Deutsch, 2013).

Normalization has been shown to be important for music similarity. Transposing melodic segments (pitch normalization) was relevant in parent work recognition (e.g., Paper II). However, this strategy seemed to be less effective when using full-length pieces or large musical excerpts (compare Paper II, sec. 5.2.1, and Paper V, sec. 4). This result might indicate that in some cases the tonal content is relevant for music similarity. In 1-D, we used segmentation approaches delivering segments of different lengths. Measuring segment similarity requires segment length equalisation (normalization). We found that normalization by zero padding produced better results than normalization by interpolation, suggesting that the structure of segments is related to their length (Paper II).

In 1-D, we evaluated the use of Euclidean, city-block and dynamic time warping (DTW) distances. We found similar results when measuring similarity with city-block and Euclidean distances. In general, DTW was not associated with the best results when finding musical patterns, and was more computationally expensive than the other two distances. We assumed that the DTW distance might prove to be useful for music presenting temporal deviations such as *ritardando* or *accelerando* (Paper IV). As Euclidean and city-block distances performed similarly in our experiments in 1-D, and as the Euclidean distance is generally used as a distance measure in image processing, we measured similarity with the Euclidean distance in the 2-D space.

When classifying with a CNN, we applied a hierarchical structure with nine filters of about 1 qn in the first layer, and five filters of about half qn in the second layer (Paper VI). In our experiments, the CNN did not outperform the state-of-the-art (Paper VI). We attributed this result to the training of the CNN with a small number of samples or a possibly reached "glass ceiling" in the evaluated task; however, the CNN's automatically learnt filters allowed us to directly relate those filters to musical patterns, gaining musical

insight (Paper VI, sec. 3.3).

12.1 On using the results of computational music analysis to support musicology

In general, musicologists might be interested in computational music analysis methods to, for example, test a theory (or theories) systematically, or to test different methods for the same problem, and to find an approach that is more appropriate for a given task.

In particular, our computational approach resembling *paradigmatic analysis* (Nattiez, 1975; Ruwet, 1966) has shown to be competitive when evaluated on a 'ground truth' analysis not produced by experts in paradigmatic analysis. This suggests that paradigmatic analysis is a relevant method for music analysis. Additionally, this suggests convergence of human music analysis given a well-defined output. Musicologists may be interested in a visualisation of the patterns obtained by our method which resembles paradigmatic analysis, and they might help us by giving feedback to the output of the approach.

Spurious composer attribution can be a practical application of our classification approach. Musicologists might be interested to compare their hypotheses to the predictions of the system. Additionally, the filters learnt by the convolutional neural network might relate to musical structures of the analysed datasets. However, this relation is not well understood and deserves future work.

12.2 1-D or 2-D?

We developed computational music analysis methods in the 1-D and 2-D space, but we did not compare the two approaches with each other. Pattern discovery was evaluated in 1-D only, and although we tested the method for music classification in 1-D and 2-D, we did not run experiments on the same datasets, and therefore, we do not have sufficient evidence to draw conclusions regarding the advantages of one approach over the other, or guidelines to select one of the two for a given task.

Although it would seem intuitive, we do not have any firm evidence that analysis in 1-D would be more appropriate for monophonic music, nor that analysis in 2-D would be generally more suited for polyphonic works. To make such conclusions, we would need to study both approaches over the same context, to evaluate, for example competitiveness and complexity. However, concerning complexity, it is evident that the approach in the 1-D space is more economical than its counterpart in the 2-D space. On the other hand, the performance of the 1-D approach might be affected by the parsing of voices for some styles of music.

12.3 On the generalizability of the methods for music and beyond

It seems possible that the methods in 1-D and 2-D can generalise well for music where timbral features are not the most important descriptors of their style. We evaluated the methods on folk music and classical music of the periods: renaissance, baroque, classical and romantic, but it might well be that music from other periods of time and genres could be effectively analysed with the proposed methods.

We did not evaluate the proposed methods outside music applications. However, the methods in 1-D for segmentation, pattern discovery and classification can be used for applications in other domains, such as finance, weather, medicine, etc., where data is represented as a 1-D signal or time-series.

In the MIR community, the evaluation of computational methods is based on specific applications, such as music classification, segmentation, chord estimation, query by singing/humming, etc., (see the Music Information Retrieval Evaluation eXchange at `http://www.music-ir.org/mirex/wiki/MIREX_HOME`). We participated in the MIREX task on the discovery of repeated themes and sections (Collins, 2014). Our method has been benchmarked against other methods, which for example represent data not as time-series but as point sets (Meredith, 2013).

Keogh & Kasetty (2003) presented a survey and evaluation of various methods for time-series applications in finance, medicine, astronomy and networking among others. However, there was no evaluation of methods for music applications.

In 2-D, our method for classification can be potentially used in non-music applications where the input are images. Indeed, we evaluated a convolutional neural network which is state-of-the-art in digit recognition.

Therefore, the main contributions of this dissertation are the design and evaluation of convolution-based methods in music applications, and the musical insight gained by their use.

13 Conclusion

The main contribution of this dissertation is the introduction of novel convolution-based methods for music analysis. We designed, implemented and evaluated an automated framework for the analysis of music in applications such as music segmentation, pattern discovery, and classification. We systematically studied and evaluated the effect of filtering and other processing techniques on representation and segmentation. Moreover, we studied and optimized the parameters of filters (Haar, Morlet, Gaussian and learnt filters), and machine learning algorithms (k-nearest neighbours, single linkage, support vector machines, convolutional neural networks) in pattern discovery and classification applications. We developed a framework for music analysis in the one-dimensional and two-dimensional spaces; however, we did not rigorously compare one-dimensional approaches with two-dimensional approaches. The proposed

methods do not use domain knowledge features and are therefore more generalizable than other methods which depend on dataset-derived or style-dependent features. We found that filtering improves recognition over non-filtered representations, and that, in particular, filtered representations are more robust to musical variation. Moreover, local processing and processing at a large scale prove to be important in music classification, and a combination of large-scale and small-scale feature extraction strategies can be complementary for ensembling. In 1-D, our convolution-based segmentation method is comparable to a state-of-the-art Gestalt-based segmentation approach in classification experiments. In the last three Music Information Retrieval Evaluation eXchange campaigns, our proposed method, which resembles paradigmatic analysis, has been evaluated on the discovery of repeated themes and sections task applied to monophonic symbolic music, and has been shown to be a competitive approach over all measures in that evaluation. In 2-D, our convolution-based ensemble of classifiers reaches the state-of-the-art on composer recognition and achieves similar performance on genre classification. Moreover, our classifiers perform equally well on symbolic and audio music data. Finally, observation of filters automatically learnt by a convolutional neural network provides musical insight.

13.1 Future work

Future work might consider the following aspects:

- Evaluation of the performance and complexity of the proposed methods in 1-D and 2-D for classification, using the same datasets.
- Extension of the 1-D pattern discovery method to audio and the polyphonic version.
- In 1-D, evaluating the approach using other filters such as Gaussian, Ricker, Daubechies.
- In 1-D and 2-D, evaluation of the use of a convolutional neural network initiating the filters with given Morlet wavelets or Gaussian filters.
- Evaluation on multi-class classification tasks (e.g., genre, composer, performer, etc.) on larger datasets.
- Evaluation of classification results in the analysis of spurious attributions.
- Obtaining feedback from musicologists and music students on the output of the methods.
- Evaluating the proposed method on non-music-related tasks and applications.

References

Antoine, J.-P. (1999). Wavelet analysis: a new tool in physics. In J. C. van den Berg (Ed.), *Wavelets in physics.* Cambridge: Cambridge University Press.

Antoine, J.-P., Carrette, P., Murenzi, R., & Piette, B. (1993). Image analysis with two-dimensional continuous wavelet transform. *Signal Processing*, *31*(3), 241–272.

Bach, J. S. (1722). *The Well-Tempered Clavier, Book I, BWV 846-869.* `http://music.kimiko-piano.com/album/bach-well-tempered-clavier-book-1`. Navona Records. (Peformed by Kimiko Ishizaka. Accessed 7-Aug-2015.)

Bach, J. S. (1742). *The Well-Tempered Clavier, BWV 846-893.* `http://www.musedata.org/encodings/bach/bg/keybd/`. (Provided in the MuseData collection. Accessed 23-Feb-2015.)

Bach, J. S. (1790). *Inventions BWV 772-786.* `http://www.musedata.org/encodings/bach/rasmuss/inventio/`. (MIDI encodings ed. by Steve Rasmussen; Accessed Apr-2011.)

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798-1828. doi: 10.1109/TPAMI.2013.50

Berger, J., Goldberg, M., & Coifman, R. (1994). A method of denoising and reconstructing audio signals. In *Proceedings of the International Computer Music Conference* (pp. 344–344). San Francisco, CA: ICMA.

Cambouropoulos, E. (1997). Musical rhythm: A formal model for determining local boundaries, accents and metre in a melodic surface. In *Music, gestalt, and computing* (pp. 277–293). Springer.

Cambouropoulos, E. (2001). The local boundary detection model (LBDM) and its application in the study of expressive timing. In *Proceedings of the International Computer Music Conference* (pp. 17–22). San Francisco, CA: ICMA.

Chechik, G., Anderson, M. J., Bar-Yosef, O., Young, E. D., Tishby, N., & Nelken, I. (2006). Reduction of information redundancy in the ascending auditory pathway. *Neuron*, *51*(3), 359 - 368. Retrieved from

http://www.sciencedirect.com/science/article/pii/S0896627306005125 doi: http://dx.doi.org/10.1016/j.neuron.2006.06.030

Collins, T. (2014). *Johannes Kepler University Patterns Development Database.* https://dl.dropbox.com/u/11997856/JKU/JKUPDD-Aug2013.zip. (Accessed 12-May-2014)

Collins, T. (2014). *MIREX task: Discovery of repeated themes and sections, 2014.* http://www.music-ir.org/mirex/wiki/2014:Discovery_of_Repeated_Themes_%26_Sections. (Accessed 12-May-2014)

Collins, T. (2016). *MIREX task: Discovery of repeated themes and sections, 2016.* http://www.music-ir.org/mirex/wiki/2016:Discovery_of_Repeated_Themes_%26_Sections. (Accessed 28-Oct-2016)

Conklin, D. (2006). Melodic analysis with segment classes. *Machine Learning*, *65*(2-3), 349–360.

Corrêa, D. C., & Rodrigues, F. A. (2016). A survey on symbolic data-based music genre classification. *Expert Systems with Applications*, *60*, 190 - 210. Retrieved from http://www.sciencedirect.com/science/article/pii/S095741741630166X doi: http://dx.doi.org/10.1016/j.eswa.2016.04.008

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. Retrieved from http://dx.doi.org/10.1007/BF00994018 doi: 10.1007/BF00994018

Costa, Y. M., Oliveira, L., Koerich, A. L., Gouyon, F., & Martins, J. (2012). Music genre classification using LBP textural features. *Signal Processing*, *92*(11), 2723–2737.

Daubechies, I., & Maes, S. (1996). A nonlinear squeezing of the continuous wavelet transform based on auditory nerve models. In A. Aldroubi & M. Unser (Eds.), *Wavelets in medicine and biology* (pp. 527–546). Boca Raton, FL: CRS Press.

Daugman, J. G. (1980). Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, *20*(10), 847–856.

Deutsch, D. (1999). Grouping mechanisms in music. In D. Deutsch (Ed.), *The psychology of music* (2nd ed., pp. 299–348). San Diego: Academic Press.

Deutsch, D. (2013). *Psychology of music* (3rd ed.). San Diego: Academic Press.

Dominguez, A. (2015). A history of the convolution operation. *Pulse IEEE*. Retrieved from http://pulse.embs.org/january-2015/history-convolution-operation/

Dreyfus, L. (1996). *Bach and the patterns of invention*. Cambridge, MA: Harvard University Press.

Ernst, M. O., & Bülthoff, H. H. (2004). Merging the senses into a robust percept. *Trends in Cognitive Sciences*, *8*(4), 162–169.

Farge, M. (1992). Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics*, *24*(1), 395–458.

Fix, E., & Hodges, J. (1951). *Discriminatory analysis, nonparametric discrimination: Consistency properties* (Technical Report No. 4). Texas: USAF School of Aviation Medicine, Randolph Field.

Fu, Z., Lu, G., Ting, K. M., & Zhang, D. (2011). A survey of audio-based music classification and annotation. *IEEE Transactions on Multimedia*, *13*(2), 303-319. doi: 10.1109/TMM.2010.2098858

Gabor, D. (1946). Theory of communication. *Electrical Engineers-Part III: Radio and Communication Engineering, Journal of the Institution of*, *93*(26), 429–441.

Gammerman, A., & Vovk, V. (2015). Alexey Chervonenkis's bibliography. *Journal of Machine Learning Research*, *16*, 2067–2080. Retrieved from `http://jmlr.org/papers/v16/gammerman15c.html`

Grijp, L. P. (2008). Introduction. In L. P. Grijp & I. van Beersum (Eds.), *Onder de groene linde–163 dutch ballads from the oral tradition* (p. 187). Amsterdam/Hilversum: Meertens Instituut Music and Words.

Haar, A. (1910). Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, *69*(3), 331–371.

Herlands, W., Der, R., Greenberg, Y., & Levin, S. (2014). A machine learning approach to musically meaningful homogeneous style classification. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada, July 27-31, 2014* (pp. 276–282). Retrieved from `http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8314`

Hidaka, S., Teramoto, W., Sugita, Y., Manaka, Y., Sakamoto, S., Suzuki, Y., & Coleman, M. (2011). Auditory motion information drives visual motion perception. *PLoS One*, *6*(3), e17499.

Hillewaere, R., Manderick, B., & Conklin, D. (2010). String quartet classification with monophonic models. In J. S. Downie & R. C. Veltkamp (Eds.), *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, Netherlands, August 9-13, 2010* (pp. 537–542). International Society for Music Information Retrieval. Retrieved from `http://ismir2010.ismir.net/proceedings/ismir2010-91.pdf`

Hontanilla, M., Pérez-Sancho, C., & Iñesta, J. M. (2013). Modeling musical style with language models for composer recognition. In J. M. Sanches, L. Micó, & J. S. Cardoso (Eds.), *Pattern Recognition and Image Analysis: 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5-7, 2013. Proceedings* (pp. 740–748). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from `http://dx.doi.org/10.1007/978-3-642-38628-2_88` doi: 10.1007/978-3-642-38628-2_88

Horton, J. (2014). *In Defence of Musical Analysis.* https://www.youtube.com/watch?v=BP7Gw3mfe-U. ([Accessed 20-Sep-2016])

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, *160*(1), 106.

Huron, D. (1996). The melodic arch in western folksongs. *Computing in Musicology*, *10*, 3–23.

Janssen, B., De Haas, W. B., Volk, A., & Van Kranenburg, P. (2013). Finding repeated patterns in music: State of knowledge, challenges, perspectives. In *International Symposium on Computer Music Modeling and Retrieval* (pp. 277–297).

Jeon, W., & Ma, C. (2011, May). Efficient search of music pitch contours using wavelet transforms and segmented dynamic time warping. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (p. 2304-2307). doi: 10.1109/ICASSP.2011.5946943

Jones, J. P., & Palmer, L. A. (1987). An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, *58*(6), 1233–1258.

Juhász, Z. (2009). Motive identification in 22 folksong corpora using dynamic time warping and self organizing maps. In K. Hirata, G. Tzanetakis, & K. Yoshii (Eds.), *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe International Conference Center, Kobe, Japan, October 26-30, 2009* (pp. 171–176). International Society for Music Information Retrieval. Retrieved from http://ismir2009.ismir.net/proceedings/PS1-20.pdf

Karmakar, A., Kumar, A., & Patney, R. (2011). Synthesis of an optimal wavelet based on auditory perception criterion. *EURASIP Journal on Advances in Signal Processing*, *2011*(1), 1–13.

Kay, K. N., Naselaris, T., Prenger, R. J., & Gallant, J. L. (2008). Identifying natural images from human brain activity. *Nature*, *452*(7185), 352–355.

Keogh, E., & Kasetty, S. (2003). On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining and Knowledge Discovery*, *7*(4), 349–371. Retrieved from http://dx.doi.org/10.1023/A:1024988512476 doi: 10.1023/A:1024988512476

Lamont, A., & Dibben, N. (2001). Motivic structure and the perception of similarity. *Music Perception: An Interdisciplinary Journal*, *18*(3), 245–274. Retrieved from http://www.jstor.org/stable/10.1525/mp.2001.18.3.245

Lartillot, O. (2014). Patminr: In-depth motivic analysis of symbolic monophonic sequences. *Music Information Retrieval Evaluation eXchange, Taipei, Taiwan*. Retrieved from http://www.music-ir.org/mirex/abstracts/2014/OL1.pdf

Lavington, S. (2012). *Alan Turing and his Contemporaries: Building the world's first computers*. Swindon, GB: BCS, The Chartered Institute for IT. Retrieved from http://www.ebrary.com

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324. doi: 10.1109/5.726791

LeCun, Y., Kavukcuoglu, K., & Farabet, C. (2010, May). Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems* (pp. 253–256). doi: 10.1109/ISCAS.2010.5537907

Lerdahl, F., & Jackendoff, R. S. (1983). *A generative theory of tonal music*. Cambridge, MA.: MIT Press.

Marčelja, S. (1980). Mathematical description of the responses of simple cortical cells. *Journal of Neuophysiology*, *70*(11), 1297–1300.

Marsden, A. (2016). Music analysis by computer: Ontology and epistemology. In D. Meredith (Ed.), *Computational music analysis* (pp. 3–28). Cham: Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-25931-4_1 doi: 10.1007/978-3-319-25931-4_1

Meredith, D. (2013). COSIATEC and SIATECCompress: Pattern discovery by geometric compression. In *Music Information Retrieval Evaluation eXchange (MIREX 2013), Curitiba, Brazil*. International Society for Music Information Retrieval. Retrieved from http://vbn.aau.dk/files/181893482/DM10.pdf

Monelle, R. (1992). *Linguistics and semiotics in music*. Chur, Harwood Academic.

Morlet, J. (1981). Sampling theory and wave propagation. In *Proceedings of the 51st Annual Meeting of the Society of Exploration Geophysicists. Los Angeles, Ca, October 1981.*

Müllensiefen, D., & Wiggins, G. (2011a). Polynomial functions as a representation of melodic phrase contour. In A. Schneider & A. van Ruschowski (Eds.), *Systematic musicology: Empirical and theoretical studies* (Vol. 28 of the Hamburger Jahrbuch für Musikwissenschaft). Peter Lang.

Müllensiefen, D., & Wiggins, G. A. (2011b). Sloboda and Parker's recall paradigm for melodic memory: a new, computational perspective. *Music and the Mind: Essays in Honour of John Sloboda*, 161.

Müller, M. (2015). *Fundamentals of music processing*. Switzerland: Springer International Publishing. doi: 10.1007/978-3-319-21945-5

Murdock Jr., B. B. (1979). Convolution and correlation in perception and memory. In L.-G. Nilsson (Ed.), *Perspectives on Learning and Memory* (pp. 105–119). Hillsdale, NJ: Erlbaum.

Nattiez, J.-J. (1975). *Fondements d'une sémiologie de la musique*. Paris: Union Générale d'Éditions.

Nieto, O., & Farbood, M. (2013). Mirex 2013: Discovering musical patterns using audio structural segmentation techniques. *Music Information Retrieval Evaluation eXchange, Curitiba, Brazil*.

Nilsson, N. J. (2010). *The quest for artificial intelligence*. New York, NY: Cambridge University Press.

Nixon, M. S., & Aguado, A. S. (2012). *Feature extraction & image processing for computer vision*. Oxford, OX: Academic Press.

Paulus, J., Müller, M., & Klapuri, A. (2010). Audio-based music structure analysis. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)* (pp. 625–636). Utrecht, The Netherlands.

Pinto, A. (2009, June). Indexing melodic sequences via wavelet transform. In *2009 IEEE International Conference on Multimedia and Expo* (p. 882-885). doi: 10.1109/ICME.2009.5202636

Ponce de León, P., & Iñesta, J. M. (2004). Statistical description models for melody analysis and characterization. In *Proceedings of the 2004 International Computer Music Conference* (pp. 149–156). University of Miami: International Computer Music Association.

Rosenblatt, F. (1957). *The perceptron–a perceiving and recognizing automaton* (Report No. 85-460-1). New York: Cornell Aeronautical Laboratory.

Ruwet, N. (1966). Méthodes d'analyse en musicologie. *Revue belge de Musicologie/Belgisch Tijdschrift voor Muziekwetenschap*, 65–90.

Schenker, H. (1935). *Der freie satz. (E. Oster, trans. by as: Free composition)*. New York, NY: Schirmer Books. (Reprint 1979, translated by E. Oster)

Schmuckler, M. A. (1999). Testing models of melodic contour similarity. *Music Perception: An Interdisciplinary Journal*, *16*(3), 295–326. Retrieved from `http://mp.ucpress.edu/content/16/3/295` doi: 10.2307/40285795

Schnupp, J. (2006). Auditory filters, features, and redundant representations. *Neuron*, *51*(3), 278 - 280. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0896627306005563` doi: http://dx.doi.org/10.1016/j.neuron.2006.07.016

Schoenberg, A. (1984). *Style and idea: Selected writings*. Berkeley: California University Press. (translated by Leo Black)

Schön, D., & Besson, M. (2005). Visually induced auditory expectancy in music reading: a behavioral and electrophysiological study. *Journal of Cognitive Neuroscience*, *17*(4), 694–705.

Shafer, J. C. (2016). Classification and pattern extraction: Applications of wavelets in music analysis. *The Ohio State University*. Retrieved from https://etd.ohiolink.edu/ (Electronic Thesis or Dissertation)

Sinaga, F., Gunawan, T. S., & Ambikairajah, E. (2003). Wavelet packet based audio coding using temporal masking. In *Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on* (Vol. 3, pp. 1380–1383).

Smith, L. M., & Honing, H. (2008). Time–frequency representation of musical rhythm by continuous wavelets. *Journal of Mathematics and Music*, *2*(2), 81–97. Retrieved from http://dx.doi.org/10.1080/17459730802305336

Srinivasan, P., & Jamieson, L. H. (1998, Apr). High-quality audio compression using an adaptive wavelet packet decomposition and psychoacoustic modeling. *IEEE Transactions on Signal Processing*, *46*(4), 1085-1093. doi: 10.1109/78.668558

Stein, L. (1979). *Structure & style: the study and analysis of musical forms*. Summy-Birchard Music.

Tuia, D., Volpi, M., Mura, M. D., Rakotomamonjy, A., & Flamary, R. (2014). Automatic feature learning for spatio-spectral image classification with sparse SVM. *IEEE Transactions on Geoscience and Remote Sensing*, *52*(10), 6062-6074. doi: 10.1109/TGRS.2013.2294724

Uitdenbogerd, A. L., & Zobel, J. (1998). Manipulation of music for melody matching. In *Proceedings of the Sixth ACM International Conference on Multimedia* (pp. 235–240). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/290747.290776 doi: 10.1145/290747.290776

Urbano, J., Lloréns, J., Morato, J., & Sánchez-Cuadrado, S. (2010). Melodic similarity through shape similarity. In *International Symposium on Computer Music Modeling and Retrieval* (pp. 338–355).

van Kranenburg, P., & Backer, E. (2004). Musical style recognition—a quantitative approach. In R. Parncutt, A. Kessler, & F. Zimmer (Eds.), *Proceedings of the Conference on Interdisciplinary Musicology (CIM04) Graz, Austria, April 15-18, 2004* (pp. 106–107).

van Kranenburg, P., Volk, A., & Wiering, F. (2013). A comparison between global and local features for computational classification of folk song melodies. *Journal of New Music Research*, *42*(1), 1-18. Retrieved from http://dx.doi.org/10.1080/09298215.2012.718790 doi: 10.1080/09298215.2012.718790

Velarde, G. (2010). Pattern identification in melody via wavelets. *Late breaking Demo, International Society of Music Information Retrieval, Utrecht, The Netherlands*. Retrieved from http://ismir2010.ismir.net/proceedings/late-breaking-demo-02.pdf

Velarde, G., Cancino Chacón, C., Meredith, D., Weyde, T., & Grachten, M. (in review). Convolution-based classification of audio and symbolic representations of music. *Submitted*.

Velarde, G., & Meredith, D. (2014). A wavelet-based approach to the discovery of themes and sections in monophonic melodies. *Music Information Retrieval Evaluation eXchange, Taipei, Taiwan*. Retrieved from http://www.music-ir.org/mirex/abstracts/2014/VM1.pdf

Velarde, G., Meredith, D., & Weyde, T. (2016). A wavelet-based approach to pattern discovery in melodies. In D. Meredith (Ed.), *Computational music analysis* (pp. 303–333). Cham: Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-25931-4_12 doi: 10.1007/978-3-319-25931-4_12

Velarde, G., & Weyde, T. (2011). Symbolic melody classification using wavelets. In *Digital Music Research Network One-day Workshop, London, 20 December 2011.* London, United Kingdom.

Velarde, G., & Weyde, T. (2012a). Melodic structure and automatic classification in Bach's 2-part inventions. In B. Abels, M. Grant, & A. Waczkat (Eds.), *Proceedings of the Conference on Interdisciplinary Musicology (CIM12), Göettingen, 4-5 September, 2012.*

Velarde, G., & Weyde, T. (2012b). On symbolic music classification using wavelet transform. In *International Conference on Applied and Theoretical Information Systems Research, 10-12 February, Taipei, Taiwan, 2012.* Academy of Taiwan Information Systems Research. (August, 2013. An error has been found in the code of the experiment that affects the results shown in tables and figures.)

Velarde, G., & Weyde, T. (2012c). The relevance of wavelet representation of melodic shape. In *Music & Shape Conference, London, 12-14 July 2012.* London, United Kingdom.

Velarde, G., & Weyde, T. (2012d). Wavelet-based melody representation and segmentation for recognition of tune families. In *Digital Music Research Network One-day Workshop, London, 18 December, 2012.* London, United Kingdom.

Velarde, G., Weyde, T., Cancino Chacón, C. E., Meredith, D., & Grachten, M. (2016). Composer recognition based on 2d-filtered piano-rolls. In M. I. Mandel, J. Devaney, D. Turnbull, & G. Tzanetakis (Eds.), *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016, New York City, United States, August 7-11, 2016* (pp. 115–121). Retrieved from https://wp.nyu.edu/ismir2016/wp-content/uploads/sites/2294/2016/07/063_Paper.pdf

Velarde, G., Weyde, T., & Meredith, D. (2013a). An approach to melodic segmentation and classification based on filtering with the haar-wavelet. *Journal of New Music Research*, *42*(4), 325–345. Retrieved from http://dx.doi.org/10.1080/09298215.2013.841713 doi: 10.1080/09298215.2013.841713

Velarde, G., Weyde, T., & Meredith, D. (2013b). Wavelet-filtering of symbolic music representations for folk tune segmentation and classification. In P. van Kranenburg, C. Anagnostopoulou, & A. Volk (Eds.), *Proceedings of the Third International Workshop on Folk Music Analysis, 6-7 June, Amsterdam, The Netherlands, 2013* (pp. 56–62).

Wu, M.-J., Chen, Z.-S., Jang, J.-S. R., Ren, J.-M., Li, Y.-H., & Lu, C.-H. (2011). Combining visual and acoustic features for music genre classification. In *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on* (Vol. 2, pp. 124–129).

Part II

Collection of Papers

Paper I. Wavelet-filtering of symbolic music representations for folk tune segmentation and classification.

Wavelet-filtering of symbolic music representations for folk tune segmentation and classification

Velarde, Gissel; Weyde, Tillman; Meredith, David

Published in:
Proceedings of the Third International Workshop on Folk Music Analysis (FMA2013)

Publication date:
2013

Document Version
Peer reviewed version

WAVELET-FILTERING OF SYMBOLIC MUSIC REPRESENTATIONS FOR FOLK TUNE SEGMENTATION AND CLASSIFICATION

Gissel Velarde
Aalborg University
gv@create.aau.dk

Tillman Weyde
City University London
T.E.Weyde@city.ac.uk

David Meredith
Aalborg University
dave@create.aau.dk

ABSTRACT

The aim of this study is to evaluate a machine-learning method in which symbolic representations of folk songs are segmented and classified into tune families with Haar-wavelet filtering. The method is compared with previously proposed Gestalt-based method. Melodies are represented as discrete symbolic pitch-time signals. We apply the continuous wavelet transform (CWT) with the Haar wavelet at specific scales, obtaining filtered versions of melodies emphasizing their information at particular time-scales. We use the filtered signal for representation and segmentation, using the wavelet coefficients' local maxima to indicate local boundaries and classify segments by means of k-nearest neighbours based on standard vector-metrics (Euclidean, cityblock), and compare the results to a Gestalt-based segmentation method and metrics applied directly to the pitch signal. We found that the wavelet based segmentation and wavelet-filtering of the pitch signal lead to better classification accuracy in cross-validated evaluation when the time-scale and other parameters are optimized.

1. INTRODUCTION

One of the aims of folk song research is the study of melodic variations caused by the process of oral transmission between generations (van Kranenburg et al., 2009). Wiering et al. (2009) propose an interdisciplinary and ongoing process between human expertise, methods and models to understand melodic variation and its mechanisms. Classification models and methods dealing with such challenges define their representation and processing to be evaluated based on some ground truth. In this paper, we present our method based on wavelet-filtering and evaluate it on a collection of Dutch folk songs ("Onder de groene linde", Grijp, 2008), in which songs were classified into tune families according to expert similarity assessments, mainly based on rhythm, contour and motifs (Wiering et al., 2009; Volk & van Kranenburg, 2012).

The collection of folks songs that we study in this paper, is a monophonic collection of Dutch folk melodies encoded in MIDI files, so that we have pitches encoded as integer numbers, ranging from 0 to 127, and onsets and durations in quarter notes and subdivisions. In order to analyse these files via wavelets, we sample each melody as a one dimensional (1D) signal. Graphically, the melodic contour of 1D pitch signal can be drawn in a pitch over time plot, with the horizontal axis representing time in quarter notes, and the vertical axis representing pitch numbers. This contour representation of melodies has been linked to human melodic processing, using contour classes (Huron, 1996), interpolation lines (Steinbeck, 1982) and polynomial functions (Müllensiefen & Wiggins, 2011; Müllensiefen, Bonometti, Stewart & Wiggins, 2009). However, the contour representation does not give direct access to some aspects that are important for music similarity. Large-scale changes, like transposition of a melody lead to a completely different set of values although the melody is not substantially different. Similarly, small-scale changes like ornaments can lead to different pitch values even if the main essential shape of the melody is preserved.

Wavelet coefficients are obtained as the inner product of a 1D signal and a wavelet (i.e., a short signal with zero average and defined energy). The wavelet is shifted along the time axis and for each time position a coefficient is calculated. This is equivalent to a convolution with the wavelet flipped along the time axis, and thus to a finite impulse response filtering of the signal. The wavelet can be stretched on the time axis, leading to coefficients at different time-scales, corresponding to different filters. This process can also be understood as comparing the melodic shape with the wavelet shape, so that the coefficients represent similarity values at different time-positions and time-scales. The process of producing a full set of wavelet coefficients for a signal is known as the wavelet transform (WT), of which there are different variants. The transformed signal is represented as a set of coefficient signals at different scales. We use the Haar wavelet, which is a function of time t that takes values of 1 if $0 \leq t < 0.5$, or $0.5 \leq t < 1$, and 0 otherwise.

We use the information of the wavelet coefficients to define and compare melodic segments. Local maxima of the wavelet coefficients occur when the inner product of the melody and the wavelet is maximal in that position. In the case of the Haar wavelet this occurs when there is a locally maximal change of pitch - averaged over half the length of the wavelet - in the melody. Therefore, we use the local maxima of wavelet coefficients to indicate segmentation points. If the found segments correlate with human structural perception and music theory, we assume that they can be used to classify melodies containing similar segments. A melodic fragment and its transposed version will be represented by the same wavelet coefficients (except for very beginning of the melody).

Musical similarity in folk music is a hard problem to define (Wiering et al., 2009). We can understand it as a partial identity, where entities share some properties that can be measured (Cambouropoulos, 2009). With wavelet-filtering we apply a process that selectively focuses on a specific time-scale. It is a preprocessing step before determining segment similarities, which we calculate based on distance metrics. In the following section we will discuss some computational models and methods that have been used to model melodic similarity in symbolic music representation and have been applied to classify folk melodies.

2. RELATED WORK

2.1 Modelling melodic variations

Computational models applied to modelling melodic variations in symbolic music representations of folk songs include string matching methods and multidimensional feature vectors to represent global properties of melodies (Hillewaere, Manderick & Conklin, 2009; Hillewaere, Manderick & Conklin, 2012; van Kranenburg, 2010). In origin and genre classification, global representations perform only slightly worse than string-based methods (Hillewaere et al., 2009 and 2012). However, methods based on global representation depend heavily on the choice of features, which can lead to reduce generalizability.

Van Kranenburg, Volk & Wiering (2013) showed that sequence alignment algorithms using local features prove successful in classifying folk song melodies to tune families defined by experts. Sequence alignment algorithms are used to quantify similarity of sequences by computing the operations needed to transform one sequence into another, by means of substitutions, insertions and deletions (Manderick & Conklin, 2012; van Kranenburg, 2010). Although van Kranenburg's (2010) method was very successful when used to classify melodies from the Dutch folk-song corpus into tune families, its representation requires 14 attributes for each note in a melodic sequence (see van Kranenburg, 2010, pp. 94-95), apart from the standard information that is encoded in MIDI format (pitch number, onset and duration), meaning that this approach might not be applicable for classification using MIDI files only. In the following section we present our method, which can be applied to any data set encoded in MIDI format, or any other format containing pitch, onset and duration information for each note in a melody.

2.2 Gestalt-based segmentation

Segmentation is a core activity for musical processing and cognition (Lerdahl & Jackendoff, 1983). In order to study this mechanism, some authors adapt concepts of visual processing to study musical processing. Cambouropoulos (1997, 2001) presents a segmentation model based on Gestalt principles of similarity and proximity, known as the local boundary detection model (LBDM). The LBDM computes a profile of segmentation strength in the range [0, 1], based pitch intervals, inter-onset-intervals and rests. When the strength exceeds a threshold, a segmentation point is introduced. (Cambouropoulos, 2001). We use the LBDM here as a baseline for our model.

2.3 The use of wavelets in the symbolic domain

Wavelet analysis has been applied to diverse time series datasets. A time series is a set of observations recorded at a specified time (Brockwell & Davis, 2009). The use of wavelets for time series processing and analysis can be found in different areas, i.e. meteorological (Torrence & Compo, 1998), political (Aguiar-Conraria, Magalhaes, Soares, 2012), medical (Hsu, 2010), financial (Hsieh, Hsiao, & Yeh, 2011). Wavelets are also well known in audio music information retrieval (Andén & Mallat, 2011; Jeon & Ma, 2011; Smith & Honing, 2008; Tzanetakis, Essl, & Cook, 2001), but they have been scarcely applied on symbolic music representations. The only example of wavelets applied to symbolic music representation, apart from our previous study (Velarde & Weyde, 2012), is presented by Pinto (2009), demonstrating that it is possible to index melodic sequences with few wavelet coefficients, obtaining improved retrieval results compared to the direct use of melodic sequences. The method used by Pinto can be exploited for compression purposes, whereas our method is used for structural analysis and classification.

3. THE METHOD

We extend the method introduced in Velarde and Weyde (2012) by exploring segmentation based on the information of the wavelet coefficients' local maxima, and evaluate it on the classification of folk tunes into tune families. Our previous study (Velarde & Weyde, 2012) showed good results in a different classification task using the 15 Two-Part Inventions by J. S. Bach.

3.1 Representation

We represent melodies as normalized pitch signals or by the wavelet coefficients of the pitch signals. Discrete pitch signals $v[l]$ with length L are sampled from MIDI files at a rate r (given in number of samples per quarter note), so that we have a pitch value for every time point, expressed as $v[t]$. Rests are replaced by the following procedure: if a rest occurs at the beginning of a sequence, it is replaced by the first pitch number that appears in the sequence, otherwise it is replaced by the pitch number of the last note that precedes it.

Normalized pitch signal representation (vr). We normalize pitch signals segments, by subtracting the average pitch in order to make the representation transposition-invariant. The normalization is applied after the segmentation.

Wavelet representation (wr). We apply the continuous wavelet transform (CWT) (Mallat, 2009), expressed in a discretized version as the inner product of the pitch signal $v[l]$ and the Haar wavelet $\psi_{s,u}[l]$, at position u and scale s:

$$w_s[u] = \sum_{l=1}^{L} \psi_{s,u}[l] v[l] \quad (1)$$

To avoid edge effects due to finite-length sequences (Torrence & Compo, 1998), we pad on both ends with a mirror image of the pitch signal (Woody & Brown, 2007). Once the coefficients are obtained, the segment that corresponds to the padding is removed, so that the signal maintains its original length.

3.2 Segmentation

Wavelet segmentation (ws). Local maxima of the wavelet coefficients occur when the inner product of the melody and the wavelet is maximal. This occurs with the Haar wavelet, when there is a locally maximal change of pitch (averaged over half the length of the wavelet) in the melody. We use local maxima of wavelet coefficients to determine local boundaries.

3.3 Classification

The melodic segments are used as the data points for classification. A melody is represented as a set of segments, and we use the *k*-Nearest-Neighbour (kNN) method for classification (Mitchell, 1997). We use two different distance measures: cityblock distance and Euclidean distance. We define the maximal length n of all segments to be compared and pad shorter segments as necessary with zeros at the end.

4. EXPERIMENT

In our experiment we address the question of how filtering the representation of melodic segments affects the folk tune family classification. We assumed that if segments represent meaningful melodic structures, they can be used to identify tunes belonging to a tune family and that some time-scales of the melodic contour might be more discriminative than others.

We ran the experiment[1] using the collection "Onder de groene linde" (Grijp, 2008). This collection is a high quality data set of 360 monophonic songs classified into 26 families according to field-experts' similarity assessments in terms of melodic, rhythmic and motivic content (Volk & van Kranenburg, 2012). The MIDI files of this collection are sampled into pitch signals with a sampling rate of 8 samples per quarter note (qn). We apply the CWT with the Haar wavelet using a dyadic set of 8 scales. Melodies are represented as normalized pitch signals (vr) or as the resulting wavelet coefficients (wr). Signals are segmented by the wavelet coefficients' local maxima (ws), or by the local boundary detection model (LBDM; Cambouropoulos, 1997, 2001) using thresholds from 0.1 to 0.8 in steps of 0.1. We explored the parameter space with a grid search testing all combinations of representations and segmentations: wavelet representation (wr), normalized pitch signal representation (vr), wavelet segmentation (ws), LBDM (LBDM) segmentation and 1 to 5 nearest neighbours. Segments are used to build classifiers from training sets and that are tested on unseen folk melodies. We evaluate the classification accuracy with cityblock and Euclidean distances in leave-one-out cross validation.

5. RESULTS

The results of the experiment can be seen in Figures 1 to 4. Alternatively, Tables 1 and 2 shows the best and worst classification values over all parameters for each combination of representation-segmentation, for each value of *k* in the kNN method, and for Euclidean and cityblock distance metrics. The results show that wavelet filtering of the melodies can improve classification performance compared to using the pitch signal directly. Independently of the segmentation method, wavelet representation proves to be more discriminative than pitch signals. For this corpus and experimental setup, we have used single time-scales and evaluated this melodic discrimination performance. The classification performance varies, obtaining best results at small scales and poor results at large scales, with exception of the largest scale which recovers its performance to some extent.

In terms of segmentation, it is possible to observe that shorter segments produce better results when used with wavelet representation. This is contrary to the results of the LBDM applied to pitch signals, where shorter segments produce worse results than larger ones. We observe an improvement towards threshold 0.4 and a gradual improvement towards the threshold of 0.8, which corresponds to larger segments, meaning that using the complete melodic sequences or a combination of complete melodies and melodic segments, can lead to better classification results when using pitch signals.

[1] The algorithms are implemented in MATLAB (The Mathworks, Inc) using the Wavelet Toolbox and the MIDI Toolbox for the implementation of the LBDM (Eerola & Toiviainen, 2004), and we use an update of Christine Smit's read_midi function (http://www.ee.columbia.edu/~csmit/matlab_midi.html, accessed 4 October 2012).

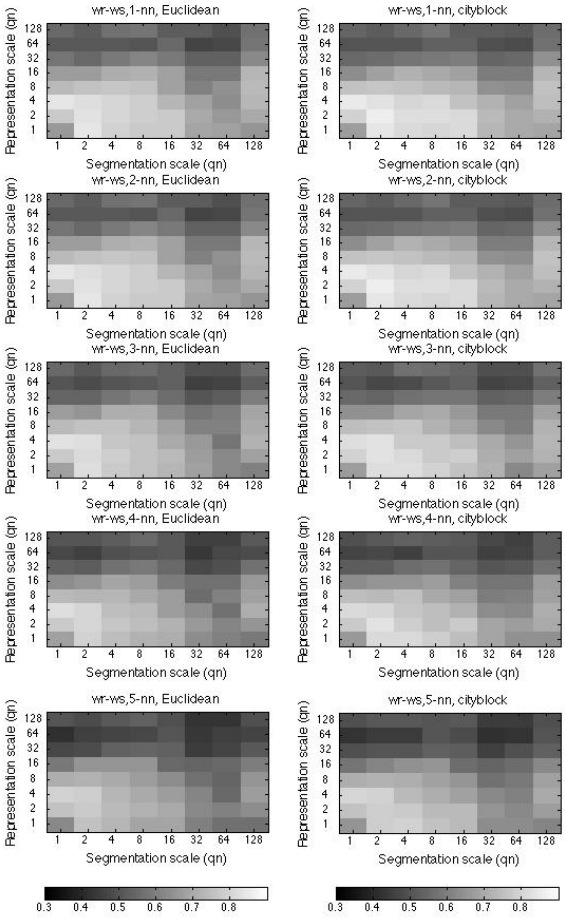


Figure 1. Accuracies for the combination of wavelet representation (wr) and wavelet segmentation (ws).

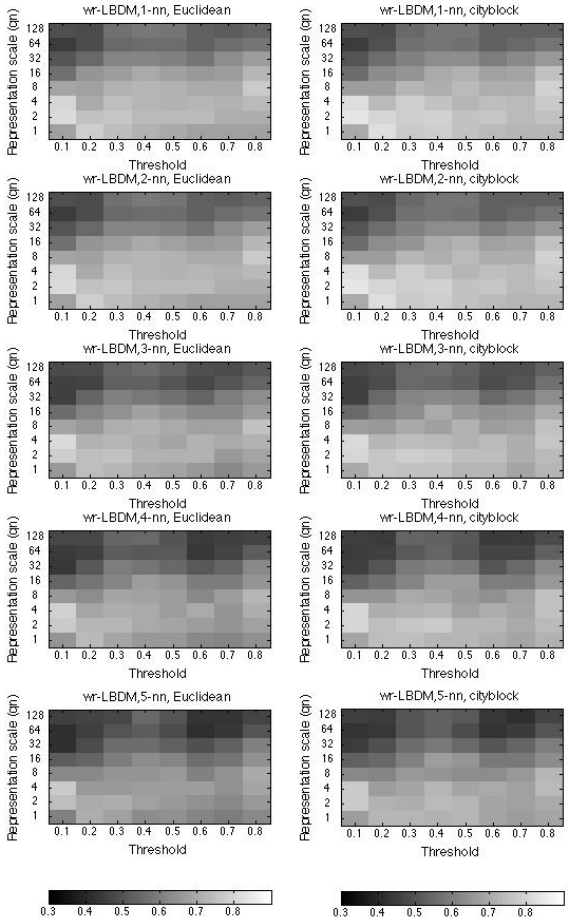


Figure 2. Accuracies for the combination of wavelet representation (wr) and local boundary detection model (LBDM).

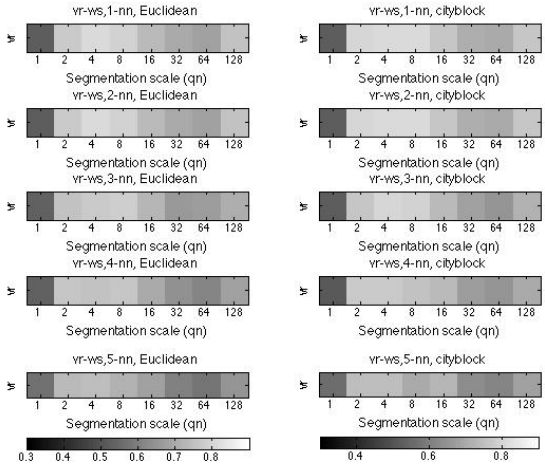


Figure 3. Accuracies for the combination of pitch signal representation (vr) and wavelet segmentation (ws).

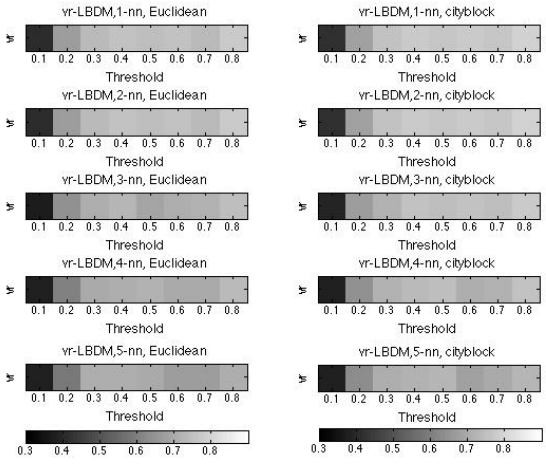


Figure 4. Accuracies for the combination of pitch signal representation (vr) and local boundary detection model (LBDM).

In general, similarity measured by cityblock distance proves more accurate than by Euclidean distance in pitch signals over time or wavelet representations, and the effect of using cityblock distance makes the difference between segmentation methods less important. The number of *k*-nearest neighbours shows that one or two neighbours produce the best results and when *k* increases further the accuracy decreases.

Euclidean distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8417**	**0.8417**	0.8306	0.8194	0.7917
	worst	0.4667	0.4667	0.4583	0.4333	0.4167
wr-LBDM	best	0.8111	0.8111	0.8083	0.7889	0.7694
	worst	0.4472	0.4472	0.4528	0.4333	0.4139
vr-ws	best	0.8083	0.8083	0.7806	0.7667	0.7444
	worst	0.5194	0.5194	0.5333	0.525	0.5639
vr-LBDM	best	0.7778	0.7778	0.7444	0.7333	0.7083
	worst	0.4111	0.4111	0.3722	0.3806	0.3806

Table 1. Classification accuracies best and worst values for each combinations using Euclidean distance.

Cityblock distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8556**	**0.8556**	0.8333	0.8306	0.7972
	worst	0.4833	0.4833	0.4639	0.45	0.4167
wr-LBDM	best	0.8417	0.8417	0.8083	0.8028	0.7778
	worst	0.4417	0.4417	0.4556	0.4417	0.4139
vr-ws	best	0.8139	0.8139	0.7972	0.7778	0.7472
	worst	0.5194	0.5194	0.5194	0.5139	0.5583
vr-LBDM	best	0.7889	0.7889	0.7778	0.75	0.725
	worst	0.4139	0.4139	0.3861	0.3778	0.3806

Table 2. Classification accuracies best and worst values for each combinations using cityblock distance.

6. DISCUSSION AND FUTURE DIRECTIONS

The best classification accuracies based on wavelet segmentation are only slightly better than the best accuracies obtained by the LBDM. The parameter exploration shows however, that wavelet segmentation performs better across different scales than the LBDM across different thresholds. Interestingly, these comparable methods meet the criteria of measuring local changes in melodic contour. While the LBDM measures the degree of change between successive values, the wavelet segmentation finds locally maximal falls of average pitch in melodies using different scales. The fact that small scales perform better than larger scales corroborates the findings of van Kranenburg et al. (2013) that local processing is most important in melodic similarity.

In terms of representation, wavelet-representation proves more discriminative than raw pitch signals. We assume that this is due to the transposition invariance of the wavelet representation and the emphasis on a specific time-scale.

Our best results are far less accurate than the results reported by van Kranenburg et al. (2013) using alignment methods on the same corpus. Our method uses only the information that is encoded in MIDI format (pitch number, onset and duration). It requires less encoded expert knowledge than the method used by van Kranenburg (2010), making it applicable to other corpuses of folk songs encoded in MIDI format or similar. In order to make a more reliable comparison, our method would need to include the expert based features used by van Kranenburg (2010). For instance, annotated phrase information seems to improve importantly the results obtained by sequence alignment algorithms. This information could be used to improve the scale selection. Also, our method uses only the information about contained segments, and not the order of the segments, leaving room for further work.

We used one default setup for the whole corpus, i.e. one best performing scale for all songs. In a future study, we are interested to address wavelet scale selection derived from individual songs' periodicities.

7. CONCLUSION

The main contribution of this research is the evaluation of wavelet-filtered signals for melodic segmentation and classification on a corpus of folk songs in MIDI format. Wavelet-filtering proves more discriminative than direct representation of pitch signals or pitch-time series. Segmentation by local maxima of wavelet coefficients performs slightly better than LBDM segmentation when processing at individual scales. Small scales perform better than large scales, indicating that local processing may be more relevant for melodic similarity in classification tasks.

The method presented here can be applied to other corpora and other symbolic formats that encode melodies. Possible ways to improve the classification performance of the method presented in this paper could be using alignment of wavelet representations of complete melodies, using selective combination of scales and exploring metrical information derived from songs' periodicities.

Acknowledgements

We thank Peter van Kranenburg (Meertens Institute, Amsterdam) for sharing the Dutch Tune Families data set. Gissel Velarde is supported by the Department of Architecture, Design and Media Technology at Aalborg University.

8. REFERENCES

Andén, J., & Mallat, S. (2011). Multiscale scattering for audio classification. In: *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), Utrecht, NL.: ISMIR*, pp. 657–662. Available online at http://ismir2011.ismir.net/papers/PS6-1.pdf

Aguiar-Conraria, L., Magalhães, P. C. and Soares, M. J. (2012), Cycles in politics: wavelet analysis of political time series. *American Journal of Political Science*, 56: 500–518. doi: 10.1111/j.1540-5907.2011.00566.x

Brockwell, P. & Davis, R. (2009). Time series: theory and methods. Springer series in statistics. Second edition.

Cambouropoulos, E. (1997). Musical rhythm: a formal model for determining local boundaries, accents and metre in a melodic surface. In: *M. Leman (Ed.), Music, Gestalt and Computing: Studies in Cognitive and Systematic Musicology*. Berlin: Springer, pp. 277-293.

Cambouropoulos, E. (2001). The local boundary detection model (LBDM) and its application in the study of expressive timing. In: *Proceedings of the International Computer Music Conference. San Francisco, CA: ICMA*, pp. 17-22.

Cambouropoulos, E. (2009). How similar is similar?. *Musicæ Scientiæ*. Discussion Forum 4B, pp. 7-24

Eerola, T. & Toiviainen, P. (2004). MIDI Toolbox: MATLAB Tools for Music Research. University of Jyväskylä. Available at http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/miditoolbox/.

Grijp, L.P.. (2008). Introduction. In: L.P. Grijp & I. van Beersum (Eds.), Onder de groene linde. 163 verhalende liederen uit de mondelinge overlevering, opgenomen door Ate Doornbosch e.a./Under the green linden. 163 Dutch Ballads from the oral tradition recorded by Ate Doornbosch a.o. (Boek + 9 cd's + 1 dvd). Amsterdam/Hilversum : Meertens Instituut & Music and Words. pp. 18-27.

Hillewaere, R, Manderick, B., & Conklin, D. (2009). Global feature versus event models for folk song classification. In: *10th International Society for Music Information Retrieval Conference (ISMIR 2009), Kobe, Japan*. pp. 729-733. Available online at http://ismir2009.ismir.net/proceedings/OS9-1.pdf.

Hillewaere, R., Manderick, B. and Conklin, D. (2012) String methods for folk music classification. In: *13th International Society for Music Information Retrieval Conference (ISMIR 2012)*.

Huron, D. (1996) The melodic arch in western folksongs. *Computing in Musicology*, Vol. 10, pp. 3-23.

Hsieh, TJ. Hsiao, H.f., & Yeh, WC. (2011). Forecasting stock markets using wavelet transforms and recurrent neural networks: an integrated system based on artificial bee colony algorithm. *Applied soft computing*, Vol. 11 Issue 2. March 2011, pp. 2510–2525. Elsevier

Hsu, WY. (2010). EEG-based motor imagery classification using neuro-fuzzy prediction and wavelet fractal features. *J. Neurosci Methods*. 2010 Jun 15;189(2):295-302. doi: 10.1016/j.jneumeth.2010.03.030. Epub 2010 Apr 8.

Jeon, W., & Ma, C. (2011). Efficient search of music pitch contours using wavelet transforms and segmented dynamic time warping. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011)*, pp. 2304–2307.

Lerdahl, F., & Jackendoff, R. (1983). A Generative Theory of Tonal Music.Cambridge, MA.: MIT Press.

Mallat, S. (2009). A wavelet tour of signal processing - The sparse way. Academic Press, Third Edition, 2009.

Mitchell, T. (1997). Machine Learning, (McGraw-Hill).

Müllensiefen, D., Bonometti, M., Stewart, L., and Wiggins, G. (2009). Testing Different Models of Melodic Contour. *7th Triennial Conference of the European Society of the Cognitive Sciences of Music (ESCOM 2009), University of Jyäskylä, Finalnd.*

Müllensiefen, D, and Wiggins, G. (2011). Polynomial functions as a representation of melodic phrase contour. In *A. Schneider & A. von Ruschkowski (Eds.), Systematic Musicology: Empirical and Theoretical Studies.* pp. 63-88. Frankfurt a.M.: Peter Lang.

Pinto, A. (2009). Indexing melodic sequences via wavelet transform. In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'09*). pp. 882–885.

Smith, L.M., & Honing, H. (2008). Time-frequency representation of musical rhythm by continuous wavelets. *Journal of Mathematics and Music*, Vol. 2, No. 2, pp. 81–97.

Steinbeck, W. (1982). Struktur und Ähnlichkeit: Methoden automatisierter Melodieanalyse. Kassel: Bärenreiter.

Tzanetakis, G., Essl, G., & Cook, P. (2001). Audio analysis using the discrete wavelet transform. In: *Proc. WSES Int. Conf. Acoustics and Music: Theory and Applications (AMTA 2001), Skiathos, Greece*. Available online at http://webhome.cs.uvic.ca/~gtzan/work/pubs/amta01gtzan.pdf

van Kranenburg, P. (2010) A Computational Approach to Content-Based Retrieval of Folk Song Melodies. [S.l.] : [s.n.], 2010. Full text: http://depot.knaw.nl/8400

van Kranenburg, P., Garbers, J., Volk, A., Wiering, F. Grijp, L.P. and Veltkamp, R. (2009). Collaboration perspectives for folk Song research and music information retrieval: The indispensable role of computational musicology, *Journal of Interdisciplinary Music Studies*. (2009), doi: 10.4407/jims.2009.12.030

van Kranenburg, P., Volk, A., & Wiering, F. (2013): A Comparison between Global and Local Features for Computational Classification of Folk Song Melodies, *Journal of New Music Research*, 42:1, 1-18

Velarde, G., & Weyde, T. (2012). On symbolic music classification using wavelet transform. In: *International Conference on Applied and Theoretical Information Systems Research, Taipei, Taiwan*.

Volk, A. & van Kranenburg, P. (2012). Melodic similarity among folk songs: An annotation study on similarity-based categorization in music. *Musicae Scientiae*, November 2012 vol. 16 no. 3, pp. 317-339.

Wiering, F., Veltkamp, R., Garbers, J., Volk, A. and van Kranenburg, P. (2009). Modelling Folksong Melodies. *Interdiciplinary Science Reviews*, Vol 34, No. 2-3, 154-171.

Woody, N. A. & Brown, S. D. (2007) Selecting wavelet transform scales for multivariate classification. *J. Chemometrics*, 21: 357–363. doi: 10.1002/cem.1060

Paper II. An approach to melodic segmentation and classification based on filtering with the Haar-wavelet.

An approach to melodic segmentation and classification based on filtering with the Haar-wavelet

Velarde, Gissel; Weyde, Tillman; Meredith, David

Published in:
Journal of New Music Research

DOI (link to publication from Publisher):
10.1080/09298215.2013.841713

Publication date:
2013

Document Version
Accepted manuscript, peer reviewed version

This is an Author's Original Manuscript of an article whose final and definitive form, the Version of Record, has been published in the Journal of New Music Research, available online at:
http://www.tandfonline.com/doi/full/10.1080/09298215.2013.841713

An approach to melodic segmentation and classification based on filtering with the Haar-wavelet

Gissel Velarde*, Tillman Weyde** and David Meredith*

*Aalborg University, Denmark; **City University London, UK

Correspondence: Gissel Velarde, Aalborg University, Department of Architecture, Design and Media Technology, Sofiendalsvej 11, 9200 Aalborg SV, Denmark. Email: gv@create.aau.dk

Abstract

We present a novel method of classification and segmentation of melodies in symbolic representation. The method is based on filtering pitch as a signal over time with the Haar-wavelet, and we evaluate it on two tasks. The filtered signal corresponds to a single-scale signal w_s from the continuous Haar wavelet transform. The melodies are first segmented using local maxima or zero-crossings of w_s. The segments of w_s are then classified using the *k*–nearest neighbour algorithm with Euclidian and city-block distances. The method proves more effective than using unfiltered pitch signals and Gestalt-based segmentation when used to recognize the parent works of segments from Bach's *Two-Part Inventions* (BWV 772–786). When used to classify 360 Dutch folk tunes into 26 tune families, the performance of the method is comparable to the use of pitch signals, but not as good as that of string-matching methods based on multiple features.

Keywords: *Music analysis, wavelet analysis, classification, symbolic music, melodic analysis, information retrieval, folk song analysis, melodic segmentation*

1 Introduction

Melodic classification models depend strongly on melodic representation. Computational models that work on symbolic data (e.g., MIDI) usually transform the data into a suitable representation before applying any machine learning technique.

Most computational approaches for melodies use string methods, treating melodies as sequences of notes or intervals, and modelling distributions and transitions of note properties (Knopke & Jürgensen, 2009; Hillewaere, Manderick, & Conklin, 2009). Other approaches use multidimensional feature vectors to represent global properties of melodies, assigning coefficients to various musical dimensions (Ponce de Léon & Iñesta, 2004; Hillewaere, Manderick, & Conklin, 2012; van Kranenburg, 2010).

We present below a method for analysing and classifying monophonic melodies, which involves filtering symbolic representations of melodies with the Haar wavelet. We evaluate it on two classification tasks, each using a different MIDI dataset. In the first task, we use the approach to identify the parent works of segments from the parts of the fifteen *Two-Part Inventions* (BWV 772–786) by Johann Sebastian Bach (1685-1750)[1]. In the second task, the method is used to classify 360 Dutch folk songs into 26 tune families (Grijp, 2008). We compare our wavelet-based approach to the use of unfiltered pitch signals and a previous Gestalt-based model of segmentation (Cambouropoulos, 1997, 2001).

2 Background

2.1 The wavelet transform

The *wavelet transform* (WT) is a mathematical tool that was born from a multidisciplinary effort in mathematics, physics, computer science and engineering. Having developed rapidly since the second half of the 1980s, wavelets have been used for numerous applications (Daubechies, 1996; Mallat, 2009) and are today a standard tool in audio and image processing.

In the context of time-based one-dimensional (1D) signals, a wavelet is a signal that has finite energy concentrated over a short amount of time and that is zero or almost zero everywhere else. Mathematically, a wavelet is normally characterized by a total energy of 1 and an average of 0, with its energy centred around time 0 (Mallat, 2009). The WT decomposes a signal into a sum of components based on different versions of a so-called *mother wavelet* and often an additional scaling function, also called the *father wavelet*. We focus here on the mother wavelet and the coefficients that are based on shifted and scaled versions of the mother wavelet.

[1] We used the Musedata encodings of Bach's *Two-Part Inventions*, available at http://www.musedata.org.

Shifting refers to the position of the wavelet in time, while scaling refers to the degree of compression of the wavelet shape on the time axis, along with a normalization factor to maintain an energy of 1 (Antoine, 1999; Daubechies, 1996). The scaled and shifted versions of the wavelet are weighted by coefficients, determined by the inner product with the wavelet, so that they add up to the original signal. The wavelet transformation can also be viewed as using a filter-bank, where the coefficients at each scale correspond to a different band-pass filter that emphasises a specific scale in the signal (see Farge, 1992, pp. 449–450).

The WT is similar to the Fourier transform, with Fourier frequency corresponding to the inverse scale in wavelets. The sine and cosine functions used in Fourier analysis are periodic signals, so that the Fourier components are not localized in time within the signal being analysed. Wavelets, by contrast, have localized energy and use several shifted and scaled versions, so that wavelet coefficients become more localized in time when the scale decreases, at the expense of scale resolution. Wavelet analysis offers a trade-off between better time resolution for small scales, corresponding to high frequencies, and better scale resolution for large scales, corresponding to low frequencies (Antoine, 1999; Farge, 1992; Torrence & Compo, 1998).

There are different types of wavelets with different properties and the choice of wavelet to analyse a signal depends on the type of the signal and the features that are relevant to the analysis. There are two main forms of the WT, the *continuous wavelet transform* (CWT) and the *discrete wavelet transform* (DWT), and the two different forms tend to be used for different purposes. The CWT is mostly used for signal analysis (i.e., pattern identification or feature detection), while the DWT is used for compression and reconstruction (Antoine, 1999; Mallat, 2009). Our method is based on the CWT, which will be described below.

In audio music information retrieval (MIR), both the continuous and discrete WT have been applied extensively in tasks such as rhythmic content analysis (Smith & Honing, 2008), feature extraction for music genre classification (Andén & Mallat, 2011; Grimaldi, Cunningham, & Kokaram, 2003; Tsunoo, Ono, & Sagayama, 2009; Tzanetakis, Essl, & Cook, 2001), pitch contour extraction and melodic indexing in "query-by-humming" systems (Jeon, Ma, & Ming Cheng, 2009; Jeon & Ma, 2011), denoising (Berger, Coifman, & Goldberg, 1994; Yu, Mallat, & Bacry, 2008) and

audio compression (Dobson, Yang, Whitney, Smart, & Rigstaa, 1996; Srinivasan & Jamieson, 1998).

Wavelets exhibit similarities to many information-processing steps in the human brain and have been extensively used in modelling vision (see, e.g., Kay, Naselaris, Prenger, & Gallant, 2008; Zhang, Zhang, Huang, & Tian, 2005; Zhang, Shan, Qing, Chen, & Gao, 2009). In hearing, auditory perception in the cochlea and the auditory pathway has been modelled using bandpass filters based on the CWT and other wavelet-based techniques (Daubechies & Maes, 1996; Sinaga, Gunawan & Ambikairajah, 2003; Karmakar, Kumar & Patney, 2011). The interesting mathematical properties of wavelets and their applicability to modelling neural mechanisms motivate us to explore here the applicability of wavelets to the symbolic level of music description (i.e., to notes and their properties).

2.2 Symbolic music representation and analysis with wavelets

Although wavelets have been used extensively for analysing music audio, the use of the WT is scarce in the symbolic domain. One isolated example is Pinto's (2009) use of the DWT to index melodic sequences with few wavelet coefficients, obtaining improved retrieval results compared to the direct use of the melodies.

A Western staff-notation score depicts a piece of music as a set of notes, specifying (amongst other things) the pitch, relative onset time and relative duration of each note. In a MIDI file, the pitch of each note is specified by its MIDI note number, which represents its chromatic pitch (see Meredith, 2006, pp. 126–129). For the purpose of wavelet analysis, a melody can be represented as a 1D signal, called a *pitch signal*, that indicates the chromatic pitch (MIDI note number) of the melody at each tatum time-point. The pitch signal can then be transformed into coefficients at different scales using the WT. A similar representation using Fourier analysis has been shown by Schmuckler (1999) to capture relevant information for melodic similarity.

2.2.1 Melodic segmentation

Music unfolds over time. This characteristic is the most prominent difference between music and visual art, engaging our brains in a prediction-expectation game of events occurring over time (Huron, 2006; Levitin, 2006). We do not know how a piece will develop or end until it finishes. However, as the music unfolds, we constantly identify segments that start somewhere, develop and end. Finding coherent segments, or

groups, at various different time scales is a basic, automatic aspect of music cognition (Lerdahl & Jackendoff, 1983).

Most theoretical work in music perception has concentrated on the perceived associations of events, based on grouping, adapting visual Gestalt principles of similarity and proximity to musical perception. These theories include Tenney and Polansky's (1980) theory of *temporal Gestalt-units*, Lerdahl and Jackendoff's (1983) grouping structure theory and the *Local Boundary Detection Model* (LBDM) of Cambouropoulos (1997, 2001), which sets local boundaries according to change and proximity rules. The rules in these models address both local changes and longer-term averages, so that representing melodic movements at different scales with wavelet filters, leading to different levels of localization on the time-axis, appears to be an appropriate approach for deriving group boundaries.

2.2.2 Relation to neural mechanisms

Recent neuroscientific imaging work based on EEG, fMRI and MEG provides evidence that musical structure constantly engages the brain in a game of prediction, expectation and reward, based on long-term memory and statistical regularities of coded features (Trainor & Zatorre, 2009). Moreover, it has been observed that brain activity increases transiently at musical movement boundaries, as well as other non-musical event boundaries, and it has been suggested that segmentation is thus an essential perceptual component, occurring simultaneously at multiple time-scales as an adaptive mechanism that integrates recent past information to improve predictions about the near future (Kurby & Zacks, 2008).

Perceptual boundary detection has been successfully modelled with wavelets. For example, Gabor wavelets have been used to model the early stages of the visual pathway (Kay *et al.*, 2008; Nixon & Aguado, 2012; Zhang *et al*., 2005; Zhang *et al*., 2009). It therefore seems reasonable to hypothesise that a similar wavelet-based approach might successfully be used to model group boundary perception in melodies.

2.2.3 Melodic theory

Huron (1996) proposes a reductionist approach to melodic classification, summarizing the contour of a folk song by its first and final pitches, along with an average of all the pitches in between. He demonstrates that folk songs have arc-like contours, with an inverted 'U' shape being the most common. In his study, a melody is classified into

one of nine types, depending on whether it describes a trajectory that is ascending, descending, horizontal or a combination of these basic types.

In Schenkerian analysis (Brown, 2005; Forte & Gilbert, 1982; Schenker, 1935), the musical surface or *foreground* is recursively reduced to a *fundamental structure* (*Ursatz*) by removing notes of progressively greater structural importance. In a wavelet representation, small-scale structures that occur only in the foreground (e.g., ornaments) will be represented only in the small-scale coefficients; whereas the higher structural levels (corresponding loosely to the background or fundamental structure) will be represented by the coefficients at greater time scales. In this way, wavelets at different scales can be used to extract structure at what would correspond to different transformational levels (*Schichten*) in the Schenkerian approach.

It is possible to understand many musical works as having been generated by the reverse of this hierarchical reduction process—that is, by the successive elaboration of a fundamental structure with less structural notes, until the detailed foreground or musical surface emerges. Wavelet filters emphasise different temporal scales in a pitch signal, thus providing a tool to focus on and discover musical structure at a variety of different temporal scales.

3 Method

We investigate the effectiveness of the WT to represent relevant properties of melodies in segmentation and classification tasks. Our input data are sequences of notes, represented as pitch signals. To these we apply the CWT and obtain a time-scale representation for structural analysis in classification tasks. Figure 1 a) presents the score representation of a melodic fragment, Figure 1 b) is the 1D pitch signal that represents it, and Figure 1 c) is its CWT by Haar wavelet, in a scalogram plotting the absolute coefficients, using darker colours for smaller values and brighter colours for larger values.

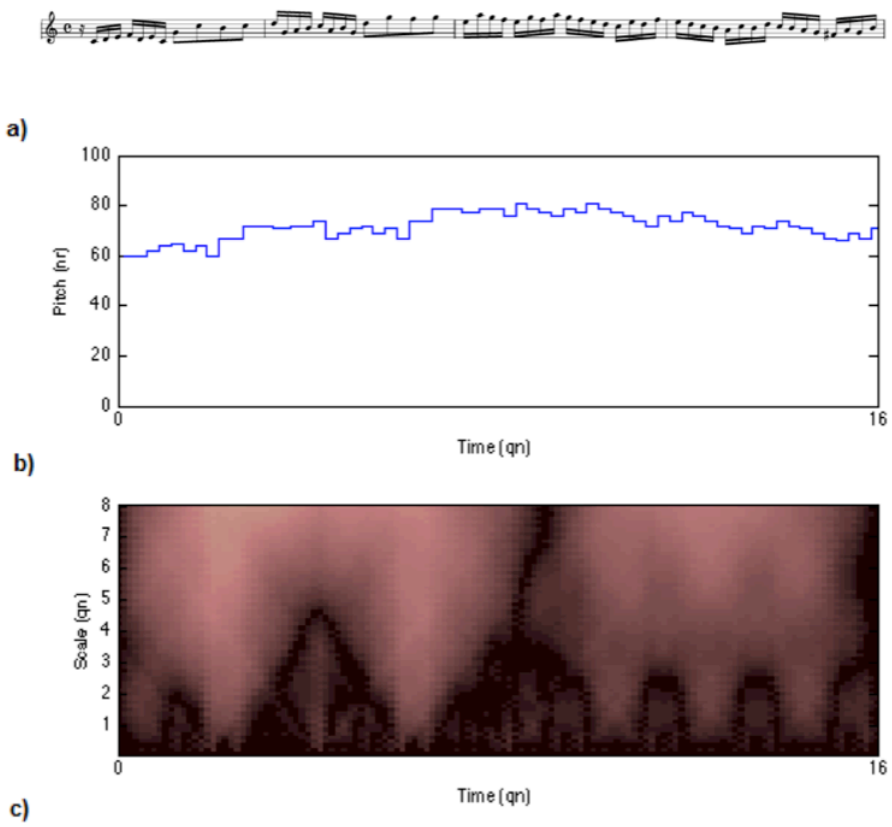


Figure 1. The opening bars of the upper part of J. S. Bach's *Invention* in C major (BWV 772), represented as a) a score, b) a pitch signal and c) a scalogram of the CWT (i.e., the absolute values of the coefficients).

3.1 Representation

We represent melodies as pitch signals or by the wavelet coefficients of the pitch signals.

3.1.1 Pitch signal representation

A discrete *pitch signal* v with length L is sampled from MIDI files at a rate r in number of samples per quarter note (qn), so that we have a pitch value for every time point, expressed as $v[t]$. We use two different ways of treating rests: they are either *represented* by the value 0, or they are *removed* from the representation by the following procedure: if a rest occurs at the beginning of a sequence, it is replaced by the first pitch number that appears in the sequence, otherwise it is replaced with the pitch number of the note that immediately precedes it.

Normalized pitch signal representation. We normalize pitch signal segments by subtracting the average pitch in order to make the representation invariant to transposition. The normalization is applied after the segmentation.

3.1.2 Wavelet representation

The CWT[2] transforms a 1D signal into a set of coefficients $w_{s,u}$ using an analysing function $\psi_{s,u}(t)$, which is derived from the mother wavelet ψ by scaling by a factor $s > 0$ and shifting in time by an amount $u \in \mathbb{R}$:

$$\psi_{s,u}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right). \qquad (1)$$

The coefficients $w_{s,u}$ are calculated for real valued wavelets as the inner product of the signal $v(t)$ and the analysing function $\psi_{s,u}(t)$:

$$w_{s,u} = \left\langle v, \psi_{s,u} \right\rangle = \int_{-\infty}^{+\infty} v(t)\psi_{s,u}(t)dt. \qquad (2)$$

To avoid edge effects due to finite-length sequences (Torrence & Compo, 1998), we pad on both ends with a mirror image of the pitch signal (Woody & Brown, 2007). Once the coefficients are obtained, the segment that corresponds to the padding is removed, so that the signal maintains its original length.

We can treat coefficients on one scale as a function of the shift parameter with $w_s(u) = w_{s,u}$. Then the CWT acts as a *filter*, equivalent to the convolution of v with the scaled and flipped real-valued wavelet. The CWT calculates the wavelet coefficients at all points u, so that the complete information of the pitch signal is still retained in the coefficients at one scale and it can be recovered using deconvolution, given a suitable wavelet.

For implementation on a computer, we can write equation (2) in a discretized version, where we compute the convolution for each translation u and scale s:

$$w_s[u] = \sum_{l=1}^{L} \psi_{s,u}[l]v[l]. \qquad (3)$$

[2] We follow the presentation by Antoine (1999). Signals processed by digital computers have to be discretized. The term "continuous" refers to the fact that all sample positions are used as shift values, as opposed to the discrete wavelet transform where shift values are much sparser.

The summation index only needs to run over the support of ψ, i.e., between the maximum and minimum time-points for which ψ is not zero, which is typically considerably shorter than the signal *v*.

3.2 Wavelet choice

The selection of wavelet or analysing function depends on the kind of information that we want to extract from the signal, considering that the transform's coefficients combine information about the signal and the wavelet (Farge, 1992). The wavelet should give a compact representation of the variation in the signal that we are interested in. We use the Haar wavelet, which is defined by

$$\psi(t) = \begin{cases} 1, & if \quad 0 \le t < 1/2 \\ -1, & if \quad 1/2 \le t < 1 \\ 0, & otherwise \end{cases} \qquad (4)$$

and has a shape as shown in Figure 2.[3]

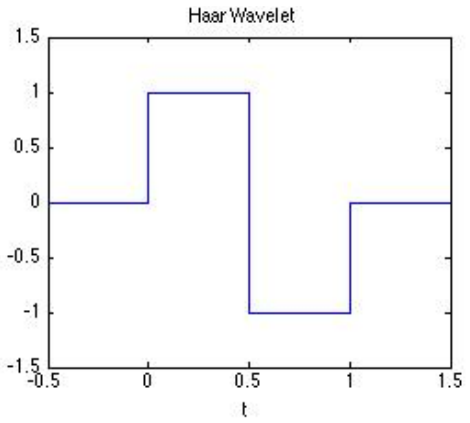


Figure 2. The shape of the Haar wavelet.

We selected the Haar wavelet because it matches the discontinuous, step-wise nature of the pitch signal. A continuous wavelet would require a combination of many small-scale components to represent the step transitions between pitches, obscuring the representation of pitch changes. On the other hand, the Haar wavelet is not suitable for continuous pitch data, which could represent vibrato, glissando, melismatic ornamentation, etc.

[3] The Haar function was introduced by Haar in 1910 (Haar, 1910). Equation (4) uses Mallat's (2009) notation.

The Haar wavelet has support on the time interval [0,*s*), and the inner product with the Haar wavelet calculates the difference between the averages of pitch in the first and second halves of that interval. In other words, the coefficient $w_{s,u}$ gives a measure of whether the melody is moving upwards or downwards over the scale period starting at position *u*.

Figure 3 illustrates the Haar wavelet shifted and scaled. In each of the three rows of sub-figures, different wavelet shifts can be seen (first vs. second column). The scale is 0.5 in the first row and 0.25 in the second and third rows.

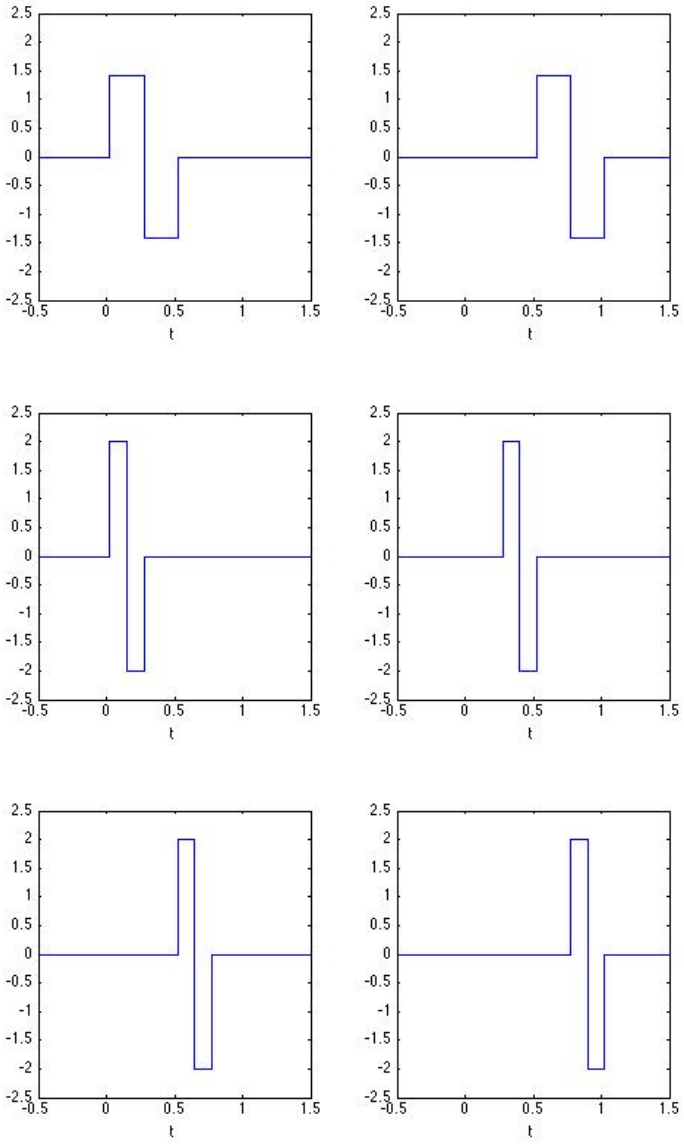


Figure 3. The Haar wavelet shifted and scaled.

3.3 Segmentation

We use the wavelet coefficients to determine melodic segments in two different ways, setting segmentation points either at local maxima or at zero crossings of the wavelet coefficients. Default segmentation points are set at the beginning and at the end of signals.

3.3.1 Zero crossing segmentation

Zero crossings occur when the inner product between the melody and the Haar wavelet is zero. This means that the average pitch in the first half of the scale period is equal to the average pitch in the second half of the scale period. See Figure 4 for illustration.

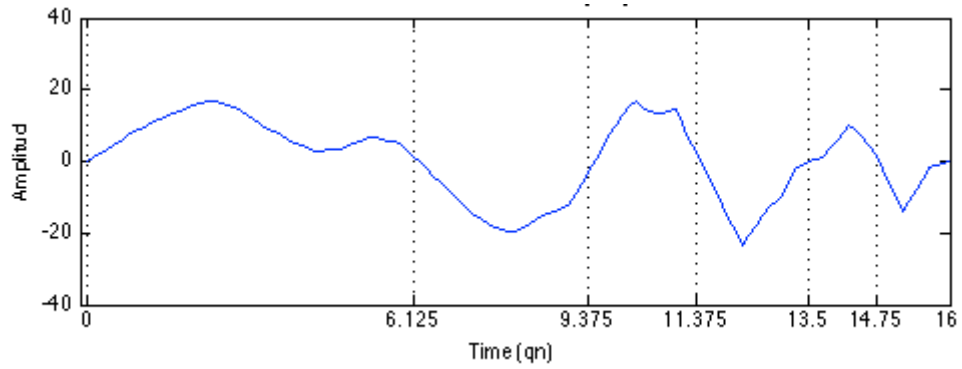


Figure 4. Wavelet coefficient signal at the scale of 4 for the first 16 qns of the sixth *Invention* in E major (BWV 777). Locations of zero-crossings are indicated by dotted vertical lines.

3.3.2 Local maxima segmentation

Local maxima in the wavelet representation occur when the shapes of the melody and the Haar wavelet correlate most. The inner product with the Haar wavelet of length *s* can also be described as the difference of the average pitch during the first half of the wavelet minus the average pitch over the second half of the wavelet times *s*. Local maxima occur, therefore, where there is a locally maximal fall in average pitch content at the scale of the wavelet used. See Figure 5 for illustration.

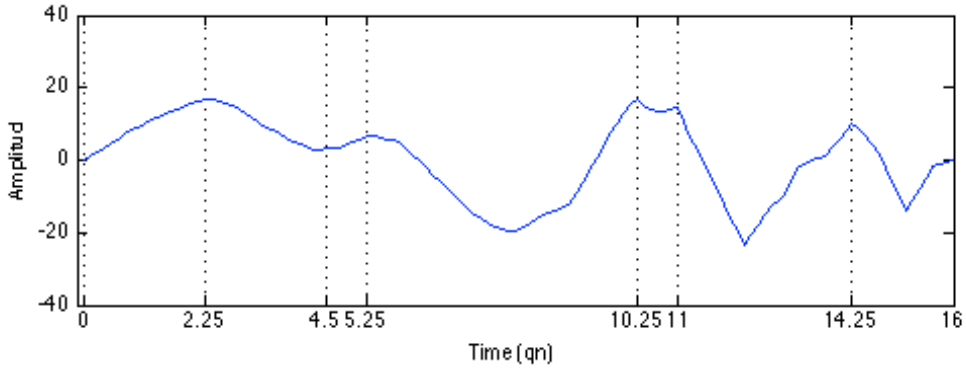


Figure 5. Wavelet coefficient signal at the scale of 4 for the first 16 qns of the sixth *Invention* in E major (BWV 777). Local maxima are indicated by dotted vertical lines.

3.3.3 Segment length normalization

In the evaluation tasks described below, the segments identified need to be classified, for which we introduce similarity measures on segments. We use the Euclidean and city-block distances, which entails that the segments need to be represented as vectors of equal length. However, segments are not generally of the same length when using the segmentation approaches described here. In order to obtain segments of equal length, we use two different procedures: we normalize the length of segments to the maximal segment length, or we define a maximal length for all segments and pad shorter segments as necessary with zeros at the end.

For comparison, we also segment using Eerola and Toiviainen's (2004) implementation of Cambouropoulos' (1997, 2001) LBDM (see above). The LBDM calculates a normalized boundary strength between 0 and 1 for the interval between each pair of consecutive notes in a melody (Cambouropoulos, 2001). In order to generate a specific segmentation, it therefore requires a threshold value between 0 and 1 to be defined.

3.4 Scale selection

In this study, we use the wavelet coefficients at only one scale, as we focus only on a single level of segmentation. By representing melodies by their wavelet coefficients at only one scale, we emphasise information on that time-scale in the signal, as discussed above. Small scales focus on short-term movements, while large scales emphasise the longer-term trend of the melody. We have tested dyadic multiples of quarter notes as scale values and selected those that yield the best classification results.

3.5 Classification

We use the wavelet representation and segmentation to perform classification of melodies with a *k*-Nearest-Neighbour (kNN) classifier. A kNN classifier is defined by a set of labelled items and a distance measure. It then assigns labels to a new item *x* by finding the *k* items that are closest to *x* according to the distance measure and choosing the label that occurs most often among these *k* items.

We use two different distance measures, *city-block* distance and *Euclidean* distance. The Euclidean distance between two segments, *st* and *sc*, is given by

$$d^{E}_{stsc} = \sqrt{\sum_{j=1}^{n} (st[j] - sc[j])^2} \ .$$

The city-block distance is given by

$$d^{C}_{stsc} = \sum_{j=1}^{n} \left| st[j] - sc[j] \right| .$$

3.6 Example

For illustration, Figure 6 presents an example of similarity measurements between a target segment (row 2) from a melody represented as pitch signal (row 1) and four test segments (rows 3 to 6). Test segment 3 has the smallest distance to the target segment when segment length normalization by zero padding is applied. On the other hand, if segment length normalization by interpolation is applied, the segment that has the smallest distance to the target segment is test segment 1.

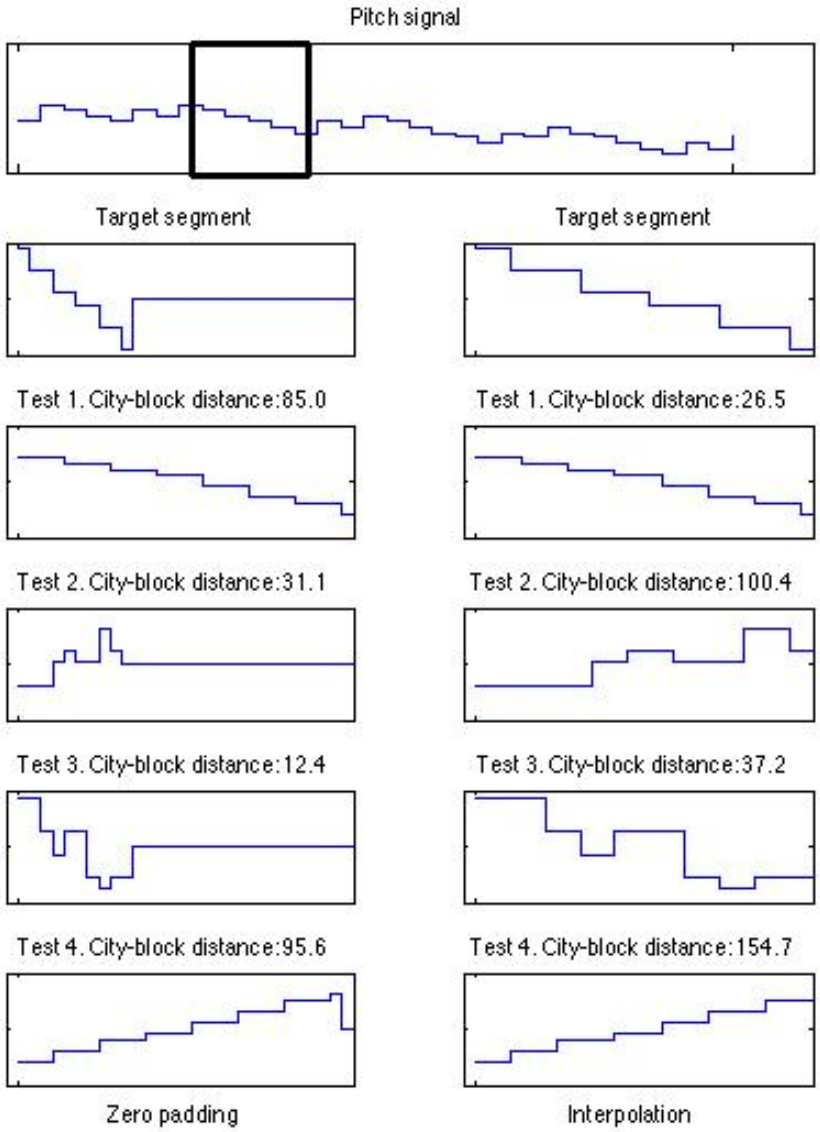


Figure 6. Illustration of a melodic segment (row 1) and similarity measurements between a target segment (row 2) and four test segments (rows 3 to 6). Segment length normalization by zero padding (left column) vs. segment length normalization by interpolation (right column). The black square in row 1 denotes the target segment.

4 Classification experiments

In this section, we present two experiments on different data sets[4]. One experiment is on recognizing the parent works of segments from Bach's *Two-Part Inventions* (BWV 772–786). The second experiment is on recognizing the tune families to which Dutch folk songs belong, using the Dutch Song Database (Grijp 2008; The Meertens Institute, 2012).

4.1 Experiment 1: Classification of segments from J. S. Bach's *Two-Part Inventions*

Music theorists describe J. S. Bach's *Inventions* as being coherently developed from a theme, the subject, that dominates each piece (see, e.g., Dreyfus, 1996). The *Invention's* subject is presented in the exposition, and it is contrapuntally treated across the (usually three) other sections (Stein, 1979). From this point of view, we hypothesize that the parent work of one of the later sections of an *Invention* can be successfully identified by finding the *Invention* with the exposition that the section resembles most closely in terms of melodic segments used.

For the 15 *Two-Part Inventions*, the classifier set C is built from segments $sc_{i,j}$ from the expositions of all *Inventions*, where each segment can stem from either the upper or the lower part. $sc_{i,j}$ is the j^{th} segment in *Invention i*. We define the length of the exposition as 16 qn, which is, of course, not accurate in all cases, but rather corresponds to the longest exposition in order to avoid including exposition material in the test sets possibly however, including material of the following section in the classifier. After the first 16 qn, each invention is divided into 3 sections of equal length to build the test sets. Each test set T is built from segments *st*, where each *st* can stem from either the upper or the lower part. We denote the j^{th} segment in *Invention i* by $st_{i,j.}$ To classify a segment *st* to one of the 15 classes, we apply 1-NN classification. That is, we compute the distances between *st* and all *sc* in C, and classify *st* to the class *i* of the $sc_{i,j}$ that has the smallest distance to *st*. The section is assigned the class most frequently predicted by its segments. In both cases we use the next nearest point to break ties.

[4] The algorithms are implemented in MATLAB (R2012b, The Mathworks, Inc) using the Wavelet Toolbox and the MIDI Toolbox (Eerola & Toiviainen, 2004). We use the LBDM implementation of the MIDI Toolbox, and an update of Christine Smit's read_midi function (http://www.ee.columbia.edu/~csmit/matlab_midi.html, accessed 4 October 2012).

We test the classification accuracy of classifiers built from the first 4, 8 or 16 qn, on three, equally-divided sections after the exposition (see Figure 7), to study the development of the method's performance over the course of the *Invention*. We expect the classification rates to first decrease, reflecting the increasing degrees of variation of the original material and to increase towards the end, where the original material typically returns. We also compare different representations, segmentations and distance measures, as the performance can inform us about the suitability of these measures for representing the motivic coherence that music theorists describe in the *Inventions*.

We also test the effect of including contrapuntal variations in the classifier, because music theorists claim that these techniques are used for variation in the *Inventions* (and generally in imitative styles of music) (see, e.g., Dreyfus, 1996). Specifically, we considered inversion (reflection in a constant-pitch axis), retrograde (reflection in a constant-time axis) and retrograde inversion (rotation through a half turn) (see Figure 8). Contrapuntal variations are added as classes to the kNN classifier and we therefore have 4 times the number of classes.

We compare the wavelet representation with the normalized pitch signal representation, as described above. We evaluate the case when classifier and test sets contain one segment for each part and section, i.e. "*without segmentation*", and the case of applying a segmentation algorithm to create several segments from each part and section, which we call "*with segmentation*". We compare the results of zero crossing wavelet segmentation with two other segmentation methods: segmentation into segments of constant length, as a simplistic baseline segmentation, and segmentation with Cambouropoulos' LBDM as mentioned above. Local maxima wavelet segmentation was not used in this experiment as preliminary tests showed that segmenting at zero crossings produced better results in general for this dataset. Figure 9 shows, as an example, the first 16 qn of the upper voice of the first *Invention* (BWV 772) in the different combinations used for the experiments.

64

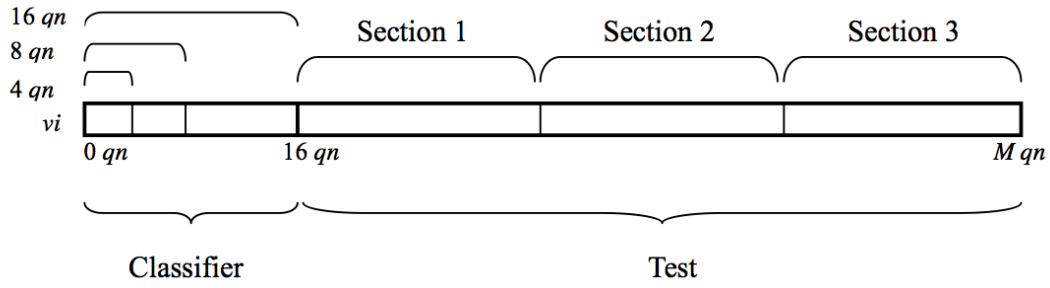


Figure 7. Scheme of classifier and test construction based on signal v_i.

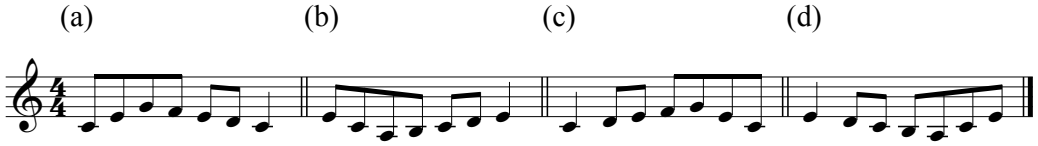


Figure 8. Contrapuntal variations: (a) prime form, (b) inversion, (c) retrograde and (d) retrograde inversion.

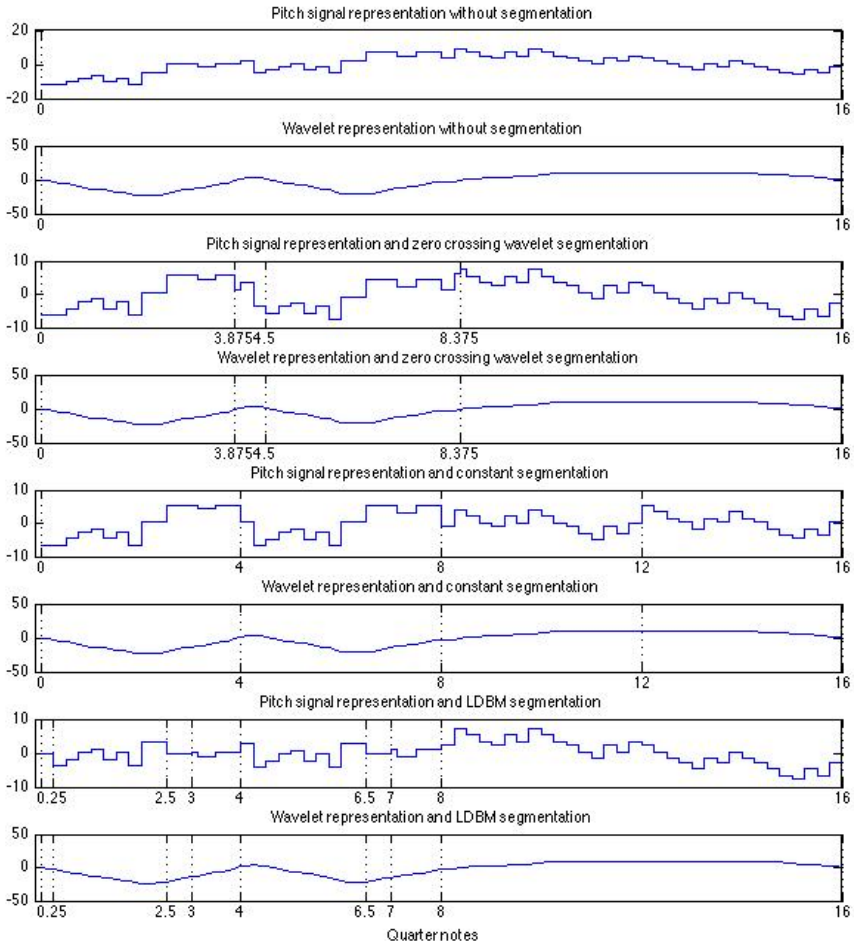


Figure 9. The first 16 qn of the upper voice of *Invention* 1 (BWV 772) in different combinations of representation and segmentation (the segmentation points are shown as vertical dotted lines): Normalized pitch signal representation (odd rows) and wavelet representation at scale of 4 qn (even rows), without segmentation (rows 1 and 2), wavelet segmentation at scale of 4 qn (rows 3 and 4), constant segmentation at 4 qn (rows 5 and 6), and LBDM with a threshold of 0.4 (rows 7 and 8). Pitch signal normalization takes places after segmentation, leading to pitch shifts between the original melody and the segments.

When the segments' lengths are normalized by zero padding, the length of segments is set to the maximal segment length, and shorter segments are padded as necessary with zeros at the end, even if they are segmented by constant length segmentation. In this case the sampling rate is not affected. When the segments' lengths are normalized by interpolation, the lengths of segments are resized to the

maximal segment length by nearest neighbour interpolation (de Boor, 1978). This, of course, changes the sampling rate in most cases.

We used pitch signals initially sampled at 8 samples per quarter note (qn) and varied the following parameters to optimize classification performance:

- two melodic representations: normalized pitch signal representation (vr) and wavelet representation at scale of 1 qn (wr),
- without segmentation and with three segmentation methods: constant segmentation (cs) at 1 and 4 qn, LBDM with thresholds of 0.2 and 0.4 and zero crossing wavelet segmentation (ws) at scale 1 and 4 qn,
- segment length normalization by zero padding and by interpolation and
- Euclidean and city-block distance.

The optimal values of these parameters and the effect of representation, segmentation and contrapuntal variations will be presented in the results section.

4.2 Experiment 2: Classification of Dutch Folk Tunes

Folk tunes are a cultural heritage and interesting to study in the context of melodic classification because:

1) they present variation due to the process of oral transmission between generations;
2) understanding variations can help us understand cultural developments in music; and
3) there is a substantial body of research and data to support experiments and comparisons.

The Meertens Institute in Amsterdam hosts a collection of Dutch folk songs that has been digitized and classified into tune families according to similarity assessments done by experts (van Kranenburg, 2010). The Dutch Song Database we use contains 360 folk songs in 26 tune families, and is a subset of the collection known as “Onder de groene linde” (Grijp, 2008; The Meertens Institute, 2012). Automatic classification methods based on global features and string matching have been extensively tested by van Kranenburg (2010), and he concluded that recurrence of common motives is the most important musical factor in defining tune families.

For the Dutch tune family classification task, we designed two experiments, testing, among other parameters, the effect of segmentation. We use complete melodies or segments of melodies for classification.

4.2.1 Experiment 2-1: Classification without segmentation

In this experiment, we use complete melodies without segmentation. The songs of the Dutch Song Database are sampled to pitch signals of length 2^{10}. We evaluate rest representation[5] and pitch normalization, as described in section 3.1. Moreover, we evaluate melodies as pitch signals or as wavelet coefficients. When melodies are represented as wavelet coefficients, we apply the CWT with Haar wavelet at a single scale. We evaluate classification accuracy with 1NN using city-block and Euclidean distances in leave-one-out cross validation on the corpus of 360 folk songs.

4.2.2 Experiment 2-2: Classification with segmentation

We build the classifier set C from all segments sc_j of the whole corpus minus one—that is 359 labeled songs. The remaining song is used for testing. We use *kNN* classification, where k=1 to 5. We thus compute the distances between a test segment st_j and all segments in C, and assign the segment to the most frequent class of the k segments with the smallest distances and the tune to the most frequent class of its segments. We calculate the classifiers' accuracies using all segments of all songs belonging to a tune family with 1 to 5 nearest neighbours and with two distance measures (Euclidean and city-block) in leave-one-out cross validation on the corpus of 360 folk songs.

In this second experiment with segmentation, we use once again the two types of melodic representations (normalized pitch signal and wavelet coefficients at one scale) but only two segmentation models: LBDM and local maxima of wavelet coefficients. Zero crossings were not used in this experiment as preliminary tests showed that segmenting at local maxima produced better results in general for this dataset[6]. The MIDI files of this collection are initially sampled at 8 samples per qn. We apply the CWT with the Haar wavelet using a dyadic set of 8 scales. Melodies are represented as normalized pitch signals (vr) or as the resulting wavelet coefficients (wr). Signals are segmented by the wavelet coefficients' local maxima (ws), or by the local boundary detection model LBDM using thresholds from 0.1 to 0.8 in steps of 0.1. We explore the parameter space with a grid search, testing all combinations of

[5] We also tested the way that rests are represented in normalized pitch signals by assigning the value zero to rests, subtracting the average pitch (excluding rests) and assigning the value zero to rests again after normalization. This practice produced worse results than the way that rests are represented in the normalized pitch signal representation described in section 3.1.

[6] We ran some tests with segmentation points at local extrema (i.e., local minima and maxima), but, in general, results with local maxima were better.

representations and segmentations: wavelet representation (wr), normalized pitch signal representation (vr), wavelet segmentation (ws), LBDM (LBDM) segmentation. Segment length normalization is done by zero padding and by interpolation.

5 Results and discussion

5.1 Results of experiment 1: Classification of segments from J. S. Bach's *Two-Part Inventions*

5.1.1 Experiment 1-1. Classification without segmentation

Table 1 shows the best accuracies with a corpus of the 15 *Two-Part Inventions* by J. S. Bach (BWV 772–786) without segmentation. The parameters used to achieve the values shown in Table 1 are:

- pitch signals sampled at 8 samples per qn,
- normalized pitch signal representation,
- wavelet representation at the scale of 1 qn,
- 1-nearest neighbour classifier with city-block or Euclidean distance, and
- length normalization by zero padding or by interpolation.

	City-block		Euclidean	
	(wr)	(vr)	(wr)	(vr)
Mean NC	0.1778	0.0889	0.1333	0.0889
Std-Dev. NC	0.0385	0.0770	0.0667	0.1018
Mean CP	0.1333	0.1556	0.0667	0.1333
Std-Dev. CP	0.0667	0.1388	0.0000	0.1155

Table 1. Experiment without segmentation. Summary of the best classification accuracies over three sections of the inventions, mean and standard deviation (Std-Dev.) of the classifiers build from the first 16 qn. Classifier built from the exposition (NC), and the classifier built from the exposition and its contrapuntal variations (CP). Combinations: wavelet representation (wr), normalized pitch signal representation (vr)..Appendix A, Table A3 shows the results of all combinations tested in the experiment.

This approach is a baseline experiment, which does not use segment information or alignment, and the observed accuracies are above chance level (6.66%) but very low as expected.

5.1.2 Experiment 1-2. Classification with segmentation

For this corpus and experiment, segmentation improves the classification rates substantially. Figures 10 and 11 show the classification performance on each section, the effect of segmentation and representation (rows vs. columns), the effect of including contrapuntal techniques (Figure 10 vs. Figure 11) and the number of quarter notes used for the classifiers (red, green and blue lines). The remaining fixed parameter values were chosen such that the best results were achieved in the majority of the cases shown (Appendix A, Tables A1 and A2 summarize the results of all other parameterisations). The used parameter values are:

- normalized pitch signal representation,
- wavelet representation at the scale of 1 qn,
- zero crossing wavelet segmentation at the scale of 1 qn,
- LBDM segmentation at a threshold of 0.2,
- constant segmentation at 1 qn,
- 1-nearest neighbour classifier with city-block distance, and
- segment length normalization by zero padding.

The classification results vary widely, with segmentation method having a stronger effect than representation type. Wavelet segmentation combined with wavelet representation produces the best classification results when using 16 quarter notes of the exposition.

Including contrapuntal variations is clearly detrimental when using wavelet segmentation and to some degree when using LBDM, but improves performance with constant segmentation. This result was unexpected, as a common view in musicology is that inversion, retrograde and retrograde inversion are important principles of variation in J. S. Bach's inventions (e.g. Stein, 1979) and would therefore help in recognising the inventions. However, the lower-than-expected recognition rates achieved with our contrapuntal variation classifier may be due to the fact that we use chromatic pitch representations rather than ones based on diatonic (or "morphetic") pitch (see Meredith, 2006, pp. 126–9).

The classification performance generally decreases from the 1st to the 2nd sections and it rises from the 2nd to the 3rd sections, to some degree conforming to the expectation of increased similarity between the final section and the exposition.

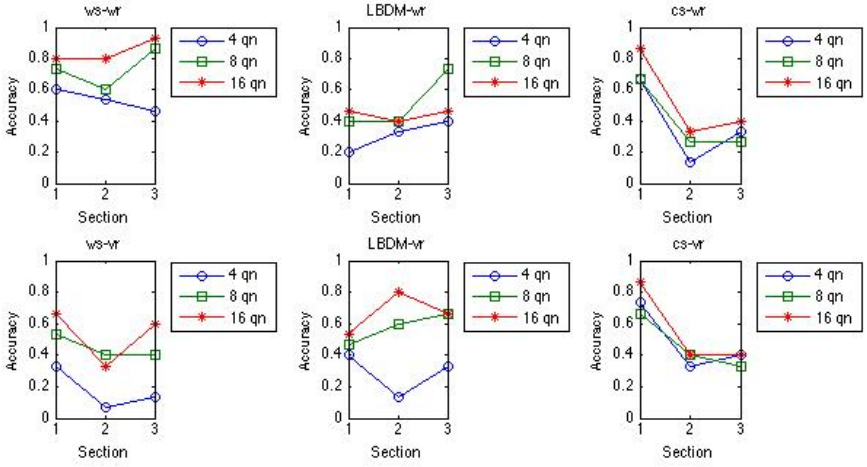


Figure 10. Performance for each section with the classifier based on the exposition.

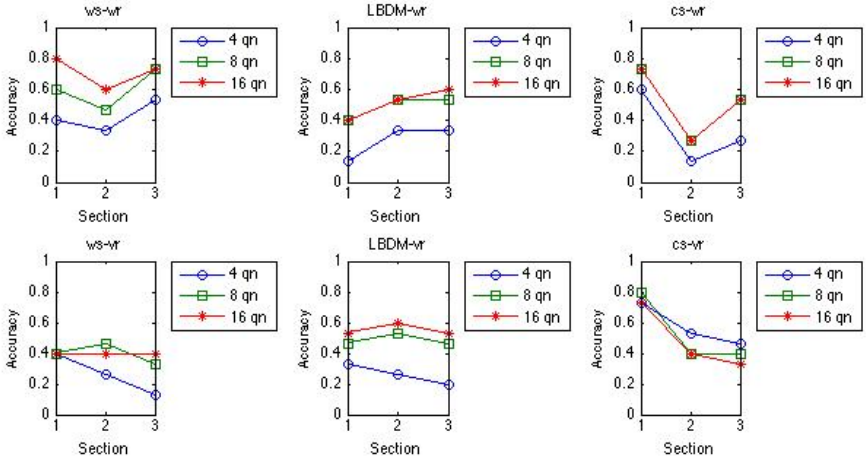


Figure 11. Performance for each section with the classifier based on the exposition and its contrapuntal variations.

5.2 Results of experiment 2: Classification of Dutch Folk Tunes

5.2.1 Classification without segmentation

Table 2.1 shows the classification rates obtained in the experiment on the corpus of 360 Dutch Folk songs without segmentation, using complete melodies. The parameter values are:

- pitch signals of length 2^{10},
- normalized pitch signal representation,
- wavelet representations at a single scale and
- classification in leave-one-out cross validation with 1 nearest neighbours using Euclidean and city-block distances.

	City-block	Euclidean	City-block	Euclidean
	Rests removed		Rests represented by zeros	
(vr)	0.8806	0.8694	0.7944	0.7056
(wr)	0.8556	0.8306	0.7472	0.7222

Table 2. Classification accuracy observed for different methods. Pitch signal representation (vr) and wavelet representation (wr) combined with different distance measures and rest treatment.

For this experiment, removing rests from the representation produced better classification accuracies. We therefore removed rests from the representation for the experiment with segmentation. The use of complete melodies represented as pitch signals without filtering produces the best results.

5.2.2 Classification with segmentation

Contrary to the effect seen in experiment 1, segmentation did not produce a significant change in the classification rates, even varying several parameters. Figure 12 shows the classification rates obtained with segmentation, where brighter colours indicate higher rates. The parameter values are:

- pitch signals initially sampled at 8 samples per qn,
- normalized pitch signal representation,
- wavelet representations using a dyadic set of 8 scales,
- local maxima wavelet segmentation using a dyadic set of 8 scales,
- LBDM segmentation using thresholds from 0.1 to 0.8 in steps of 0.1,
- classification with 1 to 5 nearest neighbours using city-block distances, and
- segment length normalization by zero padding.

Table 3 summarizes the best and worst classification rates with the parameters mentioned above. The effect of using segment length normalization by interpolation produces slightly lower results than segment length normalization by zero padding (see Table 4).

The results show that wavelet filtering of the melodic segments can improve classification performance compared to using the pitch signal directly. When segmentation is used, wavelet representation proves to be more discriminative than pitch signals independently of the segmentation method. The classification performance varies, obtaining best results at small representation scales and poor results at large scales, with the exception of the largest scale, which recovers its performance to some extent (see Figure 12).

In terms of segmentation, we observe that shorter segments produce better results when used with wavelet representation. This is contrary to the results of the LBDM applied to pitch signals, where shorter segments produce worse results than larger ones. We observe an improvement towards threshold 0.4 and a gradual improvement towards the threshold of 0.8, which corresponds to larger segments, meaning that using the complete melodic sequences or a combination of complete melodies and melodic segments can lead to better classification results. Indeed, as shown in the first part of this second experiment using the Dutch Song Database, the classification rates improve when using complete melodies represented as pitch signals.

In general, the city-block distance performs slightly better than Euclidean distance and the wavelet representation works better than the normalized pitch signal representation. In addition, we studied the effect of using more than one nearest neighbour. It can be observed that using one and two nearest neighbours produced the best results. Different effects are seen when using values greater than 2 for *k* in the *k*NN, but in general the performance decreases as *k* increases.

The best classification rates are achieved by using the wavelet representation and segmentation using 1 or 2 nearest neighbours at small scales. This suggests that the melodies in this corpus contain typically several similar segments that are typical for that family. This agrees with van Kranenburg's (2010) claim that recurrent motives are important for determining the family of a folk song in the Dutch Song Database. On the other hand, the results of van Kranenburg *et al*. (2013) using string-matching are considerably better, suggesting that information on the order of the segments also plays an important role.

City-block distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8556**	**0.8556**	0.8333	0.8306	0.7972
	worst	0.4833	0.4833	0.4639	0.45	0.4167
wr-LBDM	best	0.8417	0.8417	0.8083	0.8028	0.7778
	worst	0.4417	0.4417	0.4556	0.4417	0.4139
vr-ws	best	0.8139	0.8139	0.7972	0.7778	0.7472
	worst	0.5194	0.5194	0.5194	0.5139	0.5583
vr-LBDM	best	0.7889	0.7889	0.7778	0.75	0.725
	worst	0.4139	0.4139	0.3861	0.3778	0.3806

Euclidean distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8417**	**0.8417**	0.8306	0.8194	0.7917
	worst	0.4667	0.4667	0.4583	0.4333	0.4167
wr-LBDM	best	0.8111	0.8111	0.8083	0.7889	0.7694
	worst	0.4472	0.4472	0.4528	0.4333	0.4139
vr-ws	best	0.8083	0.8083	0.7806	0.7667	0.7444
	worst	0.5194	0.5194	0.5333	0.525	0.5639
vr-LBDM	best	0.7778	0.7778	0.7444	0.7333	0.7083
	worst	0.4111	0.4111	0.3722	0.3806	0.3806

Table 3. Summary of the accuracies for the combinations: wavelet representation and wavelet segmentation (wr-ws), wavelet representation and local boundary detection model (wr-LBDM), pitch signal representation and wavelet segmentation (vr-ws), pitch signal representation and local boundary detection model (vr-LBDM), segment length normalization by zero padding.

Euclidean distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8361**	**0.8361**	0.8194	0.7944	0.7722
	worst	0.4306	0.4306	0.3944	0.3889	0.3722
wr-LBDM	best	0.8056	0.8056	0.7972	0.7611	0.7389
	worst	0.3611	0.3611	0.3528	0.3306	0.3083
vr-ws	best	0.7833	0.7833	0.7694	0.7806	0.7556
	worst	0.5111	0.5111	0.5444	0.5167	0.5000
vr-LBDM	best	0.7833	0.7833	0.7639	0.7667	0.7500
	worst	0.3917	0.3917	0.3667	0.3639	0.3611

City-block distance						
represent.-segment.	Value	Nearest Neighbours				
		1	2	3	4	5
wr-ws	best	**0.8472**	**0.8472**	0.8333	0.8111	0.7944
	worst	0.4306	0.4306	0.4222	0.4000	0.3556
wr-LBDM	best	0.8139	0.8139	0.7917	0.7806	0.7528
	worst	0.3306	0.3306	0.3333	0.3083	0.3083
vr-ws	best	0.7944	0.7944	0.7861	0.7917	0.7583
	worst	0.5083	0.5083	0.5389	0.5306	0.5583
vr-LBDM	best	0.8028	0.8028	0.7833	0.7806	0.7528
	worst	0.4028	0.4028	0.3722	0.3639	0.3694

Table 4. Summary of the accuracies for the combinations: wavelet representation and wavelet segmentation (wr-ws), wavelet representation and local boundary detection model (wr-LBDM), pitch signal representation and wavelet segmentation (vr-ws), pitch signal representation and local boundary detection model (vr-LBDM), segment length normalization by interpolation.

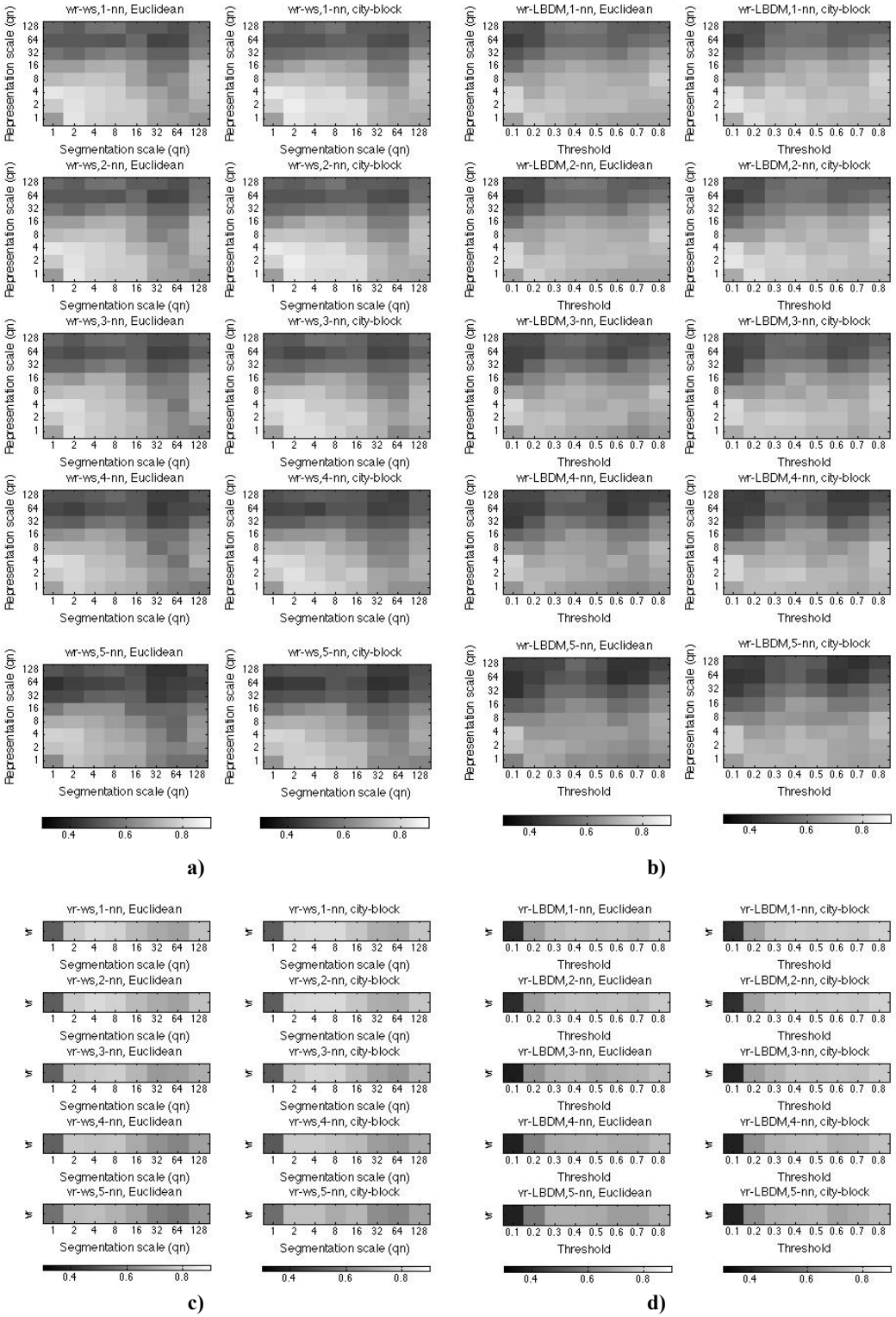


Figure 12. Accuracies for the combinations: a) wavelet representation (wr) and wavelet segmentation (ws), b) wavelet representation (wr) and local boundary detection model (LBDM), c) pitch signal representation (vr) and wavelet segmentation (ws), pitch signal representation (vr) and local boundary detection model (LBDM), segment length normalization by zero padding.

5.3 Discussion

We have presented two experiments, in which continuous Haar-wavelet filtering was applied in two musicologically motivated classification tasks. The results of the first experiment support the view that there are strong, intra-opus, motivic relations within Bach's *Two-Part Inventions* that allow for the parent works of sections from these pieces to be identified, depending on the amount of material used from the exposition, along with the approaches used to segment and represent the music. The negative effect of adding contrapuntal variations in the classifiers in connection with wavelet segmentation is interesting and may suggest that the similarities captured by wavelets are different to and in some way incompatible with contrapuntal variations we have used in the experiment. On the other hand, this effect could also be an artefact of the specific type of pitch representation used—we intend to explore this further in future work.

When the wavelet-based approach was used to identify the tune families of songs in a database of Dutch folk songs, it proved to work slightly better than using the LBDM with direct melody comparison and slightly worse than using complete melodies without filtering. However, results with string-matching methods reported by van Kranenburg *et al*. (2013) are considerably better. This indicates that the overall sequential structure of the melody is relevant for this task, which is ignored in the segmentation approach. This is supported by the observation that the wavelet-based classifier performs similarly at small and large scales, with different k values and for different distance metrics, indicating that the relevant information may not be just in the segments.

Segment length normalization by zero padding produces slightly better results than normalization by interpolation. This suggests that the structure of segments is related to their length and the effect of zero padding does not negatively influence the reliability of similarity measurement.

Melodic segmentation has a different effect between the two experiments possibly due to musical differences between the Dutch folk tunes and Bach's *Inventions* or due to different principles determining whether two tunes should be in the same tune family or whether two melodic excerpts belong in the same piece.

6 Conclusion

In this paper, we have presented a method for using wavelets to represent and segment melodies for classification and we have evaluated it on two different musicological classification tasks. Our main contribution has been to introduce and demonstrate the potential of a novel, wavelet-based approach to modelling melodic structure.

The results of the experiments reported here suggest that a method employing a wavelet-based approach to representing and segmenting the data can out-perform one that uses a direct pitch-time representation and Gestalt-based or constant-duration segmentation in the task of predicting which work in a collection contains a given query segment. When the task was to identify the musicologically defined tune family to which a given folk song belongs, our wavelet-based approach worked only slightly better than one based on Gestalt principles and slightly worse than one without segmentation using pitch melodies. However, it was clearly out-performed by string-matching methods, which is probably due to the fact that, in this task, the overall structure of the compared melodies contains relevant information that our classification method is not using, regardless of whether or not wavelets are used.

We propose that the positive results of wavelet representation and segmentation can be understood by viewing the wavelets in terms of the pitch trend over the scale duration. Focusing on an appropriate time-scale, giving less weight to short-term movement as well as the average pitch (i.e., transposition), can make relevant parts of the melodic contour more prominent in the distance measure.

7 Future work

There are several further aspects of modelling melodic perception with wavelets that have not been explored in this study, including the problem of automatic scale determination, and the relation between musical style and features in wavelet coefficient representations.

Understanding the wavelet analysis better in terms of musical properties may help improve the results for melodic similarity. Multiple scales could be used for hierarchical segmentation. Using a selective combination of scales and exploring metrical information derived from songs' periodicities could be used to develop a method for scale selection. Applying machine learning to develop more complex

wavelet-based feature extraction from melodies could also be a very interesting way to use the wavelet representation on symbolic music data.

We also aim to identify the cognitive mechanisms that underlie the effectiveness of the wavelet-filtering approach and to explain why coefficient zero-crossings work better in some classification tasks while coefficient local maxima work better in others.

We generally aim in future research to gain a deeper understanding of the musical meaning and perceptual relevance of wavelet-based music representation and segmentation.

References

Andén, J., & Mallat, S. (2011). Multiscale scattering for audio classification. In: *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011),* Utrecht, NL.: ISMIR, pp. 657–662. Available online at http://ismir2011.ismir.net/papers/PS6-1.pdf

Antoine, J.-P. (1999). Wavelet analysis: A new tool in physics. In: J.C. van den Berg (Ed.), *Wavelets in Physics*. Cambridge: Cambridge University Press, pp. 9–22.

Berger, J., Coifman, R., & Goldberg, M. (1994). Removing noise from music using local trigonometric bases and wavelet packets. *J. Audio Eng. Soc*. Vol. 42, No. 10, pp. 808–818.

Brown, M. (2005). *Explaining Tonality: Schenkerian Theory and Beyond*. Rochester, NY.: University of Rochester Press.

Cambouropoulos, E. (1997). Musical rhythm: A formal model for determining local boundaries, accents and metre in a melodic surface. In: M. Leman (Ed.), *Music, Gestalt and Computing: Studies in Cognitive and Systematic Musicology*. Berlin: Springer, pp. 277–293.

Cambouropoulos, E. (2001). The local boundary detection model (LBDM) and its application in the study of expressive timing. In: *Proceedings of the International Computer Music Conference*. San Francisco, CA: ICMA, pp. 17–22.

Daubechies, I. (1996). Where do wavelets come from? A personal point of view. In: *Proceedings of the IEEE*, Vol. 84, No. 4, pp. 510–513.

Daubechies, I., & Maes, S. (1996). A nonlinear squeezing of the continuous wavelet transform based on auditory nerve models. In: A. Aldroubi and M. Unser (Eds.), *Wavelets in Medicine and Biology*. Boca Raton, FL.: CRC Press, pp. 527–546.

de Boor, C. (1978). *A practical guide to splines*. Springer-Verlag.

Dobson, K., Yang, J., Whitney, N., Smart, K., & Rigstaa, P. (1996). A low complexity wavelet based audio compression method. In: *Proceedings of the Data Compression Conference (DCC '96)*.

Dreyfus, L. (1996). *Bach and the Patterns of Invention*. Cambridge, MA.: Harvard University Press.

Eerola, T., & Toiviainen, P. (2004). *MIDI Toolbox: MATLAB Tools for Music Research*. University of Jyväskylä. Available at http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/miditoolbox/.

Farge, M. (1992). Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics*, Vol. 24. pp. 395–457.

Forte, A., & Gilbert, S. (1982). *Introduction to Schenkerian Analysis*. New York, NY.: Norton.

Grijp, L.P.. (2008). Introduction. In: L.P. Grijp & I. van Beersum (Eds.), *Onder de groene linde. 163 verhalende liederen uit de mondelinge overlevering, opgenomen door Ate Doornbosch e.a./Under the green linden. 163 Dutch Ballads from the oral tradition recorded by Ate Doornbosch a.o. (Boek + 9 cd's + 1 dvd)*. Amsterdam/Hilversum: Meertens Instituut & Music and Words. pp. 18–27.

Grimaldi, M., Cunningham, P., & Kokaram, A. (2003). A wavelet packet representation of audio signals for music genre classification using different ensemble and feature selection techniques. In: *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR '03)*. pp. 102–108.

Haar, A. (1910). Zur Theorie der orthogonalen Funktionensyteme. *Mathematische Annalen*, Vol. 69, pp. 331–371.

Hillewaere, R., Manderick, B., & Conklin, D. (2009). Global feature versus event models for folk song classification. In: *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, Kobe, Japan: ISMIR. pp. 729-733. Available online at http://ismir2009.ismir.net/proceedings/OS9-1.pdf

Hillewaere, R., Manderick, B., & Conklin, D. (2012). String methods for folk music classification. In: *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR 2012)*. Porto, Portugal. pp. 217–222. Available online at http://ismir2012.ismir.net/event/papers/217-ismir-2012.pdf

Huron, D. (1996). The melodic arc in western folk songs. *Computing in Musicology*, Vol. 10, pp. 3–23.

Huron, D. (2006). *Sweet Anticipation: Music and the Psychology of Expectation*. Cambridge, MA: MIT Press.

Jeon, W., Ma, C., & Ming Cheng, Y. (2009). An efficient signal-matching approach to melody indexing and search using continuous pitch contours and wavelets. In: *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, Kobe, Japan. pp. 681–686. Available online at http://ismir2009.ismir.net/proceedings/PS4-18.pdf

Jeon, W., & Ma, C. (2011). Efficient search of music pitch contours using wavelet transforms and segmented dynamic time warping. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011),* pp. 2304–2307.

Karmakar, A., Kumar, A., & Patney, R. (2011). Synthesis of an optimal wavelet based on auditory perception criterion. *EURASIP Journal on Advances in Signal Processing*, Vol. 2011, Article ID 170927.

Kay, K.N., Naselaris, T., Prenger, R.J., & Gallant, J.L. (2008). Identifying natural images from human brain activity. *Nature*, Vol. 452, 20 March 2008, doi:10.1038/nature06713. pp. 352–356.

Knopke, I., & Jürgensen, K. (2009). A system for identifying common melodic phrases in the masses of Palestrina. *Journal of New Music Research*, Vol. 38, No. 2, pp. 171–181.

Kurby, C.A., & Zacks, J.M. (2008). Segmentation in the perception and memory of events. *Trends in Cognitive Sciences*, Vol.12, No. 2, pp. 72-79.

Lerdahl, F., & Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. Cambridge, MA.: MIT Press.

Levitin, D.J. (2006). *This is Your Brain on Music: The Science of a Human Obsession*. New York: Penguin.

Mallat, S. (2009). *A Wavelet Tour of Signal Processing: The Sparse Way*. 3rd edition. Burlington, MA.: Academic Press.

The Meertens Institute. (2012). Dutch Song Database. http://www.liederenbank.nl/index.php?lan=en

Meredith, D. (2006). The *ps13* pitch spelling algorithm. *Journal of New Music Research*, Vol. 35, No. 2, pp. 121–159.

Nixon, M.S., & Aguado, A.S. (2012). *Feature Extraction and Image Processing for Computer Vision*. 3rd edition. Kidlington: Academic Press.

Pinto, A. (2009). Indexing melodic sequences via wavelet transform. In: *Proceedings of the IEEE International Conference on Multimedia and Expo* (ICME'09). pp. 882–885.

Ponce de Léon, P.J., & Iñesta, J.M. (2004). Statistical description models for melody analysis and characterization. In: *Proceedings of the International Computer Music Conference (ICMC 2004)*, Miami, FL.: ICMA. pp. 149–156.

Schmuckler, M.A. (1999). Testing models of melodic contour similarity. *Music Perception*, Vol. 16, No. 3, pp. 295–326.

Schenker, H, (1935). *Der freie Satz*. Trans. By E. Oster as: *Free Composition*. New York, NY.: Schirmer Books, 1979.

Sinaga, F., Gunawan, T.S., & Ambikairajah, E. (2003). Wavelet packet based audio coding using temporal masking. *The Fourth International Conference on Information, Communications and Signal Processing and Pacific-Rim Conference on Multimedia (ICICS-PCM '03)*, Singapore, Vol. 3, pp. 1380–1383.

Smith, L.M., & Honing, H. (2008). Time-frequency representation of musical rhythm by continuous wavelets. *Journal of Mathematics and Music*, Vol. 2, No. 2, pp. 81–97.

Srinivasan, P., & Jamieson, L. (1998). High quality audio compression using an adaptive wavelet packet decomposition and psychoacoustic modelling. *IEEE Transactions on Signal Processing*, Vol. 46. Issue: 4. pp. 1085–1093, 1998

Stein, L. (1979). *Structure and Style: The Study and Analysis of Musical Forms*. Expanded edition. Miami, FL.: Summy-Birchard Music.

Tenney, J., & Polansky, L. (1980). Temporal gestalt perception in music. *Journal of Music Theory*. Vol 24, No. 2, pp. 205–241.

Torrence, C., & Compo, G.P. (1998). A practical guide to wavelet analysis. *Bulletin of the American Meteorological Society*, Vol. 79, No. 1, pp. 61–78.

Trainor, L. J., & Zatorre, R. J. (2009). The neurobiological basis of musical expectation. In: S. Hallam, I. Cross, & M. Thaut (Eds.), *The Oxford Handbook of Music Psychology*. Oxford: Oxford University Press, pp. 171–183.

Tsunoo, E., Ono, N., & Sagayama, S. (2009). Musical bass-line pattern clustering and its application to audio genre classification. In: *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, Kobe, Japan, pp. 219–224. Available online at http://ismir2009.ismir.net/proceedings/PS2-5.pdf.

Tzanetakis, G., Essl, G., & Cook, P. (2001). Audio analysis using the discrete wavelet transform. In: *Proc. WSES Int. Conf. Acoustics and Music: Theory and Applications (AMTA 2001)*, Skiathos, Greece. Available online at http://webhome.cs.uvic.ca/~gtzan/work/pubs/amta01gtzan.pdf.

van Kranenburg, P. (2010). A Computational Approach to Content-Based Retrieval of Folk Song Melodies. PhD thesis, Meertens Institute, Royal Netherlands Academy of Arts and Sciences (KNAW), NL. Full text available at http://depot.knaw.nl/8400

van Kranenburg, P., Volk, A., & Wiering, F. (2013): A Comparison between Global and Local Features for Computational Classification of Folk Song Melodies, *Journal of New Music Research*, 42:1, 1-18

Woody, N. A. & Brown, S. D. (2007) Selecting wavelet transform scales for multivariate classification. *J. Chemometrics*, 21: 357–363. doi: 10.1002/cem.1060

Yu, G., Mallat, S., & Bacry, E. (2008). Audio denoising by time-frequency block thresholding. *IEEE Transactions on Signal processing,* Vol. 56, No. 5, pp. 1830–1839.

Zhang, H., Zhang, B., Huang, W., & Tian, Q. (2005). Gabor wavelet associative memory for face recognition. *IEEE Transactions on Neural Networks*, Vol. 16, No. 1, pp. 275–278.

Zhang, W., Shan, S., Qing, L., Chen, X., & Gao, W. (2009). Are Gabor phases really useless for face recognition? *Pattern Analysis Applications*, Vol. 12, No. 3, pp. 301–307.

Appendix A

			Wavelet rep. (wr)			Pitch signal rep. (vr)		
			(ws)	(LBDM)	(cs)	(ws)	(LBDM)	(cs)
A	Rests represented	Mean NC	0.8444	0.4444	0.5333	0.5333	0.6667	0.5556
		Std-Dev. NC	0.0770	0.0385	0.2906	0.1764	0.1333	0.2694
		Mean CP	0.7111	0.5111	0.5111	0.4000	0.5556	0.4889
		Std-Dev. CP	0.1018	0.1018	0.2341	0.0000	0.0385	0.2143
	Rests removed	Mean NC	0.7556	0.6000	0.5333	0.4000	0.6222	0.4889
		Std-Dev. NC	0.0770	0.0667	0.2309	0.1333	0.0770	0.2776
		Mean CP	0.5778	0.6000	0.5556	0.2667	0.4889	0.4000
		Std-Dev. CP	0.2037	0.0667	0.2694	0.1155	0.0385	0.2309
B	Rests represented	Mean NC	0.4889	0.4889	0.3556	0.5556	0.3556	0.3556
		Std-Dev. NC	0.1018	0.0770	0.2694	0.0385	0.1018	0.2776
		Mean CP	0.3556	0.4000	0.3556	0.4222	0.3333	0.4000
		Std-Dev. CP	0.1018	0.0667	0.2694	0.1018	0.0667	0.2000
	Rests removed	Mean NC	0.4000	0.6000	0.3556	0.3111	0.3778	0.4222
		Std-Dev. NC	0.0000	0.0000	0.3289	0.0770	0.1018	0.3289
		Mean CP	0.4000	0.5111	0.3778	0.3778	0.4000	0.3778
		Std-Dev. CP	0.0667	0.1678	0.3791	0.1388	0.1155	0.1925
LN-Zero padding								
			Wavelet rep. (wr)			Pitch signal rep. (vr)		
			(ws)	(LBDM)	(cs)	(ws)	(LBDM)	(cs)
A	Rests represented	Mean NC	0.8000	0.4444	0.5778	0.6444	0.4444	0.5778
		Std-Dev. NC	0.0667	0.0385	0.3151	0.0385	0.1678	0.3079
		Mean CP	0.6889	0.3778	0.5778	0.3556	0.4667	0.5111
		Std-Dev. CP	0.0385	0.0385	0.2524	0.1018	0.0667	0.2694
	Rests removed	Mean NC	0.8000	0.5333	0.5556	0.3556	0.4667	0.4667
		Std-Dev. NC	0.0667	0.0000	0.3421	0.1678	0.1333	0.2906
		Mean CP	0.6444	0.4444	0.5778	0.2222	0.4667	0.4000
		Std-Dev. CP	0.1018	0.0385	0.2694	0.1388	0.0667	0.2906
B	Rests represented	Mean NC	0.3556	0.1778	0.3556	0.4444	0.1778	0.3778
		Std-Dev. NC	0.1018	0.1388	0.2694	0.0770	0.1540	0.2037
		Mean CP	0.3778	0.2000	0.3778	0.4889	0.1778	0.4222

		Std-Dev. CP	0.0385	0.1333	0.1925	0.0385	0.0385	0.2037
	Rests removed	Mean NC	0.4000	0.2667	0.3333	0.2889	0.2222	0.4000
		Std-Dev. NC	0.1333	0.1764	0.2906	0.1018	0.1018	0.2000
		Mean CP	0.3111	0.1778	0.4222	0.3111	0.2222	0.3333
		Std-Dev. CP	0.1388	0.1018	0.2694	0.1018	0.1018	0.2309
LN-Interpolation								

Table A1. Classification accuracies over three sections of the inventions, mean and standard deviation (Std-Dev.) values of the classifiers build from the first 16 qn, using only the exposition (NC) and the exposition and its contrapuntal variations (CP) for wavelet representation at the scale of 1 qn and normalized pitch signal representation using city-block distance and 1NN. (A) corresponds to segmentation: (ws) at 1 qn, (LBDM) at threshold 0.2 and (cs) at 1 qn. (B) corresponds to segmentation (ws) at 4 qn, (LBDM) at threshold 0.4 and (cs) at 4 qn.

			Wavelet rep. (wr)			Pitch signal rep. (vr)		
			(ws)	(LBDM)	(cs)	(ws)	(LBDM)	(cs)
A	Rests represented	Mean NC	0.8667	0.4667	0.5333	0.5111	0.6222	0.6444
		Std-Dev. NC	0.0667	0.1333	0.2906	0.1678	0.0385	0.2524
		Mean CP	0.7111	0.4444	0.5111	0.3778	0.6000	0.5333
		Std-Dev. CP	0.0385	0.0770	0.1678	0.0385	0.0000	0.2404
	Rests removed	Mean NC	0.7333	0.5333	0.5111	0.4000	0.6000	0.5556
		Std-Dev. NC	0.0667	0.0000	0.2524	0.1333	0.0667	0.2341
		Mean CP	0.6000	0.5778	0.5778	0.2444	0.4444	0.4444
		Std-Dev. CP	0.2000	0.0385	0.2341	0.1018	0.0385	0.2524
B	Rests represented	Mean NC	0.4667	0.4444	0.3333	0.3556	0.3778	0.4222
		Std-Dev. NC	0.1155	0.1388	0.2309	0.1018	0.1018	0.2341
		Mean CP	0.3778	0.3333	0.4444	0.3778	0.3556	0.4222
		Std-Dev. CP	0.1678	0.0000	0.3079	0.1678	0.0385	0.2143
	Rests removed	Mean NC	0.3778	0.4222	0.3333	0.3333	0.4000	0.3556
		Std-Dev. NC	0.1018	0.1388	0.3528	0.2000	0.1333	0.3289
		Mean CP	0.3778	0.4000	0.3333	0.2667	0.3778	0.3556
		Std-Dev. CP	0.1388	0.1155	0.3528	0.1764	0.0770	0.2694
LN-Zero padding								
			Wavelet rep. (wr)			Pitch signal rep. (vr)		
			(ws)	(LBDM)	(cs)	(ws)	(LBDM)	(cs)
A	Rests represented	Mean NC	0.7778	0.4667	0.4889	0.6000	0.4444	0.6222
		Std-Dev. NC	0.1018	0.0667	0.3289	0.0667	0.1018	0.2776
		Mean CP	0.7111	0.3778	0.5778	0.3556	0.4444	0.4889
		Std-Dev. CP	0.0385	0.0385	0.2037	0.0770	0.0770	0.2143
	Rests removed	Mean NC	0.8000	0.4667	0.4889	0.3778	0.4667	0.5778
		Std-Dev. NC	0.0667	0.1155	0.3421	0.1018	0.1333	0.3079
		Mean CP	0.6667	0.4000	0.5333	0.2222	0.4000	0.4000
		Std-Dev. CP	0.1155	0.0667	0.3055	0.1388	0.1155	0.2906

		Mean NC	0.3333	0.2222	0.4667	0.4222	0.2222	0.3556
		Std-Dev. NC	0.0667	0.1018	0.2667	0.0385	0.0770	0.2143
		Mean CP	0.3333	0.2889	0.4222	0.3778	0.2000	0.4444
	Rests represented	Std-Dev. CP	0.0000	0.1540	0.2776	0.0770	0.0667	0.1388
		Mean NC	0.3778	0.2222	0.3556	0.3111	0.3111	0.3333
		Std-Dev. NC	0.1018	0.1678	0.3906	0.1388	0.0770	0.2404
		Mean CP	0.3333	0.1778	0.2889	0.2000	0.2667	0.3778
	Rests removed	Std-Dev. CP	0.0667	0.0770	0.3289	0.1333	0.1764	0.2037
B	LN-Interpolation							

Table A2. Classification accuracies over three sections of the inventions, mean and standard deviation (Std-Dev.) values of the classifiers build from the first 16 qn, using only the exposition (NC) and the exposition and its contrapuntal variations (CP) for wavelet representation at the scale of 1 qn and normalized pitch signal representation using Euclidean distance and 1NN. (A) corresponds to segmentation: (ws) at 1 qn, (LBDM) at threshold 0.2 and (cs) at 1 qn. (B) corresponds to segmentation (ws) at 4 qn, (LBDM) at threshold 0.4 and (cs) at 4 qn.

			City-block		Euclidean	
			(wr)	(vr)	(wr)	(vr)
		Mean NC	0.1778	0.0889	0.1333	0.0889
		Std-Dev. NC	0.0385	0.0770	0.0667	0.1018
	Rests represented	Mean CP	0.1333	0.1556	0.0667	0.1333
		Std-Dev. CP	0.0667	0.1388	0.0000	0.1155
		Mean NC	0.1333	0.0667	0.0667	0.1111
		Std-Dev. NC	0.1155	0.0000	0.0000	0.0385
		Mean CP	0.1556	0.1111	0.1333	0.0667
P	Rests removed	Std-Dev. CP	0.0385	0.1388	0.0000	0.0667
		Mean NC	0.0667	0.0889	0.0889	0.0444
		Std-Dev. NC	0.0667	0.0385	0.0770	0.0385
	Rests represented	Mean CP	0.0889	0.1333	0.1111	0.0667
		Std-Dev. CP	0.0770	0.0667	0.1018	0.0667
		Mean NC	0.0222	0.0667	0.0222	0.0667
		Std-Dev. NC	0.0385	0.0000	0.0385	0.0000
		Mean CP	0.0222	0.1556	0.0444	0.0889
I	Rests removed	Std-Dev. CP	0.0385	0.1018	0.0385	0.0385

Table A3. Classification accuracies without segmentation over three sections of the inventions, mean and standard deviation (Std-Dev.) values of the classifiers build from the first 16 qn, using only the exposition (NC) and the exposition and its contrapuntal variations (CP) for wavelet representation at the scale of 1 qn (wr) and normalized pitch signal representation (vr) using city-block and Euclidean distances and 1nn. (P) corresponds to length normalization by zero padding. (I) corresponds to interpolation length normalization by interpolation.

Paper III. A wavelet-based approach to the discovery of themes and sections in monophonic melodies.

A wavelet-based approach to the discovery of themes and sections in monophonic melodies

Velarde, Gissel; Meredith, David

A WAVELET-BASED APPROACH TO THE DISCOVERY OF THEMES AND SECTIONS IN MONOPHONIC MELODIES

Gissel Velarde
Aalborg University
gv@create.aau.dk

David Meredith
Aalborg University
dave@create.aau.dk

ABSTRACT

We present the computational method submitted to the MIREX 2014 Discovery of Repeated Themes & Sections task, and the results on the monophonic version of the JKU Patterns Development Database. In the context of pattern discovery in monophonic music, the idea behind our method is that, with a good melodic structure in terms of segments, it should be possible to gather similar segments into clusters and rank their salience within the piece. We present an approach to this problem and how we address it. In general terms, we represent melodies either as raw 1D pitch signals or as these signals filtered with the continuous wavelet transform (CWT) using the Haar wavelet. We then segment the signal either into constant duration segments or at the resulting coefficients' modulus local maxima. Segments are concatenated based on their contiguous city-block distance. The concatenated segments are compared using city-block distance and clustered using an agglomerative hierarchical cluster tree. Finally, clusters are ranked according the sum of the length of segments' occurrences. We present the results of our method on the JKU Patterns Development Database.

1. INTRODUCTION

We present the computational method[1] submitted to the MIREX 2014 Discovery of Repeated Themes & Sections task, and the results on the monophonic version of the JKU Patterns Development Database[2]. In the context of pattern discovery of monophonic pieces, the idea behind our method is that, with a good melodic structure in terms of segments, it should be possible to gather together similar segments to rank their salience within the piece (See 'paradigmatic analysis' [3]). We also consider other aspects of the problem, in particular, representation, segmentation, measuring similarity, clustering of segments and ranking segments according to salience.

[1]The algorithm is implemented in MATLAB (R2013b, The Mathworks,Inc), using the following toolboxes: Signal Processing, Statistics, Symbolic Math, Wavelet, and the MIDI Toolbox (Eerola & Toiviainen, 2004).

[2] https://dl.dropbox.com/u/11997856/JKU/JKUPDD-noAudio-Aug2013.zip. Accessed 12 May 2014

In the context of this MIREX task, a good melodic structure is considered to be one that is closer to the ground truth analysis, which specifies certain patterns identified by expert analysts as being important or noticeable. These patterns may be nested or hierarchically related (see [1]). We use an agglomerative technique to cluster segments by similarity. Clusters are then ranked according to a perceptually motivated criterion.

2. METHOD

The method follows and extends our previously reported approach to melodic segmentation and classification based on filtering with the Haar wavelet [4] and uses some ideas from a generic motif discovery algorithm for sequential data [2]. It follows [4] in terms of representation and segmentation, extending the segmentation method. As [2] is very generic, we use the idea of computing a similarity matrix for "window connectivity information" as described in section 2.2.4.

2.1 Representation

As in [4], we represent melodies either as raw 1D pitch signals or as these signals filtered with the continuous wavelet transform (CWT) using the Haar wavelet at a single time scale. The melodic contour of a melody is sampled using chromatic MIDI pitch information at a defined sampling rate. In the case of pitch signal representation, after segmentation, melodic segments are normalized by subtracting the average pitch.

2.2 Segmentation

2.2.1 First stage segmentation

We use some of the segmentation methods described in [4] and additionally use modulus maxima segmentation. The segmentation methods are:

- constant segmentation, i.e., segmentation into segments of constant length, or
- modulus maxima, where segmentation points are set at local modulus maxima of the wavelet coefficients.

2.2.2 Segment length normalization

The segments obtained using these methods generally have different lengths. In order to normalize their length for the purpose of measuring their city-block distances,

and therefore have segments of equal length we define a maximal length for all segments and pad shorter segments as necessary with zeros at the end.

2.2.3 Comparison

Segments are compared by building a distance matrix giving all pair-wise distances between segments in terms of normalized city-block distance. The normalization consists of dividing the pairwise distance by the length of the smallest segment before segment length normalization by zero padding.

2.2.4 Concatenation of segments

We binarize the distance matrix setting a threshold: values lower than or equal to the threshold take the value of 1 or true, otherwise the value is 0 or false. We concatenate segments of contiguous true values of the diagonals, to form longer segments.

2.3 Comparison

This time we use the segments that have been concatenated as described in 2.2.4. The comparison is the same as in 2.2.3.

2.4 Clustering

The distance matrix obtained in 2.3 is used for clustering. We use agglomerative clusters from an agglomerative hierarchical cluster tree. Finally, clusters are ranked according to the sum of the length of segments' occurrences.

3. EXPERIMENTS

We tested the following parameter combinations:

- Melodies sampled at 16 samples per quarter note (qn)
- Representation: normalized pitch signal or wavelet coefficients filtered at the scale of 1 qn
- Segmentation: constant segmentation or modulus maxima
- Scale segmentation at 1 or 4 qn
- Threshold for concatenating segments: 0.1 or 1
- Distance for both comparisons: city-block
- Number of clusters: 7
- Ranking criterion: Sum of the length of occurrences

4. RESULTS

We used the evaluation metrics defined by Collins and Meredith in [1] and Collins' Matlab implementation to compute the results. The results are obtained applying our method on the JKU Patterns Development Database monophonic version, which contains five melodies for training: Bach's Fugue BWV 889, Beethoven's Sonata Op. 2, No. 1, Movement 3, Chopin's Mazurka Op. 24, No. 4, Gibbons's Silver Swan, and Mozart's Sonata K.282, Movement 2. Table 1 and Table 2 present the results of our two submissions VM1 and VM2 respectively. In our experiments we have tested all combinations mentioned in section 3, and selected two configurations to submit to MIREX. VM1 differs from VM2 in the following parameters:

- Normalized pitch signal representation,
- Constant segmentation at the scale of 1 qn,
- Threshold for concatenation 0.1.

VM2 differs from VM1 in the following parameters:

- Wavelet coefficients representation filtered at the scale of 1 qn
- Modulus maxima segmentation at the scale of 4 qn
- Threshold for concatenation 1

According to Friedman's test ($\chi^2(1)=1.8$, $p=0.1797$) VM1 and VM2 show no significant difference in the results of the "three-layer" F1 score. However, for discovering exact occurrences, VM1 outperforms VM2, ($\chi^2(1)=4$, $p=0.045$). On the other hand, there is a statistically significant difference in the runtime, suggesting that VM2 should be preferable for fast computation, ($\chi^2(1)=5$, $p=0.0253$).

In general, recall values are slightly higher than precision values, and the standard deviation of the recall values are slightly lower than the standard deviation of the precision values. For standard precision, recall and F1 score, the standard deviation is highest, compared to the standard deviation of establishment and occurrence measures. These results suggest that VM1 and VM2 perform consistent on the training dataset over establishment and occurrence values, and VM1 performs less consistent on the standard measures.

5. CONCLUSIONS

We present a novel computational method for the discovery of repeated themes and sections in monophonic melodies and the results of our two submissions on the same task, considering that VM1 and VM2 perform similarly on the 'three-layer" measures, but VM1 should be preferable for standard measures and VM2 should be preferable for runtime computation.

6. ACKNOWLEDGEMENT

Gissel Velarde is supported by the Department of Architecture, Design and Media Technology at Aalborg University. The contribution of David Meredith to the work reported here was made as part of the "Learning to Create" project (Lrn2Cre8) which acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 610859.

7. REFERENCES

[1] T. Collins. Mirex 2014 competition: Discovery of repeated themes and sections, 2014. http://www.music-ir.org/mirex/wiki/2014:Discovery_of_Repeated_Themes_%26_Sections. Accessed on 12 May 2014.

[2] K. Jensen, M. Styczynski, I. Rigoutsos and G. Stephanopoulos: "A generic motif discovery algorithm for sequential data", Bioinformatics, 22:1, pp. 21-28, 2006.

[3] R. Monelle: *Linguistics and Semiotics in Music*, Harwood Academic Publishers, Chur, 1992.

[4] G. Velarde, T. Weyde and D. Meredith: "An approach to melodic segmentation and classification based on filtering with the Haar-wavelet", *Journal of New Music Research*, 42:4, 325-345, 2013.

Piece	n_P	n_Q	P_est	R_est	F1_est	P_occ (c=.75)	R_occ (c=.75)	F1_occ (c=.75)	P_3	R_3	F1_3	Runtime (s)	FFTP_ est	FFP	P_occ (c=.5)	R_occ (c=.5)	F1_occ (c=.5)	P	R	F1
Bach	3	7	0.87	0.95	0.91	0.63	0.72	0.67	0.51	0.65	0.57	8.50	0.95	0.60	0.63	0.72	0.67	0.14	0.33	0.20
Beethoven	7	7	0.92	0.92	0.92	0.98	0.98	0.98	0.86	0.91	0.88	31.00	0.76	0.80	0.89	0.93	0.91	0.57	0.57	0.57
Chopin	4	7	0.53	0.86	0.66	0.66	0.86	0.75	0.48	0.70	0.57	34.20	0.68	0.47	0.46	0.83	0.60	0.00	0.00	0.00
Gibbons	8	7	0.95	0.95	0.95	0.66	0.93	0.77	0.85	0.79	0.82	17.76	0.77	0.79	0.66	0.93	0.77	0.29	0.25	0.27
Mozart	9	7	0.92	0.79	0.85	0.82	0.96	0.88	0.79	0.69	0.73	23.61	0.67	0.73	0.72	0.92	0.81	0.57	0.44	0.50
mean	**6.2**	**7**	**0.84**	**0.89**	**0.86**	**0.75**	**0.89**	**0.81**	**0.70**	**0.75**	**0.71**	**23.01**	**0.77**	**0.68**	**0.67**	**0.87**	**0.75**	**0.31**	**0.32**	**0.31**
SD	2.59	0	0.17	0.07	0.12	0.15	0.11	0.12	0.19	0.10	0.14	10.34	0.11	0.14	0.15	0.09	0.12	0.26	0.22	0.23

Table 1. Results of VM1 on the JKU Patterns Development Database.

Piece	n_P	n_Q	P_est	R_est	F1_est	P_occ (c=.75)	R_occ (c=.75)	F1_occ (c=.75)	P_3	R_3	F1_3	Runtime (s)	FFTP_ est	FFP	P_occ (c=.5)	R_occ (c=.5)	F1_occ (c=.5)	P	R	F1
Bach	3	7	0.56	0.65	0.60	0.89	0.43	0.58	0.39	0.41	0.40	5.07	0.59	0.37	0.56	0.46	0.50	0.00	0.00	0.00
Beethoven	7	7	0.90	0.90	0.90	0.79	0.89	0.84	0.82	0.86	0.84	5.54	0.67	0.75	0.83	0.90	0.86	0.00	0.00	0.00
Chopin	4	7	0.58	0.86	0.69	0.69	0.83	0.75	0.53	0.78	0.64	5.83	0.65	0.44	0.67	0.65	0.66	0.00	0.00	0.00
Gibbons	8	7	0.92	0.88	0.90	0.79	0.84	0.82	0.81	0.73	0.77	2.22	0.70	0.76	0.72	0.69	0.71	0.14	0.13	0.13
Mozart	9	7	0.83	0.71	0.77	0.93	0.93	0.93	0.77	0.63	0.69	5.70	0.56	0.68	0.84	0.88	0.86	0.00	0.00	0.00
mean	**6.2**	**7**	**0.76**	**0.80**	**0.77**	**0.82**	**0.78**	**0.78**	**0.66**	**0.68**	**0.67**	**4.87**	**0.63**	**0.60**	**0.72**	**0.71**	**0.72**	**0.03**	**0.03**	**0.03**
SD	2.59	0	0.17	0.11	0.13	0.09	0.20	0.13	0.19	0.17	0.17	1.51	0.06	0.18	0.12	0.18	0.15	0.06	0.06	0.06

Table 2. Results of VM2 on the JKU Patterns Development Database.

Paper IV. A Wavelet-Based Approach to Pattern Discovery in Melodies.

A Wavelet-Based Approach to Pattern Discovery in Melodies

Velarde, Gissel; Meredith, David; Weyde, Tillman

Published in:
Computational Music Analysis

DOI (link to publication from Publisher):
10.1007/978-3-319-25931-4_12

Publication date:
2015

Document Version
Peer reviewed version

Chapter 12
A Wavelet-Based Approach to Pattern Discovery in Melodies

Gissel Velarde, David Meredith, and Tillman Weyde

Abstract We present a computational method for pattern discovery based on the application of the wavelet transform to symbolic representations of melodies or monophonic voices. We model the importance of a discovered pattern in terms of the compression ratio that can be achieved by using it to describe that part of the melody covered by its occurrences. The proposed method resembles that of paradigmatic analysis developed by Ruwet (1966) and Nattiez (1975). In our approach, melodies are represented either as 'raw' 1-dimensional pitch signals or as these signals filtered with the *continuous wavelet transform* (CWT) at a single scale using the Haar wavelet. These representations are segmented using various approaches and the segments are then concatenated based on their similarity. The concatenated segments are compared, clustered and ranked. The method was evaluated on two musicological tasks: discovering themes and sections in the JKU Patterns Development Database and determining the parent compositions of excerpts from J. S. Bach's Two-Part Inventions (BWV 772–786). The results indicate that the new approach performs well at finding noticeable and/or important patterns in melodies and that filtering makes the method robust to melodic variation.

12.1 Introduction

Since the 19th century, music theorists have placed great importance on the analysis of motivic repetition and variation (Marx, 1837; Reicha, 1814; Riemann, 1912;

Gissel Velarde · David Meredith
Department of Architecture, Design and Media Technology, Aalborg University, Aalborg, Denmark
e-mail: {gv, dave}@create.aau.dk

Tillman Weyde
Department of Computer Science, City University London, London, UK
e-mail: t.e.weyde@city.ac.uk

Schoenberg, 1967), leading to the development of *paradigmatic analysis* by Ruwet (1966) and Nattiez (1986) during the latter half of the 20th century. Ruwet's method consists of an exhaustive similarity comparison of small units or segments in order to generate a structural description (see Monelle, 1992). Paradigmatic analysis focuses on clustering similar segments in a melody into "paradigms", regardless of where these segments might occur. It is typically carried out in parallel with *syntagmatic analysis* which focuses on identifying sequential relationships between consecutive segments. Syntagmatic and paradigmatic analysis can be seen as complementary tools for exploring the *semiotic* structure of a melody.

Almost three decades after the work by Ruwet, the first computational models to automate paradigmatic analysis of music appeared (Adiloglu et al., 2006; Anagnostopoulou and Westermann, 1997; Cambouropoulos, 1998; Cambouropoulos and Widmer, 2000; Conklin, 2006; Conklin and Anagnostopoulou, 2006; Grilo et al., 2001; Höthker et al., 2001; Weyde, 2001). However, it is difficult to evaluate these models, as some are not fully automated (e.g., require a user-supplied segmentation), the implementations are generally not public and they have not been tested on a common ground truth. Although the notion of defining a ground truth at all for a musical analysis is controversial, the MIREX task on discovery of repeated themes and sections (Collins, 2014) offers a practical opportunity to evaluate thematic analysis algorithms. However, it should be noted that the 'ground truth' analyses used in this task do not include any analyses by experts in paradigmatic analysis.

In this chapter, we focus on describing a fully automated method of musical analysis that closely resembles paradigmatic analysis. It has been implemented in Matlab and it is publicly available.[1] The method is based on segmenting melodies, clustering the resulting segments by similarity and then ranking the clusters obtained. In Sect. 12.3 we present the results obtained when our method was used for discovering repeating themes and sections in the Johannes Kepler University Patterns Development Database (JKU PDD).[2] We also compare these results with those obtained using other methods. In order to test the generalizability of the proposed method, we also evaluated it on a second musicological task, namely, that of identifying the parent compositions of excerpts from J. S. Bach's Two-Part Inventions (BWV 772–786).[3]

[1] Available at http://www.create.aau.dk/music/software/. It is implemented in MATLAB (R2014a, The Mathworks, Inc), using the following toolboxes: Signal Processing, Statistics, Symbolic Math, Wavelet, and the MIDI Toolbox (Eerola and Toiviainen, 2004). We also used an implementation of the dynamic time warping algorithm (DTW) by Paul Micó, accessed on 30-April-2013 from http://www.mathworks.com/matlabcentral/fileexchange/16350-continuous-dynamic-time-warping.

[2] https://dl.dropbox.com/u/11997856/JKU/JKUPDD-Aug2013.zip. Accessed on 12-May-2014.

[3] MIDI encodings edited by Steve Rasmussen,
http://www.musedata.org/encodings/bach/rasmuss/inventio/. Accessed April 2011

12.1.1 Melodic Structure and Wavelet Analysis

Our understanding of melodic structure has benefited from work that has been carried out in a number of fields, including music theory, psychology neuroscience and computer science. For example, melodic contour has been studied by Huron (1996), who classified melodies into 9 types according to their shapes (e.g., ascending, descending, arc-like, etc.) by considering the first, last and average pitches of a melody. In contrast, Schenkerian analysis aims to recursively reduce the musical surface or *foreground* to a *fundamental structure (Urzatz)* via one or more *middleground* levels *(Schichten)* (Schenker, 1935). Furthermore, listeners typically hear melodies to be "chunked" into *segments* or, more generally, *groups* (Cambouropoulos, 1997; Lerdahl and Jackendoff, 1983; Tenney and Polansky, 1980). Neuroscientific evidence from fMRI studies suggests that brain activity increases when subjects perceive boundaries between musical movements, and, indeed, boundaries between events in other, non-musical domains (Kurby and Zacks, 2008). Such evidence strongly supports the notion that segmentation is an essential component of perception, occurring simultaneously at multiple timescales. Psychological approaches focus on perception and memory and have tried to determine relevant melodic structures empirically (see, e.g., Lamont and Dibben, 2001; Müllensiefen and Wiggins, 2011b).

Computational approaches to the analysis of melodic structure include geometric approaches to pattern discovery, grammars, statistical descriptors, Gestalt features and data mining (see, e.g., Conklin, 2006; Mazzola et al., 2002; Meredith et al., 2002; Weyde, 2002). Wavelet analysis is a relatively new approach that has been widely used in audio signal processing. However, to our knowledge, it has been scarcely used on symbolic music representations, except by Smith and Honing (2008), who used wavelets to elicit rhythmic content from sparse sequences of impulses of a piece, and Pinto (2009), who used wavelets for melodic indexing as a compression technique.

As mentioned above, the wavelet-based method that we present below is closely related to paradigmatic analysis. It is based on the assumption that, if a melody is segmented appropriately, then it should be possible to produce a high-quality analysis by gathering together similar segments into clusters and then ranking these clusters by their importance or salience. In our study, we were particularly interested in exploring the effectiveness of the *wavelet transform* (WT) (Antoine, 1999; Farge, 1992; Mallat, 2009; Torrence and Compo, 1998) for representing relevant properties of melodies in segmentation, classification and pattern detection.

Wavelet analysis is a mathematical tool that compares a time-series with a wavelet at different positions and time scales, returning similarity coefficients. There are two main forms of the WT, the *continuous wavelet transform* (CWT) and the *discrete wavelet transform* (DWT). The CWT is mostly used for pattern analysis or feature detection in signal analysis (e.g., Smith and Honing, 2008), while the DWT is used for compression and reconstruction (e.g., Antoine, 1999; Mallat, 2009; Pinto, 2009). In our method, we sample symbolic representations of melodies or monophonic voices to produce one-dimensional (1D) *pitch signals*. We then apply the continuous wavelet transform (CWT) to these pitch signals, filtering with the Haar wavelet (Haar, 1910). Filtering with wavelets at different scales resembles the mechanism by

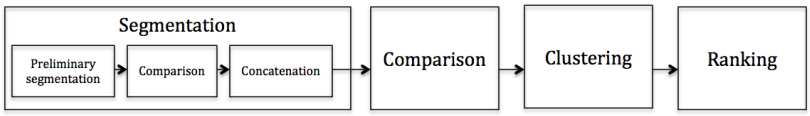


Fig. 12.1 A schematic overview of the main stages of the proposed method

which neurons, such as orientation-selective simple cells in the primary visual cortex, gather information from their receptive fields (Hubel and Wiesel, 1962). Indeed, more recently, Gabor wavelet pyramids have been used to model the perception of visual features in natural scenes (Kay et al., 2008).

Wavelet coefficient encodings seem to be particularly appropriate for melodic analysis as they provide a transposition-invariant representation. We also use wavelet coefficient representations to determine local segment boundaries at different time scales, which accords well with the notion that listeners automatically organize the musical surface into coherent segments, or groups, at various time scales (Lerdahl and Jackendoff, 1983).

12.2 Method

The method presented in this chapter extends our previously reported approach to melodic segmentation and classification based on filtering with the Haar wavelet (Velarde et al., 2013), and also incorporates an approach to segment construction similar to that developed by Aucouturier and Sandler (2002) for discovering patterns in audio data. A schematic overview of the method is shown in Fig. 12.1. In the following sub-sections we explain the method in detail.

12.2.1 Representation

A wide variety of different strategies have been adopted in music informatics for representing melodies, including (among others) viewpoints (Conklin, 2006), strings (McGettrick, 1997), contours (Huron, 1996), polynomial functions (Müllensiefen and Wiggins, 2011a), point sets (Meredith et al., 2002), spline curves (Urbano, 2013), Fourier coefficients (Schmuckler, 1999) and global features (van Kranenburg et al., 2013).

The representations used in this study are illustrated in Fig. 12.2. The top graph in this figure shows what we call a *raw pitch signal*. This is a discrete pitch signal, v, with length, L, constructed by sampling from MIDI files at a rate, r, in samples per quarter note (qn). MIDI files encode pitches as MIDI Note Numbers (MIDI NN). We denote the pitch value at time point t by $v[t]$. This representation is not used for

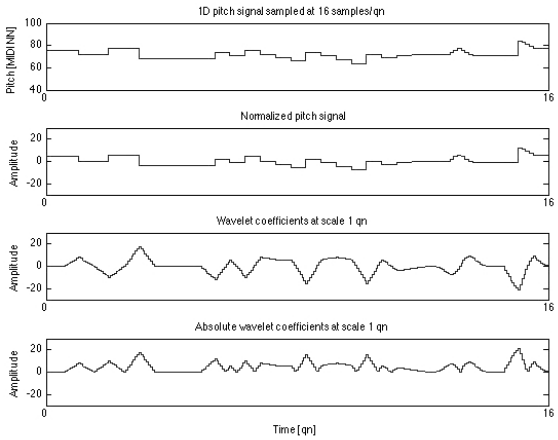


Fig. 12.2 Representations used in the method. From top to bottom: a raw pitch signal, a normalized pitch signal, a wavelet coefficient representation and an absolute wavelet coefficient representation

segment comparison directly. It is either filtered by the Haar wavelet or transformed into what we call a *normalized pitch signal* in order to obtain a transposition-invariant representation which is then segmented.

The second graph in Fig. 12.2 shows a *normalized pitch signal*, obtained by subtracting the average pitch of a segment from the pitch values in that segment. This process is applied to each segment individually after segmentation. It serves to reduce the measured dissimilarity between segments that have very similar contour but occur at different pitch heights (i.e., have different transpositions).

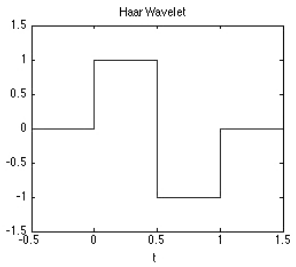
The third graph in Fig. 12.2 shows a *wavelet coefficient representation* resulting from carrying out a continuous wavelet transform (CWT) on the pitch signal with the Haar wavelet at a single time scale. This process tends to highlight structural features at the scale of the wavelet. The Haar wavelet (Haar, 1910) is used because it measures the movement direction of the melody and because its shape reflects the step-wise nature of symbolic pitch signals. Figure 12.3 shows an example of a Haar wavelet.

The CWT computed at a single time scale acts as a *filter* by the convolution of v, the pitch signal, with the scaled and flipped real-valued wavelet for each translation, u, and scale, s:

$$w_s[u] = \sum_{\ell=1}^{L} \psi_{s,u}[\ell] v[\ell] \; . \tag{12.1}$$

To avoid edge effects due to finite-length sequences (Torrence and Compo, 1998), we pad on both ends with a mirror image of v (Woody and Brown, 2007). To maintain

Fig. 12.3 The Haar wavelet



the signal's original length, the segments that correspond to the padding on both ends are removed after convolution.

The bottom graph in Fig. 12.2 shows an *absolute wavelet coefficient* representation. The value at each time point in this representation is the absolute value of the wavelet coefficient at that time point.

The type of wavelet to use depends on the kind of information one wishes to extract from the signal, since the wavelet coefficients combine information about the signal and the analysing function (Farge, 1992). We use the Haar wavelet (Haar, 1910) as the analysing function, as defined by Mallat (2009):

$$\psi_t = \begin{cases} 1, & \text{if } 0 \leq t < 0.5, \\ -1, & \text{if } 0.5 \leq t < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{12.2}$$

The choice of time scale depends on the scale of structure in which one is interested. Local structure is best analysed using short time scales, while longer-term structure can be revealed by using wavelets at longer time scales. When features of the wavelet-based representations are used for segmentation (as will be described in Sect. 12.2.2), using a shorter wavelet leads to smaller segments in general. We therefore expect shorter wavelets to be more appropriate for finding smaller melodic structural units such as motives, while longer wavelets might be expected to produce segments at longer time scales such as the phrase level and above. In the experiments reported below, we used a variety of different scales in order to explore the effect of time scale on performance.

12.2.2 Segmentation

Segmentation is a central component of music perception, occurring simultaneously at multiple timescales as an adaptive mechanism of the brain. It has been shown that brain activity increases transiently at musical movement boundaries, as well as other non-musical event boundaries (Kurby and Zacks, 2008). In agreement with the neuroscientific evidence, most theories of music perception and cognition note the

importance of segmentation, or grouping at various different time scales. Typically, such theories concentrate on the perceived associations of events, relating visual Gestalt principles to the musical domain. Examples of such theories include Tenney and Polansky's theory of temporal Gestalt-units (Tenney and Polansky, 1980), Lerdahl and Jackendoff's theory of grouping structure (Lerdahl and Jackendoff, 1983) and Cambouropoulos' Local Boundary Detection Model (LBDM) (Cambouropoulos, 1997, 2001). The rules in these models address changes in both local parameters and longer-term averages. Similarly, wavelet filters could be used to represent melodic movements at different scales, leading to different levels of localization on the time-axis for deriving group boundaries. Conklin (2006) also stresses the importance of melodic analysis on segmentation. He additionally demonstrates the effect of different symbolic melodic representations called *viewpoints* at different time scales (note, beat, bar, phrase and piece level) in the context of style discrimination.

As shown in Fig. 12.1, the *Segmentation* phase of our method is split into three subphases: *Preliminary segmentation*, *Comparison* and *Concatenation*. Each of these subphases will now be described.

12.2.2.1 Preliminary Segmentation

In this study, we explored three strategies for producing a preliminary segmentation: constant-duration segmentation; segmentation at zero crossings in the wavelet coefficient and absolute wavelet coefficient representations; and segmentation at local maxima in the absolute wavelet coefficient representation. The lower three graphs in Fig. 12.4 show three of the possible combinations of representation and segmentation.

The simplest segmentation strategy that we explore is *constant-duration segmentation* in which the signal is chunked into segments of constant duration (with the possible exception of the final segment which could be shorter than the other segments). The second graph in Fig. 12.4 shows an example of this type of segmentation combined with a normalized pitch signal representation.

We also experiment with *zero-crossings* segmentation in the wavelet-based representations, where segment boundaries are set at time points with value zero in the representation. Zero-crossings occur when the inner product between the melody and the Haar wavelet is zero. This means that the average pitch in the first half of the scale period is equal to the average pitch in the second half of the scale period.

The third segmentation strategy we use is *absolute maxima* segmentation, where segment boundaries are set at time points corresponding to local maxima in the absolute wavelet coefficient representation. These maxima occur when the inner product of the wavelet and the signal is locally maximal. In our case, this corresponds to time points when there is a maximal positive or negative correlation between the shape of the melody and the Haar wavelet. These points occur when there is a locally maximal fall or rise in average pitch content at the scale of the wavelet used. The absolute maxima of a real wavelet such as the Haar wavelet are a special case of the *modulus maxima* of a wavelet transform in general. The latter were used by Muzy

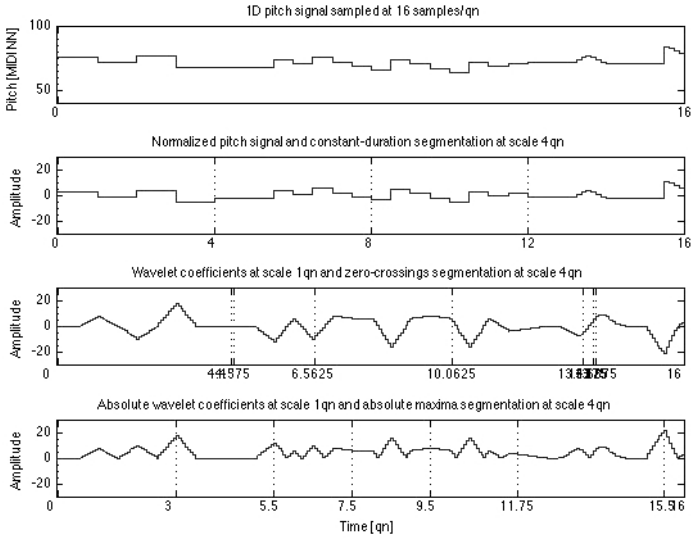


Fig. 12.4 Segmentation approaches used in the method, from top to bottom: a raw pitch signal without segmentation; normalized pitch signal and constant-duration segmentation at a scale of 4 qn; wavelet coefficient representation filtered with the Haar wavelet at a scale of 1 qn and segmented at zero-crossings at a scale of 4 qn; absolute wavelet coefficient representation filtered at a scale of 1 qn and segmented at absolute maxima at a scale of 4 qn. Note that the wavelet scales used to generate the *representations* shown in the third and fourth graphs are different from those used to produce the *segmentations*. The segmentation points therefore do not necessarily coincide with zero-crossings or maxima in the wavelet coefficient representations shown

et al. (1991) to show the structure of fractal signals and by Mallat and Hwang (1992) to indicate the location of edges in images. The bottom graph in Fig. 12.4 shows an example of absolute maxima segmentation of an absolute wavelet coefficient representation.

The segments obtained using these three strategies generally have different durations. However, in order to measure similarity between them using standard metrics such as *city block* or *Euclidean distance*, it is necessary for the segments to be the same length. We achieve this by defining a maximal length for all segments and padding shorter segments as necessary with zeros at the end.

12.2.2.2 Comparison

Segments are compared by building an $m \times m$ distance matrix, H, giving all pair-wise distances between segments in terms of normalized distance. m is the number of

segments. We use three different distance measures: *Euclidean distance*, *city block distance* and *dynamic time warping* (DTW). For city block and Euclidean distances, the segments compared must be of equal length and in these cases the normalization consists of dividing the pairwise distance by the length of the smallest segment before segment-length equalization by zero padding. When using DTW, which is an alignment-based method, it is not necessary to equalize the lengths of the segments being compared. In this case, therefore, the normalization consists of dividing the distance by the length of the aligned segments.

We use the *Euclidean distance* $d_{\mathrm{E}}(x,y)$ between two segments, x and y, which is defined as follows:

$$d_{\mathrm{E}}(x,y) = \sqrt{\sum_{j=1}^{n} (x[j]-y[j])^2}\,, \tag{12.3}$$

and the *city block distance* $d_{\mathrm{C}}(x,y)$ between x and y:

$$d_{\mathrm{C}}(x,y) = \sum_{j=1}^{n} |x[j]-y[j]|\ . \tag{12.4}$$

The *dynamic time warping distance* (DTW), $d_{\mathrm{D}}(x,y)$, is the minimal cost of a *warping path* between sequences x and y. A warping path of length, L, is a sequence of pairs $p = ((n_1,m_1),...,(n_L,m_L))$, where n_i is an index into x and m_i is an index into y. p needs to satisfy several conditions which ensure that it can be interpreted as an alignment between x and y that allows skipping elements in either sequence (see Müller, 2007, p. 70). The DTW distance, $d_{\mathrm{D}}(x,y)$, is then defined to be the total cost of a warping path, defined to be the sum of a local cost measure, $c(x[n_i],y[m_i])$, along the path:

$$d_{\mathrm{D}}(x,y) = \sum_{i=1}^{L} c(x[n_i],y[m_i])\,, \tag{12.5}$$

where, here, $c(x[n_i],y[m_i])$ is defined to be simply the absolute difference, $|x[n_i]-y[m_i]|$.

Having computed all the pairwise distances in the matrix, H, these values are then normalized in the range $[0,1]$ by dividing each pairwise distance by the largest distance in the matrix for that distance type.

12.2.2.3 Concatenation of Segments

The final subphase of the segmentation phase is to concatenate consecutive segments found in the preliminary segmentation to form larger units that are then compared, clustered and ranked in the subsequent phases of the method.

The first subphase of the segmentation phase gives a preliminary segmentation of the melody. It is preliminary, as it may be the case that a repeated (or approximately repeated) segment discovered in the preliminary segmentation only occurs as part of a longer repeated segment, such that a paradigmatic relation is found. In such

cases, one would generally only be interested in the longer repeated segment (this relates to the concept of "closed patterns" (see Lartillot, 2005, and Chap. 11, this volume) and Meredith et al.'s (2002) concept of "maximal translatable patterns" (see also Chap. 13, this volume). One would only want to report the shorter segment if it also occurred independently of the longer segment. In the third subphase of the segmentation phase, we therefore concatenate, or merge locally, the preliminary segments derived in the preliminary segmentation into generally longer units, that are then passed on to the later phases of the method.

Segments are concatenated based on their similarity. We therefore set a threshold, τ, that defines the level of similarity between preliminary segments required to allow concatenation. The $m \times m$ distance matrix, H, is therefore binarized as follows:

$$H(i,j) = \begin{cases} 1, & \text{if } H(i,j) \leq \tau, \\ 0, & \text{otherwise,} \end{cases} \tag{12.6}$$

for $1 \leq i \leq m$ and $i \leq j \leq m$ (note that we use 1-based indexing in this chapter).

Segments are concatenated to form units based on the information contained in the upper triangle including the leading diagonal in the binarized similarity matrix, H, scanning the matrix horizontally and diagonally. A *unit*, $\overline{(i,j)}$, $i \leq j$, consists of the concatenated segments $i,\ldots,j$, and we use two concatenation processes to generate units.

A process of *horizontal concatenation* generates units that consist of consecutive occurrences of the "same" segment (i.e., corresponding to horizontal sequences of consecutive 1s in the binarized similarity matrix, H). The units, $\overline{(i,k)}$, generated by this process are those for which $hor(i,k)$ is true, where

$$hor(i,k) \iff (hor(i,k-1) \wedge H(k-1,k) = 1) \vee (i = k)\,. \tag{12.7}$$

A process of *diagonal concatenation* generates units that are repeated in the piece, and $dia(i,j)$ must be true, where

$$\begin{aligned} dia(i,j) \iff & (dia(i,j-1) \wedge \exists \ell,k \mid \ell - k = j - i \wedge dia(k,\ell-1) \wedge H(j-1,\ell-1) = H(j,\ell) = 1) \\ & \vee (j - i = 1 \wedge \exists \ell \mid H(i,\ell-1) = H(j,\ell) = 1)\,. \end{aligned} \tag{12.8}$$

Any $hor(i,j)$ or $dia(i,j)$ that is not a strict subset of another generates a unit $\overline{(i,j)}$. Subsets will be identified as *trivial units*.

When these two concatenation processes are carried out on the matrix in Fig. 12.5, horizontal concatenation generates the unit $\overline{(9,10)}$ and diagonal concatenation generates the units $\overline{(1,2)}$, $\overline{(4,5)}$ and $\overline{(7,8)}$.

The concatenation method presented here is similar to the one described by Aucouturier and Sandler (2002).

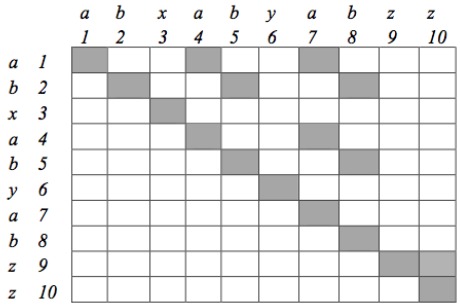


Fig. 12.5 Upper triangular matrix, grey means 1 and white 0. It corresponds to the binarized distance matrix H of the sequence $v_1 = abxabyabzz$

12.2.3 Comparison and Clustering of Units

In this second comparison, the units constructed in the previous concatenation step (Sect. 12.2.2.3) are compared using the same process of similarity measurement as that described in Sect. 12.2.2.2. Any two units $\overline{(\ell, j)}$ and $\overline{(p, r)}$ obtained by concatenation, will then be units x and y respectively, to be compared in this second comparison.

Having obtained values for the pairwise similarity between units, these similarity values are then used to cluster the units into classes. To achieve this, we use a simple hierarchical agglomerative clustering method called *single linkage*, or *nearest-neighbour*, which produces a series of successive fusions of the data, starting from N single-member clusters that fuse together to form larger clusters (Everitt et al., 2011; Florek et al., 1951; Johnson, 1967; Sneath, 1957). Here, the distance matrix obtained from the comparison as described in Sect. 12.2.3 is used for clustering. *Single linkage* takes the smallest distance between any two units, one from each group or cluster. The distance $D(X,Y)$ between clusters X and Y is described as

$$D(X,Y) = \min_{x \in X, y \in Y} d(x,y) \,, \tag{12.9}$$

where clusters X and Y are formed by the fusion of two clusters, x and y, and $d(x,y)$ denotes the distance between the two units x and y (Everitt et al., 2011). Consider the case of five units or clusters v, w, x, y and z, as shown on the left in Fig. 12.6 as points in a Euclidean space. The minimal distance occurs for x and y, and for z and w. Then, two new clusters are formed, a cluster s consisting of x and y and a cluster t consisting of z and w. The next minimal distance occurs for v and t, forming a new cluster u consisting of v and t. Finally, clusters s and u are grouped together into a cluster c. The right plot in Fig. 12.6 shows a dendrogram of the formed clusters. The y-axis corresponds to the distances between clusters; for instance, clusters x and y

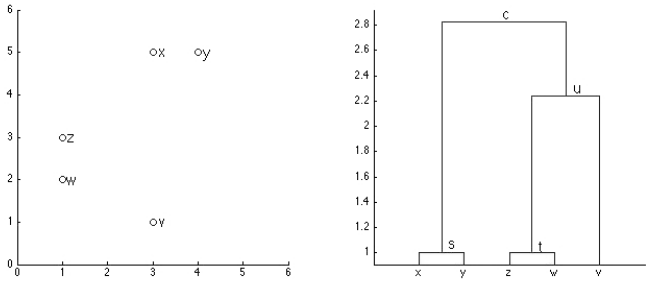


Fig. 12.6 Example of the hierarchical clustering of units or clusters *v*, *w*, *x*, *y* and *z*. Left plot shows the units in a Euclidean space. Right plot shows a dendrogram of the formed clusters

have a distance of 1, and clusters t and u have a distance of 2.2. In this example, the number of clusters ranges from 1, where all units form a single cluster, to 5, where each cluster contains just one unit. The number of clusters can be set to be three, having clusters s, t and u or it can be set to two, giving clusters s and u. Finally, the number of clusters is set to yield the best classification results.[4]

12.2.4 Ranking

In general, if X and Y are two parts of some object, then one can describe $X \cup Y$ in an *in extenso* fashion simply by specifying the properties of each atomic component in X and Y. Alternatively, if there exists a sufficiently simple transformation, T, that maps X onto Y, then it may be possible to provide a compact description of $X \cup Y$ by providing an in extenso description of X along with a description of T.[5]

In the current context, each cluster generated by the previous stage of the method contains units (i.e., parts of a melody) that are similar to each other. If every member of a cluster can be generated by a simple transformation of one member (e.g., if all the units within a cluster are exact repeats of the first occurrence), then the portion of the melody *covered* by the cluster (i.e., the union of the units in the cluster) can be represented by giving an explicit description of the first occurrence along with the positions of the other occurrences. If the members of the cluster do not overlap, then such a representation can be compact because the starting position of a unit can usually be specified using fewer bits than explicitly describing the content of the unit. This would give a losslessly compressed encoding of the part of the melody

[4] When preliminary experiments were performed on the JKU PDD, using between 3 and 10 clusters, the best classification results were obtained using 7 clusters. We therefore used 7 clusters in the experiments reported in Sect. 12.3 below.

[5] This idea is discussed in more detail in Chap. 13, this volume.

covered by the union of the units in the cluster. This is the essential idea behind the compression-driven geometric pattern discovery algorithms described by Meredith (2006, 2013) and Forth (2012). If we represent the music to be analysed as a set of points in pitch-time space and if a cluster (or 'paradigm'), C, only contains the *exact* occurrences of a pattern, p, then the compression ratio achieved is

$$CR(C) = \frac{|\bigcup_{q \in C}\{q\}|}{|p| + |C| - 1}\,, \tag{12.10}$$

where $|\cdot|$ denotes the cardinality of a set. Here, however, the units within a cluster are not necessarily exact repetitions of some single pattern. This means that the degree of compression achievable with one of the clusters generated in the previous sections will not, in general, be as high as in (12.10).

Collins et al. (2011) have provided empirical evidence that the compression ratio achievable in this way by a set of occurrences of a pattern can be used to help predict which patterns in a piece of music are heard to be noticeable and/or important. In the method presented in this chapter, we therefore adapt (12.10) to serve as a measure of importance or noticeability for the clusters generated in the previous phase of the method. Here, we define the "compression ratio", CR_k, of cluster k as follows:

$$CR_k = \frac{\sum_{i=1}^{n_k} S_i}{(n_k + \bar{S}_k)}\,, \tag{12.11}$$

where n_k is the number of units in cluster k, S_i is the length in sample points of unit i in cluster k, and $\bar{S}_k$ is the mean length of a unit in cluster k. Clusters are ranked into descending order by this value of "compression ratio". All clusters are kept in the final output.

12.3 Experiments

The method described above was evaluated on two tasks: discovering repeated themes and sections in monophonic music; and identifying the parent works of excerpts from J. S. Bach's Two-Part Inventions (BWV 772–786). The methods used and results obtained in these experiments will now be presented.

12.3.1 Experiment 1: Discovering Repeated Themes and Sections in Monophonic Music

Various computational methods for discovering patterns in music have been developed over the past two decades (see Janssen et al., 2013, for a recent review), but only recently have attempts been made to compare their outputs in a rigorous way.

Notable among such attempts are the two tasks on discovering repeated themes and sections that have been held at the Music Information Retrieval Evaluation eXchange (MIREX) in 2013 and 2014 (Collins, 2014). In these tasks, algorithms have been run on a set of five pieces and the analyses generated by the algorithms have been compared with ground truth analyses by expert analysts. A number of measures were devised for evaluating the performance of pattern discovery algorithms in this competition and comparing the output of an algorithm with a ground truth analysis (Collins, 2014). Collins has also provided a training database, the JKU PDD, which exists in both monophonic and polyphonic versions. The JKU PDD consists of the following five pieces along with ground truth analyses:

- Orlando Gibbons' madrigal, "Silver Swan" (1612);
- the fugue from J. S. Bach's Prelude and Fugue in A minor (BWV 889) from Book 2 of *Das wohltemperirte Clavier* (1742);
- the second movement of Mozart's Piano Sonata in E flat major (K. 282) (1774);
- the third movement of Beethoven's Piano Sonata in F minor, Op. 2, No. 1 (1795); and
- Chopin's Mazurka in B flat minor, Op. 24, No. 4 (1836).

The monophonic versions of the pieces by Beethoven, Mozart and Chopin were produced by selecting the notes in the most salient part (usually the top part) at each point in the music. For the contrapuntal pieces by Bach and Gibbons, the monophonic encodings were produced by concatenating the voices (Collins, 2014).

We used the JKU PDD as a training set for determining optimal values for the parameters of the analysis method described above. Heuristics based on knowledge gained from previous experiments (Velarde et al., 2013) were used to start tuning the parameters. Then, in an attempt to approach optimal values, all parameters were kept fixed, except one which was varied along a defined range to find an optimal adjustment. This process was repeated for all parameters. Finally, the method was run on the JKU PDD with 162 different parameter value combinations, consisting of all possible combinations of the following:

- 1 sampling rate: 16 samples per qn
- 3 representations: normalized pitch signal, wavelet coefficients filtered at the scale of 1 qn, absolute wavelet coefficients filtered at the scale of 1 qn
- 3 segmentation strategies: constant-duration segmentation, segmentation at zero-crossings, segmentation at absolute maxima
- 2 scales for segmentation: 1 qn and 4 qn
- 1 threshold for binarizing the similarity matrix: 0.001
- 3 distances for measuring similarity between segments on the first comparison: city block (CB), Euclidean (Eu) and dynamic time warping (DTW)
- 3 distances for measuring similarity between segments on the second comparison: city block (CB), Euclidean (Eu) and dynamic time warping (DTW)
- 1 strategy for equalizing the lengths of segments for comparison: segment length normalization by zero padding
- 1 clustering method: Single linkage (nearest neighbour)
- 1 value for the number of clusters: 7

- 1 criterion for ranking clusters: compression ratio

12.3.1.1 Results

We used the monophonic version of the JKU PDD with the evaluation metrics defined by Collins (2014) and Meredith (2015), which we computed using Collins' Matlab implementation.[6] The evaluation metrics consist of a number of variants on standard precision, recall and F_1 score, designed to allow algorithms to gain credit for generating sets of occurrences of patterns that are similar but not identical to those in the ground truth. The standard versions of the metrics are not adequate for evaluating pattern discovery algorithms because they return 0 for a computed pattern even if it differs from a ground truth pattern by only one note.

The more robust versions of the precision, recall and F_1 score are designed to measure (1) the extent to which an algorithm finds at least one occurrence of a pattern (*establishment recall/precision/F_1 score*); (2) the extent to which an algorithm finds all the occurrences of a pattern (*occurrence recall/precision/F_1 score*); and (3) the overall similarity between the set of occurrence sets generated by an algorithm and the set of occurrence sets in a ground truth analysis (*three-layer precision/recall/F_1 score*). As these different metrics reveal different aspects of the method's strengths or weaknesses, we decided to evaluate our method based on the standard F_1 score, where P is precision and R is recall

$$F_1 = \frac{2PR}{P+R} \tag{12.12}$$

and on the mean of establishment F_1 ($F1\_est$), occurrence F_1 at (c=.75) ($F1\_occ_{(c=.75)}$), occurrence F_1 at (c=.5) ($F1\_occ_{(c=.5)}$) (Collins, 2014), and three-layer F_1 ($F1\_TL$) (Meredith, 2015):

$$F1\_mean = \frac{F1\_est + F1\_occ_{(c=.75)} + F1\_occ_{(c=.5)}) + F1\_TL}{4}. \tag{12.13}$$

Figure 12.7 shows the highest mean F_1 scores ($F1\_mean$) for each combination, considering segmentation scale, representation type and segmentation type. The left plot shows nine combinations where the segmentation scale was 1 qn, while the right plot shows the scores of nine combinations where the segmentation scale was 4 qn. For each plot in Fig. 12.7, there are 3 bars grouped for each segmentation method, where the grey tones (dark grey, light grey and white) indicate the three representation types, and finally, the distance measures associated with the first and second comparison (e.g., "EU,EU", "CB,CB", etc.). Figure 12.8 shows the corresponding standard F_1 scores for the same combinations. Finally, Fig. 12.9 shows the runtimes in seconds obtained with our implementations of the method, associated with each combination.

[6] https://dl.dropbox.com/u/11997856/JKU/JKUPDD-Aug2013.zip. Accessed on 12-May-2014.

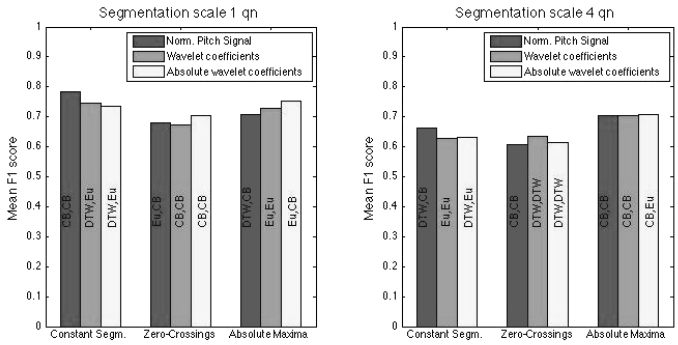


Fig. 12.7 Mean F_1 score ($F1\_mean$)

We ran the experiment twice, the first time keeping trivial units and the second time discarding trivial units. Figures 12.7, 12.8 and 12.9 show the results when keeping trivial units. A Wilcoxon signed rank test indicated that keeping or discarding trivial units did not significantly affect the results of mean F_1 scores ($Z = -1.2439$, $p = 0.2135$), standard F_1 scores ($Z = -1.633$, $p = 0.1025$), or runtimes ($Z = -0.8885$, $p = 0.3743$), for a segmentation scale of 1 qn. Similarly, no difference was found in the results when keeping or discarding trivial units for a scale of 4 qn for mean F_1 scores ($Z = 1.007$, $p = 0.3139$), standard F_1 scores ($Z = 0$, $p = 1$), or runtimes ($Z = -0.53331$, $p = 0.5940$). Therefore, only the results of the first run are shown and explained in the following paragraphs.

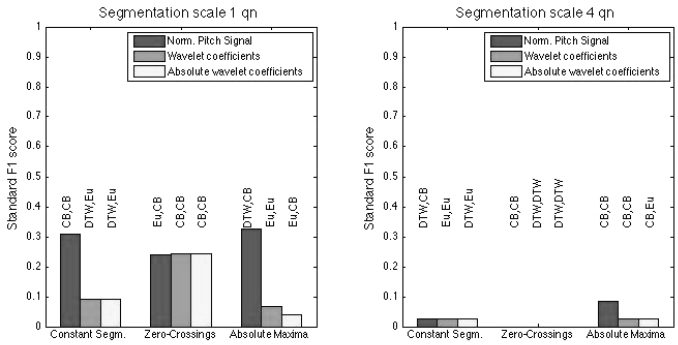


Fig. 12.8 Standard F_1 score

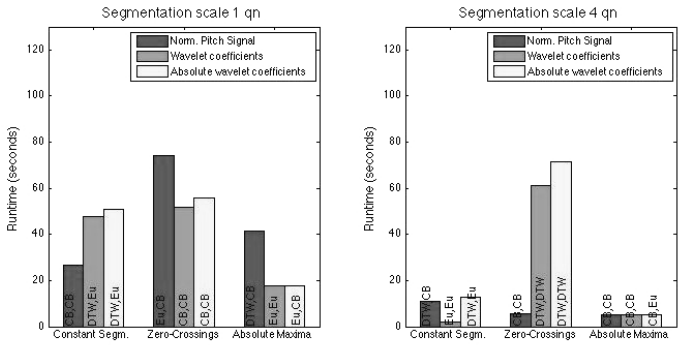


Fig. 12.9 Runtimes in seconds obtained using our implementation of the method. The implementation was programmed using Matlab 2014a and run on a MacBook Pro using MAC OS X with a 2.3 GHz, Intel Core i7 processor and 8 GB 1600 MHz DDR3 RAM

According to the parameters tested, we observe that the segmentation scale used in the preliminary segmentation phase has a greater effect on the results. Figures 12.8 and 12.9 show that using a smaller segmentation scale of 1 qn as opposed to 4 qn was in general slower but produced better results. A Wilcoxon signed rank test indicated there is a statistically significant difference between the use of a smaller and larger scale ($Z = 2.6656$, $p = 0.007$), suggesting that a scale of 1 qn should be used in the preliminary segmentation phase, for higher (mean or standard) F_1 scores.

In terms of mean F_1 score (Fig. 12.7), the normalized pitch signal representation worked slightly better than the wavelet representations when constant-duration segmentation was used. We speculate that only with additional pattern data containing greater variation between occurrences would the benefit of wavelet over normalized pitch representations emerge (see Sect. 12.3.2.1 for more discussion on this point). DTW was used less frequently than Euclidean or city block distance in the best-performing combinations. It seems possible that DTW might have proved more useful if the input representations had included temporal deviations such as *ritardando* or *accelerando* such as might occur in an encoding generated from a live performance.

From Figs. 12.7, 12.8 and 12.9 it is not possible to determine whether the running time is more dependent on the segmentation approach or on the distance measure used. Tables 12.1 and 12.2, show the highest mean F_1 scores of combinations using the same distance measure for both comparison phases, averaged by representation approach. From Table 12.1, it is possible to observe that when using a scale of 1 qn for the preliminary segmentation phase, Euclidean and city-block distances have similar performance, and their F_1 scores are higher than the ones delivered when using DTW distance. However, this gap becomes smaller when the scale is 4 qn. The results in Table 12.2 show that the running times using DTW are more than 8 times slower than those obtained using Euclidean or city-block distances. Evaluating runtimes according to segmentation approaches, it is possible to observe that for the smaller

Table 12.1 Mean F_1 scores averaged over representations, combinations of same distance measure for both comparisons. The rows correspond to the different combinations of distances (CB = city-block, Eu = Euclidean and DTW = dynamic time warping), while the columns correspond to the segmentation approaches (CS = constant-duration segmentation, ZC = zero-crossings segmentation, and AM = absolute maxima segmentation). Mean and standard deviation values are shown per row and per column

	Segmentation scale 1 qn					Segmentation scale 4 qn				
	CS	ZC	AM	**Mean**	SD	CS	ZC	AM	**Mean**	SD
CB-CB	0.74	0.69	0.75	**0.73**	0.03	0.65	0.60	0.70	**0.65**	0.05
Eu-Eu	0.73	0.68	0.72	**0.71**	0.03	0.63	0.59	0.70	**0.64**	0.05
DTW-DTW	0.57	0.64	0.60	**0.60**	0.04	0.59	0.61	0.66	**0.62**	0.03
Mean	**0.68**	**0.67**	**0.60**			**0.62**	**0.60**	**0.69**		
SD	0.10	0.03	0.08			0.03	0.01	0.03		

Table 12.2 Corresponding mean running times in seconds of the combinations in Table 12.1

	Segmentation scale 1 qn					Segmentation scale 4 qn				
	CS	ZC	AM	**Mean**	SD	CS	ZC	AM	**Mean**	SD
CB-CB	24.3	60.8	17.9	**34.32**	23.17	2.2	5.4	5.2	**4.23**	1.80
Eu-Eu	24.4	57.1	17.8	**33.10**	21.02	2.1	5.3	5.1	**4.16**	1.79
DTW-DTW	664.4	2248.2	720.1	**1210.91**	898.77	21.5	61.5	67.4	**50.14**	25.01
Mean	**237.69**	**788.70**	**251.93**			**8.58**	**24.04**	**25.92**		
SD	369.56	1263.98	405.44			11.17	32.44	35.97		

Table 12.3 Mean F_1 scores averaged over representations, when the concatenation phase is not performed. The rows of the Table indicate the distances used for comparison (CB = city-block, Eu = Euclidean and DTW = dynamic time warping), while the columns correspond to the segmentation approaches (CS = constant-duration segmentation, ZC = zero-crossings segmentation, and AM = absolute maxima segmentation). Mean and standard deviation values are shown per rows and per columns

	Segmentation scale 1 qn					Segmentation scale 4 qn				
	CS	ZC	AM	**Mean**	SD	CS	ZC	AM	**Mean**	SD
CB	0.10	0.18	0.11	**0.13**	0.04	0.22	0.23	0.18	**0.21**	0.03
Eu	0.10	0.14	0.10	**0.11**	0.02	0.22	0.21	0.16	**0.20**	0.03
DTW	0.10	0.09	0.11	**0.10**	0.01	0.22	0.20	0.18	**0.20**	0.02
Mean	**0.10**	**0.14**	**0.11**			**0.22**	**0.21**	**0.18**		
SD	0.00	0.04	0.01			0.00	0.02	0.01		

scale of 1 qn in the preliminary segmentation phase, the runtimes of constant-duration segmentation and wavelet absolute maxima segmentation are similar and about twice as fast as the runtimes of the zero-crossings segmentation. On the other hand, for a larger scale of 4 qn in the preliminary segmentation phase, constant-duration segmentation is three times faster than wavelet segmentation approaches.

Table 12.3 shows the effect of not using the concatenation phase: melodies undergo the preliminary segmentation phase, but skip the first comparison and the concatenation phases, such that all preliminary segments are used for the comparison, clustering and ranking phases. The results in Table 12.3 show that omitting the concatenation phase severely reduces the performance of the method on this task. In this

case, when segments are not concatenated, a segmentation scale of 4 qn is, in almost all combinations, twice as good as a segmentation scale of 1 qn. On the other hand, as seen in Table 12.1, a preliminary segmentation phase with a finer segmentation scale, helps to improve the identification of patterns in this dataset.

12.3.1.2 Comparison with Other Computational Methods

The other computational methods addressing the MIREX task on Discovery of repeated themes and sections, included geometric approaches (Meredith, 2013), incremental mining methods (Lartillot, 2014) and methods based on audio techniques (Nieto and Farbood, 2013, 2014).[7] For comparison, we selected our submission VM1, as this configuration was also selected for comparison in the published results of the task. The details of the parameters settings of VM1 are described by Velarde and Meredith (2014).

Table 12.4 shows the results obtained by the different algorithms in the 2014 MIREX task on the monophonic version of the JKU Patterns Test Database (PTD). As can be seen in this table, our method ranked highest at discovering at least one occurrence of each ground truth pattern ($F1\_est$) as well as being the fastest method. Lartillot's method (OL1) performed better at finding inexact occurrences of patterns ($F1\_occ_{(c=.75)}$) but is considerably slower. VM1 and OL1 performed at a similar level with respect to finding exact occurrences of the patterns, and, in both cases, the standard deviation was high. The addition of more pieces to training and test databases over time will enable researchers to investigate the generalizability of their methods.

Table 12.4 Results on the JKU test set. NF1 (Nieto and Farbood, 2014), OL1 (Lartillot, 2014), VM1 (Velarde and Meredith, 2014) and DM10 (Meredith, 2013)

		$F1\_est$	$F1\_occ_{(c=.75)}$	TL_F1	$F1$	*Runtime*
NF1	Mean	0.50	0.41	0.33	0.02	480.80
	SD	0.14	0.27	0.12	0.05	558.43
OL1	Mean	0.50	**0.81**	0.43	0.12	35508.82
	SD	0.17	0.12	0.13	0.13	52556.11
VM1	Mean	**0.73**	0.60	**0.49**	**0.16**	**100.80**
	SD	0.14	0.09	0.14	0.15	119.18
DM10	Mean	0.55	0.62	0.43	0.03	161.40
	SD	0.06	0.09	0.08	0.04	194.87

[7] Results of the annual MIREX competitions on Discovery of Repeated Themes and Sections can be found on the MIREX website at at http://www.music-ir.org/.

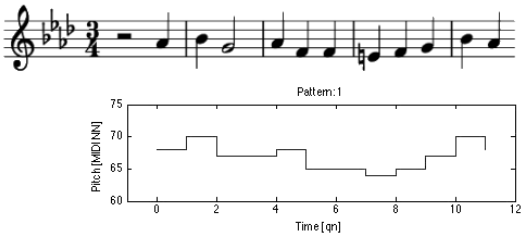


Fig. 12.10 Notation and pitch-signal representations of the first ground truth pattern for the third movement of Beethoven's Piano Sonata in F minor, Op. 2, No. 1 (1795)

12.3.1.3 Comparing Patterns Discovered Automatically with Patterns Identified by Experts

In this section, we present the output of the computational method compared to the JKU PDD ground truth analysis of the monophonic version of the third movement of Beethoven's Piano Sonata in F minor, Op. 2, No. 1 (1795). In order to visualize the ground truth and computationally discovered patterns and their occurrences, we will present them as pitch signals rather than in notation. To help with understanding the correspondence between the pitch signal representation and notation, Fig. 12.10 shows both representations of the first ground truth pattern.

The ground truth analysis for this piece identifies seven patterns and their occurrences as shown in Fig. 12.11. In this figure, plots on the left correspond to patterns, while plots on the right correspond to pattern occurrences. Each pattern occurrence is marked with vertical dotted lines in the graphs on the right side of the figure. All pitch signals have been shifted to start at time 0. The patterns are ordered, from top to bottom, in decreasing order of salience. The lengths of these seven ground truth patterns range from 12 to 119 qn. Some occurrences of the patterns overlap as is the case for the occurrences of pattern 1 and pattern 3, or pattern 2 and pattern 5.

The computational analysis of the piece can be seen in Fig. 12.12. The parameters used are the following:

- 1 sampling rate: 16 samples per qn
- representations: absolute wavelet coefficients filtered at the scale of 1 qn
- segmentation at absolute maxima
- scales for segmentation: 1 qn
- threshold for binarizing the similarity matrix: 0.001
- distance for measuring similarity between segments on the first comparison: city block (CB)
- distance for measuring similarity between segments on the second comparison: city block (CB)
- clustering method: Single linkage (nearest neighbour)
- value for the number of clusters: 7

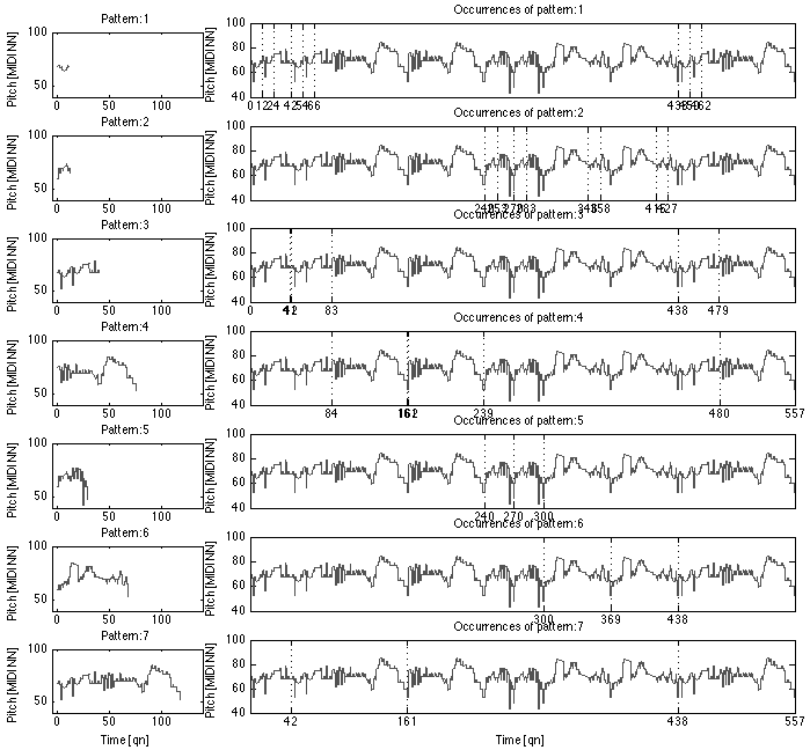


Fig. 12.11 JKU PDD Ground truth patterns for the third movement of Beethoven's Piano Sonata in F minor, Op. 2, No. 1 (1795). Pitch signal representation, with signals shifted to start at time 0. Plots on the left correspond to the patterns, while plots on the right correspond to the entire piece, with each pattern occurrence marked with a vertical dotted line at its starting and ending position

- criterion for ranking clusters: compression ratio

In this example, the number of clusters is the same as the number of patterns in the ground truth. Once again, the salience of patterns can be seen from top to bottom, where the most salient pattern is shown in the top plot. Six out of seven pattern shapes match approximately the ground truth pattern shapes (in some cases, some notes may be missing at the beginning or end of a pattern). The pattern that has been ranked as the most salient, corresponds to pattern 2 in the ground truth analysis, and all its four occurrences have been found. The shape of the second most salient computed pattern, does not resemble the shape of any of the patterns in the ground truth. Pattern 2 is a short-duration pattern, whose cluster contains several melodic units, including segments that approximate the occurrences of pattern 1 in the ground truth (this cannot be seen in Fig. 12.12). The remaining computed pattern

shapes (patterns 3–7) can be found in the ground truth, each with the same number of occurrences. The ranking of salience is not exactly the same as in the ground truth, but it is similar in chunks, such that:

- the first two computed clusters correspond to the first two pattern occurrences in the ground truth;
- computed cluster 3 corresponds to the occurrences of ground truth pattern 3;
- computed clusters 4–6 correspond to the occurrences of ground truth patterns 4–6,
- and finally the last computed cluster corresponds to the occurrences of the last ground truth pattern.

The second cluster contains several melodic units. In future work, we would like to cluster such clusters until they satisfy a given condition and discard clusters that fail to satisfy the condition. We expect that the effect on such clusters of keeping or discarding trivial units may be more evident if we carry out this process.

12.3.2 Experiment 2: Classification of Segments from J. S. Bach's Two-Part Inventions

We also evaluated the method on a second task where the goal was to recognize the parent works of excerpts from J. S. Bach's 15 Two-Part Inventions (BWV 772–786). In contrast to the first experiment, in this task, all segments were used in the evaluation, not just concatenated units. Also, whereas in the first experiment there was room for disagreement about the validity of the ground truth, in this second task, the ground truth was not controversial—there was no doubt as to which parent Invention each test excerpt belonged to. The notion that the piece to which an excerpt belongs can be identified on the basis of the content of the excerpt is based on the premise that the musical material in the excerpt is motivically related to the rest of the piece. Specifically, in the case of Bach's Two-Part Inventions, it is well established that the opening exposition of each of these pieces presents the motivic material that is developed throughout the rest of the piece, which is typically divided into three sections (Dreyfus, 1996; Stein, 1979). In this experiment, we followed the experimental setup described by Velarde et al. (2013), building the classifier from the expositions of the pieces and the test set from the three following sections of each piece. More precisely, an initial, 16 qn segment from each piece was used to build the classifier, and the remainder of each piece was split into three sections of equal length which were used to build the test set. We could have attempted to determine the length of each exposition precisely, but we wanted to avoid making subjective analytical judgements. We therefore used a fixed length of 16 qn as the length of each "exposition" section despite the fact that the actual lengths of the expositions in the Inventions vary. This particular length was chosen because it was the length of the longest exposition in the pieces, thus ensuring that no exposition material would be included in the test set.

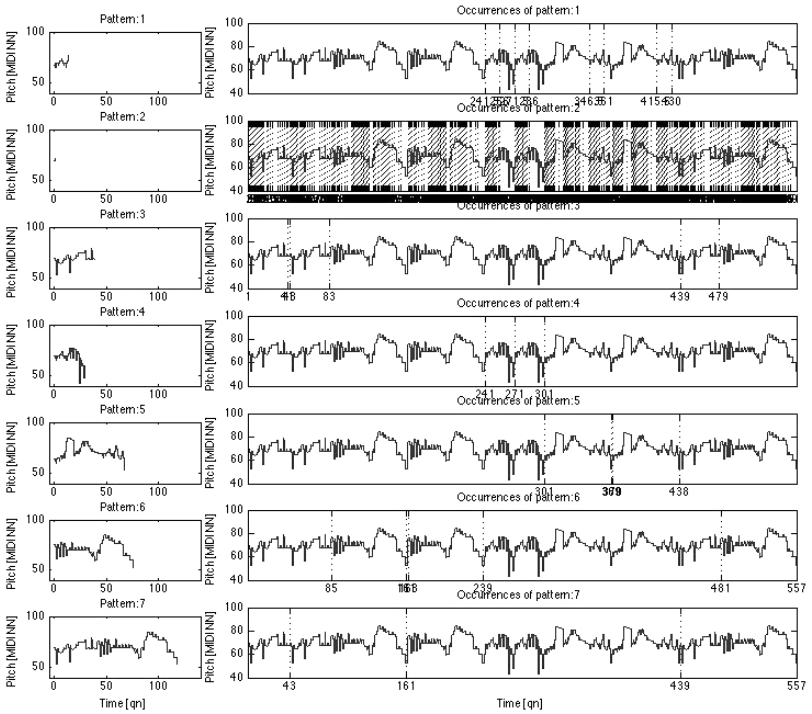


Fig. 12.12 Patterns discovered by the method for the third movement of Beethoven's Piano Sonata in F minor, Op. 2, No. 1 (1795), JKU PDD monophonic version. Pitch signal representation, with signals shifted to start at time 0. Plots on the left correspond to the patterns, while plots on the right correspond to the entire piece, with each pattern occurrence marked with a vertical dotted line at its starting and ending position

We were also interested in investigating the amount of initial expository material required to enable the parent works of excerpts to be accurately identified. We therefore constructed classifiers from the first 4, 8 and 16 qn of the pieces.

Figure 12.13 shows schematically how the classifiers and the test sets were constructed. The classifier set C was built from segments $sc_{i,j}$ from the expositions of the 15 *Inventions*, where each segment could be from either the upper or the lower voice. $sc_{i,j}$ is the jth segment in Invention i. Each test set T was built from segments st, where each st could be from either the upper or the lower voice. We denote the jth segment in Invention i by $st_{i,j}$. To classify a segment st to one of the 15 classes, we applied 1-nearest neighbour classification (Mitchell, 1997). That is, we computed the distances between st and all sc in C, and classified st to the class i of the $sc_{i,j}$ that had the smallest distance to st. Each test excerpt was assigned the class most frequently

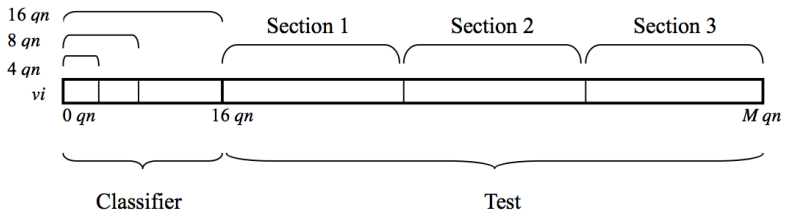


Fig. 12.13 Scheme of classifier and test construction based on signal v_i

predicted by its segments. In both cases we used the next nearest neighbour to break ties.

We expected higher classification rates with classifiers built from more exposition material, similar performance for the different combinations of wavelet-base classifiers, and higher classification rates in the first section compared to the following two, as the subject appears in the first section following the exposition at least once in each part (Stein, 1979).

The following parameters were used in the experiment:

- Sampling rate: 8 samples per qn[8]
- Representation: normalized pitch signal (WR), wavelet coefficients filtered at the scale of 1 qn (WR) and absolute wavelet coefficients filtered at the scale of 1 qn (WRA)
- Segmentation: constant-duration segmentation (CS), wavelet zero-crossing (ZC) and wavelet absolute maxima (AM)
- Scale segmentation at 1 qn
- Segment length normalization by zero padding
- Clustering: 1-nearest neighbour
- Distance measure: city block

12.3.2.1 Results

Figures 12.14 and 12.15 show the classification accuracy on each section, with the concatenation phase omitted and included, respectively. Both figures show the effect of segmentation and representation (columns vs. rows), and the number of qn used for the classifiers (asterisk, square, and circle markers). As expected, the amount of material used from the exposition (4, 8, or 16 qn) affects the classification success rates: the more material used, the higher the success rates. Moreover, segmentation has a stronger effect on the classification than representation. With respect to the results between sections, the classification rates for the first section are higher than

[8] The sampling rate was chosen to be the same as that used by Velarde et al. (2013).

those for the second and third sections. Representations associated with constant-duration segmentation are accurate in the first section after the exposition, where the subject is presented at least once in one of the voices (Stein, 1979), but far less accurate in the second and third sections where an increasing degree of variation of the original material occurs. Also, in sections 2 and 3, segment boundaries may not fall on whole-quarter-note time points, instead they may be shifted by a small amount, as an effect of the equal division of the sections. This may result in poor discriminatory information contained in segments when using constant-duration segmentation. The approach based on wavelet representation and segmentation is more robust to variation compared to constant-duration segmentation and the unfiltered pitch signal, resulting in similar classification rates for each classifier among all three sections.

A Wilcoxon signed rank test indicated that the concatenation phase did not significantly affect the results of accuracy per segmentation method (CS: $Z = 1.6036$, $p = 0.1088$, ZC: $Z = 0.4472$, $p = 0.6547$, AM: $Z = 1.6036$, $p = 0.1088$) or accuracy per representation type (VR: $Z = 1.4142$, $p = 0.1573$, WR: $Z = 1.6036$, $p = 0.1088$,

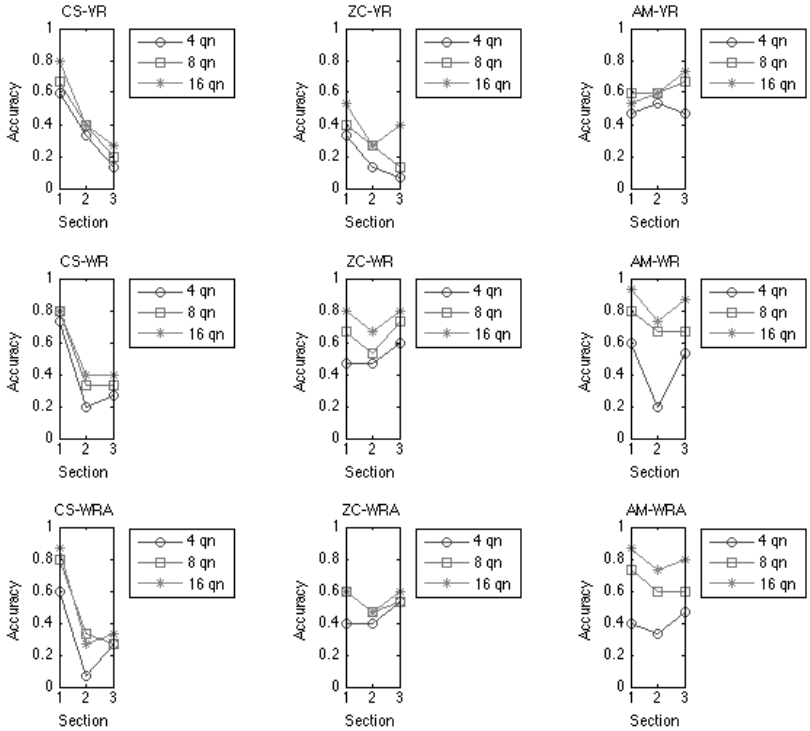


Fig. 12.14 Performance for each section with the classifier based on the exposition

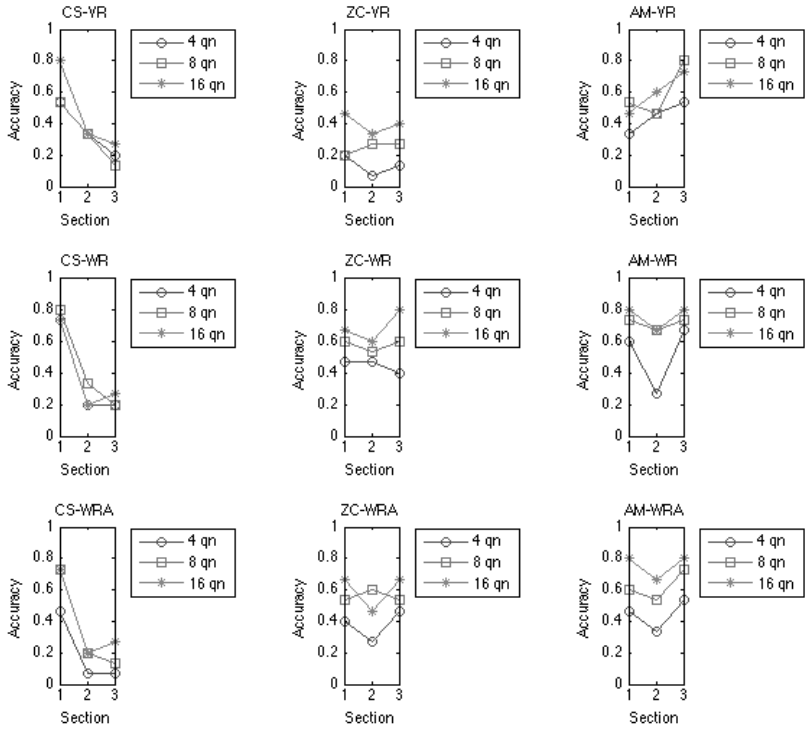


Fig. 12.15 Performance for each section with the classifier based on the exposition, and the concatenation phase included in the segmentation process

WA: $Z = 0.8165$, $p = 0.4142$) for classifiers built from the first 16 qn. However, while including the concatenation phase did not significantly affect the results, it slightly reduced the mean accuracy by 4%. We speculate that this may be a result of the concatenation phase causing some test-set segments to become much longer than the classifier segments, which would lead to segments of very unequal length being measured for similarity. This, in turn, could result in poorer classification accuracies.

12.4 Summary and Conclusions

We have presented a novel computational method for analysis and pattern discovery in melodies and monophonic voices. The method was evaluated on two musicological tasks. In the first task, the method was used to automatically discover themes and sections in the JKU Patterns Development Database. In the second task, the method

was used to determine the parent composition of excerpts from J. S. Bach's Two-Part Inventions (BWV 772–786). We explored aspects of representation, segmentation, classification and ranking of melodic units. The results of the experiments led us to conclude that the combination of constant-duration segmentation and an unfiltered, "raw", pitch-signal representation is a powerful approach for pieces where motivic and thematic material is restated with only slight variation. However, when motivic material is more extensively varied, the wavelet-based approach proves more robust to melodic variation.

The method described in this chapter could be developed further, perhaps by evaluating the quality of clusters in order to discard clusters that are too heterogeneous. Other measures of pattern quality could also be explored for ranking patterns in the algorithm output, including measures that perhaps more precisely model human perception and cognition of musical patterns. Moreover, it would be interesting to study the method's performance on a corpus of human performances of the pieces in experiment 1, in order to test, in particular, the robustness of our distance measures.

Acknowledgements Gissel Velarde is supported by the Department of Architecture, Design and Media Technology at Aalborg University. The contribution of David Meredith to the work reported here was made as part of the "Learning to Create" project (Lrn2Cre8). The project Lrn2Cre8 acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 610859.

References

Adiloglu, K., Noll, T., and Obermayer, K. (2006). A paradigmatic approach to extract the melodic structure of a musical piece. *Journal of New Music Research*, 35(3):221–236.

Anagnostopoulou, C. and Westermann, G. (1997). Classification in music: A computational model for paradigmatic analysis. In *Proceedings of the International Computer Music Conference*, pages 125–128, Thessaloniki, Greece.

Antoine, J.-P. (1999). Wavelet analysis: a new tool in physics. In van den Berg, J. C., editor, *Wavelets in Physics*. Cambridge University Press.

Aucouturier, J.-J. and Sandler, M. (2002). Finding repeating patterns in acoustic musical signals: Applications for audio thumbnailing. In *Audio Engineering Society 22nd International Conference on Virtual, Synthetic, and Entertainment Audio (AES22)*, Espoo, Finland.

Cambouropoulos, E. (1997). Musical rhythm: A formal model for determining local boundaries, accents and metre in a melodic surface. In Leman, M., editor, *Music, Gestalt, and Computing*, volume 1317 of *Lecture Notes in Artificial Intelligence*, pages 277–293. Springer.

Cambouropoulos, E. (1998). *Towards a general computational theory of musical structure*. PhD thesis, University of Edinburgh.

Cambouropoulos, E. (2001). The local boundary detection model (LBDM) and its application in the study of expressive timing. In *Proceedings of the International Computer Music Conference (ICMC'2001)*, Havana, Cuba.

Cambouropoulos, E. and Widmer, G. (2000). Automated motivic analysis via melodic clustering. *Journal of New Music Research*, 29(4):303–317.

Collins, T. (2014). MIREX 2014 Competition: Discovery of Repeated Themes and Sections. http://tinyurl.com/krnqzn5. Accessed on 9 April 2015.

Collins, T., Laney, R., Willis, A., and Garthwaite, P. H. (2011). Modeling pattern importance in Chopin's Mazurkas. *Music Perception*, 28(4):387–414.

Conklin, D. (2006). Melodic analysis with segment classes. *Machine Learning*, 65(2-3):349–360.

Conklin, D. and Anagnostopoulou, C. (2006). Segmental pattern discovery in music. *INFORMS Journal on computing*, 18(3):285–293.

Dreyfus, L. (1996). *Bach and the Patterns of Invention*. Harvard University Press.

Eerola, T. and Toiviainen, P. (2004). MIDI Toolbox: MATLAB tools for music research. Available online at http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/miditoolbox/.

Everitt, B., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis*. Wiley Series in Probability and Statistics. Wiley.

Farge, M. (1992). Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics*, 24(1):395–458.

Florek, K., Łukaszewicz, J., Perkal, J., Steinhaus, H., and Zubrzycki, S. (1951). Sur la liaison et la division des points d'un ensemble fini. *Colloquium Mathematicae*, 2(3–4):282–285.

Forth, J. (2012). *Cognitively-motivated geometric methods of pattern discovery and models of similarity in music*. PhD thesis, Goldsmiths College, University of London.

Grilo, C. F. A., Machado, F., and Cardoso, F. A. B. (2001). Paradigmatic analysis using genetic programming. In *Artificial Intelligence and Simulation of Behaviour (AISB 2001)*, York, UK.

Haar, A. (1910). Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69(3):331–371.

Höthker, K., Hörnel, D., and Anagnostopoulou, C. (2001). Investigating the influence of representations and algorithms in music classification. *Computers and the Humanities*, 35(1):65–79.

Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106.

Huron, D. (1996). The melodic arch in Western folksongs. *Computing in Musicology*, 10:3–23.

Janssen, B., De Haas, W. B., Volk, A., and Van Kranenburg, P. (2013). Discovering repeated patterns in music: state of knowledge, challenges, perspectives. In *Proceedings of the 10th International Symposium on Computer Music Multidisciplinary Research (CMMR 2010)*, Marseille, France.

Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254.

Kay, K. N., Naselaris, T., Prenger, R. J., and Gallant, J. L. (2008). Identifying natural images from human brain activity. *Nature*, 452(7185):352–355.

Kurby, C. A. and Zacks, J. M. (2008). Segmentation in the perception and memory of events. *Trends in Cognitive Sciences*, 12(2):72–79.

Lamont, A. and Dibben, N. (2001). Motivic structure and the perception of similarity. *Music Perception*, 18(3):245–274.

Lartillot, O. (2005). Efficient extraction of closed motivic patterns in multi-dimensional symbolic representations of music. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 191–198, London, UK. Available online at <http://ismir2005.ismir.net/proceedings/1082.pdf>.

Lartillot, O. (2014). PatMinr: In-depth motivic analysis of symbolic monophonic sequences. In *Music Information Retrieval Evaluation Exchange (MIREX 2014), Competition on Discovery of Repeated Themes and Sections*.

Lerdahl, F. and Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. MIT Press.

Mallat, S. (2009). *A Wavelet Tour of Signal Processing: The Sparse Way*. Academic Press, 3rd edition.

Mallat, S. and Hwang, W. L. (1992). Singularity detection and processing with wavelets. *Information Theory, IEEE Transactions on*, 38(2):617–643.

Marx, A. B. (1837). *Die Lehre von der musikalischen Komposition: praktisch-theoretisch*, volume 1. Breitkopf and Härtel.

Mazzola, G. et al. (2002). *The Topos of Music*. Birkhäuser.

McGettrick, P. (1997). *MIDIMatch: Musical pattern matching in real time*. PhD thesis, MSc. Dissertation, York University, UK.

Meredith, D. (2006). Point-set algorithms for pattern discovery and pattern matching in music. In *Proceedings of the Dagstuhl Seminar on Content-based Retrieval (No. 06171, 23–28 April, 2006)*, Schloss Dagstuhl, Germany. Available online at http://drops.dagstuhl.de/opus/volltexte/2006/652.

Meredith, D. (2013). COSIATEC and SIATECCompress: Pattern discovery by geometric compression. In *Music Information Retrieval Evaluation Exchange (MIREX)*, Curitiba, Brazil.

Meredith, D. (2015). Music analysis and point-set compression. *Journal of New Music Research*, 44(3). In press.

Meredith, D., Lemström, K., and Wiggins, G. A. (2002). Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4):321–345.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill.

Monelle, R. (1992). *Linguistics and Semiotics in Music*. Harwood Academic.

Müllensiefen, D. and Wiggins, G. (2011a). Polynomial functions as a representation of melodic phrase contour. In Schneider, A. and von Ruschkowski, A., editors, *Systematic Musicology: Empirical and Theoretical Studies*, volume 28 of *Hamburger Jahrbuch für Musikwissenschaft*. Peter Lang.

Müllensiefen, D. and Wiggins, G. A. (2011b). Sloboda and Parker's recall paradigm for melodic memory: a new, computational perspective. In Deliége, I. and Davidson, J. W., editors, *Music and the Mind: Essays in Honour of John Sloboda*, pages 161–188. Oxford University Press.

Müller, M. (2007). *Information Retrieval for Music and Motion*, volume 2. Springer.

Muzy, J., Bacry, E., and Arneodo, A. (1991). Wavelets and multifractal formalism for singular signals: application to turbulence data. *Physical Review Letters*, 67(25):3515.

Nattiez, J.-J. (1975). *Fondements d'une sémiologie de la musique*. Union Générale d'Éditions.

Nattiez, J.-J. (1986). La sémiologie musicale dix ans après. *Analyse musicale*, 2:22–33.

Nieto, O. and Farbood, M. (2013). Mirex 2013: Discovering musical patterns using audio structural segmentation techniques. In *Music Information Retrieval Evaluation eXchange (MIREX 2013)*, Curitiba, Brazil.

Nieto, O. and Farbood, M. M. (2014). Mirex 2014 entry: Music segmentation techniques and greedy path finder algorithm to discover musical patterns. In *Music Information Retrieval Evaluation Exchange (MIREX 2014)*, Taipei, Taiwan.

Pinto, A. (2009). Indexing melodic sequences via wavelet transform. In *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, pages 882–885. IEEE.

Reicha, A. (1814). *Traité de mélodie*. Chez l'auteur.

Riemann, H. (1912). *Handbuch der Phrasierung*. Hesse.

Ruwet, N. (1966). Méthodes d'analyses en musicologie. *Revue belge de musicologie*, 20(1/4):65–90.

Schenker, H. (1935). *Der freie Satz*. Universal Edition. (Published in English as E. Oster (trans., ed.) *Free Composition*, Longman, New York, 1979.).

Schmuckler, M. A. (1999). Testing models of melodic contour similarity. *Music Perception*, 16(3):295–326.

Schoenberg, A. (1967). *Fundamentals of Musical Composition*. Faber.

Smith, L. M. and Honing, H. (2008). Time–frequency representation of musical rhythm by continuous wavelets. *Journal of Mathematics and Music*, 2(2):81–97.

Sneath, P. H. (1957). The application of computers to taxonomy. *Journal of General Microbiology*, 17(1):201–226.

Stein, L. (1979). *Structure & style: the study and analysis of musical forms*. Summy-Birchard Company.

Tenney, J. and Polansky, L. (1980). Temporal gestalt perception in music. *Journal of Music Theory*, 24(2):205–241.

Torrence, C. and Compo, G. P. (1998). A practical guide to wavelet analysis. *Bulletin of the American Meteorological society*, 79(1):61–78.

Urbano, J. (2013). Mirex 2013 symbolic melodic similarity: A geometric model supported with hybrid sequence alignment. In *Music Information Retrieval Evaluation Exchange (MIREX 2013)*, Curitiba, Brazil.

van Kranenburg, P., Volk, A., and Wiering, F. (2013). A comparison between global and local features for computational classification of folk song melodies. *Journal of New Music Research*, 42(1):1–18.

Velarde, G. and Meredith, D. (2014). A wavelet-based approach to the discovery of themes and sections in monophonic melodies. In *Music Information Retrieval Evaluation Exchange (MIREX 2014)*, Taipei, Taiwan.

Velarde, G., Weyde, T., and Meredith, D. (2013). An approach to melodic segmentation and classification based on filtering with the Haar-wavelet. *Journal of New Music Research*, 42(4):325–345.

Weyde, T. (2001). Grouping, similarity and the recognition of rhythmic structure. In *Proceedings of the International Computer Music Conference (ICMC)*, Havana, Cuba.

Weyde, T. (2002). Integrating segmentation and similarity in melodic analysis. In *Proceedings of the International Conference on Music Perception and Cognition*, pages 240–243, Sydney, Australia.

Woody, N. A. and Brown, S. D. (2007). Selecting wavelet transform scales for multivariate classification. *Journal of Chemometrics*, 21(7-9):357–363.

Paper V. Composer Recognition based on 2D-Filtered Piano-Rolls.

Composer Recognition based on 2D-Filtered Piano-Rolls

Velarde, Gissel; Weyde, Tillman; Cancino Chacón, Carlos; Meredith, David; Grachten, Maarten

Published in:
Proceedings of the 17th International Society for Music Information Retrieval Conference

Publication date:
2016

Document Version
Accepted manuscript, peer reviewed version

COMPOSER RECOGNITION BASED ON 2D-FILTERED PIANO-ROLLS

Gissel Velarde[1] **Tillman Weyde**[2] **Carlos Cancino Chacón**[3]
David Meredith[1] **Maarten Grachten**[3]
[1] Department of Architecture Design & Media Technology, Aalborg University, Denmark
[2] Department of Computer Science, City University London, UK
[3] Austrian Research Institute for Artificial Intelligence, Austria
{gv, dave}@create.aau.dk, t.e.weyde@city.ac.uk, {carlos.cancino, maarten.grachten}@ofai.at

ABSTRACT

We propose a method for music classification based on the use of convolutional models on symbolic pitch–time representations (i.e. piano-rolls) which we apply to composer recognition. An excerpt of a piece to be classified is first sampled to a 2D pitch–time representation which is then subjected to various transformations, including convolution with predefined filters (Morlet or Gaussian) and classified by means of support vector machines. We combine classifiers based on different pitch representations (MIDI and morphetic pitch) and different filter types and configurations. The method does not require parsing of the music into separate voices, or extraction of any other predefined features prior to processing; instead it is based on the analysis of texture in a 2D pitch–time representation. We show that filtering significantly improves recognition and that the method proves robust to encoding, transposition and amount of information. On discriminating between Haydn and Mozart string quartet movements, our best classifier reaches state-of-the-art performance in leave-one-out cross validation.

1. INTRODUCTION

Music classification has occupied an important role in the music information retrieval (MIR) community, as it can immediately lead to musicologically interesting findings and methods, whilst also being immediately applicable in, for example, recommendation systems, music database indexing, music generation and as an aid in resolving issues of spurious authorship attribution.

Composer recognition, one of the classification tasks addressing musical style discrimination (among genre, period, origin identification, etc.), has aroused more attention in the audio than in the symbolic domain [13]. Particularly in the symbolic domain, the string quartets by Haydn and Mozart have been repeatedly studied [10, 12, 13, 24], since discriminating between Haydn and Mozart has been found to be a particularly challenging composer recognition task [24].

In this study, we propose a novel method and evaluate it on the classification of the string quartet movements by Haydn and Mozart. The method is based on the use of convolutional models on symbolic pitch–time representations (i.e. piano-rolls). An excerpt of a piece to be classified is first sampled to a 2D pitch–time representation which is then subjected to various transformations, including convolution with predefined filters (Morlet or Gaussian) and classified by means of Support Vector Machines (SVM).

2. RELATED WORK

Typically it is seen that computational methods use some kind of preprocessing to extract melody and harmony. Previous computational methods addressing composer discrimination of polyphonic works required defining sets of musical features or style makers, and/or relied on the encoding of separate parts or voices [10,12,13,24]. However, hard-coded musical features require musical expertise and may not perform similarly on different datasets [24], while the performance of methods relying on separate encoding of voice parts could be affected if voices are not encoded separately or even be unusable.

In order to avoid the requirements of previous methods, we aim to develop a more general approach studying the texture of pitch–time representations (i.e. piano-rolls) in the two-dimensional space. Previous studies did not address musical texture as it is proposed here.

Next, we review previous work that employs 2D music representations (2.1), and briefly sketch the background of the use of convolutional methods for machine perception and classification (2.2).

2.1 Representing music with 2D images

Visually motivated features generated from spectrograms have been successfully used for music classification (see [5, 28]). This success may be partly due to the fact that similar principles of perceptual organization operate in both vision and hearing [8]. The Gestalt principles of proximity, similarity and good continuation, originally developed to account for perceptual organization in vision, have also been used to explain the way that listeners organize

sonic events into streams and chunks [3, 7, 16]. Moreover, other studies suggest direct interaction between visual and auditory processing in common neural substrates of the human brain, which effectively integrates these modalities in order to establish robust representations of the world [9, 11, 21].

Graphical notation systems have been used since ancient times to transmit musical information [27]. Moreover, most Western music composed before the age of recording survives today only because of transmission by graphical notation — as staff notation, tablature, neumatic notation, etc. Standard graphical musical notation methods have proved to be extremely efficient and intuitive, possibly in part due to the natural mapping of pitch and time onto two orthogonal spatial dimensions.

2.2 Convolutional models

Convolutional models have been used extensively to model the physiology and neurology of visual perception. For example, in 1980, Daugman [6] and Marčelja [17] modeled receptive field profiles in cortical simple cells with parametrized 2D Gabor filters. In 1987, Jones and Palmer [14] showed that receptive-field profiles of simple cells in the visual cortex of a cat are well described by the real parts of complex 2D Gabor filters. More recently, Kay et al. [15] used a model based on Gabor filters to identify natural images from human brain activity. In our context, the Gabor filter is equivalent to the Morlet wavelet which we have used as a filter in the experiments described below.

Filters perform tasks like contrast enhancement or edge detection. In image classification, filtering is combined with classification algorithms such as SVM or neural networks for object or texture recognition [2, 23].

In the remainder of this paper, we present our proposed method in detail (3). Then, we report the results of our experiments (4) and finally, state our conclusions (5).

3. METHOD

Figure 1 provides an overview of our proposed method. As input, the method is presented with excerpts from pieces of music in symbolic format. Then, in the *sampling* phase, a 2D image is derived from each input file in the form of a piano-roll. After the *sampling* phase, various *tranformations* are applied to the images before carrying out the final *classification* phase, which generates a class label for the input file using an SVM. Details of each phase are given below. We begin by describing the *sampling* phase, in which symbolic music files are transformed into images of piano-rolls.

3.1 Sampling piano-roll images from symbolic representations

3.1.1 MIDI note numbers encoding

Symbolic representations of music (e.g. MIDI files) encode each note's pitch, onset and duration. We encoded pitch as an integer from 1 to 128 using MIDI note numbers (MNN), where C4 or *middle C* is mapped to MNN 60. Onset and duration are temporal attributes measured in quarter notes (qn).

3.1.2 Morphetic pitch encoding

The pitch name of a note is of the form <letter name><alteration><octave number>, e.g. C♯4. By removing the <alteration> and mapping all note names with the same <letter name> and <octave number> to the same number we reduce the space to *morphetic pitch*: an integer corresponding to the vertical position of the note on a musical staff.

We use a pitch-spelling algorithm by Meredith called *PS13s1* [18], to compute the pitch names of notes. The *PS13s1* algorithm has been shown to perform well on classical music of the type considered in this study. The settings of the *PS13s1* algorithm used here are the same as in [18], [1] with the *pre-context* parameter set to 10 notes and the *post-context* set to 42 notes. These parameters define a context window around the note to be spelt, which is used to compute the most likely pitch name for the note, based on the extent to which the context implies each possible key. When transposing a pattern within a major or minor scale (or, indeed, any scale in a diatonic mode), as is common practice in tonal (and modal) music, chromatic pitch intervals within the pattern change although the transposed pattern is still recognized by listeners as an instance of the same musical motif [8]. Morphetic pitch intervals are invariant to within-scale transpositions. We hypothesize that preserving this tonal motif identity might improve the performance of our models.

3.1.3 Piano-rolls (p_{70qn})

Symbolic representations of music are sampled to 2D binary images of size $P \times T$ pixels taking values of 0 or 1, called piano-roll representations. Our piano-roll representations are sampled from the first 70 qn of each piece, using onset in qn, duration in qn and either MNN or morphetic pitch, with a sampling rate of 8 samples per qn. We denote such representations by p_{70qn}. Each note of a piece symbolically encoded is described as an ordered tuple (onset, duration, pitch). The onsets are shifted, so that the first note starts at 0 qn. The piano-roll image is initialized with zeros and filled with ones for each sampled note. Its rows correspond to pitch and columns to samples in time. For each note, its onset and duration are multiplied by the sampling rate and rounded to the nearest integer. Note that since the tempo in terms of quarter notes per minute varies across pieces in our test corpora, the resulting samples vary in physical duration.

3.1.4 Piano-rolls (p_{400n})

As an alternative to the 70 qn piano roll excerpts, p_{70qn}, defined in 3.1.3 above, we also tested the methods on piano-roll excerpts consisting of the first 400 notes of each piece.

[1] We use a Java implementation of the *PS13s1* algorithm by David Meredith that takes MIDI files as input. **kern files are first converted to MIDI. Then we use the function *writemidi_seconds* by Christine Smit: `http://www.ee.columbia.edu/~csmit/midi/matlab/html/example_script1.html#2`

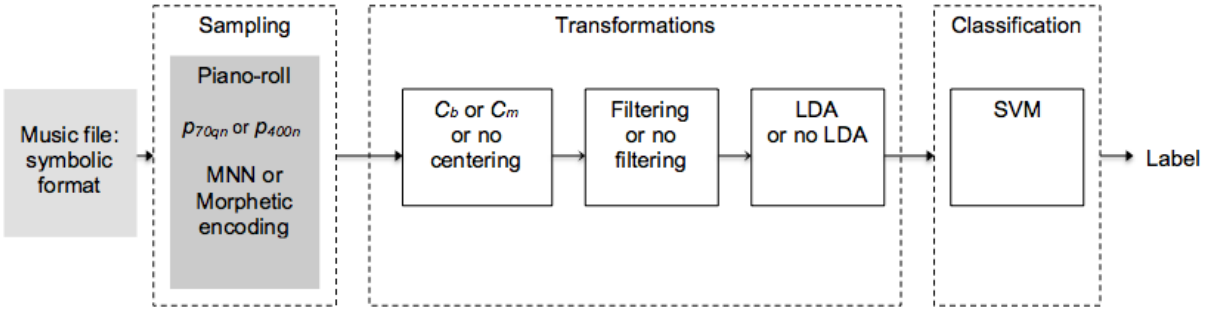


Figure 1. Overview of the method. Music, represented symbolically, is first sampled to 2D images of piano-rolls. Then, various transformations or processing steps are applied to the images, including convolution with predefined filters. The order of applying these transformations is from left to right. Finally, the images are classified with an SVM.

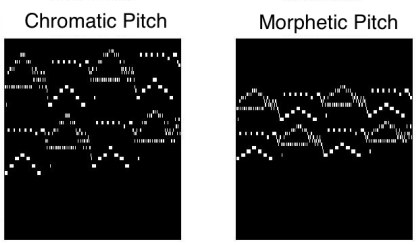


Figure 2. Piano-roll representation using MNN (left) and morphetic pitch (right) of the first 48 qn of Prelude 3 in C♯ major, BWV 848 by Bach. Note that the approximately similar inverted "V" shaped patterns in the left-hand figure are transformed into patterns of exactly the same shape in the right-hand figure.

We denote this type of representation by p_{400n}. These p_{400n} representations were produced by sampling in the way described in section 3.1.3, but using the first 400 notes instead of the first 70 qn of a piece and sampling to a size of $P \times T$ pixels. If a piece has fewer than 400 notes, all notes of the piece are represented. This representation is used to approximately normalize the amount of information per image.

In the next phase of our proposed method with a single classifier, as seen in Figure 1, various transformations or processing steps are applied which will be described as follows.

3.2 Transformations

We explore the effect of applying transformations or processing techniques to the piano-roll images. These transformations are applied in order to find a suitable normalization (i.e., alignment between the images) before classification, and to test the robustness of the method to transformations of the input data that would not be expected to reduce the performance of a human expert (cf. [22]). We now consider each of these transformations in turn.

3.2.1 Pitch range centering (C_b)

Typically, the pitch range of a piece in a piano-roll representation does not extend over the full range of possible MIDI note number values. We hypothesized that we could improve performance by transposing each piano roll so that its pitch range is centered vertically in its image. That is, for a piano-roll image of size $P \times T$ pixels, we translated the image by $y_s = (P - (y_d + y_u))/2$ pixels vertically, where y_d and y_u are the lower and upper co-ordinates, respectively, of the bounding box of the piano roll (i.e., corresponding to the minimum and maximum pitches, respectively, occurring in the piano roll). This transformation is used to test robustness to pitch transposition.

3.2.2 Center of mass centering (C_m)

An image p of size $P \times T$ pixels is translated so that the centroid of the piano roll occurs at the center of the image. We denote the centroid by $(\bar{x}, \bar{y}) = (M_{10}/M_{00}, M_{01}/M_{00})$, where $M_{ij} = \sum_x \sum_y x^i y^j p(x, y)$. The elements of the image are shifted circularly to the central coordinates (x_c, y_c) of the image, where $(x_c = T/2)$ and $(y_c = P/2)$, an amount of $(x_c - \bar{x})$ pixels on the x-axis, and $(y_c - \bar{y})$ pixels on the y-axis. In this case, circular shift is applied to rows and columns of p. In the datasets used for the experiments, in 5% of the pieces with MNN encoding, one low-pitch note was shifted down by this transformation and wrapped around so that it became a high-pitched note (in one piece there were four low-pitch notes shifted to high pitch-notes after circular shift). However, this transformation caused most pieces to be shifted and wrapped around in the time dimension so that, on average, approximately the initial 2 quarter notes of each representation were transferred to the end.

3.2.3 Linear Discriminant Analysis

We apply Linear Discriminant Analysis (LDA) [4] solving the singularity problem by Singular Value Decomposition and Tikhonov regularization to find a linear subspace for

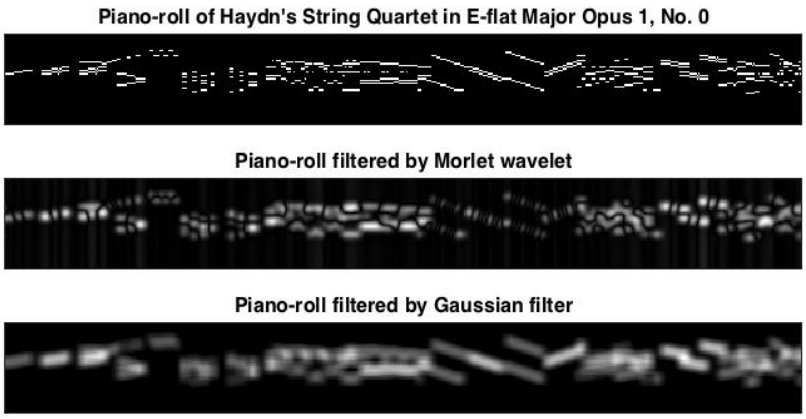


Figure 3. Piano-roll (p_{400n}) morphetic pitch representation (top) of Haydn's String Quartet in E-flat Major Opus 1, No. 0 and its transformations filtered by the Morlet wavelet at a scale of 2 pixels oriented of 90 degrees (second image), and by a Gaussian filter of size 9×9 pixels with $\sigma = 3$ (third image). p_{400n} and its filtered versions are each 56×560 pixels.

discrimination between classes. [2]

3.2.4 Filtering

Images are convolved with pre-defined filters (Morlet wavelet or a Gaussian filter). We apply the continuous wavelet transform (CWT) [1], with the Morlet wavelet ψ at fixed scale a and rotation angle θ

$$\psi_{a,\theta}(x,y) = a^{-1}\psi(a^{-1}r_{-\theta}(x,y)) \quad (1)$$

with rotation r_θ

$$r_\theta(x,y) = (x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta), 0 \leq \theta < 2\pi. \quad (2)$$

where

$$\psi(x,y) = e^{ik_0 y}e^{-\frac{1}{2}(\varepsilon^{-1}x^2+y^2)} \quad (3)$$

with frequency $k_0 = 6$ and $\varepsilon = 1$.

The filtered images are the absolute values of the real part of the wavelet coefficients. We test a defined set of scales and angles (see section 4). The selection of scale and angle of orientation are those that yield the best classification as in [25].

We also filter images with a rotationally symmetric Gaussian low-pass filter g:

$$g(x,y) = e^{\frac{-(x^2+y^2)}{2\sigma^2}} \quad (4)$$

where x and y are the distances from the origin in the horizontal and vertical axis, respectively.

We test a defined set of filter sizes h and σ values (see section 4). The selection of the size h of the filter and the value of σ are those that yield the best classification. As an example of the effect of filtering, Figure 3 shows the piano-roll image, p_{70qn} of Haydn's String Quartet in E-flat Major Opus 1, No. 0 and the filtered images obtained by the convolution with Morlet wavelet and Gaussian filter.

[2] We use Deng Cai's LDA implementation version 2.1: `http://www.cad.zju.edu.cn/home/dengcai/Data/DimensionReduction.html`.

3.3 Classification with support vector machines

For classification, we use SVM with the Sequential Minimal Optimization (SMO) method to build an optimal hyperplane that separates the training samples of each class using a linear kernel [19]. Samples are transformed images of size $P \times T$ if they are not reduced by LDA. If LDA is applied, samples are points in 1D. Each sample is normalized around its mean, and scaled to have unit standard deviation before training. The Karush–Kuhn–Tucker conditions for SMO are set to 0.001.

4. EXPERIMENTS

We used a set of movements from string quartets by Haydn and Mozart, two composers that seemed to have influenced each other on this musical form. Walthew [26] observes that "Mozart always acknowledged that it was from Haydn that he learnt how to write String Quartets" and, in his late string quartets, Haydn was directly influenced by Mozart.

Distinguishing between string quartet movements by Haydn and Mozart is a difficult task. Sapp and Liu [20] have run an online experiment to test human performance on this task and found, based on over 20000 responses, that non-experts perform only just above chance level, while self-declared experts achieve accuracies up to around 66%.

Classification accuracy—that is, the proportion of pieces in the test corpus correctly classified—has been the established evaluation measure for audio genre and composer classification since the MIREX 2005 competition [3] and also for symbolic representations [12, 13, 24].

In our experiments we used the same dataset as in [24] consisting of 54 string quartet movements by Haydn and 53 movements by Mozart, encoded as **kern files, [4] and

[3] See `http://www.music-ir.org/mirex/wiki/2005:Main_Page`.

[4] `http://www.music-cog.ohio-state.edu/Humdrum/representations/kern.html`

	Pitch–time representation	Morlet LDA	Gauss LDA	NF LDA	Morlet	Gauss	NF
Morphetic pitch	p_{70qn}	65.4	58.9	57.9	53.3	68.2	58.9
	$C_b(p_{70qn})$	65.4	60.7	47.7	57.9	63.6	51.4
	$C_m(p_{70qn})$	53.3	60.7	52.3	64.5	59.8	56.1
	p_{400n}	67.3	**80.4**	57.0	63.6	72.9	55.1
	$C_b(p_{400n})$	62.6	72.9	54.2	61.7	66.4	53.3
	$C_m(p_{400n})$	65.4	65.4	55.1	66.4	70.1	53.3
MNN	p_{70qn}	64.5	67.3	66.4	62.6	66.4	64.5
	$C_b(p_{70qn})$	**70.1**	61.7	63.6	67.3	61.7	61.7
	$C_m(p_{70qn})$	63.6	57.9	57.0	66.4	56.1	54.2
	p_{400n}	66.4	69.2	64.5	65.4	63.6	64.5
	$C_b(p_{400n})$	54.2	64.5	52.3	58.9	58.9	49.5
	$C_m(p_{400n})$	53.3	62.6	42.1	56.1	63.6	44.9

Table 1. Haydn and Mozart String Quartet classification accuracies in leave-one-out cross validation for different configurations of classifiers (NF = no filtering).

evaluated our method's classification accuracies in leave-one-out cross-validation as it was done in [24].

Table 1 shows the classification accuracies (mean values) obtained in leave-one-out cross-validation for images of size 56×560 pixels. The standard deviation values are not presented, as they are not informative. The standard deviation can be derived from the accuracy in this case (accuracy of binary classification in leave-one-out cross-validation). The filters of the classifiers were tuned according to their classification accuracy over the different pitch–time representations. The angle of orientation of the Morlet wavelet was set to 90 degrees. This orientation was chosen out of a selection of angles (0, 45, 90 and 135 degrees). The scale was set to 2 pixels, selected varying its value from 1 to 9 pixels. The Gaussian filter was tested with pixel sizes of 1 to 10 pixels, with values of σ ranging from 1 to 4 pixels. Gaussian filters were set to 9 pixels and $\sigma = 3$. The best classifier using MNN encoding corresponds to a classifier operating on pitch–time representation $C_b(p_{70qn})$, filtered by Morlet wavelet oriented 90 degrees at a scale of 2 pixels, and LDA reduction. The best classifier of all reaches state-of-the-art performance with an accuracy of 80.4%. This classifier corresponds to a pitch–time representation p_{400n} in morphetic pitch encoding, filtered by a Gaussian filter of size 9 pixels and $\sigma = 3$, and LDA reduction. It misclassified 12 movements by Haydn and 9 by Mozart. The misclassified movements (mov.) are shown in Table 2. Due to our model section, it could be that the results present some overfitting.

From the results in Table 1 we observe that filtering significantly improves recognition at 5% significance level (Wilcoxon rank sum $= 194.5$, $p = 0.0107$, $n = 12$, with Morlet wavelet), (Wilcoxon rank sum $= 203$, $p = 0.0024$, $n = 12$, with Gaussian filter), and it is not significantly different to filter with Morlet or Gaussian filters (Wilcoxon rank sum $= 133$, $p = 0.3384$, $n = 12$). On the other side, there is not sufficient evidence to conclude that LDA improves recognition (Wilcoxon rank sum $= 154$, $p = 0.8395$, $n = 12$).

Movements by Haydn	**Movements by Mozart**
Op 1, N. 0, mov. 4	K. 137, mov. 3
Op 1, N. 0, mov. 5	K. 159, mov. 3
Op 9, N. 3, mov. 1	K. 168, mov. 2
Op 20, N. 6, mov. 2	K. 168, mov. 3
Op 20, N. 6, mov. 4	K. 428, mov. 3
Op 50, N. 1, mov. 3	K. 465, mov. 2
Op 64, N. 1, mov. 2	K. 465, mov. 4
Op 64, N. 4, mov. 2	K. 499, mov. 1
Op 64, N. 4, mov. 3	K. 499, mov. 4
Op 71, N. 2, mov. 2	
Op 103, mov. 1	
Op 103, mov. 2	

Table 2. Misclassified movements of our best classifier.

We study the effect of encoding (MNN vs. morphetic pitch), transposition (not centering vs. centering with C_b) and the amount of information (p_{70qn} vs. P_{400n}). The center of mass centering C_m was not evaluated, as this transformation may affect human recognition. Considering all results in Table 1 obtained with filtering and excluding the ones obtained with C_m, the difference in encoding between MNN and morphetic pitch is not significant at %5 significance level (Wilcoxon rank sum $= 269.5$, $p = 0.8502$, $n = 16$), nor are the results significantly different with or without centering C_b (Wilcoxon rank sum $= 311.5$, $p = 0.0758$, $n = 16$), neither it is significantly different to use p_{70qn} or P_{400n} (Wilcoxon rank sum $= 242$, $p = 0.4166$, $n = 16$). These findings suggest that the method based on 2D-Filtered piano-rolls is robust to transformations such as encoding, transposition, and amount of information that are considered not to affect human perception.

In Table 3, we list all previous studies where machine-learning methods have been applied to this Haydn/Mozart discrimination task. A direct comparison can be made between the classification accuracy achieved by the method of van Kranenburg and Backer [24] and our proposed method, as we used the same dataset. The datasets used by the other approaches in Table 3 were not available for us to test our method and make direct comparisons. Hontanilla et al. [13] used a subset of the set used in [24]: 49 string quartets movements by Haydn and 46 string quartets movements by Mozart [13]. Hillewaere et al. [12] extended van Kranenburg and Backer's [24] dataset to almost double its size, including several movements from the period 1770–1790. Herlands et al. [10] used a dataset consisting of MIDI encodings of only the first movements of the string quartets.

Table 3 shows that our best classifier reaches state-of-the-art performance and that there is no significant dif-

Method	Accuracy
Proposed best classifier	**80.4**
Van Kranenburg and Backer (2004) [24]	79.4
Herlands et al. (2014) [10]*	80.0
Hillewaere et al. (2010) [12]*	75.4
Hontanilla et al. (2013) [13]*	74.7

Table 3. Classification accuracies achieved by previous computational approaches on the Haydn/Mozart discrimination task. * indicates that a different dataset was used from that used in the experiments reported here.

ference from the results obtained by van Kranenburg and Backer at 5% significance level (Wilcoxon rank sum $= 11449$, $p = 0.8661$, $n = 107$). Compared to previous approaches [10,12,13,24], our method is more general in that it does not need hard-coded musical style markers for each dataset as in [24], nor does it require global musical feature sets as in [12], nor does it depend on the music having been parsed into separate parts or voices as in [10, 12, 13].

5. CONCLUSION

We have shown that string quartets by Haydn and Mozart can be discriminated by representing pieces of music as 2-D images of their pitch–time structure and then using convolutional models to operate on these images for classification. Our approach based on classifying pitch–time representations of music does not require parsing of the music into separate voices, or extraction of any other predefined features prior to processing. It addresses musical texture of 2-D pitch–time representations in a more general form. We have shown that filtering significantly improves recognition and that the method proves robust to encoding, transposition and amount of information. Our best single classifier reaches state-of-the-art performance in leave-one-out cross validation on the task of discriminating between string quartet movements by Haydn and Mozart.

With the proposed method, it is possible to generate a wide variety of classifiers. In preliminary experiments, we have seen that diverse configurations of classifiers (i.e. different filter types, orientations, centering, etc.) seem to provide complementary information which could be potentially used to build ensembles of classifiers improving classification further. Besides, we have observed that the method can be applied to synthetic audio files and audio recordings. In this case, audio files are sampled to spectrograms instead of piano-rolls, and then follow the method's chain of transformations, filtering and classification. We are optimistic that our proposed method can perform similarly on symbolic and audio data, and might be used successfully for other style discrimination tasks such as genre, period, origin, or performer recognition.

6. ACKNOWLEDGMENTS

The work for this paper carried out by G. Velarde, C. Cancino Chacón, D. Meredith, and M. Grachten was done as part of the EC-funded collaborative project, "Learning to Create" (Lrn2Cre8). The project Lrn2Cre8 acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 610859. G. Velarde is also supported by a PhD fellowship from the Department of Architecture, Design and Media Technology, Aalborg University. The authors would like to thank Peter van Kranenburg for sharing with us the string quartets dataset and results that allowed as statistical tests, William Herlands and Yoel Greenberg for supporting the unsuccessful attempt to reconstruct the dataset used in their research, Jordi Gonzalez for comments and suggestions on an early draft of this paper, and the anonymous reviewers for their detailed insight on this work.

7. REFERENCES

[1] J-P Antoine, Pierre Carrette, R Murenzi, and Bernard Piette. Image analysis with two-dimensional continuous wavelet transform. *Signal Processing*, 31(3):241–272, 1993.

[2] Yoshua Bengio, Aaron Courville, and Pierre Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828, 2013.

[3] Albert S. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, Cambridge, MA., 1990.

[4] Deng Cai, Xiaofei He, Yuxiao Hu, Jiawei Han, and T. Huang. Learning a spatially smooth subspace for face recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–7, June 2007.

[5] Yandre MG Costa, LS Oliveira, Alessandro L Koerich, Fabien Gouyon, and JG Martins. Music genre classification using lbp textural features. *Signal Processing*, 92(11):2723–2737, 2012.

[6] John G Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20(10):847–856, 1980.

[7] Diana Deutsch. Grouping mechanisms in music. In Diana Deutsch, editor, *The Psychology of Music*, pages 299–348. Academic Press, San Diego, 2nd edition, 1999.

[8] Diana Deutsch. *Psychology of Music*. Academic Press, San Diego, 3rd edition, 2013.

[9] Marc O Ernst and Heinrich H Bülthoff. Merging the senses into a robust percept. *Trends in Cognitive Sciences*, 8(4):162–169, 2004.

[10] William Herlands, Ricky Der, Yoel Greenberg, and Simon Levin. A machine learning approach to musically meaningful homogeneous style classification. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI)*, pages 276–282, 2014.

[11] Souta Hidaka, Wataru Teramoto, Yoichi Sugita, Yuko Manaka, Shuichi Sakamoto, Yôiti Suzuki, and Melissa Coleman. Auditory motion information drives visual motion perception. *PLoS One*, 6(3):e17499, 2011.

[12] Ruben Hillewaere, Bernard Manderick, and Darrell Conklin. String quartet classification with monophonic models. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, pages 537–542, Utrecht, The Netherlands, 2010.

[13] María Hontanilla, Carlos Pérez-Sancho, and Jose M Iñesta. Modeling musical style with language models for composer recognition. In *Pattern Recognition and Image Analysis*, pages 740–748. Springer, 2013.

[14] Judson P Jones and Larry A Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1233–1258, 1987.

[15] Kendrick N Kay, Thomas Naselaris, Ryan J Prenger, and Jack L Gallant. Identifying natural images from human brain activity. *Nature*, 452(7185):352–355, 2008.

[16] Fred Lerdahl and Ray S. Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, Cambridge, MA., 1983.

[17] S Marčelja. Mathematical description of the responses of simple cortical cells. *Journal of Neuophysiology*, 70(11):1297–1300, 1980.

[18] David Meredith. The *ps13* pitch spelling algorithm. *Journal of New Music Research*, 35(2):121–159, 2006.

[19] John C. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, January 1998.

[20] Craig Sapp and Yi-Wen Liu. The Haydn/Mozart String Quartet Quiz, 2015. `http://qq.themefinder.org` (Accessed 26 December 2015).

[21] Daniele Schön and Mireille Besson. Visually induced auditory expectancy in music reading: a behavioral and electrophysiological study. *Journal of Cognitive Neuroscience*, 17(4):694–705, 2005.

[22] Bob L Sturm. A simple method to determine if a music information retrieval system is a horse. *IEEE Transactions on Multimedia*, 16(6):1636–1644, 2014.

[23] Devis Tuia, Michele Volpi, Mauro Dalla Mura, Alain Rakotomamonjy, and Remi Flamary. Automatic feature learning for spatio-spectral image classification with sparse svm. *Geoscience and Remote Sensing, IEEE Transactions on*, 52(10):6062–6074, 2014.

[24] Peter Van Kranenburg and Eric Backer. Musical style recognition—a quantitative approach. In *Proceedings of the Conference on Interdisciplinary Musicology (CIM)*, pages 106–107, 2004.

[25] Gissel Velarde, Tillman Weyde, and David Meredith. An approach to melodic segmentation and classification based on filtering with the haar-wavelet. *Journal of New Music Research*, 42(4):325–345, 2013.

[26] Richard H Walthew. String quartets. *Proceedings of the Musical Association*, pages 145–162, 1915.

[27] Martin Litchfield West. The babylonian musical notation and the hurrian melodic texts. *Music & Letters*, pages 161–179, 1994.

[28] Ming-Ju Wu, Zhi-Sheng Chen, Jyh-Shing Roger Jang, Jia-Min Ren, Yi-Hsung Li, and Chun-Hung Lu. Combining visual and acoustic features for music genre classification. In *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, volume 2, pages 124–129. IEEE, 2011.

Paper VI. Convolution-based Classification of Audio and Symbolic Representations of Music

Velarde, Gissel; Cancino Chacón, Carlos; Meredith, David; Weyde, Tillman; Maarten Grachten. Submitted manuscript.

Convolution-based Classification of Audio and Symbolic Representations of Music

Gissel Velarde[1], Carlos Cancino Chacón[2], David Meredith*[1], Tillman Weyde[3] and Maarten Grachten[2]

[1]Aalborg University, Denmark
[2]The Austrian Research Institute for Artificial Intelligence, Austria
[3]City, University of London, United Kingdom

October 22, 2016

Abstract

We present a novel convolution-based method for classification of audio and symbolic representations of music. The method assigns music to a stylistic class. Pieces of music are first sampled to pitch–time representations (piano-rolls or spectrograms), and then processed with various techniques including convolution with a Gaussian filter, before being classified by a support vector machine or by k-nearest neighbours in an ensemble of classifiers. We have evaluated the proposed method on the well-studied task of discriminating between string quartet movements by Haydn and Mozart, obtaining state-of-the art results. Moreover, we performed experiments on *The Well-Tempered Clavier* by J. S. Bach to study the method's capacity to distinguish preludes from fugues. Our experimental results show that our proposed method performs equally well on both composer and genre classification, and on symbolic representations and synthetic audio, setting our method apart from most previous approaches that have been designed for use with either audio or symbolic data, but not both. Additionally, we tested a convolutional neural network on the task of discriminating between symbolic encodings of string quartet movements by Haydn and Mozart. The inspection of the filters learnt automatically by this network allows for an analysis from a more musicological point of view.

Index terms— Classification algorithms, composer classification, genre classification, convolution, filtering

*Correspondence: David Meredith, Aalborg University, Department of Architecture, Design and Media Technology, Rendsburggade 14, Building: 6-314, 9000 Aalborg, Denmark. Email: dave@create.aau.dk.

1 Introduction

We present a method for the automatic classification of musical styles. Methods modelling style recognition are of interest in music information retrieval for their applicability in, e.g., music indexing, recommendation systems, and music generation, as well as in systematic musicology where they can foster the understanding of music.

From the computational perspective taken in this study, style can be seen as a set of distinctive features shared among the instances of a style. Perceptually, style is a phenomenon that lets us characterize artists, genres, periods of composition, etc., on the basis of distinguishing salient features of works, despite variation or evolution over time (Paul & Kaufman, 2014; Rush & Sabers, 1981). A challenging style classification task for both humans and computers is the distinction between string quartet movements by Haydn and Mozart (Sapp & Liu, 2015; van Kranenburg & Backer, 2004). The computational methods proposed to date for classifying between these two composers have been applied to symbolic representations of music (Herlands, Der, Greenberg, & Levin, 2014; Hillewaere, Manderick, & Conklin, 2010; Hontanilla, Pérez-Sancho, & Iñesta, 2013; van Kranenburg & Backer, 2004; Velarde, Weyde, Cancino Chacón, Meredith, & Grachten, 2016). Most of these methods rely on features designed by experts, making them less general, and/or require each part or voice to be encoded separately. An exception is the model proposed by Velarde et al. (2016), which is based on classifying music from two-dimensional representations such as piano-rolls.

The method proposed by Velarde et al. (2016) learns to discriminate between classes of music by using filtered images of piano-roll excerpts to predict class labels, exploiting the images' textures. However, local structures on the level of motifs prove to be very important in melodic similarity (van Kranenburg, Volk, & Wiering, 2013), and melodic segmentation using small time-scales has been shown to improve recognition in parent work identification (Velarde, Weyde, & Meredith, 2013). We hypothesize that style recognition requires the use of both large- and small-scale feature extraction mechanisms. Locality is desired to detect musical patterns even if translated in time and pitch. Therefore, in this study we extend the method of Velarde et al. (2016), introducing music segmentation, and test the effect of chunking pitch–time representations into small segments of about 1 or 2 quarter notes for classification. Finally, we experiment with combining classification strategies in ensembles.

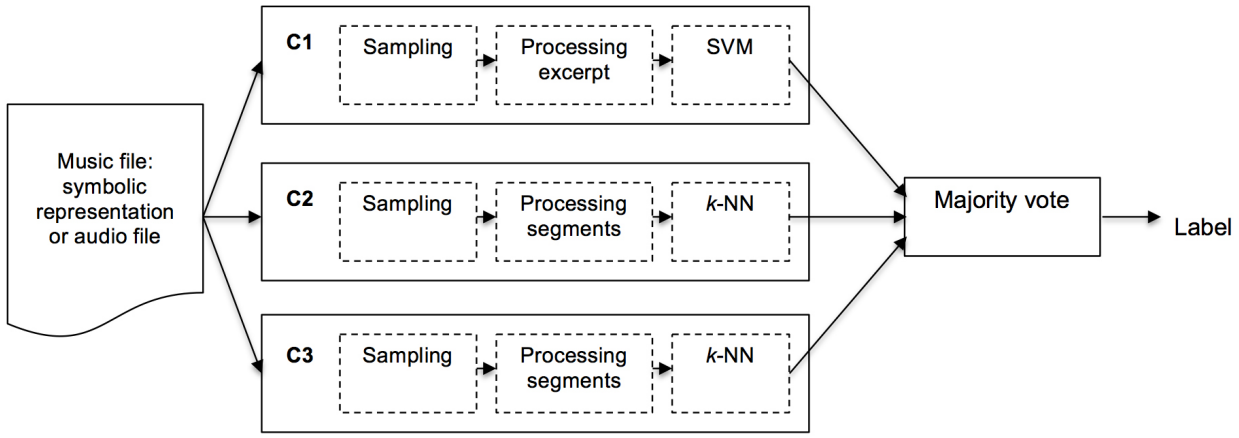


Figure 1: Diagram of the proposed method for music classification of symbolic or audio representations. The system receives a piece of music and outputs its computed class label. This system consists of an ensemble of three classifiers denoted by C1, C2 and C3, more specifically C1A, C2A and C3A for audio and C1S, C2S and C3S for symbolic music representation. Details on the configurations for each classifier are given in Table 1.

In this paper we make the following contributions:

- We propose a new classification method that can be applied to both audio recordings and symbolic representations of music.
- We report experimental results showing that our ensemble classifier produces state-of-the-art composer-identification results on two different datasets of the Haydn and Mozart string quartets.
- We report experimental results on discriminating between preludes and fugues from *The Well-Tempered Clavier* by J. S. Bach.
- We present a related classification approach based on Convolutional Neural Networks and an initial musicological interpretation of the features that have been automatically learnt.

2 Method

An overview of the proposed method is presented in the diagram in Figure 1. The system receives a piece of music as input and computes its class label as output. It consists of an ensemble of three classifiers, denoted by C1, C2 and C3. Depending on the input, we use audio-specific classifiers, henceforth denoted by C1A, C2A and C3A, or classifiers for symbolic music representations, denoted

by C1S, C2S and C3S. Each classifier consists of a sampling, a processing and a classification phase. The predictions of the three classifiers are ensembled by simple majority vote (Kuncheva, 2004) to predict the final class label.

Figure 2 shows in more detail the possible configurations of the individual classifiers. In each classifier, a piece of music is first sampled to a two dimensional (2D) piano-roll image if the input is a symbolic representation of music (e.g., MIDI file), or to a 2D magnitude spectrogram image if the input is an audio file (e.g., WAV file). After *sampling* this 2D image, either the *processing excerpt* or the *processing segments* phase follows. The main difference between the two processing phases is their output: the *processing excerpt* phase has one output per piece, while the *processing segments* phase has several outputs per piece. Finally, there is a *classification* phase employing a Support Vector Machine (SVM), a k-Nearest Neighbour (k-NN) algorithm or a Convolutional Neural Network (CNN). For pieces that follow the *processing segments* phase, the class label of a piece of music is the most frequently predicted class of its segments. Modules represented in Figure 2 by boxes with thick grey borders, are optional processing steps. Details of each phase are given below.

2.1 Sampling

A symbolic music encoding format is one that provides information similar to that given in a score and in which the atomic component is typically a note; whereas a PCM audio file represents the sound of a specific performance of a piece in terms of a sampled waveform. The *sampling* phase aims to prepare the input in such a way that music is similarly represented as a 2D pitch–time representation regardless of whether it is a symbolic encoding of a piece or an audio recording. Symbolic representations of music are sampled to piano-roll images, while audio files are sampled to spectrograms.

2.1.1 Piano-rolls

As described by Velarde et al. (2016), symbolic representations of music are sampled to piano-rolls, i.e., 2D binary images representing the pitch-time structures of music. We denote the height of such an image by P and its width by T. The piano-rolls are sampled using each note's pitch, onset, and duration. Onset and duration are encoded in quarter notes (qn). Chromatic pitch is represented by MIDI Note Number (MNN). MNN represents pitch as integer numbers from 0 to 127, C4 is mapped

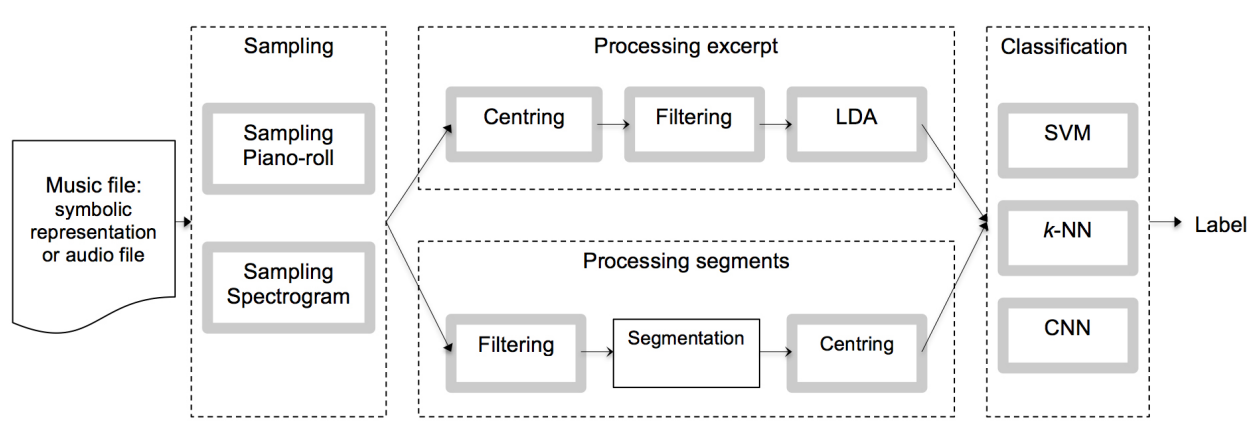


Figure 2: Diagram of the possible configurations of individual classifiers. An individual classifier receives a piece of music, which is first sampled, processed and finally classified. Modules represented by boxes with thick grey borders, are optional processing steps. In the *sampling* and *classification* phases, vertically aligned boxes are exclusive processing steps, such that only one module can be activated. If the sampled piece follows the *processing excerpt* phase, it will not follow the *processing segments* phase, and vice versa. In the *processing excerpt* phase all three modules are optional, while in the *processing segments* phase, the *segmentation* module is always activated.

to MNN 60. Alternatively, pitch is encoded as morphetic pitch (Meredith, 2006, p. 127), which is a function only of the vertical position of the note-head of a note on a staff and the clef in operation on the staff at the position where the note occurs. We compute morphetic pitch from MIDI files using a pitch spelling algorithm called *PS13s1* that requires parameters for defining a tonal context window around the note to be spelt (Meredith, 2006). The *pre-context* parameter is set to 10 notes and *post-context* is set to 42 notes, as these values performed best in Meredith's (2006) evaluation. These parameter values were also used by Velarde et al. (2016). Morphetic pitch intervals are within-scale transposition-invariant, while chromatic pitch intervals are not (cf. Velarde et al., 2016). The sampling rate for piano-rolls of full-length pieces, denoted p_{fl}, is 8 samples (i.e., pixels) per qn. Piano-rolls of excerpts are sampled with a sampling rate of 8 samples per qn from the first 70 qn of each piece. We denote a representation of this type by p_{70qn}. Alternatively, we use piano-rolls representing the first 400 notes of each piece, denoted by p_{400n}. p_{400n} piano-rolls are first sampled with a sampling rate of 8 samples per qn and then resized by nearest-neighbour interpolation (de Boor, 1978) to reach the size of $P \times T$ pixels. In this case, the sampling rate might vary for each image.

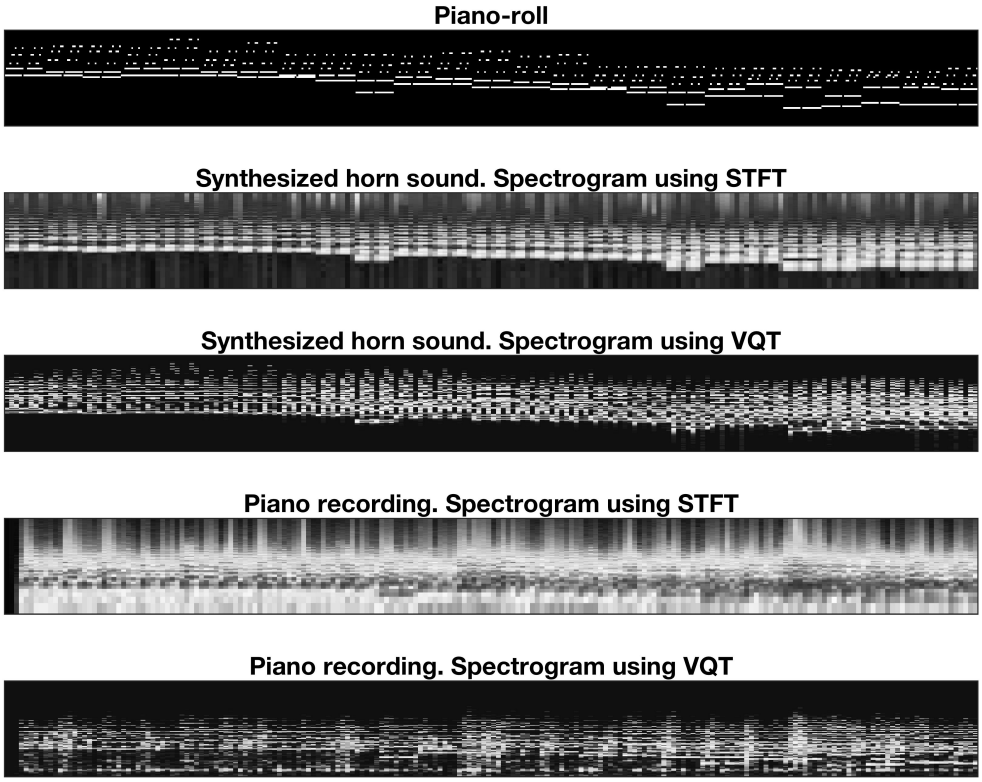


Figure 3: Pitch–time representations of an excerpt of the first 400 onsets of the Prelude in C major, BWV 846, from J. S. Bach's *Well-Tempered Clavier*. The upper image shows a piano-roll, while the second and third show spectrograms of a synthesized audio rendering using a horn sound. The fourth and fifth images show spectrograms of a piano recording by Kimiko Ishizaka.

2.1.2 Spectrograms

Spectrograms are used to present spectral information over time and have previously been used successfully for music classification (Costa, Oliveira, Koerich, Gouyon, & Martins, 2012; Wu et al., 2011). We use 2D greyscale images of spectrograms, generated from mono audio signals. Spectrograms are images of size $P \times T$ pixels taking values that fall in the interval $[0, 1]$. The audio signals we use are either recordings of human performances or synthesized from symbolic representations. The synthetic audio files are generated from the first 400 notes of each piece encoded in symbolic format, using a horn sound approximated by frequency modulation synthesis, with a sampling rate of 22050Hz.[1] The audio recordings correspond to excerpts of approximately the first 400 notes. The stereo recordings are converted to mono by taking the average of the left and right channels.

Spectrograms are obtained using the short-time Fourier transform (STFT) or the variable-Q transform (VQT) (Schörkhuber, Klapuri, Holighaus, & Dörfler, 2014).[2] STFT spectrograms are computed with a Hamming window, the window size is 1024 samples as used by Wu et al. (2011). VQT spectrograms are computed with 48 frequency bins per octave and the parameter $\gamma = 20$, which is used to increase the time resolution in the lower frequency range (cf. Schörkhuber et al., 2014).

Figure 3 shows examples of the types of pitch–time representation that we use, including a piano-roll sampled from an excerpt of a MIDI file, along with spectrograms of recorded and synthesized audio. Both STFT and VQT spectrograms are plotted with a logarithmic scale for frequency.

2.1.3 Size of images

Piano-roll images of excerpts p_{70qn} or p_{400n} are all 56×560 pixels. Piano-rolls of full-length pieces are denoted p_{fl}; the size along the time axis varies according to the length of each piece. In audio, we use only spectrograms of excerpts of music, denoted by sp_{400n}. Due to the spectral content in spectrograms, we use a higher resolution than piano-rolls, i.e., 150 pixels on the pitch axis. To approximately preserve the same amount of information as piano-rolls, we sacrifice the temporal resolution of spectrograms, downsampling them to 200 pixels, such that all spectrograms have a size

[1] We use the SYNTHTYPE function of the Matlab MIDI Toolbox (Eerola & Toiviainen, 2003). The horn sound was used as it was the best choice of the two available sounds in the toolbox that we used for rendering (the alternative was Shepard tones).

[2] Toolbox accessed from `http://www.cs.tut.fi/sgn/arg/CQT/` on 28 August 2015.

of 150×200 pixels. STFT spectrograms were downsampled from 344×398 pixels to 150×200 pixels using bicubic interpolation (Keys, 1981). VQT spectrograms were generated using the resolution of 150×200 pixels.

2.2 Processing phase

Once the piece is sampled, it can be processed as an excerpt or as segments as seen in Figure 2. Only one of the two processing phases is used in any one classifier. The input for the processing phases is a 2D pitch–time image of size $P \times T$, either a piano-roll or a spectrogram as described above.

2.2.1 Processing excerpt

The *processing excerpt* phase has three modules in the following order: centring (2.2.2.3), filtering (2.2.2.1), and Linear Discriminant Analysis (LDA) (2.2.2.4) (as in Velarde et al., 2016) (see Figure 2). Each of these three modules can be activated or deactivated depending on the given configuration, e.g., according to the selection of the parameters an image will be centred/not centred, filtered/not filtered, processed/not processed with LDA. All pitch–time images entering this phase have the same input size of $P \times T$ pixels, and correspond to excerpts of music consisting of either the first 70 qn or the first 400 notes of a piece. The output of this phase is a transformed image which preserves its input size if LDA is switched off, or it is reduced to a real number if LDA is switched on.

2.2.2 Processing segments

The *processing segments* phase has three modules in the following order: filtering (2.2.2.1), segmentation (2.2.2.2) and centring (2.2.2.3) as seen in Figure 2. Unlike the *processing excerpt* phase, where all modules can be activated, in this processing phase, the segmentation module is always active. If the *centring* module is switched on, each segment is centred individually. The *processing segments* phase outputs several segments, which are sent to the *classification* phase (2.3).

2.2.2.1 Filtering

It is well-established that filtering (and convolution in particular) is ubiquitous in the perceptual systems of animals (Snowden, Thompson, & Troscianko, 2012). For example, local processing aspects

of visual perception can be effectively described as a form of filtering or convolution (Murdock Jr., 1979; Pribram, 1986). It is therefore not surprising that this process can also form the basis of artificial methods for recognising and classifying objects. For example, in experiments involving functional neuroimaging, Gabor filters have been used to identify natural images from activity in the visual cortex (Kay, Naselaris, Prenger, & Gallant, 2008). Audition has been modeled with bandpass filters (Daubechies & Maes, 1996; Karmakar, Kumar, & Patney, 2011). Machine learning approaches use filtering combined with support vector machines (SVMs) or neural networks for image classification tasks (Bengio, Courville, & Vincent, 2013; LeCun, Kavukcuoglu, & Farabet, 2010; Tuia, Volpi, Mura, Rakotomamonjy, & Flamary, 2014)

In music classification, filtering has been shown to significantly improve recognition (Velarde et al., 2016). For the filtering module of the processing phase, we convolve pitch–time images with a rotationally symmetric Gaussian filter g:

$$g(x, y) = e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{1}$$

where (x, y) is the position of a point relative to the origin. The Gaussian filter is of size 9×9 pixels and the standard deviation of the Gaussian distribution is $\sigma = 3$ (as in Velarde et al., 2016).

2.2.2.2 Segmentation

We introduce a *segmentation* phase, as local processing has been found to be fundamental in modelling melodic similarity for music classification (van Kranenburg et al., 2013; Velarde et al., 2013). We use constant-length segmentation, which chunks each image into segments of equal length. Given a pitch–time image of size $P \times T$ pixels, this image is segmented along the time dimension into segments with a constant length of L pixels, such that after segmentation each segment's size is $P \times L$ pixels. Let $n = \lceil T/L \rceil$. If $T \bmod L \neq 0$, the width of the last segment, i.e., the nth segment, is less than L, so we pad it to the left with $L - (T \bmod L)$ columns from the $(n-1)$th segment to ensure that the final nth segment has width L. Depending on the amount of overlap between the padded nth segment and the $(n-1)$th segment, we replace the $(n-1)$th segment with the nth segment using the following procedure: if $T \bmod L \leqslant 0.3L$, then the nth segment replaces the $(n-1)$th segment, and the output number of segments is $n-1$, otherwise the output number of segments is n.

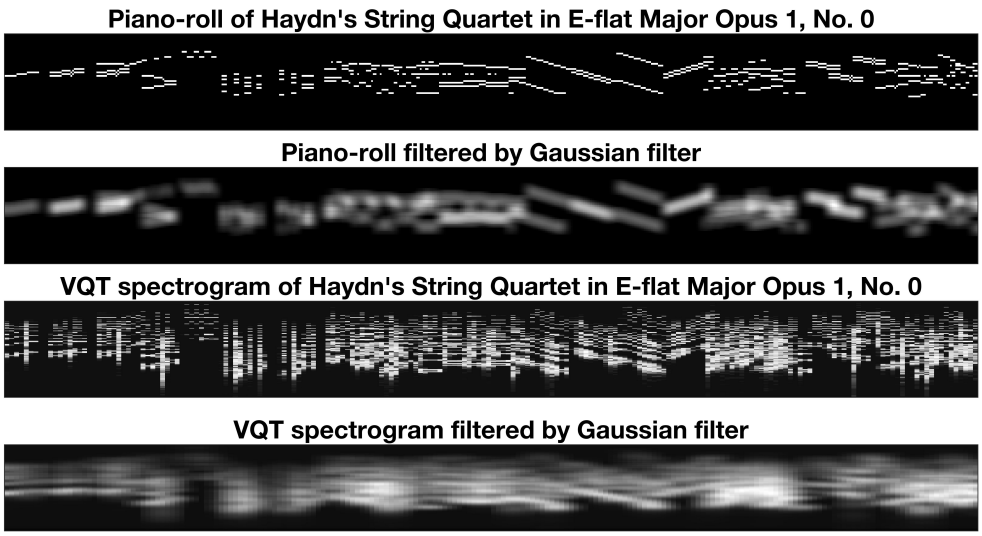


Figure 4: Excerpt of Haydn's String Quartet in E-flat Major Opus 1, No. 0, in four pitch–time representations, from the top to the bottom: Piano-roll (p_{400n}) morphetic pitch representation, followed by its convolution with a Gaussian filter (second image), VQT spectrogram of the same excerpt synthesized with a horn sound (third image), and finally the filtered version of the VQT spectrogram (fourth image).

2.2.2.3 Centring

We use the pitch range centring technique (as in Velarde et al., 2016). Pitch range centring is equivalent to pitch transposition, such that the pitch range of the image is centred vertically using a bounding box.

2.2.2.4 Linear Discriminant Analysis

LDA aims to find a linear subspace of discriminatory features between classes (Cai, He, Hu, Han, & Huang, 2007). The singularity problem is solved by Singular Value Decomposition and Tikhonov regularization.[3]

2.3 Classification

The input to the *classification* phase can be one sample if it comes from the *processing excerpt* phase, or several samples (processed segments) if they come from the *processing segments* phase. In the

[3]We use the LDA implementation by Deng Cai version 2.1: `http://www.cad.zju.edu.cn/home/dengcai/Data/DimensionReduction.html`.

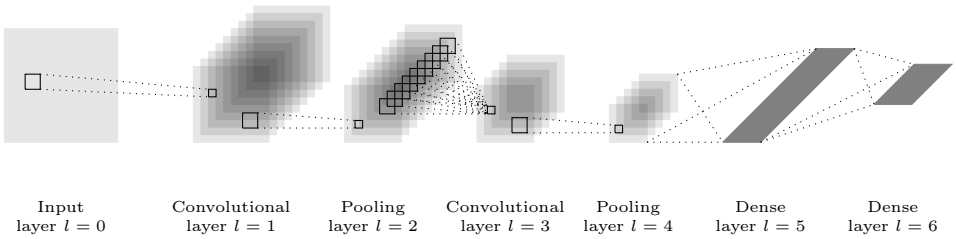


Figure 5: Architecture of the convolutional neural network used for classification.

latter case, the predicted class label of a piece of music is the most frequently predicted class of its segments.

In the *classification* phase we restricted ourselves to using an SVM, k-NN or a CNN as shown in Figure 2, even though other classification algorithms could have been used, e.g., decision trees or logistic regression.

We train a linear SVM with Sequential Minimal Optimization (SMO) with Karush–Kuhn–Tucker conditions set to 0.001, with samples normalized around the mean, and scaled to have unit standard deviation (as in Velarde et al., 2016). The k-NN classifier is used with Euclidean distance and the next nearest point to break ties. The number of nearest neighbours was set empirically. In the next section, we describe the CNN that we use.

2.3.1 Classifying with a convolutional neural network

We use a CNN (LeCun et al., 2010) as illustrated in Figure 5. This CNN consists of two convolutional–pooling stages, and a third stage of dense layers as a classifier, making a total of 6 layers. The first stage has a convolutional layer ($l = 1$) with 9 filters of size 9×9 followed by a max-pooling layer ($l = 2$) with a pool size of 2×2; while the second stage has a convolutional layer ($l = 3$) with 5 filters of size 5×5 followed by a max-pooling layer ($l = 4$) with a pool size of 2×2. Dropout is used after this stage. The third stage consists of a dense layer ($l = 5$) with 256 rectified linear units, and a dense layer ($l = 6$) with 2 softmax units as a classifier.

The network architecture and hyper-parameters were selected empirically. The CNN parameters are initialized using the method proposed by Glorot and Bengio (2010). The network is trained using RMSProp (Dauphin, de Vries, Chung, & Bengio, 2015), a mini-batch variant of stochastic gradient descent that adaptively adjusts the learning rate by dividing the gradient by an average of its recent

Classifier	Representation	Sampling	Processing	Filter	Centring	LDA	Classification
C1S	p_{400n}	Morphetic	Excerpt	Gaussian	No	Yes	SVM
C2S	p_{fl} or p_{400n}	Morphetic	Segments, $L = 8$ pixels	Gaussian	Pitch range	No	k-NN, $k = 15$
C3S	p_{fl} or p_{400n}	MNN	Segments, $L = 16$ pixels	Gaussian	No	No	k-NN, $k = 3$
C1A	sp_{400}	VQT	Excerpt	Gaussian	No	No	SVM
C2A	sp_{400}	STFT	Segments, $L = 4$ pixels	Gaussian	No	No	k-NN, $k = 15$
C3A	sp_{400}	STFT	Segments, $L = 8$ pixels	Gaussian	No	No	k-NN, $k = 3$

Table 1: Details of the configurations of individual classifiers. Classifiers C1S, C2S and C3S are used for symbolic representations of music. C1A, C2A and C3A are used for audio files.

magnitude. Additionally, we use Nesterov's method for accelerating gradient descent (Sutskever, Martens, Dahl, & Hinton, 2013). In order to avoid overfitting, l_2-norm weight regularization, early stopping and dropout is used. Early stopping was performed by monitoring the loss function on the validation set. The learning rate for RMSProp is 10^{-5}, Nesterov's momentum is 0.5, the probability of dropout is set to 0.5, the regularization coefficient is 0.01 and the batch size is 2.

We evaluated the CNN in leave-one-out cross-validation. In each run of the leave-one-out cross-validation, we used 80% of the data for training, 20% for validation, and a single piece for testing. All networks were trained for a maximum of 1000 epochs.

2.4 The ensemble of classifiers

In the design of the ensemble, our goal is to have the same structure of individual classifiers for audio and symbolic representations: one classifier extracting features at large scale (C1), and two classifiers extracting local features at two different small time scales (C2 and C3), as seen in Figure 1. Classifiers C1S, C2S and C3S are used for symbolic representations. Correspondingly, C1A, C2A and C3A are used for audio. Details on the configurations of each classifier are given in Table 1.

We use different configurations for each classifier expecting to have diversity in their predictions when ensembled. C1A was selected as it was the best performing classifier reported by Velarde et al. (2016). To select C2S and C3S, we tested different representations, number of nearest neighbours and length of segments. We then selected the best performing classifiers, which also worked best when combined in the ensemble of three classifiers.

We noticed that k-NN worked better than SVM when pieces went through the *processing seg-*

ments phase, indicating the presence of several clusters of small musical patterns. However, we did not obtain results for using different values of k in our k-NN classifier when using the *processing excerpt* phase. We intend to explore this in future work.

For symbolic representations we used morphetic pitch or MNN, while in audio, morphetic encoding would have required the system to have some kind of transcription module, which we avoided. Instead, we used two sampling methods VQT and STFT. As the time dimension of spectrograms was downsampled to almost half the size of the piano-roll time dimension, the segment length of C2A is half that in C2S. The same holds for classifiers C3A and C3S. None of the classifiers used for audio included centring or LDA when processing because of performance reasons. For spectrograms, we experienced that LDA worked better without filtering and vice versa. We assume that filtering influenced the way that LDA extracted features, presumably not being able to distinguish between musical patterns and short-term spectral patterns. We used centring for classifier C2S, but not for C2A as it had a negative effect on its performance. In piano-rolls the top and bottom regions are very uniform (mostly pixels with value 0), such that shifting bounding boxes up or down does not cause much change in the texture at the periphery. However, in spectrograms this is not the case, and we did not apply a technique to preserve the texture at the top and bottom of the images after centring. Therefore, centring spectrograms might have introduced noise into the data.

3 Experiments

First, we present two experiments: the first experiment evaluates the performance of our method on composer recognition, while the second experiment shows results on genre classification. In both cases, we used both audio recordings and symbolic representations of music. Finally, we present a third experiment on symbolic representations only, using a CNN where the filters were learnt automatically.

The task of classifying string quartet movements by Haydn and Mozart has been extensively studied on symbolic representations of music (Herlands et al., 2014; Hillewaere et al., 2010; Hontanilla et al., 2013; van Kranenburg & Backer, 2004; Velarde et al., 2016), which enables us to benchmark our proposed method for composer classification. The second experiment on genre classification focuses on discriminating between preludes and fugues from *The Well-Tempered Clavier* by J. S. Bach. We

could not find any relevant previous work on this task. The datasets of the two experiments were selected so that the first contained pieces in the same genre by different composers, while the second contained pieces in different genres by the same composer. By doing so we can test the two aspects independently.

For the first and second experiments, we perform five-fold cross-validation with a partitioning scheme of 80% for training and 20% for testing. Moreover, we also perform leave-one-out cross-validation on the string quartet movement classification task, to compare our methods with the state-of-the-art approaches (Hillewaere et al., 2010; van Kranenburg & Backer, 2004) that use this validation strategy.

3.1 Experiment 1: Classifying string quartet movements by Haydn and Mozart

3.1.1 Dataset

A string quartet is a multi-movement work for two violins, viola and cello. The earliest string quartets were written in the 1760s by composers such as Joseph Haydn and Franz Xaver Richter, with Wolfgang Amadeus Mozart writing his earliest quartets during the 1770s. The number of movements in early quartets varied and it was only with Haydn's op.9 (1769–1770) that a standard four-movement scheme became established, consisting typically of a sonata-form movement, an adagio, a dance-like movement (often a minuet and trio), and a lively finale (Eisen, Baldassarre, & Griffiths, n.d.).

Three datasets have been used to evaluate computational methods on the recognition of the string quartet movements by Haydn and Mozart. These datasets were introduced by van Kranenburg and Backer (2004), Hillewaere et al. (2010) and Herlands et al. (2014). For our experiment, we used the two datasets available to us, which we denoted by HM107 and HM207:

- **HM107**. This dataset, introduced by van Kranenburg and Backer (2004), consists of 107 movements: 54 string quartet movements by Haydn and 53 movements by Mozart, encoded as **kern files.[4]

- **HM207**. This dataset, introduced by Hillewaere et al. (2010), extends the **HM107** dataset

[4] `http://www.music-cog.ohio-state.edu/Humdrum/representations/kern.html`

(I) Symbolic representations (full length).					
		Classifiers			
		C1S-p_{400n}	C2S-p_{fl}	C3S-p_{fl}	Ensemble
HM107	Mean	0.731	0.625	0.729	**0.758**
	SD	0.072	0.042	0.067	0.053
HM207	Mean	0.628	0.734	0.725	**0.768**
	SD	0.062	0.078	0.030	0.069
(II) Symbolic representations (excerpts).					
		Classifiers			
		C1S-p_{400n}	C2S-p_{400n}	C3S-p_{400n}	Ensemble
HM107	Mean	0.731	0.702	0.702	**0.760**
	SD	0.072	0.128	0.095	0.089
HM207	Mean	0.628	0.672	0.657	**0.701**
	SD	0.062	0.073	0.063	0.055
(III) Synthetic audio files (excerpts).					
		Classifiers			
		C1A-sp_{400n}	C2A-sp_{400n}	C3A-sp_{400n}	Ensemble
HM107	Mean	0.654	0.683	0.682	**0.721**
	SD	0.069	0.141	0.088	0.126
HM207	Mean	0.691	0.653	0.623	**0.705**
	SD	0.105	0.062	0.053	0.047

Table 2: Haydn and Mozart String Quartet classification accuracies in five-fold cross-validation using symbolic representations of music and synthetic audio files. Each classifier's mean and standard deviation (SD) are reported over the five folds of the cross-validation. In blocks (I) and (II), C1S is given piano-roll excerpts of 400 notes. In block (I), C2S and C3S are given piano-rolls of full-length pieces. In block (II), the three classifiers (C1S, C2S, C3S) are given piano-roll excerpts of 400 notes. Finally, in block (III), the classifiers (C1A, C2A, C3A) are given spectrogram excerpts of 400 notes. The highest accuracies per dataset are highlighted in bold type.

to 207 movements consisting of 112 string quartet movements by Haydn and 95 string quartet movements by Mozart, encoded as MIDI files.

For the experiments on audio data, datasets HM107 and HM207 were rendered to WAV format, synthesized as described in section 2.1.2.

3.1.2 Classification results

Table 2 presents classification accuracies in five-fold cross-validation of the classifiers shown in Table 1. First, we evaluated whether classifiers C2S and C3S would perform differently with less information, such that instead of processing full-length pieces, they would be given excerpts of music. At the 5% significance level, we found no significant difference between the performance of C2S and C3S, on either dataset (HM107 and HM207), when less information was used (Wilcoxon signed

	Classifiers					
	C1S-p_{400n}	C2S-p_{fl}	C3S-p_{fl}	Ensemble	V-2004	H-2010
HM107	**0.804**	0.664	0.729	0.785	0.794	
HM207	0.614	0.696	0.696	0.725		**0.754**

Table 3: Haydn and Mozart String Quartet classification accuracies in leave-one-out cross-validation. The table presents the classification accuracies obtained by each individual classifier C1S-p_{400n}, C2S-p_{fl} and C3S-p_{fl} and their ensembles. It also shows accuracies reported by van Kranenburg and Backer (2004) (V-2004), and Hillewaere et al. (2010) (H-2010). The highest accuracies per dataset are highlighted in bold type.

		C1S-p_{400n}	C2S-p_{fl}	C3S-p_{fl}	Ensemble
HM107	(van Kranenburg & Backer, 2004)	0.905	0.002	0.095	0.811
HM207	(Hillewaere et al., 2010)	0.000	0.063	0.063	0.334

Table 4: P-values from a two-tailed binomial test for different accuracies in leave-one-out cross-validation comparing the proposed classifiers (see Table 3) and the methods presented by van Kranenburg and Backer (2004) and Hillewaere et al. (2010).

rank = 47, $z = -1.397$, $p = 0.162$, $n = 20$), see blocks (I) and (II) in Table 2. For this experiment, we observe that ensembling has a positive effect, and makes the predictions more consistent across datasets. Then, we evaluated the performance of ensembles on symbolic representations and audio. On the results of both datasets HM107 and HM207, we found no significant difference in the performance of the ensembles when classifying music represented symbolically or as audio files (Wilcoxon signed rank = 10, $p = 0.563$, $n = 10$), see blocks (II) and (III) in Table 2.

Table 3 presents the accuracies of our proposed classifiers on composer recognition in leave-one-out cross-validation, and the approaches proposed by van Kranenburg and Backer (2004) and Hillewaere et al. (2010). The method proposed by van Kranenburg and Backer (2004) is based on the use of *style markers* (mostly counterpoint characteristics), dimensional reduction and k-NN, which achieves a classification accuracy of 0.794 on HM107, slightly below that of our best model, the C1S-p_{400n}. Hillewaere et al. (2010) propose a language model that builds an n-gram model of monophonic parts of the string quartet movements, reaching a classification accuracy of 0.754 on HM207, slightly above that of our ensemble model. The approaches reported by Hontanilla et al. (2013) and Herlands et al. (2014) are not considered in this comparison, as their test datasets were different from the ones used here.

We tested the differences in accuracies achieved by our proposed ensemble classifiers and the previous approaches of van Kranenburg and Backer (2004), and Hillewaere et al. (2010) for statistical significance with a two-sided binomial test. From the p-values in Table 4, we observe that the

	C1S-p_{400n}	C2S-p_{fl}	C3S-p_{fl}
HM107	0.350	0.004	0.114
HM207	0.001	0.204	0.204

Table 5: One-tailed, binomial test p-values testing the hypotheses that the accuracies (see Table 3) obtained by the ensemble in leave-one-out cross-validation are higher that those obtained by each individual classifier on both datasets (HM107 & HM207) of the Haydn and Mozart string quartets.

classification accuracies of van Kranenburg and Backer (2004) and Hillewaere et al. (2010) are not significantly better than the accuracy of our proposed ensemble of classifiers and C3S-p_{fl}, which can therefore be claimed to have reached state-of-the-art performance on both datasets. Note that Hillewaere et al. (2010) only evaluated their method on the HM207 dataset, which is an extended version of the HM107 dataset.

Finally, we tested the differences between our ensemble of classifiers and each of the individual classifiers in the ensemble using a one-sided binomial test. The p-values are shown in Table 5. They show that the ensemble is only in some cases significantly better than the individual classifiers (i.e., $p < 0.05$).

3.2 Experiment 2: Classifying preludes and fugues by J.S. Bach

3.2.1 Dataset

The Well-Tempered Clavier by J. S. Bach consists of two books (published in 1722 and 1742), each containing 24 preludes and fugues, one in each of the 12 major and 12 minor keys. According to Stein's (1979) analysis, preludes elaborate around a short motivic subject through harmonic exploration, but are heterogeneous in form. Some preludes are imitative and sectional in *Invention* form (Book I, Nos. 3, 4, 9, and 11), others in *Toccata* style, free in form and style (Book I, Nos. 2, 4, 6). On the other hand, fugues are imitative contrapuntal works, typically built upon a single main theme called the *subject*. The voices in a fugue start in succession by stating the subject followed by a secondary theme called the *countersubject*, designed to be played simultaneously with the subject, which then starts another voice. A fugue usually consists of a series of entries of the subject stated in one or more voices, alternating with *episodes* in which motivic material derived from the subject and countersubject is developed.

For this experiment we used two datasets, which we called JSB96 and JSB48:

- **JSB96**. This dataset consists of MIDI encodings of all 48 preludes and 48 fugues from Bach's *Well-Tempered Clavier*, Books I and II, provided in the MuseData collection.[5] For experiments on audio, JSB96 was rendered to WAV format, synthesized as described in section 2.1.2.

- **JSB48** This dataset consists of 24 preludes and 24 fugues from Book I of *The Well-Tempered Clavier* in MP3 V0 audio format, performed by pianist Kimiko Ishizaka.[6]

.

In a fugue, the voices enter one after the other over the course of the exposition, which imparts a highly distinctive textural character to the beginnings of these pieces. We hypothesized that this feature (the initial texture) could be used to reliably distinguish a fugue from a prelude. We therefore removed the initial segments of all images. The removed segment size for all piano-rolls corresponded to about the first 8 qn, more precisely the first 60 pixels, so that the size of each piano-roll became 56×500 pixels. For spectrograms, we removed the first 20 pixels, as our spectrograms have less time resolution than piano-rolls. The size of the spectrograms after removing the first 20 pixels became 150×180 pixels. We found that including the initial notes improved the mean classification accuracy of C1S (over the five folds) from 0.761 ± 0.042 to 0.936 ± 0.045, while the classification accuracies of C2S and C3S where not affected. Table 6 shows the classification accuracies for *The Well-Tempered Clavier* by J.S Bach in five-fold cross-validation, where the initial 60 pixels (for piano-roll) or 20 pixels (for spectrograms) were removed.

3.2.2 Classification results

On the symbolic dataset JSB96 (Table 6) the single C1S-p_{400n} classifier performed better than the ensemble, in contrast to the results of Experiment 1 (Table 2). In general, we observe that the classification accuracies obtained on composer recognition are similar to those obtained on genre classification (see Table 6 and Table 2).

We compared performance of the classifiers on synthetic and human performed audio representations of Book I of *The Well-Tempered Clavier* (JB96, JB48) as shown in Table 6 (lower part). The results reported on these small datasets should be considered preliminary. Nevertheless, C1S seems to work better than C2S, C3S and the ensemble. We expected that audio performances would be

[5] `http://www.musedata.org/encodings/bach/bg/keybd/`. Accessed on 23 February 2015.
[6] `http://music.kimiko-piano.com/album/bach-well-tempered-clavier-book-1`. Accessed on 7 August 2015.

		Classifiers			
		C1S-p_{400n}	C2S-p_{400n}	C3S-p_{400n}	Ensemble
JSB96 (symbolic representations)	Mean	**0.761**	0.708	0.731	0.741
	SD	0.042	0.035	0.067	0.063
		C1A-sp_{400n}	C2A-sp_{400n}	C3A-sp_{400n}	Ensemble
JSB96 (synthetic audio)	Mean	0.729	**0.748**	0.658	0.718
	SD	0.120	0.066	0.059	0.034
JSB96, Book I (synthetic audio)	Mean	**0.771**	0.684	0.627	0.687
	SD	0.178	0.158	0.166	0.212
JSB48, Book I (audio recordings)	Mean	**0.751**	0.522	0.584	0.564
	SD	0.048	0.175	0.162	0.165

Table 6: Classification accuracies for discrimination between preludes and fugues from *The Well-Tempered Clavier* using symbolic representations of music, synthetic audio files and audio recordings. Each classifier's mean and standard deviation (SD) are reported over the five folds of the cross-validation. Initial 60 pixels removed from piano-rolls. Initial 20 pixels removed from spectrograms.

Datasets	CNN-p_{400n}	C1S-p_{400n}	p-value
HM107	0.776	0.804	0.286
HM207	0.560	0.614	0.070

Table 7: Comparison of the accuracies obtained in leave-one-out cross-validation by classifiers CNN-p_{400n} and C1S-p_{400n} on two datasets of the Haydn and Mozart string quartet movements.

more difficult to classify than synthetic audio, but the differences so far are rather small. However there is not enough evidence to draw any conclusions on this.

It is noteworthy that the ensemble classifiers did not perform better than individual classifiers in this experiment. The differences are however not significant. It is possible that a more sophisticated ensemble method could achieve better results here.

3.3 Experiment 3: Experiments using a convolutional neural network

For experiments with a CNN, the pieces of music were sampled as morphetic piano-roll excerpts p_{400n} and then classified without using any of the processing modules in the *processing excerpts* phase.

Table 7 shows the classification accuracies obtained by the CNN and the C1S classifiers, both using piano-rolls p_{400n}, along with the p-values obtained with a one-tailed binomial test comparing their accuracies. Although the accuracies for C1S are higher than those obtained using a CNN, the difference is not significant at the 5% level. For practical purposes, however, the computational cost of C1S is considerably lower than that of the CNN. C1S can be seen as a single filter, single layer,

convolutional classifier, while the proposed CNN has a deeper architecture with several filter layers as proposed by LeCun et al. (2010).

The CNN is interesting because the filters in the network adapt to the data. Figure 6 shows the Gaussian filter used in all experiments, and the filters learnt by the first convolutional layer of the CNN when trained on Haydn and Mozart string quartet movements. In these greyscale images, lighter shades represent higher values. Vertical distances represent pitch intervals in diatonic steps (e.g., seconds, thirds, fourths). This enables the observation of pitch and rhythm patterns.

For example in Figure 6 (b), the filter in the middle row, left column shows triad chord structures (sets transpositionally equivalent to {C,E,G}, {D,F,A}, {F,A,C}, etc.). The filter in the bottom row, middle column shows for example {D,F} followed by {E,G}. In general, we observe that the filters shown in Figure 6 (b) show thirds. In Figure 6 (c), middle column, top row, the filter shows the interval of a sixth (e.g., {C,A}). The filters in the lower two rows are dominated by a single voice. The top-left filter shows an interval of a second (e.g., {E,F}). These patterns may relate to observations such as that by Herlands et al. (2014) about a predilection for the use of certain intervals in the string quartets by Mozart which made the melodic lines of the first violin less virtuosic than those of Haydn. However additional analysis is needed to understand this relation.

Since the CNN performs well at distinguishing the string quartet movements by Haydn and Mozart, we hypothesize that these patterns play a role in distinguishing their styles. However, how the patterns in the first and second layer contribute, and what their musical relevance is, is not yet well understood and deserves further analysis in the future.

4 Discussion

In music classification, there are only a few methods that have been designed for and evaluated on audio *and* symbolic representations of music. For example, Tzanetakis, Ermolinskyi, and Cook (2003) demonstrated the use of pitch histograms for genre classification in both domains (audio and symbolic); and Cataltepe, Yaslan, and Sonmez (2007) and Lidy, Rauber, Pertusa, and Iñesta (2007) combined symbolic and audio features to improve their classifiers on genre recognition. In this study, our aim was to design a general method for music classification applicable to symbolic representations and audio recordings.

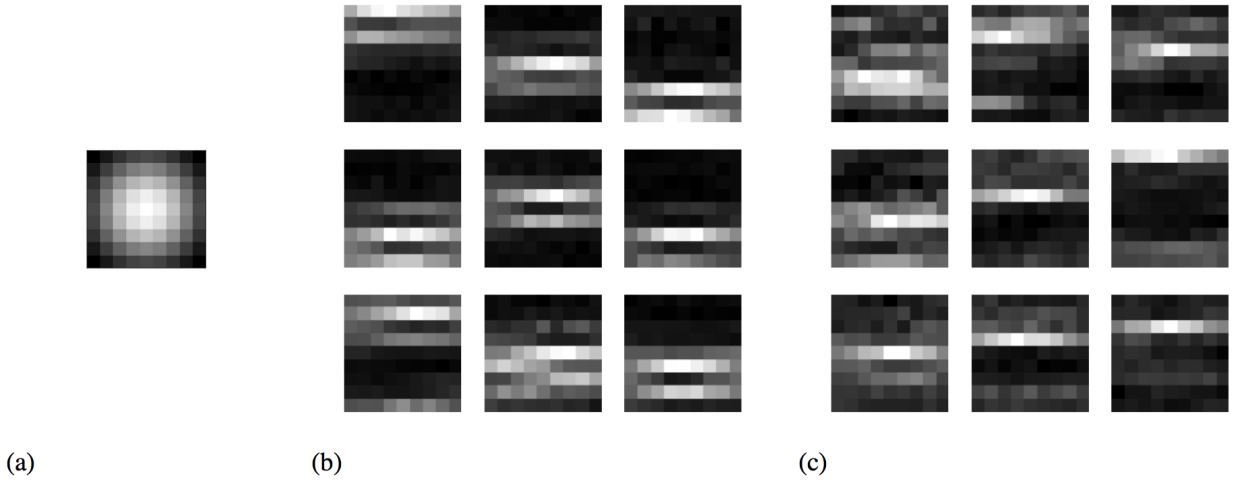


Figure 6: (a) Gaussian filter with $\sigma = 3$. (b) Filters learnt by the first convolutional layer of the CNN ($l = 1$ of p_{400n}), trained on the string quartet movements by Haydn and Mozart (HM107). (c) Filters learnt by the first convolutional layer of the CNN ($l = 1$ of p_{400n}), trained on the string quartet movements by Haydn and Mozart (HM207). Each filter is 9×9 pixels.

The experiments presented in this work suggest that robust style recognition can be achieved by a combination of feature extraction at large and small time scales. We observed that ensembling seemed to contribute towards making consistent predictions across datasets. However, the ensemble was not significantly more accurate than individual classifiers.

The effect of convolution with a Gaussian filter may highlight musical contour, making structures more discriminative after filtering.

While most audio datasets for music classification usually contain 30-second audio clips, we sampled pitch–time representations from 400 notes. Velarde et al. (2016) showed that the classifier based on excerpts (C1S) is robust to the amount of information used: either using excerpts of music of 70 quarter notes or containing 400 onsets. Moreover, in Experiment 1 (3.1) we found that classifiers using segments (C2S and C3S) performed similarly when the input was an excerpt of music or the full-length piece.

In Experiment 3, we tested a deep learning architecture that has proven to be state-of-the-art for image classification, e.g., in applications such as digit recognition. Somewhat unexpectedly, the CNN was not better than C1S in our evaluations. We suspect that for the CNN to show a significant improvement over the state-of-the-art, we would need to train it on a larger dataset, as is normally the case in image classification tasks. It could also be possible that the task of discriminating

between string quartet movements by Haydn and Mozart has reached a so-called "glass ceiling". In our experiments, one advantage of the CNN over C1S and most of the previously published methods on this task (Hillewaere et al., 2010; Hontanilla et al., 2013; van Kranenburg & Backer, 2004; Velarde et al., 2016) is the potential for gaining musical insight by exploring the information in the filters learnt by the CNN. In the future, we would like to evaluate the performance of the CNN when pieces follow the *processing segments* phase, and evaluate the CNN on audio and on larger datasets before drawing further conclusions.

5 Conclusions

We have introduced a novel convolution-based method on pitch–time representations for classification using both symbolic and audio representations of music. The effect of convolution with a Gaussian filter may highlight musical contour, making structures more discriminative after filtering. We have shown that the performance of individual classifiers based on excerpts of music is comparable to the performance of individual classifiers using small time-scale segments, and that their outputs can be complementary for ensembling. Our proposed classifiers perform well on both composer and genre classification, as well as on symbolic representations and synthetic audio, and have proven to be state-of-the art when evaluated on two datasets of the Haydn and Mozart string quartets. Additionally, we evaluated our proposed classifiers on *The Well-Tempered Clavier* by J.S Bach, demonstrating the versatility and effectiveness of our method. Our experiments were conducted on baroque and classical music, but we expect our classifiers to generalize to other styles of music, periods of time and classification tasks. Additionally, we presented the potential of a convolutional neural network to provide musical insight, based on the filters that it learns automatically. In the future, we are interested in evaluating our method on larger datasets and multi-class recognition problems.

6 Acknowledgments

The work for this paper carried out by G. Velarde, D. Meredith, C. Cancino Chacón and M. Grachten was done as part of the EC-funded collaborative project, "Learning to Create" (Lrn2Cre8). The

project Lrn2Cre8 acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 610859. G. Velarde is also supported by a PhD fellowship from the Department of Architecture, Design and Media Technology, Aalborg University. The authors would like to thank Peter van Kranenburg and Ruben Hillaweare for sharing with us the Haydn and Mozart string quartet datasets.

References

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798-1828. doi: 10.1109/TPAMI.2013.50

Cai, D., He, X., Hu, Y., Han, J., & Huang, T. (2007). Learning a spatially smooth subspace for face recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on* (pp. 1–7). doi: 10.1109/CVPR.2007.383054

Cataltepe, Z., Yaslan, Y., & Sonmez, A. (2007). Music genre classification using MIDI and audio features. *EURASIP Journal on Advances in Signal Processing*, *2007*(1), 1–8.

Costa, Y. M., Oliveira, L., Koerich, A. L., Gouyon, F., & Martins, J. (2012). Music genre classification using LBP textural features. *Signal Processing*, *92*(11), 2723–2737.

Daubechies, I., & Maes, S. (1996). A nonlinear squeezing of the continuous wavelet transform based on auditory nerve models. In A. Aldroubi & M. Unser (Eds.), *Wavelets in medicine and biology* (pp. 527–546). Boca Raton, FL: CRS Press.

Dauphin, Y. N., de Vries, H., Chung, J., & Bengio, Y. (2015). RMSProp and equilibrated adaptive learning rates for non-convex optimization. *arXiv*, *1502*, 4390.

de Boor, C. (1978). *A practical guide to splines.* Now York: Springer-Verlag.

Eerola, T., & Toiviainen, P. (2003). *MIDI toolbox: Matlab tools for music research.* Jyväskylä, Finland: University of Jyväskylä. (`http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/miditoolbox/`)

Eisen, C., Baldassarre, A., & Griffiths, P. (n.d.). *String quartet.* `http://www.oxfordmusiconline.com.zorac.aub.aau.dk/subscriber/article/grove/music/40899`. Oxford University Press. (Accessed 9-Oct-2015)

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *JNLR W&CP: Proceedings of the Thirteenth International Conference on Artificial Intelligence and*

Statistics (AISTATS 2010), Chia Laguna Resort, Sardinia, Italy (pp. 249–256).

Herlands, W., Der, R., Greenberg, Y., & Levin, S. (2014). A machine learning approach to musically meaningful homogeneous style classification. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada, July 27-31, 2014* (pp. 276–282). Retrieved from http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8314

Hillewaere, R., Manderick, B., & Conklin, D. (2010). String quartet classification with monophonic models. In J. S. Downie & R. C. Veltkamp (Eds.), *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, Netherlands, August 9-13, 2010* (pp. 537–542). International Society for Music Information Retrieval. Retrieved from http://ismir2010.ismir.net/proceedings/ismir2010-91.pdf

Hontanilla, M., Pérez-Sancho, C., & Iñesta, J. M. (2013). Modeling musical style with language models for composer recognition. In J. M. Sanches, L. Micó, & J. S. Cardoso (Eds.), *Pattern Recognition and Image Analysis: 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5-7, 2013. Proceedings* (pp. 740–748). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-642-38628-2_88 doi: 10.1007/978-3-642-38628-2_88

Karmakar, A., Kumar, A., & Patney, R. (2011). Synthesis of an optimal wavelet based on auditory perception criterion. *EURASIP Journal on Advances in Signal Processing*, *2011*(1), 1–13.

Kay, K. N., Naselaris, T., Prenger, R. J., & Gallant, J. L. (2008). Identifying natural images from human brain activity. *Nature*, *452*(7185), 352–355.

Keys, R. (1981). Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *29*(6), 1153-1160. doi: 10.1109/TASSP.1981.1163711

Kuncheva, L. I. (2004). *Combining pattern classifiers: methods and algorithms.* John Wiley & Sons.

LeCun, Y., Kavukcuoglu, K., & Farabet, C. (2010, May). Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems* (pp. 253–256). doi: 10.1109/ISCAS.2010.5537907

Lidy, T., Rauber, A., Pertusa, A., & Iñesta, J. M. (2007). Improving genre classification by combination of audio and symbolic descriptors using a transcription systems. In S. Dixon, D. Bainbridge, & R. Typke (Eds.), *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007, Vienna, Austria, September 23-27, 2007* (pp. 61–66). Austrian Computer Society. Retrieved from http://ismir2007.ismir.net/proceedings/ISMIR2007_p061_lidy.pdf

Meredith, D. (2006). The ps13 pitch spelling algorithm. *Journal of New Music Research*, *35*(2), 121–159. Retrieved from http://dx.doi.org/10.1080/09298210600834961 doi: 10.1080/09298210600834961

Murdock Jr., B. B. (1979). Convolution and correlation in perception and memory. In L.-G. Nilsson (Ed.), *Perspectives on Learning and Memory* (pp. 105–119). Hillsdale, NJ: Erlbaum.

Paul, E. S., & Kaufman, S. B. (2014). *The philosophy of creativity: New essays.* Oxford University Press.

Pribram, K. H. (1986). Convolution and matrix systems as content addressible distributed brain processes in perception and memory. *Journal of Neurolinguistics*, *2*(1), 349–364.

Rush, J. C., & Sabers, D. L. (1981). The perception of artistic style. *Studies in Art Education*, *23*(1), 24–32.

Sapp, C., & Liu, Y.-W. (2015). *The Haydn/Mozart String Quartet Quiz.* (`http://qq.themefinder.org` (Accessed 26-Dec-2015))

Schörkhuber, C., Klapuri, A., Holighaus, N., & Dörfler, M. (2014). A Matlab toolbox for efficient perfect reconstruction time-frequency transforms with log-frequency resolution. In *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio.* London, UK.

Snowden, R. J., Thompson, P., & Troscianko, T. (2012). *Basic vision: An introduction to visual perception.* Oxford: Oxford University Press.

Stein, L. (1979). *Structure & style: the study and analysis of musical forms.* Summy-Birchard Music.

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. E. (2013). On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning.* Atlanta, Georgia, USA.

Tuia, D., Volpi, M., Mura, M. D., Rakotomamonjy, A., & Flamary, R. (2014). Automatic feature learning for spatio-spectral image classification with sparse SVM. *IEEE Transactions on Geoscience and Remote Sensing*, *52*(10), 6062-6074. doi: 10.1109/TGRS.2013.2294724

Tzanetakis, G., Ermolinskyi, A., & Cook, P. (2003). Pitch histograms in audio and symbolic music information retrieval. *Journal of New Music Research*, *32*(2), 143–152.

van Kranenburg, P., & Backer, E. (2004). Musical style recognition—a quantitative approach. In R. Parncutt, A. Kessler, & F. Zimmer (Eds.), *Proceedings of the Conference on Interdisciplinary Musicology (CIM04) Graz, Austria, April 15-18, 2004* (pp. 106–107).

van Kranenburg, P., Volk, A., & Wiering, F. (2013). A comparison between global and local features for computational classification of folk song melodies. *Journal of New Music Research*, *42*(1), 1-18. Retrieved from `http://dx.doi.org/10.1080/09298215.2012.718790` doi: 10.1080/09298215.2012.718790

Velarde, G., Weyde, T., Cancino Chacón, C. E., Meredith, D., & Grachten, M. (2016). Composer recognition based on 2d-filtered piano-rolls. In M. I. Mandel, J. Devaney, D. Turnbull, & G. Tzanetakis (Eds.), *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016, New York City, United States, August 7-11, 2016* (pp. 115–121). Retrieved from `https://wp.nyu.edu/ismir2016/wp-content/uploads/sites/2294/2016/07/063_Paper.pdf`

Velarde, G., Weyde, T., & Meredith, D. (2013). An approach to melodic segmentation and classification based on filtering with the haar-wavelet. *Journal of New Music Research*, *42*(4), 325–345. Retrieved

from http://dx.doi.org/10.1080/09298215.2013.841713 doi: 10.1080/09298215.2013.841713

Wu, M.-J., Chen, Z.-S., Jang, J.-S. R., Ren, J.-M., Li, Y.-H., & Lu, C.-H. (2011). Combining visual and acoustic features for music genre classification. In *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on* (Vol. 2, pp. 124–129).

ISSN (online): 2446-1628
ISBN (online): 978-87-7112-887-1

AALBORG UNIVERSITY PRESS