Content-based Visualisation to Aid Common Navigation of Musical Audio

GAVIN JAMES WOOD

Thesis submitted for the degree of Doctor of Philosophy

Department of Computer Science
University of York

December 2005

Abstract

The amount of music available for digital archival and analysis increases steadily and swiftly. As more people listen to more music it is increasingly useful to reconsider aspects of the tools used to playback this audio.

One of these aspects is that of navigating around a music track. I propose to enhance the user experimence of audio playback software by providing with any musical audio track played a visual depiction of the contents in such a way that it may be utilised with a minimum of effort.

The amount of computation power for the processing of these musical audio signals is also increasing. On modern hardware, techniques for creating images from audio content may realistically utilise modern signal processing and machine-learning techniques. I propose a number of novel visualisation techniques drawing from state-of-the-art musical-audio signal analysis techniques.

By way of proving the thesis I prototype a number of methods to do this content-based depiction and integrate them into a common piece of software for personal music playback. I show empirically how usage of the novel visuals differs to both typical playback software with no visuals and the traditional amplitudal representation of audio.

Contents

List of Tables

List of Figures

Declaration

I declare that this thesis has been completed by myself and that, except where indicated to the contrary, the research documented is entirely my own.
Some material presented in chapters 3 and 5 has been presented at a number of conferences. A list of the publications in which the material has subsequently appeared is included in the Bibliography under Wood and O'Keefe (2004, 2005).

Gavin James Wood

Chapter 1

Introduction

"I don't know anything about music. In my line you don't have to."

—Elvis Presley (1935–1977)

1.1 Overview

The last quarter-century has seen the digital representation of music, and in particular music recordings, grow to become the most popular distribution format. From the engineering novelty of the Compact Disc by, among others, Doi and Immink (1998) it has progressed to the Internet distribution of MPEG-compressed audio (e.g. MP3 files) and the growth of portable players for this media (e.g. the *Apple iPod*).

One difference between previous formats of media and MP3 files is, like the transition from tape to disc for secondary computer storage, simultaneous digital random access.[1] A number of points in the music can be accessed and processed at once with no perceivable performance degradation; this functionality has not existed in any mainstream media before and throws open the door to a multitude of uses, one of which the present work addresses.

With the increased popularity, fixed-function playback hardware has been largely replaced with far more flexible playback software or (upgradable) firmware. People are becoming increasingly comfortable with—and reliant upon—this software for music playback. Therefore, even modest improvements in the design of such software would have a cumulatively large beneficial effect when considered in a worldwide scope.

Until recently, intra-track music navigation was limited due to the serialised nature of the playback medium (e.g. compact cassette). Navigation was limited to operations of rewinding and fast-forwarding still commonplace on CD-players and video cassette

[1] Compact Discs do allow a restricted kind of random access though only for a single read-stream and this would typically be hidden from the user.

recorders or difficult random access with the needle of a gramophone player. Other contributory factors, such as a limited visual interface and minimal numerical processing power had conspired to prevent any significant improvement in navigation facilities.

With the exception of the addition of a basic random-access interface, popular music playback software still has a relatively simplistic interface for in-track navigation. This thesis posits that these interfaces are sub-optimal, and that with the current technology superior interfaces can be devised that better allow navigation within a piece of musical audio.

1.1.1 Title Definitions

I will now discuss exactly what I mean by the given title. First by defining the terms properly then second by identifying driving factors and related fields.

Music

Discussed in the literature by for example Terhardt (1982), there are three different classes of representation of music, which indeed may be thought of as different meanings for the noun itself. These are *auditory*, concerned with our perception of the phenomenon, *acoustic* concerned with the objective observations that can be made from the sound and *symbolic* concerned with the music theoretic ideas such as notes, idealised timing relationships and so forth. By utilising only the acoustic data and foregoing any meta-data such as the score or annotated timeline, the present work focuses concretely on the former two aspects. In essence, we take music to mean music *recordings* rather than music *compositions*.

Visualisation

According to the Oxford English Dictionary, 'visualisation' is:

> "1. trans. To form a mental vision, image, or picture of (something not visible or present to the sight, or of an abstraction); to make visible to the mind or imagination.
>
> 2. absol. or intr. To form a mental picture of something not visible or present, or of an abstract thing, etc.; to construct a visual image or images in the mind.
>
> 3. trans. To render visible."—Simpson et al. (1989)

Literally we will take meaning 3; "to render visible a musical audio recording". However it is clear from the other definitions that the general meaning of *visualise* has to do with the perceiver rather than the perceived—the mind's representation over any physical form.

These definitions are helpful in describing the true goal towards which this work is merely a small step. It is an attempt, in part, to work towards a systematic method of creating an image which, when seen, is perceived as being the same 'thing' as the sound when heard. In this work we wish to create this image from the pulse-code modulation information only. Exactly what it means for a sonic-based perception to be 'the same thing' as a visual perception is an interesting question in itself and is alluded to by Spence (2001).

Navigation

This thesis is concerned with how such a thing helps one to navigate around a piece of musical audio media. For the word 'navigate', we once again defer to the OED (using the only term not about some form of transport):

> "The action or process of moving around a file, file system, website, etc."—
> Simpson et al. (1989)

Thus, by aiding navigation, the aim is to help the action of moving around within a musical recording (generally while being played back). This being an empirical study, I attempt to quantify the amount of help given in several ways, but typically with respect to some particular task to be completed.

Common

The term *common* navigation is used in order to distance the present work from the arenas of professional navigational aids and/or scientific visualisation. This work is concerned with aspects of navigation as used by people whilst listening to music in the most general sense. Though this work may be of interest and use in other, more niche, areas I do not seek to address them explicitly. The assumptions, requirements and metrics used, especially with regard to the user interface and expected tasks, vary considerably.

1.1.2 Driving Factors

A improvement in the navigation facilities of common music playback systems is desirable for practical reasons alone. In addition to providing such practical benefit, this work aims to further our understanding of musical audio information extraction by determining the characteristics of music that become apparent with a variety of audio processing methods. Such an understanding may help further work not just in terms of musical audio visualisation, but also in other fields of music and audio information retrieval where it is important to understand exactly what is being represented by the information at hand.

While the Self-Organising Map has been proposed for several music and audio-related tasks (see section 4.2.2), it has not yet been used to provide a simple representation of a single piece of music. As such this work provides ideas, information and empirical observations concerning the preprocessing, parameters and outcome of using such dimensionality reductions on musical audio data, as a method of decoding the intrinsic musical information. Related areas that will therefore benefit from this work include intra- and inter-track similarity measures.

There is a large body of work on empirical studies of navigation within audio documents in the context of speech but almost none for musical documents. The small empirical study of real and current user behaviour documents a basic understanding of humans' search and retrieval approaches. Furthermore the comparison of various performance metrics between different visualisation aids helps to understand how humans react to different annotations and exactly how these annotations manifest improved performance. Further understanding of these HCI phenomena is advantageous for continued improvements in the field.

1.1.3 Disciplines

Several disciplines are involved in the present work; I will briefly document the subset of disciplines and how they interrelate.

Computer Science

The discipline of *computer science* forms the backbone to this work. The fields of *computer audition*, *information visualisation*, *human computer interaction* and *neural networks* (NN) are all important areas that the present work draws on and in some cases contributes to.

Audio-content navigation, generally focusing on speech data, is a related field; in particular the *What You See Is What You Hear* (WYSIWYH) system of visually annotating a voice audio timeline is a proposal by Hirschberg et al. (1999) in spoken-word audio similar to that in the present work. Usage of content-based information on audio recordings provided visually in the user-interface of a (real world) playback device dates back to the nineties, with the work of Roy and Schmandt (1996).

Human-Computer Interaction (HCI) is perhaps the field in which, as a whole, the present work primarily belongs. Concerning itself with our understanding of how we, as humans, interact with machines, HCI can be seen as the field encompassing user interface-design engineering. Being concerned with the actions of humans to certain conditions, it is heavily related to the field of psychology. HCI is drawn upon and contributed to in the present work mostly with chapter 2 where in-track navigation is considered empirically.

Neural networks (NN) may as a whole be considered a discrete discipline in the field of computer science. Though there are many varieties of NN, they are generally inspired by our limited understanding of the brain. Typically, a neural network is considered a 'black

box' non-linear transformation taking some set of inputs and providing a set of more useful outputs. Before working properly they must first go through a 'training phase' whereby the context of the data is learned; the parameters of this phase will make the difference between a useful and useless end result. They may be denominated into two types; supervised networks where prototype mappings of inputs to outputs are given in the training phase and unsupervised where such mappings are not provided. The present work proposes the adoption of a particular unsupervised neural network, the Self-Organising Map, to help generate helpful visualisations.

Information Visualisation

Information visualisation, which is in essence the present work's proposal to aid navigation, is itself a rich, inter-disciplinary field with many heavily context-based avenues. Spence (2001) argues that exploration, presentation and indeed the navigation of data are each intrinsically linked to visualisation.

One should not ignore more philosophical streams of thought on visualisation such as those of Wittgenstein (2004) and in particular the arguments of Biggs (2002) and Sterrett (2004) concerning it. The representation of an abstract object or event had been, until Wittgenstein, largely a non-issue. An idea or event typically had a canonical physical form; an example might be the symbolic score for a piece of music or the script for a play. With the advent of a reusable, distributable physical form of representation for a piece of music, it becomes clearer to see how Wittgenstein's ideas came to form.

One of the key insights that Wittgenstein lends us, and which the present work implicitly draws upon, is the concept of multiple representations of the same intrinsic article, and that transformations exist in order to change from one to the other.

> "There is a general rule by means of which the musician can obtain the symphony from the score, and which makes it possible to derive the symphony from the groove on the gramophone record, and using the first rule, to derive the score again. That is what constitutes the inner similarity between these things which seem to be constructed in such entirely different ways."—Wittgenstein (2004), 4.0141.

Unlike Wittgenstein, however, the present work is concerned with the transformation between the acoustic "groove" and a second representation which may be considered more useful. Wright (1994) also offers more esoteric thoughts on visualisation. He concludes visualisations "become a focus for conceptualism, as tools for thinking. ... The scientific image can 'objectify' knowledge into visible form, but at the same time 'situating' it with respect to the forms of subjectivity implied in its reading". While the context of this work

is science and mathematics[2], it nonetheless concludes that visualisation is a useful means of dissemination of knowledge to the public.

Psychology

Psychology and psychoacoustics play an important role in providing systems and theories with which working models of our appreciation of musical audio may be created and developed. Psychoacoustics improve our understanding of the brain's representation of sound, and thus how it may be optimally represented for a knowledge-based or general signal-processing system. Psychology and (more importantly) music psychology can help us to understand how music is different from other sounds; a key idea if the overall aim is to be achieved.

Since the present work concentrates on mainly low-level aspects of the musical audio, psychology plays only a small role with psychoacoustics permeating through each of the proposals. Our analysis and evaluation of the navigation aids is largely grounded in statistical tests heavily related to and used within the field of psychology.

Electronics and Music Technology

Signal processing and the general field of informatics plays a most important part in extracting useful information from the incredibly dense data source of a sound-recording. These disciplines, together with psychoacoustics, provide a pool of understanding and knowledge which is important to consider when thinking about processing sound signals. Thus music technology, and audio engineering in general, provide a solid base from which the present work advances.

Music Information Retrieval

Music information retrieval (MIR) is a very multidisciplinary field drawing people from subjects as diverse as computer science and musicology together with librarians and music-technologists. It may be viewed as being in a similar field to computer vision as it shares many of its premises; i.e. information retrieval from source signal, high-level feature extraction and search for invariant measurements. It may also be seen as a complement to information-retrieval and dynamic-library-searching, in a similar field to the currently popular topic of data-mining. Within the fields of MIR there are several on-going research avenues including segmentation of music, classification of music, methods for browsing large music archives, and music search and retrieval.

Much of the related work discussed in chapters 3 and 4 is published within the MIR community, as indeed are portions of the present work. As such MIR contributes possibly

[2]using chaos imagery as its main example

more than any other discipline to the basis of the present work. Self-similarity matrices were presented as an MIR device (for visualisation), and much of the utilisation of a Self-Organising Map for music visualisation is inspired by the Islands of Music work, also largely a MIR-grounded project.

Music Theory

In defining our aim as including *content-based* visuals, the natural source of information for the visuals is not the music notation but rather the audio signal data that is heard by the listener. I consider the application of theory such as that in Lerdahl and Jackendoff (1983) to such a low-level form of data out of the scope of this work. Though signal to notation transcription systems do exist—and indeed have improved throughout the course of this research—they do not yet perform appropriately robust or accurate.

By addressing the playback of media files, the present work is concerned with *performances* as opposed to *compositions*. With western classical music, performers will typically try to play the piece exactly as the composer intended (though subtle variations in performance may be analysed and visualised with techniques such as those described by Dixon et al., 2002a). This is in stark contrast to jazz, especially Dixieland jazz and early folk blues music, where performances will generally attempt to create a new interpretation of the composition by varying the melody, harmonies and even time-signature.

Other idioms of music, for instance electronic art music and various forms of electronica (in the modern sense of the word meaning 'experimental dance music') including IDM[3] and post-rock, may use a particular recording as the only definitive "specification" of a given piece.[4] This results either from it being unnecessary or simply impossible to describe it in a substantial way notationally; Middleton (1990) writes "in most Western popular music since rock and roll ... 'non-notable' parameters are of great and often predominant importance.". Music which relies heavily on sampling or real-time alteration (e.g. turntablism, the art of creating music through phonograph turntables, using it in the spirit of an instrument) or whose exact composition is left undefined (e.g. indeterminate and aleatoric music) are all examples of idioms where using notation for description of a piece is troublesome. Other forms of notation have been introduced to address some of these concerns e.g. graphical and prosaic notation. Nonetheless, the limitations of common music notation mean that much of the music composed in the present-day is notated only in the form of a digital recording.

---

[3] a widely accepted, though often criticised term, being an acronym for intelligent dance music

[4] This may be seen as a blurring of the boundary between composition and performance. It seems to us a reasonable result of the culture of artists by-and-large being the only performers of their work, a trend surely following from the widespread ownership of performance playback devices (i.e. record players and their modern counterparts).

1.2 Argument

I believe that inclusion of a content-based visual in popular playback navigation interfaces speeds up certain common tasks by complementing the navigation interface. I will argue this by showing that it is easy to implement via a modern signal processing framework and furthermore very fast to compute, that it does not provide a distraction and that its appearance is not considered unattractive (a concern technically less than rigorous but of great practical importance). I will first demonstrate how it intuitively shows musically important information such as hierarchical structure and secondly conduct several moderately sized user studies in order to test this empirically.

1.2.1 Aims and Objectives

The previous section presented a brief discussion of the motivation for carrying out this project. It explained the need for the work, and where the work fits in to the general discipline. The concrete objectives I aim to achieve with this work are:

- Propose a basic working theory of users' music playback navigation in terms of purpose and strategy.
- Determine methods of automatically generating visuals from musical audio which aid the user in determining the content of the track for navigation purposes.
- Demonstrate such methods working on a mainstream popular music player with commodity hardware.

To achieve these aims, the following intermediate steps were taken:

- I reviewed the current state of the art of music track navigation aids, including the HCI technology, their current usage and the reasons for usage.
- I reviewed the current state of the art for visualising musical audio.
- I designed several techniques to project small portions of audio content (audio *texture windows*) into a colour.
- I designed a trial framework for determining how useful the proposed navigation aids were.
- I determined the degree of objectivity of certain characteristics of music in order to validate the trial framework.
- I conducted trials under this and another task-based framework, and used statistical models on the results to evaluate the techniques and make conclusions.

1.2.2 Thesis Outline

Chapter 2 concerns the nature of music track navigation in playback and its history, I discuss exactly what it is used for, and detail a case study into the real usage a typical navigation system.

Chapter 3 discusses the techniques that have been in wide use for generating visuals of musical audio. I then propose several novel techniques for generating content-based visuals, in particular techniques for transforming small blocks of audio directly to colour. I illustrate and discuss these techniques with both real-world musical audio and portions of audio created specifically to test certain aspects of visualisation.

Chapter 4 continues from chapter 3 by discussing the use of dimension-reduction techniques, in particular the Kohonen Self-Organising Map neural network, in order to produce a perceptually topologically correct mapping between audio blocks and colours. I propose a novel use of the SOM for projecting audio data into colour by mapping the audio onto a pre-set hue plane. A similar version of the technique using principal component analysis rather than the SOM is also proposed. The techniques are discussed as in chapter 3.

Chapter 5 discusses the trial framework devised to test the navigation aids. I propose two methods; a simple question-answer task list and a time-limited 'boundary' search. For the second, I discuss and present empirical evidence of objective boundaries in music. The results found by conducting the trials are then presented and discussed together with statistically significant statements that may be made.

The thesis concludes with chapter 6 which discusses the contributions made by this body of work, the empirical evidence collected and statements that may be taken from it. I go on to discuss future directions for this work in terms of HCI, visualisation and signal processing.

1.2.3 Thesis Summary

Proposals Time-to-space mapped visualisations of music facilitate navigation through audio recordings; visualisations are automatically generated using a self-organising map.

Goals

- Determine how useful (if at all) a visual navigational aid is to have in common audio playback tools.
- Determine from the musical audio visualisation methods discussed which is most useful, and the reasons why this is the case.

Contributions

- Data on, and analysis of, humans' use of intra-track navigation.
- Collection of data on and analysis of actual usage characteristics of humans doing intra-track navigation for task completion.

- Development of novel methods for creating visuals from musical audio, specifically by projecting discrete audio chunks into a colour:
  - RGB-mapped bandwise magnitude.
  - RGB-mapped bandwise rhythm magnitude.
  - PCA-projected histogram features.
  - SOM-projected histogram features.
- Comprehensive evaluation of techniques for use by humans on track navigation.

Conclusions

- People are able to effectively take cues from automatically-generated visuals and utilise them effectively to summarise and navigate through audio data.
- *Psycho-acoustic* preprocessing on the audio signal results in perceptually more meaningful visuals.
- The SOM is able to generate a visually simpler image with no loss of performance over other more direct methods.
- A traditional loudness waveform envelope visual gives a less useful representation than other texture or SOM-based methods.
- Rhythmic qualities of a music audio signal lend themselves less to a useful visualisation than basic spectral surface qualities.
- Absolute meaning in terms of colour is not a prerequisite to a useful audio visualisation; relative meaning is enough to provide a helpful image.

1.3 Chapter summary

This chapter has presented an introduction to the work of this project. It has introduced the principal topics of research and discussed the motivation for the investigation of these particular areas. The discussion of the problem resolved into a statement of the aims and objectives of the work, and how these relate to the specific requirements to be met. Following the presentation of the motivation and central themes of the thesis, an outline of the thesis as a whole has been given.

Chapter 2

Music Playback Navigation

"I see my path, but I don't know where it leads. Not knowing where I'm going is what inspires me to travel it."

—Rosalia de Castro (1837-1885)

2.1 Introduction

I begin this chapter by introducing the concept of navigation within a musical audio recording. This is important to the content of the thesis to make the reader aware of the specific reasons for navigating at all, the variety of navigation mechanisms proposed and popularly used and the difficulties that they present the user.

Chapter Summary

Following the present introduction, the chapter will begin with a review of the literature most directly concerning musical audio track navigation. I cover the popular navigation bar UI mechanism evident in all mainstream musical audio playback software. I review other techniques of navigating through musical audio such as the *Link Player* and the rhythm-metadata enabled navigation. The review is the broadened to include techniques proposed for navigation audio generally, and then in particular spoken-word audio such as voicemail processing.

By the end of the literature review I will have discussed the current state of the art in musical audio navigation. I will have demonstrated that though there is a considerable body of work dedicated to understanding user usage patterns in spoken word recording information retrieval, little has been done specifically in usage patterns of navigation for musical audio information retrieval.

The chapter continues with two studies conducted in order to better develop understanding of common aims in musical audio recording information retrieval; I describe the

questionnaire and interviews with users of musical audio playback software conducted in order to determine the sorts of tasks they think they require. I then analyse and discuss the actual actions taken by users of the random-access navigation bar when attempting to carry out a set of tasks designed around the results of the prior investigation.

By the end of the chapter the reader will have some understanding of the concept and usage of the navigation bar when used as a popular navigation aid.

Contributions

1. I present empirical data and a discussion on the uses of intra-track navigation for musical audio.

2. I present in-depth traces of the listener's experience documenting actual usage characteristics for reference task completion.

2.2 Related Work

The mature, mainstream literature (by which is meant a full volume with commentary and reference) on HCI e.g. Dix et al. (2004) and Preece et al. (1994), when considering musical audio navigation interfaces, has not progressed much further than a fairly superficial commentary on the video recorder metaphor (play/stop/fast-forward/rewind). On reflection this should not be especially surprising; PC-based musical audio playback software has not been in common use for more than around ten years, essentially since MP3 became popular through its ability to be decoded on commodity hardware. Moreover such software has only started gaining popular acceptance (i.e. outside of computer and music enthusiasts) acceptance in the time period over which the present work was produced with the advent of services like iTunes and Napster.

Musical audio (and, largely, audio) navigation interfaces can be largely broken down into two types; those that present an overall sequential and continuous metaphor, and those which provide a discretised and possibly nonlinear 'hyper-linked' metaphor. Examples of the former would be the common navigation bar and the classic fast-forward/rewind interfaces found on cassette recorders; examples of the latter would include the work by Kosonen and Eronen (2006) and ESPACE2 presented by Sawhney and Murphy (1996).

2.2.1 Navigation Bar

An early communication by Aigrain et al. (1995) gives a reasonable definition of the term *navigation bar*:

> "A scrollbar with a cursor indicating the "present" position in the document
> ... and one can directly access some relative position in the document duration

[sic] by positioning this cursor."—Aigrain et al. (1995)

Though even comprehensive HCI literature such as the volume by Dix et al. (2004) does not explicitly analyse it, it seems likely that the popular audio navigation bar stems from two well understood user interface (UI) primitives; the scroll bar and the progress bar. The progress bar is a read-only indicator of completeness. A typical example would be a download window in a modern Internet browser which will represent the amount of a file downloaded. The scroll bar is typically associated with some sort of windowed view; it is a control interface allowing the user to change (and query) which portion of a document is currently being viewed. Typically they are used because the document being viewed is larger than the limited screen space in which to view it. An example would be the scroll-bar to the right side of a web browser window, when the web page being viewed is too big to fit on screen at once.

The navigation bar combines these two to form, depending on which UI primitive you assume as parent, either a scroll bar which progresses automatically or a progress bar whose progress may be reset as desired and to which no graphical view is attached. Several examples of musical audio navigation bars may be seen in figure 2.1. One particular function that the navigation provides over the older video recorder inspired bars, is the ability to have random access into the musical audio recording, by providing a one-to-one mapping between points on the horizontal of the bar to moments in time of the music. This concept is essential for the subject material of the present work, which theorises that certain augmentations to this basic metaphor will significantly increase the utility of the UI. We will now discuss the navigation bar in more detail.

The navigation bar gives the user a visual cue, which may help to imagine (or visualise) where in the music certain events are, and where they are listening currently. This is missing from conventional playback devices such as a CD[1] and cassette recorder.

2.2.2 Musical Audio Navigation

The earliest system suggesting navigation of specifically musical audio recording is that proposed by Aigrain et al. (1995). The interface includes several representation 'strips' of the musical audio which correspond to segmentation results. Multiple segmentations are created from different feature sets including dynamics, frequencies, meter and stereo effects. The presentation of these features is somewhat cryptic and is evidently designed for the trained viewer. Navigation is performed using the familiar random access navigation-bar metaphor.

The authors value discrete representation above continuous, suggesting that a timeline

[1] a display of the current time through the track is not enough (on its own) to give an idea of how far through playing it is

(a) amaroK version 1.4, running on Ubuntu Linux 7.0.

(b) Microsoft Windows Media Player version 6, running on Microsoft Windows 2000.

(c) Nullsoft WinAMP version 5, running on Microsoft Windows XP.

Figure 2.1: Three examples of the audio navigation bar in popular musical audio playback applications. The navigation bar of the interface has been highlighted over the original image.

of "pre-organized selectable objects" are more manipulable than a continuous representation.

One of the earliest proposed systems explicitly designed for content-based navigation of musical audio is the Link Player, detailed by Blackburn and DeRoure (1998). It utilises melodic pitch-contours with a database of points where contours are found in songs. The user may ask to be directed to a different point in a different audio document matched according to the contour. As such this is not navigation between points in a track per se, but rather navigation between points in all tracks. They conclude that melodic pitch contours typically do not work especially well for this task, suggesting that information of aspects such as rhythm might be better suited.

A rhythm-metadata enabled browser was recently proposed by Kosonen and Eronen (2006). Designed for popular use on mobile devices, it allows the listener to augment the music being listened to by means of its section-skipping interface. Once in the interface, the current section will repeat seamlessly, utilising segmentation and beat-tracking technology. The user may continue playing the track from that point onwards or may move forward or backward into the neighbouring sections, replaying it until a further decision is made.

This approach is well suited to the mobile-device platform, where summary and representation methods cannot necessarily be relied upon due to the limited space for visuals. Other possible uses aside from navigation are given; users could, for instance, automatically loop their favourite parts of a track. That said, there is no significant reason to believe this form of navigation would be of popular use in a desktop environment; aside from the relatively weak discussion of the implementation the authors provide no further argument. Evidence that people would indeed utilise its functionality (e.g. a questionnaire or user study), would significantly contribute to the value of the proposal.

Aigrain (2001) was one of the first to publish a commentary explicitly referring to systems for navigation of musical audio utilising static content-based visual representations of the audio. He argues that the optimum representation-based approach would be a visual annotation that could present the content in various levels of abstraction from the acoustic to the notational:

> "But [a representation-based browser] faces the extreme difficulty of automatic transcription and also suffers from the fact that the representation hides some important features of the content (for instance, voices for speech or performance for music). Static representations of audio content take all their value when they can be presented at different scales and levels of abstraction and directly associated with sound production."—Aigrain (2001)

The present work does not attempt such an all-encompassing navigation aid for two main reasons. Firstly, and as Aigrain points out, content-based transcription is a tremendously difficult task and only applicable for relevant music. Secondly, it is an open question as to how to provide a compact interface to such information. The present work aims to provide an interface readily comprehensible and adaptable to popular playback software.

2.2.3 General Audio Browsing

General audio browsing interfaces do not restrict themselves to any particular audio content. Typically this is an easier problem to solve than information retrieval in any one particular context since the features to identify and differentiate are coarser (e.g. music from speech rather than rock from jazz). Since computation power (and general understanding of the problems involved) is forever increasing, there is a significant amount of relatively early literature dedicated to this subject.

Kimber and Wilcox (1996) present an early example of attempting to augment a basic random-access navigation system for general audio (film soundtracks is the example given) with automatic index generation from content-segmentation. The latter technology is described at length in section 3.2.3. The browser interface presents several timelines of each of the 'voices' (speakers, songs, applause etc.), which the user may utilise to direct

the random-access navigation. This would have several drawbacks for musical audio; in many types of musical audio there is not a well-defined (or usefully small, at any rate) number of voices. Furthermore, having multiple timelines takes a significant amount of the user-interface; only a far more compact representation of the content would be acceptable for a popular playback application.

Tzanetakis and Cook (1999) presented a prototype implementation of an audio browser which includes basic segment boundaries together with a CD-player like navigation interface of fast-forward/rewind along with next/previous segment. The display is representation-based giving the user an image of the audio portrayed as a basic waveform. An informal user-study suggests that automatic content-based segmentation can help with navigation through the audio document. A comparison of different persons' manual segmentation of musical audio reveals common parts labelled as segment boundaries, suggestive of an underlying 'objective' ground truth segmentation.

Comparatively, the present work explores a different approach giving the user more freedom in their control method (random-access rather than sequential jumping), though both approaches appear quite valid for ultimately easing navigation for the typical users of popular playback software.

Tzanetakis and Cook (2000b) present some follow-up work to the above. A small user study suggests once again that there is reason to expect some degree of objectivity in segmentation of musical audio, with around 70-80% of segment boundaries being agreed on by the majority of participants. The content segmented includes both musical audio and music/speech audio; no figures are given for musical audio alone.

SATIE is a piece of interactive software for expert use intended to guide the viewing of, and navigation within, musical recordings presented by Lepain (1997). In order to aid interpretation, understanding and navigation within the musical track, it presents multiple content-based and manually populated representations of the track. The navigation is a simple random access metaphor similar to the navigation bar concept previously described. At the time of publication, only the two basic representations, waveform and spectrogram, had been implemented though plans for more ambitious representations were proposed. Although founded in a somewhat different context (SATIE being developed primarily for an expert audience), this work does share the notion of providing aid for random-access navigation, though presently I will provide a more thorough investigation.

Another interesting outlook on the audio navigation problem is ESPACE2, as presented by Sawhney and Murphy (1996). Aiming to provide an accessible system for visually impaired users, it is an audio-only user interface which provides a hierarchical and spatial interface through the concept of moving in and out of virtual 'rooms' (in a similar way to the work presented by Kobayashi and Schmandt, 1997, mentioned later).

These approaches, while ambitious and possibly useful to a minority, are hardly appro-

priate for a mainstream application due to, amongst other things, extra costly equipment required (multiple speakers), the deterioration of listening experience and the training required for a relatively unfamiliar interface.

2.2.4 Speech Audio Browsing

Speech audio browsing is another form of audio navigation. The content of the audio is that of spoken-word and applications tend to focus on recordings of either single speakers (e.g. voicemail) or multiple speakers (e.g. diarization). The challenges faced when making a good speech audio browsing interface share a considerable amount in common with those focusing entirely on musical audio. The solution, argues Kimber et al. (1995), is similar to that proposed in the present work:

> "It is difficult to find specifics in audio recordings because it is necessary to listen sequentially... Although it is possible to fast forward or skip around [using a random access interface], it is difficult to know exactly where to stop and listen. For this reason, effective audio browsing requires the use of indices providing some structure to the recording... These [indices] can be displayed graphically as a navigational aid in browsing."—Kimber et al. (1995)

The interface proposed and prototyped by Kimber et al. is similar to the other audio system being proposed at the time by Kimber and Wilcox (1996), both focusing on multiple timelines depicting the onset of various voices.

Kobayashi and Schmandt (1997) suggest utilising people's natural spatial-awareness to aid the navigation of audio by mapping the time of the audio document to a spatial metaphor:

> "The motion of the sound sources maps temporal position within the audio into spatial location, so that listeners can use their memory of the spatial location to find a specific topic."—Kobayashi and Schmandt (1997)

The *NewsComm* system described by Schmandt and Roy (1996); Roy and Schmandt (1996), proposes and prototypes a device capable of aiding the browsing of audio recordings of the news. It relies upon the segmentation of the audio into one of a number of speakers or silence in order to give a discrete representation of the audio and to aid users with a next/previous speaker interface similar to that described in Tzanetakis and Cook (1999).

The *Dynamic Soundscape* project, a follow-up work to *NewsComm*, spatialises several streams of audio at various points throughout a recording (spoken word) so that they have the effect of apparently coming from various points around the user. They will naturally use the 'cocktail party effect' to listen to all streams and, when appropriate, 'home-in' on

a stream of interest by stopping all but a single stream of audio which corresponds to a section of the document in question.

The work of Hirschberg et al. (1999), Hirschberg and Choi (1998) and Nakatani et al. (1998) is particularly interesting to us at present since it details a system proposal together with empirical data about usage characteristics concerning random-access navigation of voicemail documents with visual annotation. The studies suggest, in the field of voicemail information retrieval, that random-access navigation on its own is far from optimal: "users often lose track of the current audio context, and being unable to determine the sequence and structure of different elements of the audio record". Furthermore:

> "obvious signposts such as topic/message boundaries may be less helpful than users expect them to be and perhaps even counter-productive to users acquiring a basic understanding of the data. Given this result, we are exploring alternatives to simple topic markers including... acoustic segmentation, particularly as a means of enhancing users' ability to extract the information they seek from the audio data that has been presented to them."—Hirschberg and Choi (1998)

We may take this as a (weak, since the context is somewhat different) indication that annotation from simple high-level boundaries may not be the best method to approach the problem of aiding navigation of musical audio. An acoustic-based representation could yet prove more useful in terms of usability, as indeed the present work postulates.

2.3 Analysis of the Navigation Bar

Due to the relatively sparse collection of literature explicitly addressing the real-world usage characteristics of the navigation bar for musical audio navigation, I conducted two small investigations; in the first I solicited information from users of musical audio playback software as to how they utilise the (random-access) navigation facilities. In the second I devised a set of reference tasks for eight users to carry out; the tasks were largely representative of the typical usage of navigation bars. The results and analysis of five of these tasks are given in the second section.

2.3.1 User Study

I conducted a small user study, soliciting answers from 16 potential users of the navigation system from a questionnaire. The aim was to find out:

- What sort of tasks the navigation interface would be used for.

- If the tasks were for information-retrieval purposes, what sort of information would be retrieved.

This initial questionnaire soliciting views from users is in line with typical HCI methodology of determining an objective method for evaluating the benefit of one interface over another. The *reference task agenda* proposed by Whittaker et al. (2000) was used as a broad guideline when forming this pilot study. To determine the options in the questionnaire, I included both my own expectations of usage as well as the thoughts of others I solicited informally. A copy of the questionnaire given is available in appendix C.

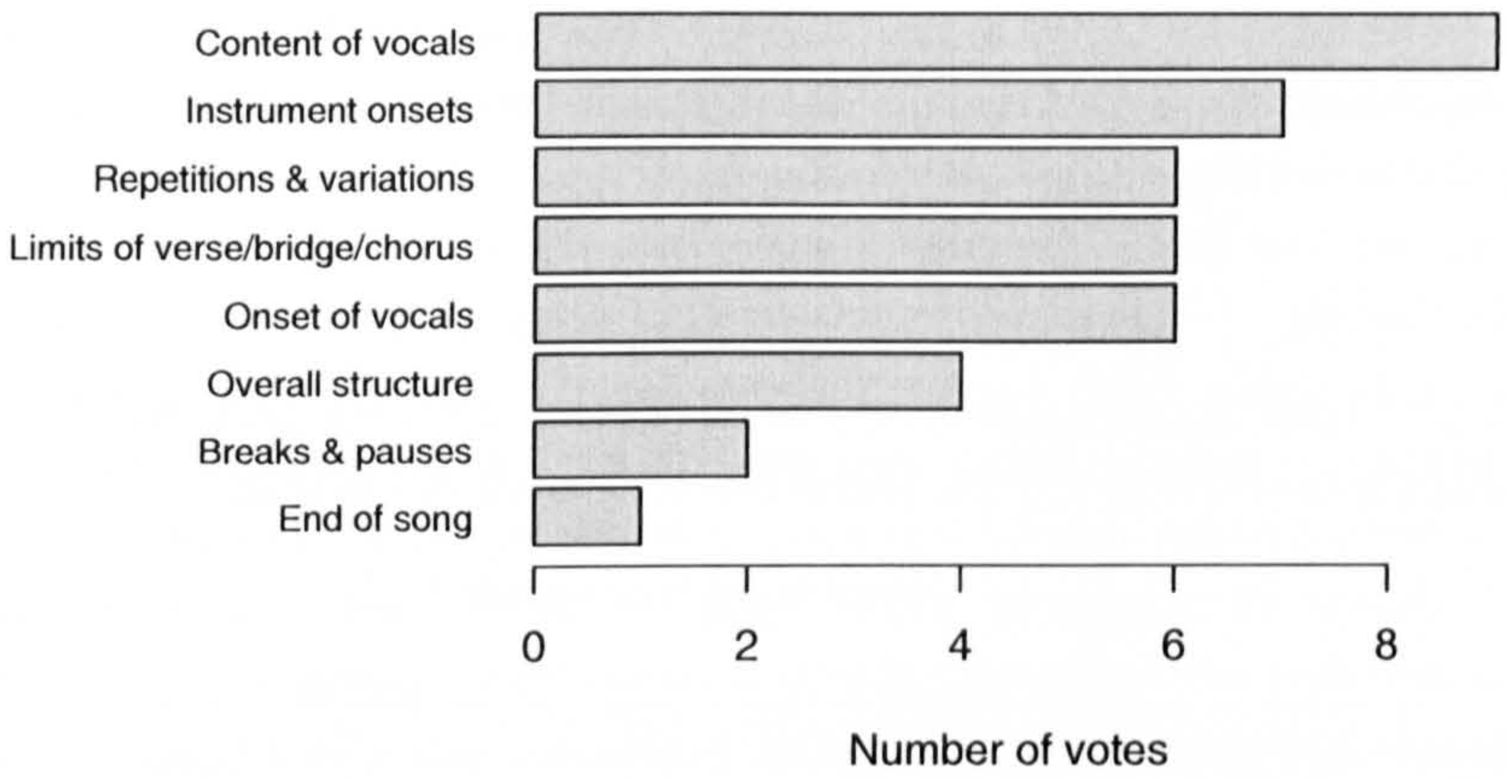


Figure 2.2: A bar plot of the results from the questionnaire given to a number of potential users asking what they utilise a navigation interface for.

The results are plotted in figure 2.2. It is reasonably clear to see from the small study that navigation is used typically to find the content of vocals in presumably popular music. Attempting to find overall structure in the music was apparently not a significant priority for *navigation*, suggesting some users imagine they would already have some idea of the broad content of the track before attempting to navigate. Finding the end of the song (i.e. the outro) and any breaks and pauses inside the track appeared similarly unneeded. Each of the other possibilities suggested appear roughly similar in popularity.

2.3.2 Task Trace Analysis

In this section I will discuss and analyse the actions taken by candidates asked to perform tasks broadly representative of those classes resulting from the previous section. This will not be considered a pure HCI experiment and as such I will not begin by building a model; this chapter aims to provide the reader with well-presented data and a discussion noting any clear trends. It is from this overview that I will consider the actions to be taken for improvement and thereby build a hypothesis proper.

Although 23 tasks were given in total, through discarding those tasks whose entire set of participants failed to answer correctly, only 22 are left. Five of these will be presented here and discussed, but the interested reader may refer to Appendix B for traces relating the other 17 tasks. Four of the tasks were finding the time of a particular onset (of either a section, instrument or vocals), and one was collecting information about the structure of the entire recording.

In all cases, the participant was alone when carrying out the task. They were told they should complete the task as quickly as possible, but that they should take every effort to ensure they gave the correct answer. They were given six minutes to practise and become familiar with the user interface and were told that the navigation bar should be used to reduce the time taken. After completing a task and entering an answer, they were *not* told if they answered correctly.

Vocals Search

Figure 2.3 shows the rock-electronica track *What I Miss the Most* by *The Aloof*. The candidates attempted to find the initial onset of vocals in the track.

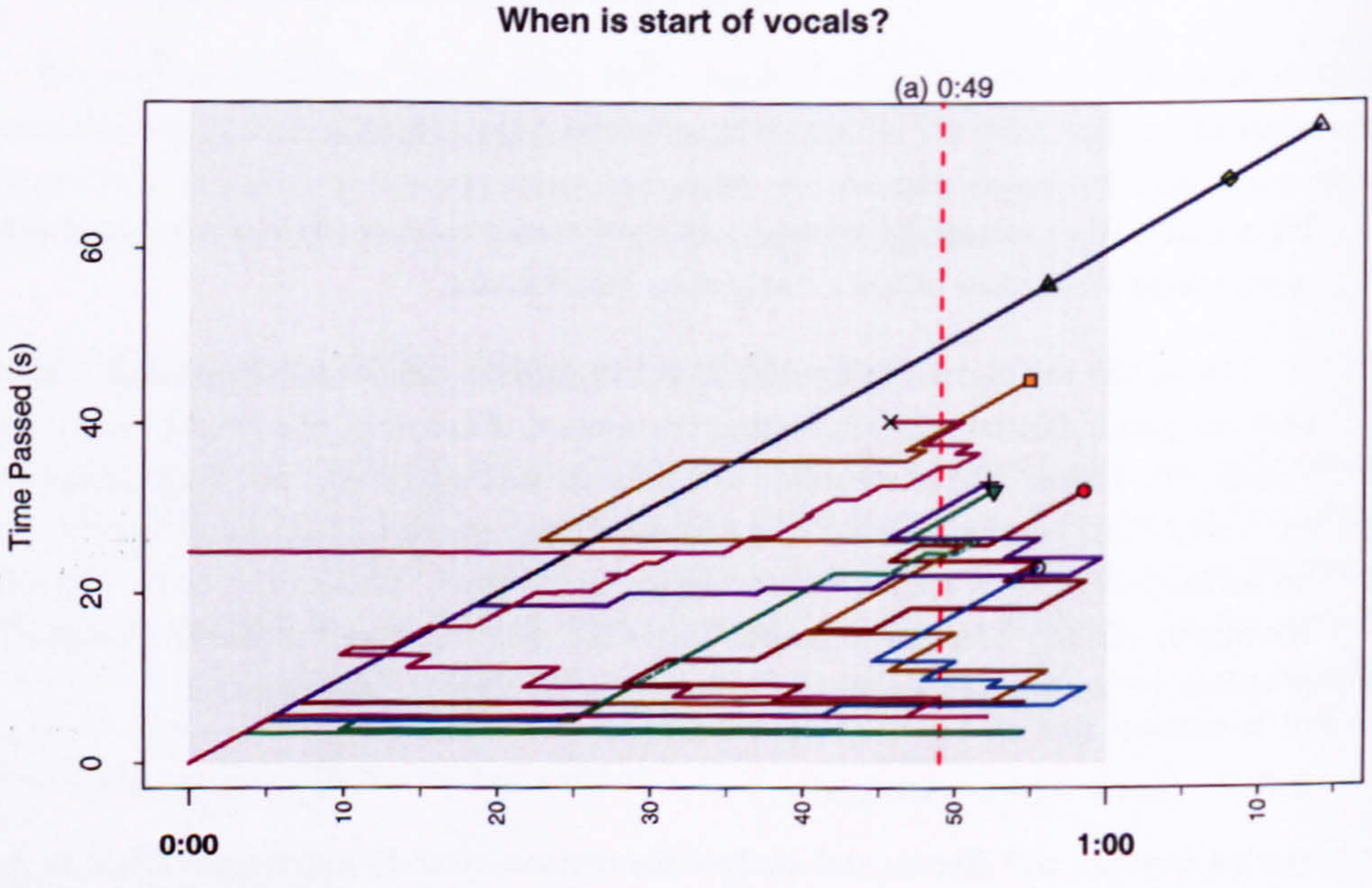


Figure 2.3: Candidates were asked to find the start of the vocals in *What I Miss the Most* by *The Aloof*. (a) marks this point.

Of all participants, only one (hollow circle) appeared to follow an effective systematic approach using random-access navigation. She made three large jumps of around 15-20

seconds each, listening for only a few seconds between each until she arrived in a vocal section; what is essentially a trinary chop search delivered her at the required point.

A second participant (filled circle) seemed to be comfortable utilising the random-access potential of the UI, but her search is less than efficient; she overshoots, undershoots and reviews several portions of the track in quick succession. Interestingly she never actually listens to the onset itself, apparently guessing correctly by picking a point somewhere between the vocals and instrumental sections.

Three participants (the upright triangles and diamond) did not utilise the random access nature of the navigation bar at all, deciding to listen through almost a minute without interruption. It seems unlikely that all three actually liked the music so much to abandon usage of the navigation bar, so it seems likely they took this course of action due to the absence of clues as to where the vocals might begin (except that they should be near the beginning). Another (upturned triangle) jumped a small way into the track and essentially listened through, possibly under a similar reasoning.

One participant (the square) found the correct onset point within 15 seconds, faster than any other. However, rather than stopping their search they instead appeared to verify it was indeed the *initial* onset by listening to a larger portion before it (0:40 to 0:50), and then skipping even further back to 0:22 and listening for another 10 seconds. This theme of requiring multiple runs for verification continues throughout the task traces.

In conclusion, it seems most participants either thought they were unable to use the navigation bar without more immediate information to the contents (and so listened through large portions of the song) or, despite trying, found it difficult to make particularly effective use of it.

Structure Assessment

As a follow up task to be carried out directly after the previous task, participants were asked how many choruses the song had. This required two key facts; firstly they needed to assess the song and find the portion representative of the verse. Secondly they had to determine on how many separate occasions this part of the music was played.

Figure 2.4 shows the trace of the six participants who correctly answered the task '3'. Each of the three choruses are shown, notably there is a bridge following the third verse (i.e. where one might expect a third chorus to be) between (b) and (c) at around 3:30–4:00. Three participants (upturned triangle, square and diamond) utilised the random access to review this portion of the track presumably to check that it was not another chorus.

In general, three of the six participants (triangles and square) tended to use the random access to effect a sort of fast-forward-like mechanism, quickly building a series of representative blocks (around five seconds each) of the song by skipping around ten seconds between them. This is a reasonable usage for random access given that the question covers

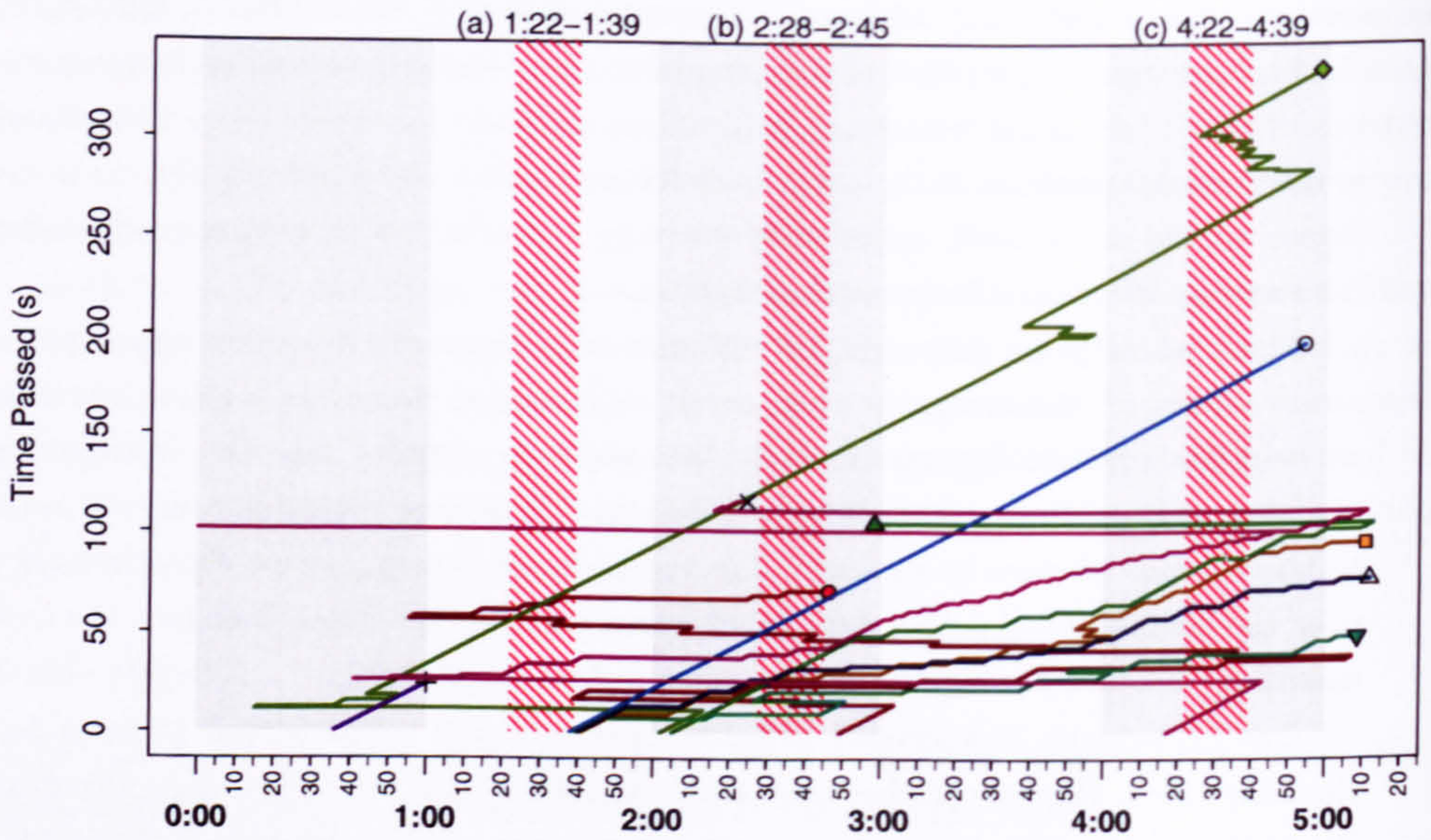


Figure 2.4: Continuing in the same track from figure 2.3, candidates were asked how many times the chorus was played in total. (a), (b) and (c) each mark the chorus sections.

the whole song and no further information is given.

Two participants (diamond and hollow circle) listened through the entire track, duplicating the approach shown by three in the previous task. One of them does however briefly review two sections (the aforementioned bridge and the final verse). This suggests usage of the random access for verification.

One of the candidates (filled circle) skipped through the track by about 30 seconds each time, listening for around five. After getting to the end of the track, she reversed direction until finding the first chorus, then the second. It seems she aimed to get a broad overview as quickly as possible, listening to only very short sections and making large jumps between them. Unlike the other participant (upturned triangle), she apparently required verification of the middle structure before making the final answer.

Overall trends in this task seem to be very suggestive of utilising the navigation bar to skip around large sections quickly for gaining a long-term summary and verification of suppositions for the structure. This 'whole track canvassing' appears to play a consistent role throughout the task traces.

Instrument Introduction

We move on to another piece of music now, this time the classical track *Warm Air* composed by *Mike Batt* and performed by *Royal Philharmonic Orchestra* with "child prodigy" *Vanessa Mae* taking a lead role on the violin.

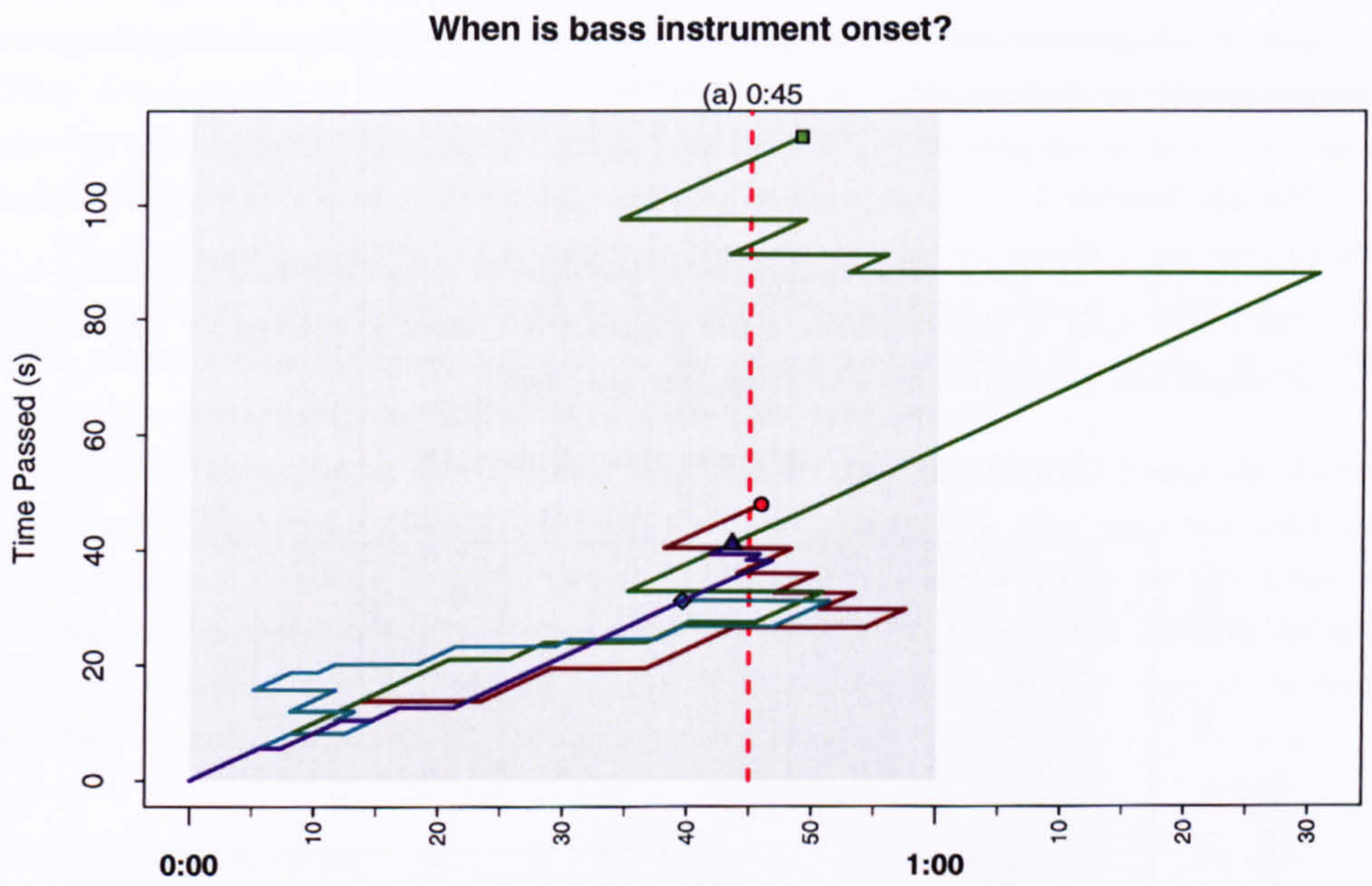


Figure 2.5: Candidates were asked to find the onset of the bass instrument in *Warm Air* (*Vanessa Mae*).

The participant to finish the task first (diamond), aside from several jumps at the beginning to apparently check she hadn't passed it around 10 seconds in, finds it very simply through skipping forward and apparently making a lucky guess at how much before the destination of her penultimate jump it starts. A second participant (circle) had a similar approach, though maintains a roughly equal skip-listen ratio (about 6-7 each time). She, however, overshot by considerably more (around eight seconds past) and was very close at the point of the jump before, so while almost making a perfect binary search it still took her several jumps to locate the onset. Nonetheless, her trace demonstrates an effective and systematic search.

The third participant to complete the task correctly (square) skips to exactly the point of onset after around 25 seconds. Presumably suspecting this, she reviews the previous 15 seconds. Her performance would be similar to the other two were she not then to continue listening through almost another minute of the track. I would advance two possible reasons for this; either she neglected the directives for the task and simply wanted to enjoy the

music at the cost of a quick completion, or she was unsure about whether the onset she heard was the appropriate one. In any case she then reviewed the onset twice more before recording her answer, undershooting herself by around ten seconds with the initial jump back.

This task demonstrates an efficient use of random access in searching, but through a lack of information overshooting and undershooting known points for reviewing content appears to be an issue.

Vocals Search 2

Candidates were again asked to determine the time of initial introduction of the vocalist. The music used in this occasion is the reggae track *Money*, written by *Pink Floyd* and arranged and performed by *The Easy Star All-Stars*.

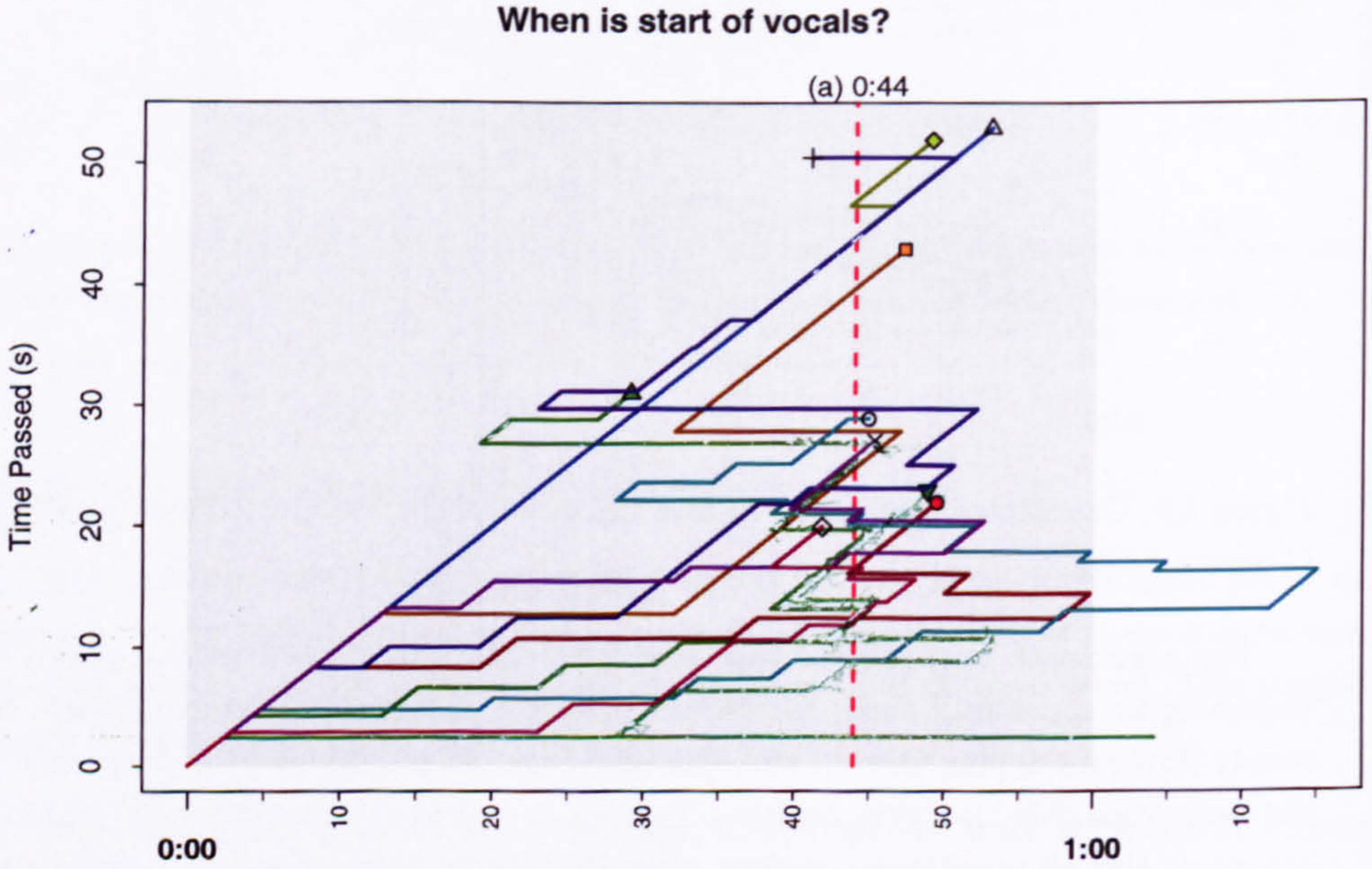


Figure 2.6: Candidates were asked to find the start of the vocals in *Money* by *The Easy Star All-Stars*. This point is marked (a).

The two participants who took the longest time to complete the task (diamond and hollow triangle) listen through the track without skipping until the vocals start. A third participant (square), though clearly hearing a long run up to the vocals (0:32 to 0:46), opts to replay the same portion of the track once again; the only explanation for this course of action I can offer is that they felt the need to become comfortable with the early contents of the track before committing to an answer.

The two fastest-completing participants (filled circle and inverted triangle) skip around fairly radically listening to about 2 seconds before skipping about 10 seconds. When they inevitably overshoot they reverse direction eventually arriving at the correct point. Notably, one of the two (inverted triangle) reviews the onset point before answering correctly.

Despite having already found where vocals start, one participant (hollow circle) canvasses a significant portion of the track (up to around 1:15) before navigating backwards. They again overshoot into the instrumental part, but again appear to want a 'broader view' of the track since they jump further back (to 0:27), reviewing the early portion again before skipping forwards and correctly arriving at the onset.

The final participant (filled triangle) is played the onset of vocals three whole times after canvassing the beginning and finding a perfect excerpt. Despite listening to the same period before the onset three times she navigates back to 0:18, almost 30 seconds backwards, presumably to verify that it is the first onset point.

The overriding theme here is one of verification; three participants found the correct point before skipping backwards through the track, suggesting they were not confident that they had not missed an earlier onset. Canvassing also seems to be a recurring theme, whereby participants make long sweeps, skipping significant portions and listening for only a few seconds at a time. I would expect this is in order to get a broader view of the track possibly to back up their mental image of its structure.

The Beginning of the End

The final task I will review here is a search for the outro of the rock track *Can't Get Enough* by *Suede*. The task was phrased to give participants the knowledge that the outro had no vocals and comprised the singer singing a recognisable 'aaah'. This, unlike other searches, gave them a solid search strategy (i.e. find the transition between vocals and 'aaah'); once vocals had been found, the transition may be located efficiently through a binary chop search with the end of the track.

One participant apparently realises this premise and acts accordingly; she skips using a reasonably precise binary chop with the end, listening for around 3-4 seconds per jump. On hearing 'aaah' first time (4:02), she retraces her steps. Interestingly, another participant (triangle) answered correctly first; she employed a similar strategy, though was more aggressive, skipping far more of the song initially. Having listened to a portion of the 'aaah', she notably decided to jump forwards again, listening for another five seconds before retracing and arriving at the transition. A further participant exhibited similar behaviour, apparently requiring a broader context of the 'aaah' section before being happy to retrace.

The participant to finish penultimately, actually managed an almost perfect first jump, overshooting the transition by only five seconds. Having listened to the 'aaah' of the outro

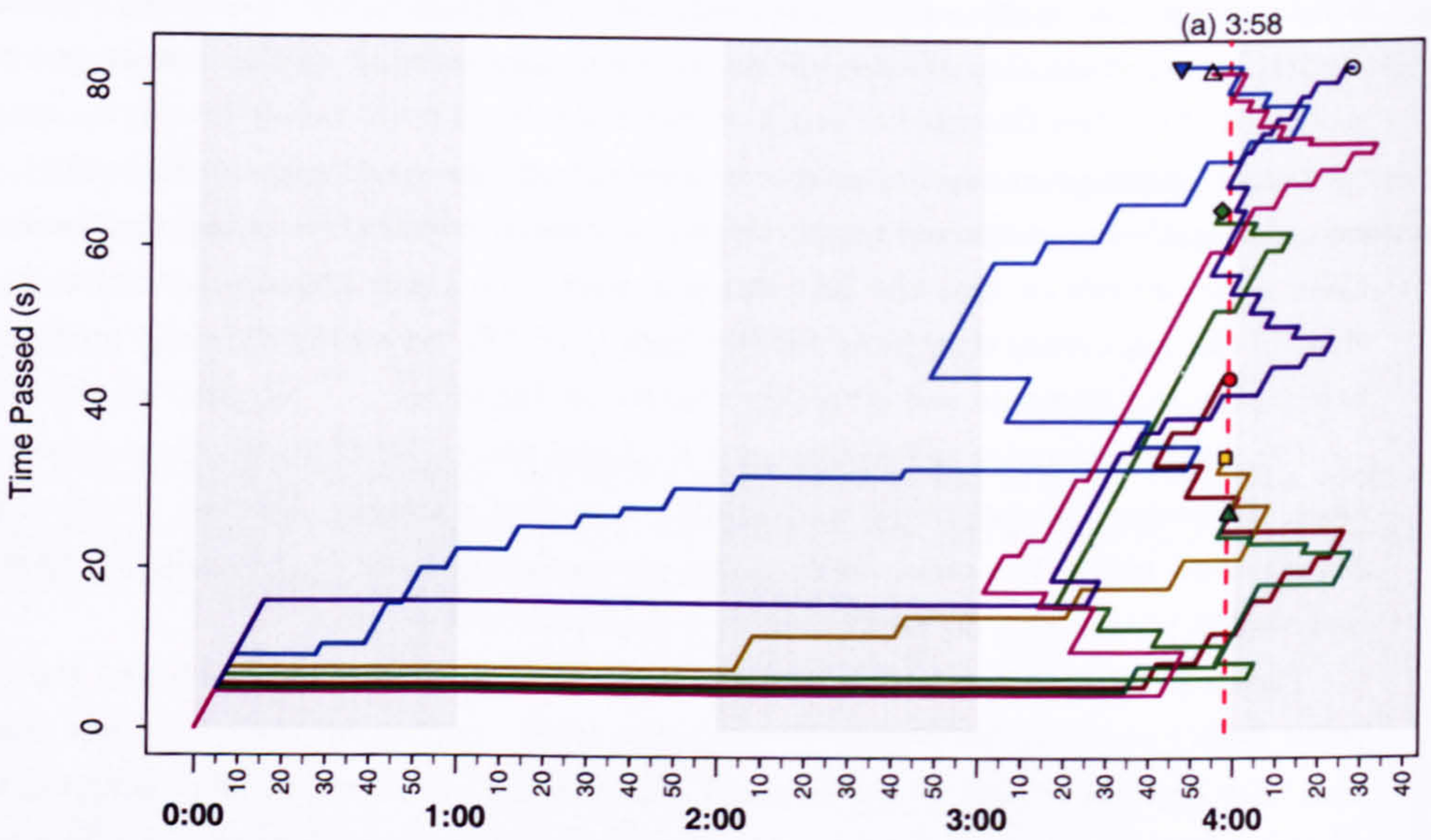


Figure 2.7: Candidates were asked to find the beginning of the outro of rock track *Can't Get Enough* by *Suede* (a), being informed it started with and consisted of a vocal 'aaah'.

for around three seconds, she then skipped back a small amount and actually heard the start of the transition. Despite this she skipped backwards, further into the vocals several times over, before eventually allowing the end of the track to play through to the transition point. This is again suggestive of the lack of confidence the participants have in a 3-5 second context.

The participant finishing last (inverted triangle) canvassed slowly for around 30 seconds before jumping to within around ten seconds of the transition point and, finding only vocals, retraces her steps twice. Having arrived two seconds into the outro, she again jumps further in, before retracing and finding the transition point. This behaviour is once again suggestive of an unwillingness to trust the logical combination of a single sampling with information pertaining to the location of the point in question. Because of this, it costs several more jumps and seconds of listening before the search can continue properly and the point found.

2.4 Conclusions

I reviewed the current state of the art of popular music track navigation aids, including the HCI technology, their current usage and the reasons for usage. I conducted a study

resulting in a histogram of task types that are commonly found among users of audio playback software. Given this I created a set of reference tasks of which five I analysed the behaviour of various users when carrying out. From this I can make two main conclusions concerning the behaviour of users while utilising the random-access navigation bar.

Firstly, of those that gave the correct answer, those who used the random access to actively search out an answer generally did better than those who did not, letting the content play without interruption or jump. In absence of any other information, I would suggest the reluctance to use the random access properties of the bar is due to a lack of confidence that they would not skip something important. Thus I favour the idea that giving an indication, where possible, that they would not skip anything important would lead to users being more confident with using the bar for random access.

Secondly, when users managed to use random-access properly and skip to the right place, there is a tendency that if the skip destination is within about three seconds of the very point they are attempting to locate, they will skip further backwards by several seconds. This suggests they are unhappy having found an apparent "local optimum" and require a review of the point in a wider scope to make sure it is in fact correct. Where prudent, reinforcing the idea (perhaps by visual indication) that the content is relatively constant around the apparent "local optimum" may help them avoid any unnecessary skipping and listening for a wide-area review.

This concludes the review of musical audio navigation technology. In the next chapter I will focus on techniques for visualisation of musical audio, in order to determine how best the navigation bar user-interface metaphor can be augmented in order to provide 'visual indications' that I would argue can improve the utility of the navigation bar for the reference tasks.

Chapter 3

Musical Audio Visualisation

"We have to remember that what we observe is not nature herself, but nature exposed to our method of questioning."

—Werner Heisenberg (1901–1976)

3.1 Introduction

Thus far in my argument, I have shown that people find navigation within a musical track useful. I will demonstrate that the mechanism used for popular audio navigation can be augmented with a visual generated directly from the audio signal. At the end of this chapter I hope to have shown that these images are, by and large, reasonable *visualisations* of the musical content of the audio signal, in so much as they are representative of the kinds of aspects we would want to identify.

Chapter Summary

Following the present introduction, the chapter will begin with a review of the literature most directly concerning musical audio visualisation. Relatively basic methods like the amplitude graph and the spectrogram are covered. More involved techniques such as self-similarity matrices and timbregrams follow. Techniques meant for performance analysis (e.g. the performance worm) and for browsing and selection (e.g. music icons) are then reviewed. Also covered are techniques meant for professional analysis used in various software packages as well as techniques developed primarily for their aesthetic value and not for representative content.

General analysis techniques are then reviewed which, though not explicitly proposed as visualisations, appear to be potential candidates for the refinement process nonetheless (e.g. segmentation, novelty). A brief review is made of the literature not directly connected

with musical audio visualisation, but concerning the visualisation of music generally; the problem of representing musical content.

By the end of the literature review I will have demonstrated that a technique accomplishing exactly the sort of visualisation I would want for the navigation metaphor has not yet been proposed[1] though several techniques might, with modification, be possible candidates. I also hope to have demonstrated the challenges of reducing such a complex entity as a musical recording to a representative image, both in terms of meaning and in more practical terms of generation from content.

I then propose a general methodology of creating content-based visuals for popular navigation, which relies on a 1-dimensional series of colours to represent the content of the music as the recording plays through. I propose several methods to generate those colours, which I term *chromatic projection* of the audio. I demonstrate each method before making an example-by-example comparison between methods over various genres together with a few contrived 'test tones'.

By the end of the chapter the reader will understand the strengths and weaknesses of each of the methods over several types of music.

Contributions

- A formalisation of a general technique for generating visualisations together with three novel concrete techniques.
- Discussion of these techniques, their advantages and problems and the relations to existing techniques.

3.2 Related Work

I will break down the various visualisations into five groups:

Traditional signal visualisation A field review would not be complete without the traditional and widely adopted methods for visualising not just musical audio but audio and signals in general.

Self-similarity The self-similarity matrix is perhaps the purest piece of work related to musical signal visualisation. It was introduced as a method specifically for visualising musical audio and has spawned several techniques based upon it.

Structure extraction This field concerns work in music thumbnailing, fingerprinting and segmentation. Having such a high-level refinement of the data is clearly ad-

[1] or, perhaps more accurately, had not been proposed when the initial survey was made

vantageous for visualisation from the sample graphics in the publications from this field.

Visualisation and representation of music composition I conduct a brief study on the thoughts and currents running through the field of music notation. This is not particularly *practical* in terms of musical audio visualisation, but does give insight into the sorts of representations (aside from traditional music notation) that have been proposed.

ad hoc visuals in software The most practical pieces of work in this field survey; I review a selection of common and niche audio and music software packages to discuss the extent of their visualisations.

For an overview of musical audio signal processing generally, I would refer the reader to the thesis of Hainsworth (2004), who spends most of the document going into far more depth then is required presently.

A Note on Information Visualisation

Spence (2001) tells us that visualisation can be broken down into several discrete processes, however it is useful to define visualisation in terms of two of these in particular, since it will be the two that I concentrate most on in this work:

$$\text{Visualisation} = \text{Data refinement} + \text{Data presentation}$$

Refinement is the stripping and transformation of the initial data into a usually but not necessarily smaller data set. It is in this process that aspects of what the viewer wishes to see from the data is extracted. The efficacy of a refinement might be illustrated in loose terms of precision and recall; a good refinement will contain as much of the data relating to the information to visualise as possible (recall) and as little data as such unrelated (precision). As such this stage is entirely context dependent, since no decisions can be made about what data to be cast aside or amalgamated unless one knows what the data means and thus how it relates to the information contained within.

Data presentation relates largely to putting this extracted data into a view most befitting the situation, which comprises at least the phenomenon to be visualised and the expected viewer. As such, data presentation is less dependent on the context of the data, and is more dependent on human factors. The use of positions, colours, sizes, dimensions, layout, topology and symbols play a large part in this stage. It is the responsibility of this stage to present the refined data as clearly as possible to the viewer.

In terms of musical audio signals this becomes:

Musical audio visual = (Signal preprocessing + Audio analysis) + Visual construction

Signal processing goes largely understated in music visualisation techniques. It will typically be of the general form of conversion to some frequency-domain representation with some psychoacoustic processing typically either MFCC or critical banding and some perceptual loudness scale.

A Note on Formal Description of Visualisations

For each explicit method of visualisation, I will consider the type of visual construction it generates. I describe this by formally describing its space-time dimensionality; I denote this by writing a single S for each space dimension and a T if the visualisation is time-based (i.e. animated). I also denote the degrees of freedom for each component making the visual; generally this will be either 1 (monochromatic/linear scaling) or 3 (full colour), though (as will be demonstrated in the next chapter) could conceivably be 2. I will define 0 to mean a purely binary value. A spectrogram, for instance is denoted *SS-1* since it has two spacial dimensions but is viewed statically (i.e. unchanging through time) and each point in space is represented as a linear value. A static waveform, by contrast would be *SS-0*, meaning it is still represented as a 2-dimension image, but that each point in the image may either be part of the waveform graphic or not.

3.2.1 Traditional Audio Visualisation

Here I review the two main visualisation techniques which can be used on musical audio signals (and, indeed, any signal) which are widespread but relatively simple; the waveform and the spectrogram.

Waveform

The waveform is the single most canonical method of viewing audio. It is essentially a time-amplitude graph of the displacement wave which gives rise to the sound. In terms of a method, it is nothing but a visual construction from the source audio data (hence canonical). Unlike any of the other approaches here it needs no signal preprocessing or analysis. The visualisation's formal form is *SS-0*, though it may be seen in an animated form where short excerpts from the signal are visualised at a time (this is often found in real-time applications such as the screen of an oscilloscope). In this case the visualisation becomes extruded through time, formally making it *SST-0*.[2]

[2] I would note with a small amount of humour, the spiral groove of a vinyl record could be considered an SSS-0 'visualisation'—or perhaps actualisation—of the waveform.

Variations on this visualisation method generally involve a change of scale or meaning of amplitude. A common variant is to use the RMS (root mean-squared) amplitude of the wave. Another is to use a logarithmic scale of amplitude to construct the visualisation; both of these are utilised in the Audacity sound editor (Mazzoni and Dannenberg, 2005).

Figure 3.1: An example of a amplitude waveform (top) and the same audio on a dB scale (bottom). The audio for *Prague Radio* by *Plaid* was visualised. While the blue area is taken up by the wave itself, the light blue areas represent the RMS of the wave. Constructed using the Audacity sound editor.

When used to visualise musical audio, a rather uninspiring image is found. Certain aspects of the audio such as large changes in dynamics or spectral content may possibly be visible, but the visualisation is difficult for anyone but an expert to understand.[3] Due to its ubiquity and simplicity this representation is used as a baseline against which other techniques may be compared.

Basic amplitude has some bearing on perceptual loudness but psychoacoustic methods are much more accurate, and smoothing through time helps to give the viewer a clearer idea of mid and long-term changes by reducing the apparency of short-term dynamics. The visualisation is still limited in a musical sense; timbral, melodic and harmonic content are largely invisible. Manually created diagrams such as those proposed by Brinkman and Mesiti (1991) use the basic concept of a time-loudness plot to demonstrate the dynamics progression throughout an orchestral piece for each of the instruments. As such, short and medium-term 'noise' is never introduced; the result being a diagram showing only the most long-term and (according to the author of any particular diagram) subjectively-important changes to the dynamics.

Spectrograms

The spectrogram is a time-frequency image of an audio signal. It is related, but not equivalent, to a Fourier transform of a signal. Spectrograms show the power or magnitude of a particular frequency of sinusoid at a particular time in the signal, whereas the Fourier spectrum of a signal shows the magnitude of a given frequency of sinusoid throughout

[3] An example of such an expert might be Arthur G. Lintgen who, according to Holland (19 November 1981), is able to 'read' vinyl records from the patterns of grooves.

the signal; since the advent of the Fast Fourier Transform (FFT), finding the Fourier series (as well as other related transforms such as the Discrete Cosine Transform) has been computationally cheap.

The spectrogram is therefore a 2-dimensional spatial representation of a piece of musical audio; each component of the figure may take only a linear value (i.e. the magnitude) and so O formally declare it *SS-1*. Figure 3.2 shows a spectrogram. The 'dancing bars' animated visuals often seen with electronic audio playback systems, which formally I would call a spectrum analyser, is directly connected to this visualisation method in that is an *SST-0* projection of otherwise the same basic data.

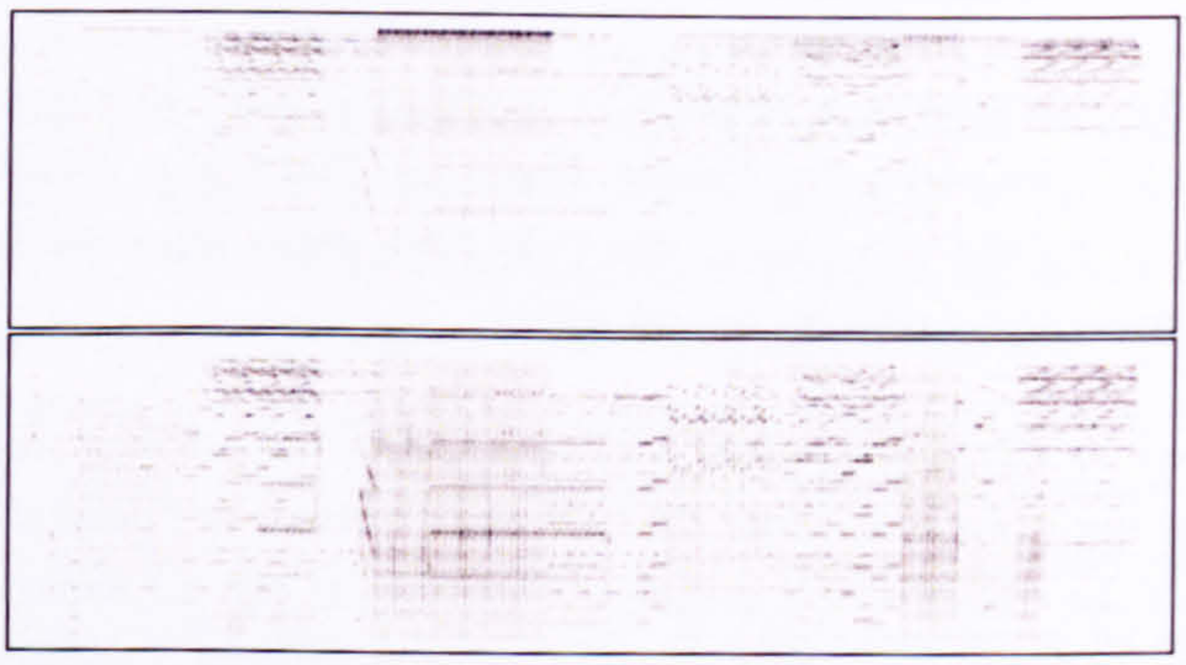


Figure 3.2: An example of a spectrogram (top) and the same audio on a Phon scale (bottom). The audio for *Prague Radio* by *Plaid* was visualised. Constructed using the Geddei Nite audio analysis tool.

In terms of the visualisation stages, spectrograms will typically have no analysis stage; the signal will be preprocessed by some time to frequency-domain transform and it is this data which will be used to construct the image directly.

In practice, the FFT is used to calculate the spectrogram using the process of *windowing*, whereby the signal is split into multiple portions and each portion's Fourier series is calculated separately. The process of constructing a time-frequency structure in this way in known as the Short-Time Fourier Transform, or STFT. Special windowing functions are applied to each portion in turn in order to reduce *edge effects* or distortions of the Fourier series caused by having a finite 'audio block' as a signal. Common window functions used are *Hamming* and *von Hann*[4]. There is an introductory text on this subject by Hamming (1998) himself.

The process of STFT is very typical in the audio analysis literature, though relatively recently a newer form of spectral analysis has gained some popularity, known as the Continuous Wavelet Transform (CWT). The STFT has one fundamental problem; the window-size is fixed for all frequencies, despite higher frequencies being able to be analysed

[4]often confused and inaccurately called 'Hanning'

with certainty within far smaller windows. The CWT circumvents this problem by varying (*dilating*) the window size for different frequencies. The outcome (a time-frequency graph) remains largely unchanged, but higher frequencies will generally have better time resolution and the overall frequency resolution is higher. Due to the dilation characteristics, the frequency axis for the CWT is naturally logarithmic. An concise description together with mathematical definition and some discussion as to how it relates to music can be found by Alm and Walker (2002).

There are several variations on basic spectrogram which still remain true to the time-frequency graph; the frequency resolution may be scaled differently in order to make certain varieties of sound or aspects of music more accessible. *Scalograms* have their frequency axis rescaled logarithmically (as mentioned above), and displayed using an octave-based scale, similar to the well-tempered piano scale but of higher frequency resolution.

The frequency resolution may also be broken down through summation into the *critical bands* of human hearing in the Bark Scale as described by Scharf (1970) and as found throughout the literature on psychoacoustics. These are a set of empirically derived frequency ranges to which the human auditory system has particular sensitivity. Another example of a psychoacoustic scale would be the mel scale, used in the popular MFCC transform, discussed later.

Other variations include changing the magnitude scale; a logarithmic scale (dB) gives a simplistic loudness-like scale allowing much more of the content to be seen immediately. More perceptually accurate scales include the Phon scale, which gives a constant logarithmic loudness scale over all pure-tone frequencies, and the Sone scale, a metric that scales linearly with perceptual magnitude. A short description of these techniques can be found by Guessford et al. (2004).

Time-frequency plots are not necessarily limited to determining frequencies of sinusoids at particular times; there are techniques to find frequencies of beats or rhythms over time; such plots might be called *beat-spectrographs.* This concept will be discussed more fully in section 3.2.2. Other variations on the concept of the spectrograph include the simplified score rendition described by Brinkman and Mesiti (1991), whereby a time-frequency graph is used once more but the data used to populate it is taken from the score itself; this work is closely related to that of the music animation machine detailed by Malinowski (2001) and discussed later in this section.

Using spectrograms directly for music analysis is not uncommon and there is a considerable amount of literature devoted to it. Don and Walker (2006) demonstrate that music can be analysed directly with scalograms in a spirit reminiscent of that proposed by Lerdahl and Jackendoff (1983). Cogan (1984) dedicates an entire volume to the analysis of the spectrograms of the performances of a wide range of music. In her PhD thesis on the subject of short-time Fourier transforms of musical audio, Dorfler (2002) states "diagrams

resulting from time-frequency analysis ... can even be interpreted as a generalised musical notation".

While the comprehensiveness of using spectrograms to visualise music and the utility of such graphics to experts is beyond doubt, they are not without faults; they are exceedingly complex to analyse in even the best of circumstances, and they suffer from other representations by being too general; like wave form graphics they can describe speech, noise and non-musical sounds as generally and precisely as music. In this respect they suffer from information overload which makes them difficult to understand and interpret directly.

As such, the main issue with using spectrograms for a visualisation aid to music navigation is clear; they are too cryptic for a casual or novice user to become proficient in easily.

3.2.2 Self-Similarity Visualisation

There are two basic methods for visualisation of data by self-similarity in the literature; the self-similarity matrix and recurrence plots. Recurrence plots, discussed by Eckmann et al. (1987), are an older form of numerical analysis technique used in analysis of fractals and chaotic systems. They could be said to be a specialisation of a self-similarity matrix, though this goes unnoted in the literature pertaining to self-similarity matrices. Self-similarity visualisations generally present the viewer with a square image; in the case of the self-similarity matrix the visualisation is formally *SS-1*, whereas the recurrence plot, presently discussed gives an *SS-0* image.

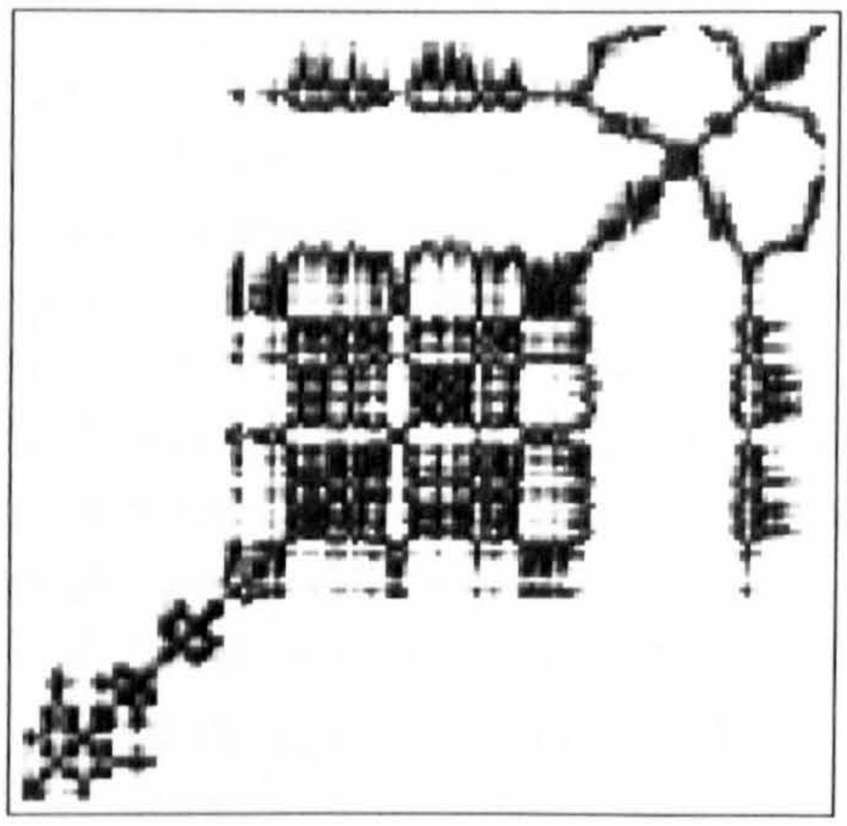


Figure 3.3: A recurrence plot of an auto-regressive process. Original image constructed by N. Marwan, *used with permission.*

Recurrence plots can be expressed formally as the Boolean matrix $\mathbf{R}$ such that:

$$\mathbf{R}_{i,j} = \Theta(\varepsilon - \|\vec{x}(i) - \vec{x}(j)\|), \quad \vec{x}(i) \in \mathbb{R}^m, \quad i,j = 1,\ldots,N, \tag{3.1}$$

where N is the number of signal states (i.e. samples) $\vec{x}(i)$, ε is a relatively small threshold distance, $\|\cdot\|$ the Euclidean norm and $\Theta(\cdot)$ the Heaviside step function.

This Boolean matrix which may be expressed visually as a bitmap will have a mark at any two times where the samples are roughly (at most ε apart) similar. Figure 3.3 shows an example of this construction.

The Self-Similarity Matrix

Self-similarity analysis is a transformation on a signal that generates a *self-similarity matrix* (SSM), a two-dimensional representation of the signal over time. Foote (1999b) proposes this transformation as a useful visualisation when the signal is that of musical audio. He argues this representation makes directly visible aspects of the audio signal such as verse and chorus repetition, thematic repetitions and variations, note transitions, on-going beats and points of novelty.

The approach prescribed by Foote (1999b) (the seminal work in this field) involves extraction of some feature-vectors on a frame-by-frame analysis of the audio. Foote used the STFT with 50% windowing and a Hamming function to reduce edge effects. On the resultant spectra he uses the mel-frequency cepstrum coefficients (MFCC) transformation, though different transformations may be applied in order to view the similarity of other aspects of the musical audio, for example chroma, loudness and so forth.

The basic SSM is defined by Foote (1999b) as being the matrix $\mathbf{S}$:

$$\mathbf{S}_{i,j} = s_w(\vec{F}_i, \vec{F}_j), \quad i,j = 1,\ldots,N \tag{3.2}$$

where N is the number of signal states (i.e. audio feature vectors), F is the series of audio feature vectors and $s(i,j)$ is the similarity function.

The key difference between the construction of the SSM and the recurrence plot is that the similarity function of the SSM is left undefined. In order to view phase differences, the recurrence plot defines a Boolean similarity function as the value equality (within limits). The example of the self-similarity matrix in figure 3.4 may be used to see the form that the two share.

The basic similarity function is defined as being the Cosine distance between the feature-vectors of the two audio blocks. Foote suggests using either the basic frequency spectrum from an STFT of the signal or the MFCC features of the signal.

$$s_w(\vec{S}_x, \vec{S}_y) \equiv \frac{\vec{S}_x \bullet \vec{S}_y}{\|\vec{S}_x\| \, \|\vec{S}_y\|} \tag{3.3}$$

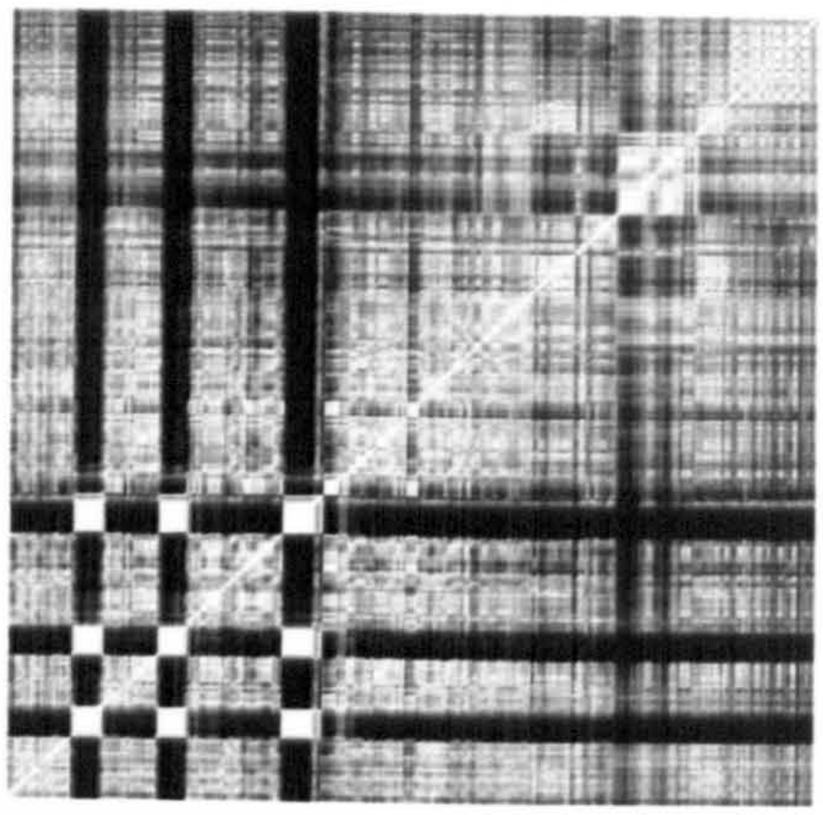


Figure 3.4: A self-similarity matrix of an excerpt from *Plaid*'s *Prague Radio*. Formed using the Cosine-distance similarity function on the Bark critical band summations as feature vectors.

As such the matrix is calculated through the cross combination of the array of feature-vectors by some 'similarity' function. He also advocates an improvement to the similarity function which uses several sequential vectors to give the (correct) ordering of the "sound" vectors a (positive) influence over the similarity. There is no mention of any other similarity measures tested[5] or otherwise consulted. There is potential for further work here to test the efficacy of the dot product similarity measure, or even whether other similarity measures can make the matrix useful for extracting or representing any other signal phenomena.

Foote postulates that retrieval could not only be made by acoustic similarity (how a piece sounds) and what appears to be the staple of the associated literature (work by Tzanetakis et al., 2001, Logan and Salomon, 2001 and Pampalk et al., 2002), but by *structural* similarity. This means that potentially it could match the same piece of music played with a different instrument.

The self-similarity matrix therefore has several advantages over other (musical) signal analysis techniques put forward. It has no need for user-specified cut-offs or other parameters. It does not rely upon absolute musical events (e.g. note onsets) to generate feedback but rather relative events (e.g. periods of lag-correlation), and thus is highly generalised.

Unfortunately, identifying discrete 'events' in the matrix tends to be somewhat more involved due to the increased amount of data to look at. Like the spectrogram it has a complex form to get accustomed to using and, perhaps more importantly it is a planar rather than linear representation of time. The matrix can give an enlightening view on the data though as it stands, due to its complexity, it is probably best left to specific expert

[5] 'tested' being a rather invalid term here, since he makes no quantitative measure of his experiments

tasks rather than general use.

Furthermore, as with many signal-based visualisations, it is clear that it works far better on monophonic music than on more complex polyphonic music. Music in general, however, tends to be realised with multiple instruments, presenting the grave difficulty of the matrix becoming so "cluttered" with features that it would make it hard to read easily. In a purely practical and aesthetic concern, integrating a naturally square visualisation into a metaphor that favours linearity and therefore a rectangular image may also be reason for concern.

Analyses on the Matrix

Though not presented as visualisations, there are two graphs from the SSM one can make which are of interest to us as potential visualisation tools; a graph measuring audio *novelty* and the *beat spectrum*. The progression of novelty over a signal is given as the sum of a Gaussian-tapered checkerboard kernel when multiplied, element-wise, by subsequent sub-matrices falling on the SSM's main diagonal and summed. The kernel matrix is given by the function K:

$$K(x,y) \equiv \begin{cases} G(x,y), & (x>0)=(y>0) \\ -G(x,y), & (x>0)\neq(y>0) \end{cases} \tag{3.4}$$

where

$$G(x,y) \equiv Gaussian(\| \left(\frac{2x}{s}, \frac{2y}{s}\right) \|) \tag{3.5}$$

where x and y both fall in the range $[-\frac{s}{2}, \frac{s}{2}]$ and the kernel matrix is of width s.

The novelty score becomes high when the submatrix is centred around a point before and after which the signal is self-similar, but around which is dissimilar. Foote (2000a) showed this can be used to segment audio and it has later been used for segmentation of music tracks which I discuss in section 3.2.3.

The beat spectrum is formed by summing the contents of the super-diagonals across the matrix; these sums form a series of lag-correlation scores for a number of lag times from the distance between successive feature vectors to the Nyquist, which here is given by half the total length of the matrix. Formally it can be described as the series S:

$$S(l) \equiv \sum_{k=0}^{s-l} \mathrm{M}(k, k+l) \tag{3.6}$$

where $\mathbf{M}$ is the self-similarity matrix over which the best spectrum is to be found.

There are many techniques to extract a beat or rhythm spectrum throughout the literature, especially in the time over which the present work was taking place. For a

comprehensive review of each of them the reader may refer to Hainsworth (2004). Foote argues that conventional methods of tempo/beat extraction must rely upon some sort of trigger, an audio characteristic signalling the onset of a beat. Using self-similarity allows a far more general method for detecting repetitive patterns. This is because the only audio characteristic necessary is that the signal must be "similar to itself", and the similarity must be periodic. This is a prerequisite for a beat to be present, similar to some of the rules regarding rhythmic well-formedness declared by Lerdahl and Jackendoff (1983) in their Generative Theory of Tonal Music.

Though not mentioned in the literature, subsequent matrices may be windowed and overlapped in order to form a *beat spectrogram*, an example of which can be seen in figure 3.5.

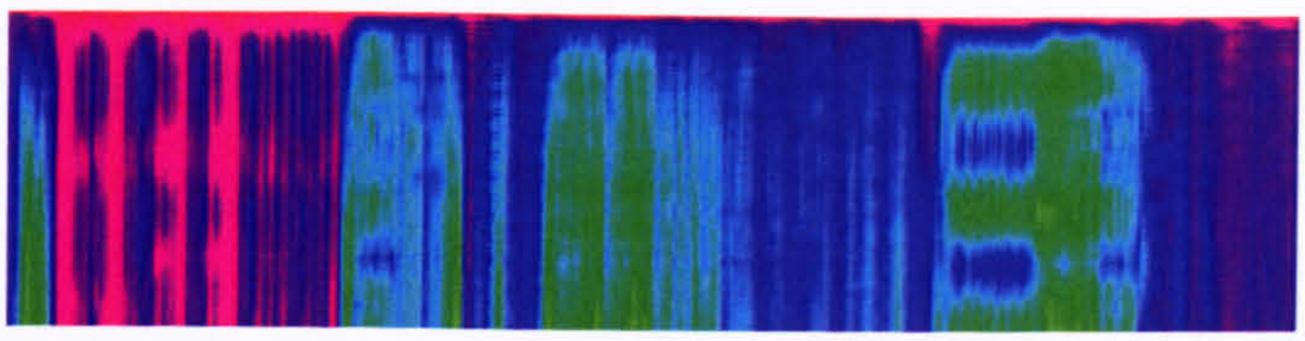


Figure 3.5: A rhythm spectrogram of an excerpt from *Plaid*'s *Prague Radio*. Formed using the Cosine-distance similarity function on the Bark critical band summations as feature vectors. Rhythm strength varies from red (highest) through blue to green.

3.2.3 Visualisation from Segmentation

In this section I discuss briefly the techniques for automatic content-based structure extraction from musical audio. Structural extraction from music attempts to determine a (possibly hierarchical, possibly labeled) description regarding the perceived structure of the underlying music. Determining exactly what the structure of any given piece music is and how one might systematically approach it is a musicological quandary, which aside from referring the reader to Lerdahl and Jackendoff (1983) I will defer until given concrete examples later in this chapter. Most researchers in the field are happy to forgo the theory and appeal to a popular opinion by comparing their results to those of humans given a similar task.

Though not explicitly proposed in the literature as visualisation methods (and thus missing the latter stage of the visualisation pipeline), they are so well suited to visualisation that authors, in describing the results of their segmentation algorithms, tend to accidentally provide visualisations of the audio tracks. I will use this accidental visualisation as my definition of the latter stage here and thus formally label it *S-1*[6].

[6] *S-0.5* may be a slightly more precise labelling, since there are typically very few segment types, and

Visualisations from segmentations are advantageous in one particular way shared by no other content-based visualisation; they give a high-level overview to the structure of the song easily with the clear and precise visual cues provided (assuming the colours chosen for the segments contrast well). Figure 3.6 is an example of such a visualisation.

Figure 3.6: A simple segmentation visualisation created from *What I Miss The Most* by *The Aloof*. Formed using the algorithm detailed by Abdallah et al. (2006). Time runs along the x axis; segments of the same colour should represent multiple repetitions or variations on the musical theme.

There are however problems with this naive way of visualising. The colours have no meaning for what they represent; only that a difference in colours represents a difference in segment type. Furthermore segments whose content is more similar than others' will not be represented. As Lerdahl and Jackendoff (1983) state and as noted by Paulus and Klapuri (2006), musical structure is typically best represented as a hierarchy giving rise to multiple levels of segmentation; typical audio segmentation approaches return a single level of segmentation only. This is likely due to the evaluation methods which typically require only a single set of segments for any track.

The above paragraph may be seen not so much as criticisms but as ideas for arriving, given segmentation technology, at a useful visualisation. The present work, however, does not follow this particular route, though I discuss it as a future direction in the conclusions.

Currents in Segmentation

Early work in the field of thumbnailing by Bartsch and Wakefield (2001) and largely reported again by Bartsch and Wakefield (2005) used a chroma-based self-similarity matrix to generate a lag-correlation matrix. This could be used to determine and retrieve the repetitive parts of the audio; given some structural assumptions on the audio (such as a verse-chorus structure) they used it to determine a representative portion on the song. This technique did not recover structure as such, but was a first step to solving a problem that would later by attempted though more thorough analysis and extraction.

Logan and Chu (2000) report of basic clustering clearly outperforming HMMs. Notably, the HMMs could not be shown to be better than random in their early attempts at structure extraction.

General audio segmentation techniques and musical audio structural analysis are heavily linked fields. Early work such as that by Tzanetakis and Cook (1999) was used for

---

thus any given time, in belonging to only one segment type, will take one of a very limited finite range of values.

audio segmentation, such as annotating boundaries between music and speech in radio broadcasts. This is useful as it can be done automatically by the application, ahead of listening. In their augmented sound editor, users may supplement automatic segmentation with their own notes.

Other work such as that by Foote (2000a), Raphael (1999) and Logan and Chu (2000) show the roots from which modern music structural analysis has grown. It is a large field with many applications; Aucouturier and Sandler (2001) have used the approach of segmentation to advance early work by Foote (2000b) in search and retrieval and numerous. Burges et al. (2005) use it for duplicate detection. Other uses include thumbnail generation (determining a representative excerpt), video synchronisation (compiling transition points and fitting them to a video stream) and section alignment.

I will briefly discuss three approaches to segmentation aimed towards the application of thumbnailing (either explicitly or apparently); the work of Foote (2000a) & Foote and Cooper (2003) and the related work of Cooper and Foote (2003) together with the work of Abdallah et al. (2005) and of Chai and Vercoe (2003b), Logan and Chu (2000) & Paulus and Klapuri (2006).

Maximising Cross-Dissimilarity

The approach proposed by Foote (2000a) uses the *novelty* measure described in section 3.2.2 to determine the segmentation boundaries. In later work, Foote and Cooper (2003) propose using a spectral clustering method to group the similar segments, creating a system of similar purpose and scope to others described. No quantitative results comparative with other techniques were presented, making it difficult to ascertain exactly how effective these techniques were.

Cooper and Foote (2003) attempt to determine segment boundaries in a similar manner through the novelty graph, though determines the labels for the segments through a further self-similarity matrix of each of the segments themselves. The similarity measure for this new matrix is calculated from the Kullback-Leibler distance. Singular value decomposition is used to determine the final labels while ignoring fine or unrepeated structures. The results published are brief but promising, though no further analysis of the technique has since been published.

Repetition Detection

Chai and Vercoe (2003b) use dynamic programming in order to deduce small segments (around 4.5 seconds) which repeat throughout a track most often and most precisely. The repeating segments that the dynamic programming are typically smaller then the musical phrase they belong to (which is what is 'really' repeating), and thus a further step is taken to merge them into fully-fledged sections.

The dynamic programming algorithm can be viewed as doing a similar job to a lag-correlation matrix determined from the self-similarity matrix as used by Bartsch and Wakefield (2001). In both cases a distance measure is used and in both cases the output is a a likelihood of repetition of data given a specific period. Two distance measures were compared in this work, concluding that a slightly modified spectral cosine-distance measure (almost identical to that used by Foote, 1999a) was generally better than a slightly more musically-orientated pitch-distance measure.

Further results of this technique are reported by Chai and Vercoe (2003a) using a chroma-based similarity measure much like that proposed by Bartsch and Wakefield (2001) on a different evaluation corpus. The results show chroma being consistently worse than the original two similarity measures. The spectral similarity measure again generally performed best.

Texture-Cluster Grouping

The most widespread approach for thumbnailing is to preprocess the audio into some relatively low-dimensionality series of 'texture' tuples. A limited set of texture 'prototypes' is generated such that each texture tuple may be classified as a prototype. Segments are sequences of texture tuples that share the same prototype classification. The key problems are finding the limited number of texture prototypes to describe the audio best (roughly known as the *dissimilarity*) and to reduce the number of segments in the classification to only those that are significant (known as minimising *complexity* and *unlabeled segments*).

The approaches vary in their exact implementation; typically the audio is transformed to a series of frames of some perceptually significant acoustic quality. Aucouturier and Sandler (2001) used the mel-scaled frequency cepstrum coefficients, whereas Abdallah et al. (2005) used the first 20 principal components of the frequency spectrum.

One approach used by Logan and Chu (2000) clusters the features directly. Other approaches use a Hidden Markov Model trained with the series of frames with Viterbi decoding to determine the prototype textures. While Aucouturier and Sandler (2001) and Logan and Chu (2000) classified segments directly with these textures, Abdallah et al. (2005) used them as the input to capture features over a longer time-scale by taking histograms of texture types over successive frames with a moving window. The histograms are then clustered with a version of the soft k-means algorithm modified to favour neighbouring histograms to share classification. Importantly, they have observed that it is the effecting of a longer time scale (they use seven beats) which produces good data for clustering rather than their particular approach with histograms of the frames' texture classifications.

Work summarised by Rhodes et al. (2006) and reported fully by Abdallah et al. (2006) advances this approach with the use of segment duration priors in a modified version of the Wolff algorithm, which affect the fitness of a given segmentation by incorporating an

expectation to the length of a segment. The authors concluded that by introducing such a prior, the problem of segment fragmentation, whereby rapidly changing portions of the audio are themselves segmented, is solved. Of course the prior is entirely 'artificial' and must be experimentally determined. However the authors found that a "suitably broad prior" was able to generate a realistically wide range of segment lengths.

Paulus and Klapuri (2006) presented a system with an interactive cost function allowing the fitness of segmentations to be dictated by varying relative values of the segmentation's *complexity* (total number of segments), *unlabeled segments* (number of textures that are not satisfactorily classed as any texture prototype) and *dissimilarity* (variance of textures classified to the same texture prototype). By varying this fitness function, the "efficient" segmentation algorithm favours differing segmentations. The interactivity allows an individual to play with the cost function to determine an optimum for any given recording. Exactly why this level of interactivity is of any use is left unexplained, and the accuracy and precision of segmentations are not benchmarked against other systems.

3.2.4 Visualisation of Tonality

Tonality visualisation as described by Gómez and Bonada (2005) is a method of music visualisation for analysing the various aspects of tonality. The musical key of an audio block is estimated from the audio signal by combining several low-level spectral features with a (presumably music-theory-based) tonal model. Gómez provides several involved visualisations for analysing a given piece of music.

This work is clearly meant for musicians since the content of the visualisations is analytic and fairly complex; though the authors do note that a foreseen use would be in studying the musical content of multiple tracks at once. One particularly interesting visualisation he presents is the *KeyScape*, which is an *SS-1* visualisation plotting the key as a hue in a time-locality space. This is similar to the earlier work of Sapp (2001). The long term key (i.e. over the whole track) determines the colour of the top of the image; the next row down is split into two portions whose keys are self-similar and coloured according to the keys found in either. This splitting and colouring continues on down the y-axis, with the time-scale getting gradually smaller and thus locality getting greater. The bottom of the image denotes the keys of the individual chords of the track. The use of the y-axis as a time-scale dimension is reminiscent of the tree visualisation for structural decomposition in Lerdahl and Jackendoff (1983).

3.2.5 Visualisation for Content-Indication

The current of work pertaining to the visualisation of audio for indications of that content of a track started with the *Timbregrams* of Tzanetakis and Cook (2000c). Timbregrams are a small part of a larger body of work known collectively as *Marsyas*; the software

and systems which Tzanetakis reported on for his PhD, and which formed a considerable part of the initial body of work of the field now known as music information retrieval. Tzanetakis introduced Timbregrams noting that being a content-based visual they could give the viewer a cue of whether the audio in question was speech or music; this later included telling apart certain genres of music such as classical and rock.

With the Timbregram, the audio is depicted through a reduction of the data from a small number of features extracted. These features are several simple statistics based upon the spectrum, together with several other values describing aspects of the rhythm. The latter values are derived from a beat-histogram, which is formed by counting successive values of lag found to have the greatest autocorrelation in each of the spectral bands. A basic dimensionality reduction technique known as Principle Component Analysis (PCA) is used to reduce the dimensionality to the three dimensions, in a rotation of the feature space that best encompasses the variance of the dataset (the dataset being the music track). These three features are normalised and used as the three components in a colour, either red/green/blue or hue/saturation/value. This process is done for each second of the audio signal. The resultant series colours are compiled into a vertically striped image where time runs from left to right. As such the visualisation is formally *S-3*.

A small and informal user study done on the timbregrams supported the suggestion that the descriptive icons could be used to give the user an idea of what audio the corresponding track contains. No further work was openly published on this technology, however the concept of automatically generating images representative of the content has been continued:

Kolhoff et al. (2006) present a system capable of generating visually attractive graphic "blooms" varying in form according to the content of musical audio tracks. Figure 3.7 has an example of such blooms. This is integrated with the operating system in order to augment the native browsing interface. The blooms vary in form, size and colour (an *SS-3* visualisation) according to the output of back-propagated multi-layer perceptron neural network. The supervised network is trained by humans selecting tracks and image combinations; the data from all tracks is then fed back into the trained network for calculating the icon parameters for each track.

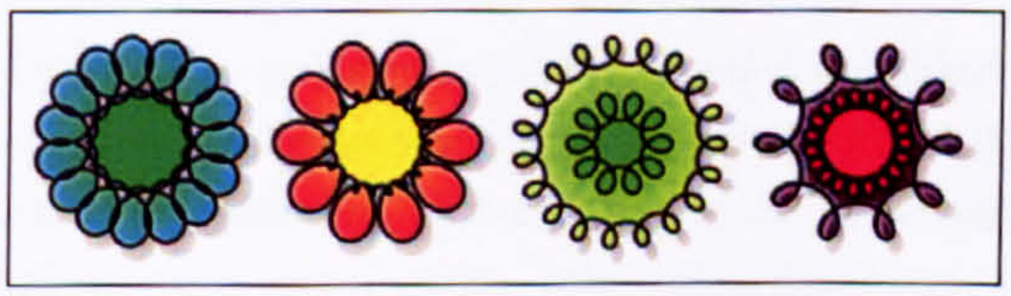


Figure 3.7: Examples of Kolhoff's blooms generated according to the content of audio and training parameters given by the user.

Unlike the Timbregram, the colours are determined with the training of the user; this

of course means more effort on the part of the user, but should result in icons that are easier to interpret. The use of the bloom shape is not only more visually enticing than the stripey rectangular bar but it also fits within the square space that an icon should inhabit far more naturally. A small user study found that the system had very respectable real-world results, with visual similarity of icons and audible similarity of music content agreeing around 70% of the time.

Other work in this area would include that of Hiraga and Matsuda (2004a), who presented a system capable of extracting information from audio to produce a rectangle of colours (thereby an *SS-3* visualisation) designed to capture and visualise the mood of the track. Their work stems from that in performance visualisation which I discuss next.

3.2.6 Animated Visualisations

Performance Visualisation

Performance visualisation attempts to visualise the particulars of expressive music performance over any other aspects of the musical content of the audio. Hiraga (2006) writes performance visualisation is used to:

- "Share understanding of a performance between co-players."
- "Compare expressions of performances."
- "Understand musical intent of performer."
- "Data-mining through the mood of performance."

Typically the analysis of performance revolves around qualities that tend to be varied from a strict interpretation of the score in classical music. Other genres of music, such as jazz, may be somewhat trickier to visualise effectively, due to the wide array of deviance from the score that performances may make.

Due to the concentration on classical music, these aspects visualised are typically limited to the relative dynamics and relative tempo of a piece. Characteristics such as timbre, absolute tempo & loudness, key and melody are essentially ignored. Assumptions such as the relations of onset intervals (such as in Dixon et al., 2003), or that a precise MIDI-encoded version of the performance is available, (such as in Hiraga's work) may be made in order to ease analysis.

Several methods for visualising performance exist; there are two main currents—the work done by Hiraga et al. (described by Hiraga et al., 2002a, Hiraga et al., 2002b and Hiraga and Matsuda, 2004b) and that of Dixon et al. A particular visualisation recently proposed by Hiraga and Matsuda (2004b) is one of form *SS-3*, comprising a series

of rectangles organised as a horizontal series. The axes of the visualisation are time-loudness. The rectangles' horizontal spacing denotes relative articulation in the music and their relative size denotes relative tempo. Absolute tempo is not shown. A user study gave mixed results for this visualisation, suggesting that except where the differences are extremely pronounced (in terms of deviation from regularity) it may not adequately reflect the viewer impression. The implementation also relied upon an accurate MIDI rendition of the performance, implementing it to use an audio signal may be non-trivial.

The visualisation proposed by Dixon et al. (2002b), called the performance worm, is based upon a tempo-loudness graph with the curve on it extending through the time of the piece in question. The visualisation was designed to be viewed in an animated fashion in real time (making it *SST-0*), though the animation can be folded down to a single image, plotting the entire track's trajectory on a single graph, making it an *SS-0* visualisation. The implementation uses a smoothed series of inter-onset-intervals (IOIs) to determine the trajectory of the curve (which in the real-time visualisation appears as a 'blob with a tail'). To do this they make an assumption as to the underlying regularity of the music that is being played which, for the body of classical pieces they have tested, is reported to work well. The authors imagine it to extend well into non-classical and non-tonal music, though no empirical tests have been reported to strengthen this claim.

A general and recent overview of performance analysis techniques and literature review is made by Widmer and Goebl (2004), and for further information the reader may find this useful.

Non-analytic Visualisations

There are several interesting though not directly relevant methods of visualising music information. Many assume MIDI data is available and generate event-based 3D words to view the music in a discrete manner; an early proposal by Smith and Williams (1997) would be an example of this, which generates an *SSST-3* type visualisation. The CAVE Automatic Virtual Environment was a somewhat grander scheme, to generate a similar type of visualisation again from score data in an immersive environment (the *Cave*), presented by Kaper (1998).

Malinowski (2001) describes his *Music Animation Machine* (MAM). This is an *SST-1* visualisation synchronised to the music performance, which uses the MIDI data of the performance to visualise the music on a time-pitch graph which itself rolls through time. It shares some similarities to a usual piano roll which finds itself in time-pitch space. Different instruments are plotted on the same stave, with individual colours to separate them. One interesting point is that unlike other time-pitch-based visualisations, it rescales the pitch axis for each instrument; this makes only relative melodic movements important and, through sacrificing the ability to compare pitch between instruments, simplifies the

view.

In a similar vein to the MAM and the tonal visualisations is the work presented by Chew and François (2003) called *MuSA.RT*. It displays an animated 3D spiral projection of chroma (*SST-3*), with each quarter revolution being equal to a Major Third (i.e. a frequency ratio of about 1.1892). This again uses MIDI data to animate the visual. The 3D spiral view has the advantage of clearly displaying triads as triangles, though it appears be be restricted to monophonic display and does not visualise aspects of the music such as timbre or rhythm.

Projects such as ImproViz by Snydal and Hearst (2005) show the utility of visual aids in music software, though it is a niche tool for jazz players and not a general purpose tool; it works only on monophonic MIDI data.

A quick search on the World Wide Web with Google (2007) reveals several popular pieces of software which attempt content-based musical audio visuals, some of which are quite sophisticated. *sndpeek* and *Armadillo* both give large analytical views of the audio using 3D to give visualisations; such as spectrograms of varying dimensionality, waveforms, and text noting certain statistical values.

Lillie's Music Visualisation

Lillie (2007) has proposed some as yet unpublished but nonetheless interesting work at MIT on visualising music. The form of the visualisation is *SST-2*, though it can also be amalgamated to *SS-2*. The visualisation is formed from a time-tone graph, with the y-axis being discretised into the 12 semitones of the equal-tempered scale. Two images are provided as an example in figure 3.8.

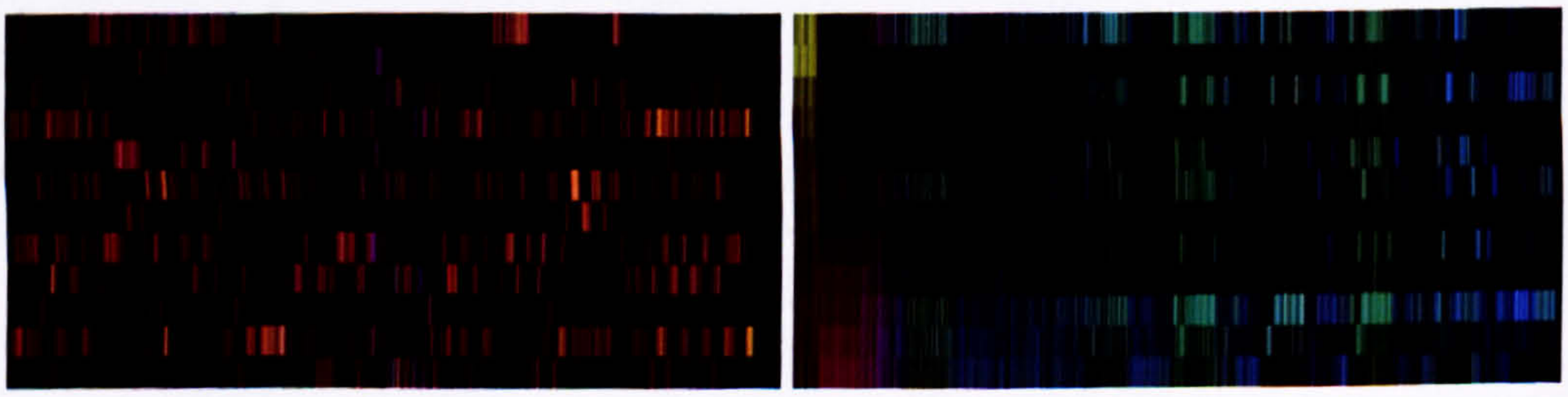


Figure 3.8: Beethoven's Moonlight Sonata (left) and Daft Punk's Superheroes (right) with Anita Lillie's music visualisation. *Reproduced with permission.*

The colour spanning any given column of the graph is determined by some timbrel characteristics of the signal, whereas the relative loudness of each semi-tone is given by the brightness. Chords are visible as repetitive combinations of loudness in particular rows.

3.2.7 Scientific and Professional Applications

There are several scientific, professional, and otherwise niche audio playback and editing tools that include musical audio visualisation components. Some follow a multi-pane approach where the time is once again presented horizontally and whereby various features are drawn accordingly, each on their own subsection of the y-axis. The *CLAM Music Annotator* presented by Amatriain et al. (2005), *WaveSurfer* presented by Sjölander and Beskow (2000) and the newer *Sonic Visualiser* presented by Cannam et al. (2006) are each examples of this. Each incorporate various visualisations of the same signal; Sonic Visualiser is a particularly interesting project with its extensible *plugin* architecture, allowing future music signal analysis techniques to be extended into its interface. The relevant techniques used to visualise the audio have, however, been discussed elsewhere in this section.

Variations2

Other visualisation tools, such as *Variations2*, as reported by Isaacson (2003), addresses the problem of visualising music content through synchronisation of various analytical views with performance playback. These views might take the form of music notation, having been analysed in a manner proposed by Schenker; a number of arbitrary text labels at specific points through the music, or an attractive hierarchical construction of segments, populated perhaps, after the rules proposed by Lerdahl and Jackendoff (1983). Notably all such views would have to be made manually; none of the visualisation is content-based.

Augmented Sound Editor

As part of the *Marsyas* project, Tzanetakis and Cook (2000b) discuss the Marsyas *Augmented Sound Editor*; it functions as a standard sound editor, where an audio file is depicted in the usual waveform graphic with time mapped from left to right. However, the wave is actively coloured dependant on the audio at that particular point, making it an *SS-1* visualisation rather than the normal *SS-0*. Aside from this niche application, the technology never progressed any further, implying either the lack of utility or an opportunity missed.

3.2.8 Representation Issues

As I noted in the introduction, the representation of music does not end at the common music notation. In many ways for many tasks, common music notation is simply not the right tool. It is cumbersome, limiting or impossible to note aspects such as hierarchy (Lerdahl and Jackendoff in their work on music theory opted for a tree representation to better analyse motific portions score), timbre, rhythm, pitch nuance and gradation.

Because of this we see various attempts at creating other methods of representing music e.g. the map-based notation of Weyde (2005) and Weyde and Wissmann (2004) which visualises music structure where linear and even hierarchical methods will not suffice.

Couprie (2004) gives an interesting discussion on possible intuitive graphical representations of music. He contends that electroacoustic music, not being score-based, has more problems with analysis than, for example, classical music. He argues that spectrograms are too complex to ease analysis and that some other form is necessary. The display comprises multiple discrete but iconic elements, with each element having a continuous form. This display has attributes of a symbolic notation (such as common music notation with multiple discrete elements), but also of a continuous visualisation (such as a spectrogram). He argues that navigational aids are helpful in numerous situations, with the final sound bite "the creation of the visual provokes an enrichment of the listening".

Isaacson (2005) reviews techniques to visualise music from a music theoretic point of view, pointing out that good visualisation techniques should be backed up with accepted music theory. Middleton (1990) argues quite persuasively that music theory is a "less than useful resource" for popular music. In agreement, many scholars in the area of content-based visualisation and musical audio analysis, when dealing with popular music, typically take a more relaxed and empirical approach, opting to collect evidence not through an explicit appeal to music theory but rather through empirical experiments. The present work attempts to take an approach respecting both views; empirical evidence will be sought, but I will discuss the visualisations features with regard to the relevant aspects of texts such as Lerdahl and Jackendoff (1983).

Isaacson writes that there are "many facets of music to be visualised". Unfortunately, he does not go into the specifics of how each of these facets might be useful; in particular there is no mention of the music navigation metaphor and how the visualisations might fit into this, despite navigation being an immediate and obvious use for visualisation (enough for Spence, 2001 to devote an entire chapter of his book to it).

Dannenberg (1992) tackles the *de facto* standard in digital music composition representation, Musical Instrument Digital Interface or MIDI. He notes that while being the standard and doing its initial job adequately (it was designed as an interface between digital music equipment in order to transfer such information as note onsets), it hardly fulfils a need for the representation of higher-level features such as structural depictions of music.

3.2.9 Colour

The primary defintion of colour is:

> "The quality or attribute in virtue of which objects present different appearances to the eye, when considered with regard only to the kind of light

reflected from their surfaces."—Simpson et al. (1989)

We can therefore see that *colour*, in a similar manner to *visualisation*, is a term concerning our perception. Colour arises, like sound, from our interpretation of a wave. Like sound, a spectrogram can be drawn of this wave; giving the intensities of the various frequencies which make it up. Similarly, these frequencies are typically limited to the range to which humans are sensitive; this is called the *visible spectrum*. Whereas sound runs from roughly 20 Hz to 20 KHz (for the average human child), colour runs from approximately 1.4 MHz (700 nm, 'red') to 2.5 MHz (400 nm, 'violet').

The visible spectrum is not an accurate representation of the degrees of freedom our perception has however. In fact, with regard to colour, humans have only three degrees of freedom; colour-sensitive (*cone*) cells in the retina are sensitive to (roughly) the blue ($\approx$ 445 nm), green ($\approx$ 535 nm) and orange ($\approx$ 575 nm) areas of the spectrum. The perception of the 'colour' of the visible spectrum comes from the ratios of these three quantities.[7] Since the visible spectrum may be made up of any combination of strengths of frequencies, humans perceive many different combinations of spectral light as being equivalent; for example a single frequency of orange light could easily appear indistinguishable from two frequencies of light which would, on their own, be perceived as yellow and red. This contrasts to sound, where two pure tones would rarely be indistinguishable from a single tone of their average frequencies.

The collection of colours which humans can perceive is called the *human gamut*. Figure 4.6 gives an illustration of the human gamut; it is bounded by the visible spectrum. On the gamut the intensity of the colour (i.e. how dark or light it is) is ignored, providing a 2-dimensional (i.e. planar) representation of colour. Much of the gamut can be emcompassed by carefully selecting a number of colour points (*primaries*) on the visible spectrum and combining them to form a *composite* colour. A systematic combination to synthesise a colour is called a *colour model*.

Colour models in themselves do not properly define colours, since they describe only the basic methodology of creating a colour rather than the specifics. Examples of colour models include red/green/blue (*RGB*, a colour is created by combining differing amount of red, green and blue light), hue/saturation/value (*HSV* a colour is defined by where it falls in the visible spectrum, its brightness, and how 'faded' it appears) and cyan/magenta/yellow/black (*CMYK*, commonly used in printing; a colour is defined by how much of four colours of paint should be combined). Modern computer systems often use the RGB model, making it convenient for transmission to visual display units which synthesise colour by mixing these three primaries.

[7] There is actually a fourth cell in the retina for perceiving light (a *rod* cell), though it is sensitive only to the intensity of light rather than individual frequencies, and is used by the eye for accurate judgement of brightness as well as periphery vision, where accurate colour perception is less important.

For accurate specification of a colour, a *colour space* must be utilised. A colour space encompasses a coluor model and extends it by defining the parameters properly (e.g. the wavelengths of the primaries). Examples of colour spaces include CIE 1931 XYZ (a perceptually motivated space, discussed later) and *sRGB*, an industry-dominant, properly defined version of the RGB colour space. Since our system will be evaluated on basic consumer hardware only, this colour space was chosen to best represent our RGB colours. Exactly what colours are presented on the display device depends on a wide array of factors (e.g. the individual device's characteristics , contrast/brightness settings on the device itself, hardware drivers of the graphics display software (e.g. X-Windows), gamma correction in the system software). As such it is unlikely that the sRGB colour will be reproduced accurately, but it does at least give a theoretical stationary target.

3.3 Proposed Visualisation Methods

3.3.1 Visual Construction Method

In at least one piece of work (that by Tzanetakis and Cook, 2000a), a visualisation of a music recording has been created by colouring the points of a plane according to the point in the recording analogous to the horizontal position of the given point. Indeed, this is done almost by accident when depicting a segmentation of musical audio, as was shown in the last section.

When combined with linear progress-bar style GUI navigation widget, however, the colours afforded by this visualisation method become an obvious metaphor for the music. Colours on the y axis are constant and therefore it seems clear that the graphic is linear, and reasonably obvious that it is a 1-1 mapping from time to the x axis. This might be contrasted to the spectrogram where both axes change; novice and casual users may not see this directly.

This technique may be formalised by naming a function $\mathcal{P}$, such that it converts from the domain of audio blocks to that of colours. For convenience, we will define the audio block $\mathbf{c}$ in terms of our common signal preprocessing. In all instances, the signal's spectra were first calculated by using a series of STFTs over the audio recording. The window size used was 1024 samples, with a 50% overlap. With the input signal being the CD standard 44100 Hz, this puts the lowest frequency to be detected at around 43 Hz, with windows around 11 ms apart. The stereo signals were first downmixed into mono, to prevent any problematic stereo separation effects.

$$\mathbf{c} \equiv \mathcal{P}(\mathbf{b}) \tag{3.7}$$

where $\mathbf{b} \in$ the set of all frequency spectra as output by the above STFT and $\mathbf{c} \in$ the set of all colours.

Some techniques naturally rely upon multiple audio blocks to create a colour (perhaps in order to build a context); projection function is therefore extended to $\mathcal{P}'$:

$$v \equiv \mathcal{P}'(c, \vec{\mathbf{b}}) \tag{3.8}$$

Where $\vec{\mathbf{b}}$ is a series of spectra.

For convenience and relying upon the fact colours can be determined through three values, one for each of three primaries, the chroma-projection is defined instead in terms of a two-parameter function $\mathcal{P}''$:

$$v \equiv \mathcal{P}''(c, \vec{\mathbf{b}}) \tag{3.9}$$

such that

$$C_{sRGB}(\mathcal{P}''(0, \vec{\mathbf{b}}), \mathcal{P}''(1, \vec{\mathbf{b}}), \mathcal{P}''(2, \vec{\mathbf{b}})) \equiv \mathcal{P}'(\vec{\mathbf{b}}) \tag{3.10}$$

where $\mathbf{b} \in$ the set of all audio blocks and $\mathbf{c} \in 0, 1, 2$. For c, 0 represents the red channel, 1 green and 2 blue of the sRGB colour space. The colour space function C_{sRGB} simply converts from the three primary colour intensities (red, green and blue) to the corresponding element in the set of all colours. This conversion, though likely rendered inaccurate by the hardware, is a precisely defined colour space, and is a convenient form, used implicitly in most computer video systems. Thus $\mathcal{P}$ converts from audio blocks and colour channels to an intensity of the given channel.

With the projection function defined, the visualisation technique may be defined as the planar shader S, which maps the points of any given plane (in x, y coordinates where both are bounded between 0 and 1):

$$S(x, y) \equiv \mathcal{P}'(\vec{\mathbf{B}}_{t \ldots t+w}), \quad t \equiv x(l - w) \tag{3.11}$$

where $\vec{\mathbf{B}}$ is the audio track's series of spectra, l is the length of the audio track in spectra and $\vec{v}_{i..j}$ is the vector containing the ith to jth elements of $\vec{v}$. See figure 3.9 for an illustration.

I call this general method an *audio-colour projection* (*ACP*), with $\mathcal{P}'$ being the *ACP function*. As such, different individual techniques need only to define their particular projection function to be completely defined as a visualisation method.

Formally, the ACP method is of type *S-3*, though if a projection function neglects the c parameter it becomes of type *S-1*. Although it cannot ever formally be *S-2* or *S-0*, it may still be useful to consider projection functions that restrict themselves to only two dimensions of colour.

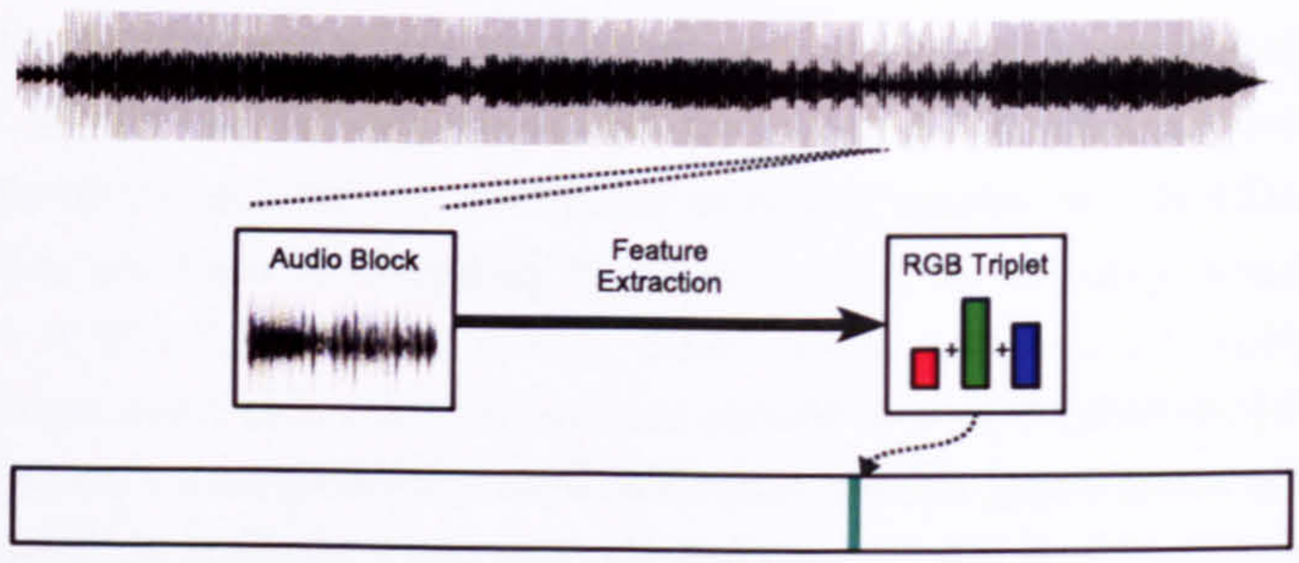


Figure 3.9: An illustration of the core visualisation method.

3.3.2 Signal Postprocessing

In order to utilise the full gamut of colour, a quantile stretch of the data is done into the full colour space. A 90% quantile stretch was used, given by v':

$$v' \equiv \frac{(v - q_{0.95})}{q_{0.95} - q_{0.05}}$$

where q_n gives the nth quantile of the dataset of values v. This particular normalisation technique was used above others to reduce any extreme outliers affecting the distribution of brightness adversely, which can be a problem with other methods (e.g. max-min, mean/normalisation) when the distribution is heavily skewed or not normal.

3.3.3 Psychoacoustics

In order to help generate a perceptually accurate image, the audio is preprocessed with certain psychoacoustic transforms, designed to account for the process that sounds must go through to reach the brain. These transforms are derived from empirical data collected on humans. In particular three transforms are used; equal-loudness contours for phon scaling, critical-band summation, and specific loudness sensation for sone scaling.

Bark

The Bark scale is a non-linear scale of 'critical' frequency bands, between which we have a similar perception of frequency difference. They are based upon the inner ear, which can be considered as a complex set of band-pass filters; each of the centre frequencies of the Bark bands are related to said filters. Figure 3.10 illustrates the edges of each critical band. As can be seen, the bands increase monotonically between 100 and 500 Hz, showing the human ear's sensitivity to changes in this part of the spectrum. The width of the bands then rises sharply after this, denoting the ear's indifference to small changes at higher frequencies.

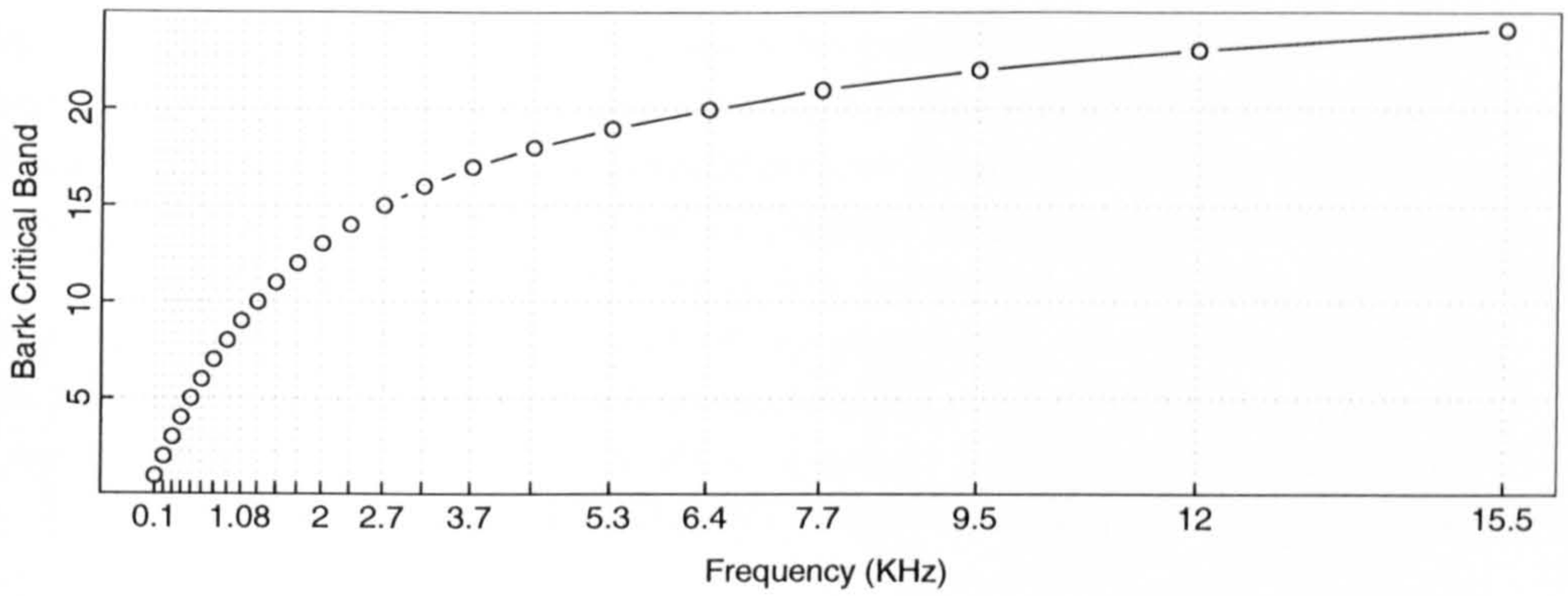


Figure 3.10: The Bark critical bands and where they fall on the audio spectrum.

Summing a basic power spectrum into a Bark critical banded spectrum has several benefits; aside from being a perceptually motivated frequency scaling, it drastically reduces the amount of data it is necessary to process; typically up to two orders of magnitude. Formally, I define the function $Bark(\mathbf{s})$ as the bark spectrum $\mathbf{s}$ of 20 bands $\mathbf{s}_i$, $0 < i \leq 21$ where:

$$\mathbf{s}_i \equiv \sum_{b_{i-1} < f_j \leq b_i} \mathbf{x}_j \tag{3.12}$$

where b_i is the frequency of the ith critical band, f_i is the frequency of the ith band of the spectrum $\mathbf{x}$ and $b_0 = 0$. The band of any given frequency may be estimated with the function Z_{bark}, proposed by

$$Z_{bark}(f_{kHz}) = 13tan^{-1}(0.76f) + 3.5tan^{-1}(f/7.5)^2 \tag{3.13}$$

Decibel

The level of sound may be objectively measured as a ratio between the pressure of the signal in question compared to some agreed base line pressure; the unit of pressure is the Pascal (Pa). Since ratios of levels tend to vary both very little and very greatly, a logarithmic scaling is used to represent the ratio, decibels (dB), which is defined as being the ten times the 10th logarithm of the ratio of value a to base line b thus:

$$a_{dB-b} \equiv 10log_{10}(\frac{a}{b}) \tag{3.14}$$

Typically in audio processing, it is useful to agree upon a standard baseline to minimise confusion between parties. The absolute level 20 μPa, is used as such, which is considered to be the lowest possible amount of sound pressure that the human ear can sense. If a

logarithmic ratio is given with this level as the baseline, then the meta-unit dB_{SPL} is used to mean *decibels (Sound Pressure Level).* Unfortunately, this presents a problem for musical audio signal processing, in that recordings typically do not denote where on the amplitude scale $20\mu Pa$ falls. Without this information, there is no way to transform accurately the raw audio spectral data into dB_{SPL} and psychoacoustic scales based thereon.

For the present work, I arbitrarily set the levels such that the maximum level within a piece of music is 90 dB_{SPL}; any levels falling below zero were raised to zero. This should mimic a listener playing back the music at the highest healthy level. Thus I define dB, which operates on a power spectrum of values in positive unity range, as:

$$dB(\mathbf{x}) \equiv \mathbf{d}, \text{where} \quad \forall i, 0 < i \leq n : \mathbf{d}_i = 10log_{10}(\mathbf{x}_i) + 90 \tag{3.15}$$

where $\mathbf{x}_i$ is the ith band of $\mathbf{x}$ and n is the total number of bands in $\mathbf{x}$.

Phon

Humans perceive tones of differing frequencies at differing levels of loudness and are most sensitive to frequencies around 3 KHz. The phon scale is a scale of frequency-independent loudness. Any pure tone of loudness p phon is defined as being perceptually as loud as a tone of 1 KHz at p dB_{SPL}. To transform (approximately), one linearly interpolates between the equal-loudness curves given by Pampalk (2001). Figure 3.11 illustrates these curves.

Sone

The sone is a scale of specific loudness sensation. It is designed to give a linear rise with respect to perceived loudness, unlike decibels which give a well spaced a scale for many different uses. Thus a doubling in the scale of sone should represent a doubling of perceived loudness. If it operates upon the phon scale, it is a frequency-independent measure. As such this represents the final stage in the present work's psychoacoustic processing chain. Formally, I define the equation $Sone(p)$, taken from Bladon and Lindblom (1981):

$$Sone(p) \equiv \begin{cases} 2^{(p-40)/10}, & p \geq 40 \\ (\frac{p}{40})^{2.642}, & \text{otherwise} \end{cases} \tag{3.16}$$

Figure 3.12 illustrates the scale; notice that until around 30 phon it grows slowly; after this it increases in sone at a much faster rate.

We may therefore define the Loudness function L, which operates on a spectrum $\mathbf{x}$ to give the first 20 Bark critical bands on a specific loudness sensation scale:

$$L(\mathbf{x}) \equiv Sone(Phon(Bark(dB(\mathbf{x})))) \tag{3.17}$$

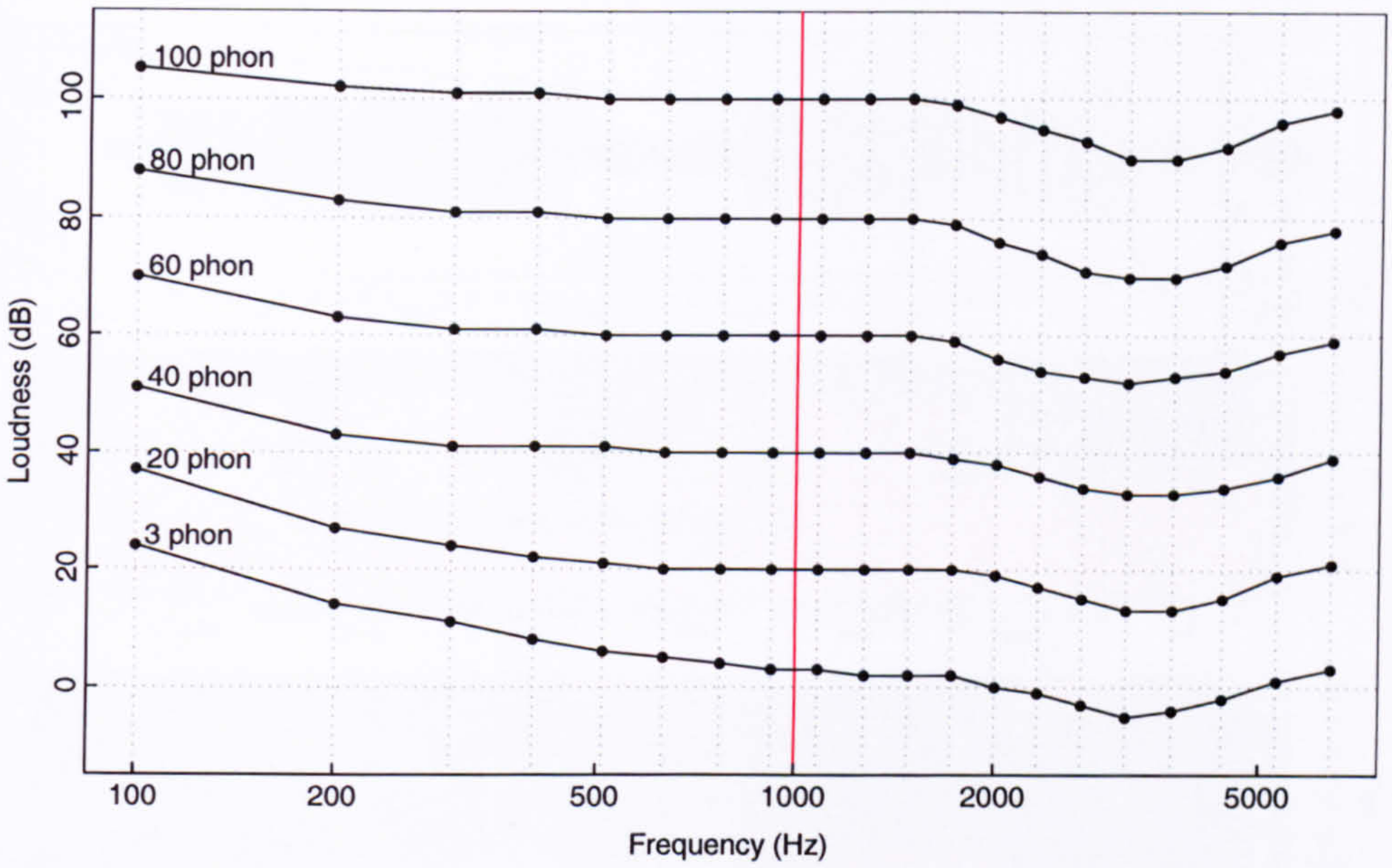


Figure 3.11: The equal-loudness curves. Each curve denotes a constant level of perceived loudness for all frequencies.

3.3.4 Bandwise Loudness Magnitude

Definition

Initially, the spectrum is processed with a psychoacoustic pipeline: The spectrum is converted to use a perceptual loudness scale Phon, which gives a frequency-independent scale of loudness. After this it is summed into the critical bands on the Bark scale. This significantly cuts down on the computation cost in many areas, since the 512 bands of the FFT output is reduced to only 24 critical bands, and it also gives a good frequency scale which can be easily split into three portions later. The final bands of the Bark scale are then converted to a perceptually linear scale, Sone.

A windowing technique is then used; as supposed by Abdallah et al. (2005), we found that a moving window mean over the signal was most helpful in refining the output to become musically relevant. The exact window width we used was an experimentally-determined three seconds, with it being moved one second between successive windows. The bandwise mean is taken over the window of spectra.

Each perceptual spectrum of 24 critical bands is then split into three separate channels for red, green and blue respectively. Each sub-spectrum is then used as an eight-dimensional vector to which the magnitude is calculated (as the Euclidean distance from

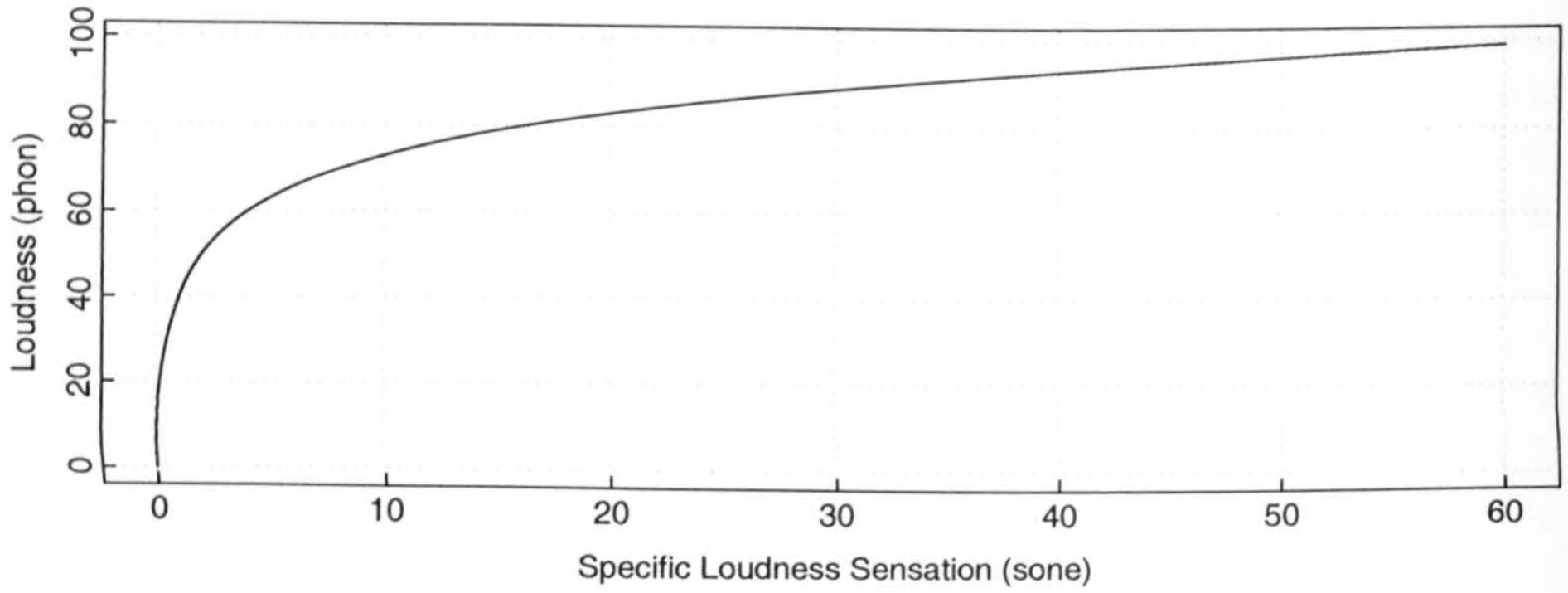


Figure 3.12: The sone scale in terms of phons.

zero). Formally, I define $\mathcal{P}_{BLSM}$:

$$\mathcal{P}_{BLSM}(c, \vec{\mathbf{b}}) \equiv \|\overline{L(\vec{\mathbf{b}})}_{8c \ldots 8c+8}\| \tag{3.18}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively) and $\mathbf{b}$ is a vector of three seconds of spectra.

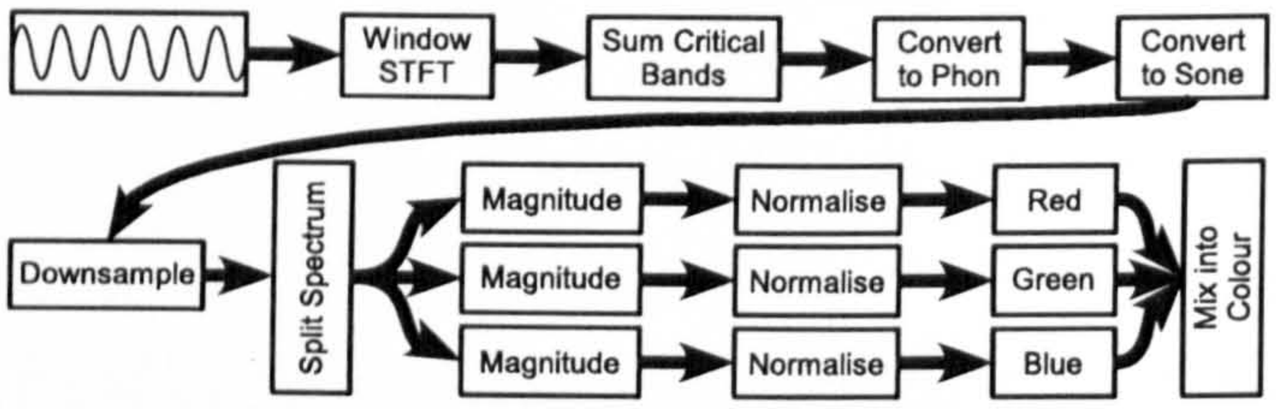


Figure 3.13: An activity flow chart of the bandwise loudness smoothed magnitude (BLSM) technique.

Figure 3.13 gives a data-flow representation of the technique, which directly corresponds to a network graph in the audio analysis framework I used.

Expectations and Discussion

Figure 3.14 gives an illustration of the various processing stages while preprocessing the track *Clubbed to Death*. Starting with the basic waveform of the audio, the basic spectral representation is shown followed by the critical-band spectrum in units of dB, Phon and Sone. The bottom graph shows the BLSM intensities in each of the three bands (red corresponding to the lower Bark bands, blue to the higher and green in the mid-range). The final visualisation for this track is shown in 3.16.

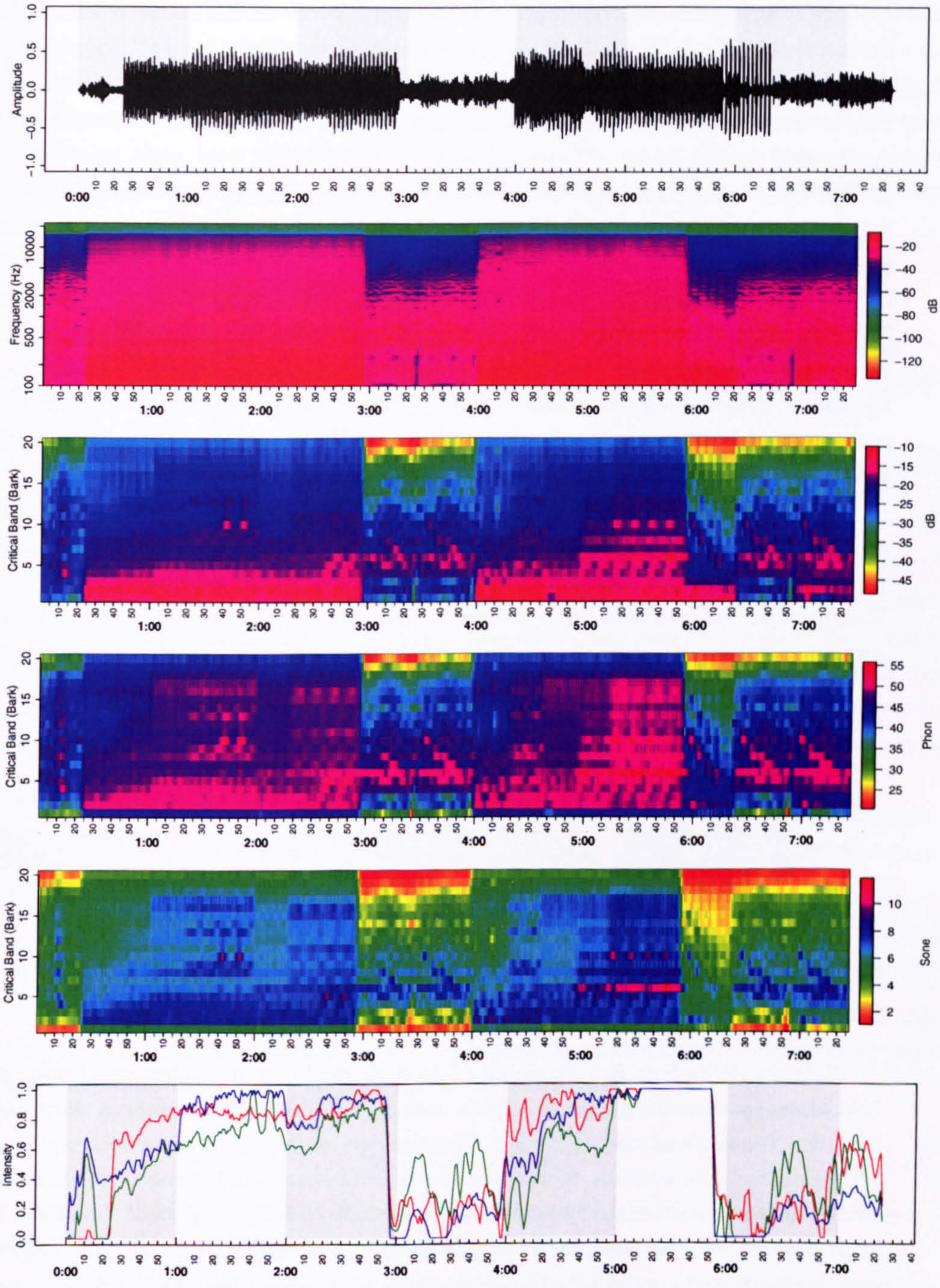


Figure 3.14: A depiction of the track *Clubbed to Death* at the various stages of the BLSM transform.

This is primarily a timbre-based refinement. Colours should be very descriptive in timbrel terms. Music where the timbre is very important benefits from this representation e.g. electronica and rock, but other types where aspects such as rhythm, harmony and melody weigh more such as classical, hip-hop and jazz music, will have their 'meaning' represented poorly.

The hue of the colour should be an indication of the relative "brightness" of the sound. A redder hue will denote more power in the low frequency portion. A greener hue denotes more mid-range content and a bluer hue would denote high-range. The lightness denotes overall relative power, as in the standard spectral magnitude measurement. Clear large-scale dynamics will be represented by clear banding of dark and light areas, whereas tracks whose power changes little will be more uniform in appearance. Finally the saturation of the colour would denote the relative balance of power in the spectrum. A spectrum that contains much of its power in a particular place should give rise to a very saturated colour, since it is likely that the power will be engulfed into one of the three subspectra.

Because of this, it should depict the spectral surface changes, such as the onset of instruments, and dynamics changes (e.g. crescendos) clearly. Other aspects of the music such as tempo changes, and more generally rhythm, we would expect to be less obvious. Similar colours should arise from certain combinations of instruments in particular keys. As such, repetitions and small variations should be visible as portions of similar colour.

Demonstration

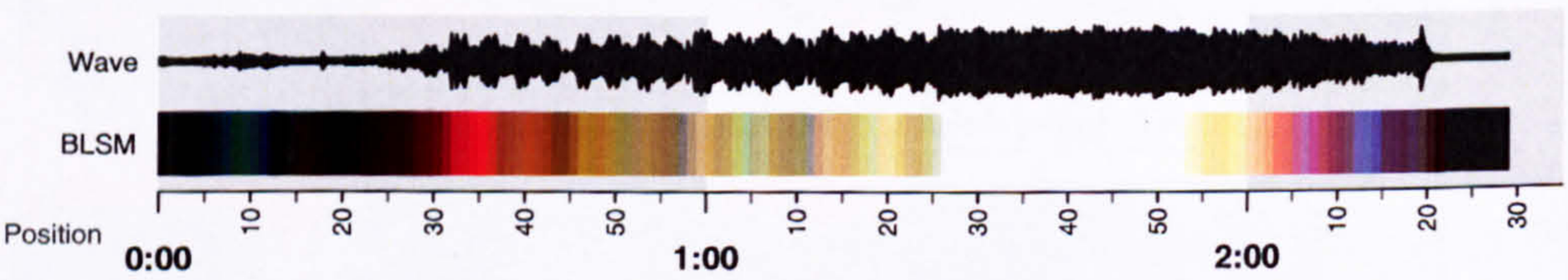


Figure 3.15: *Altitude (Red Square Reprise)* by *Hybrid* visualised as a basic wave (Wave) and with bandwise loudness smoothed magnitude (BLSM).

Hybrid's *Altitude* & *Rob Dougan*'s *Clubbed To Death* are both electronic/classical hybrid pieces incorporating aspects of both musical styles. *Clubbed* contains a short excerpt from the *Theme (Andante)* of Elgar's *Variations on an Original Theme for orchestra, Op. 36* ("Enigma"). In addition to two main sections of strings and drums with various DSP effects and samples, there are two more piano parts, largely reminiscent of Variations 1 and 12 of *Enigma*, but composed entirely by Dougan himself. *Altitude* is a simple string theme mixed with breakbeat, with both the dynamics and the rhythm building to a crescendo and holding briefly before dying away.

In *Altitude* (figure 3.15), it is clear to see the crescendo at 1:28 repeating its figure

four times until 1:53, before reducing and fading. More interestingly the visualisation depicts the general brightness of the sound on its way through the track. It starts (until around 0:24) with some non-tonal sampled content with a high-pass filter, restricting it to the treble region of the spectrum. This is represented by the blue hue. The track then introduces some quiet strings with a high-pass filter on, which get progressively louder and whose filter gets progressively more relaxed. This is visible as the red (denoting bass content) fades into murky brown, cream and eventually white.

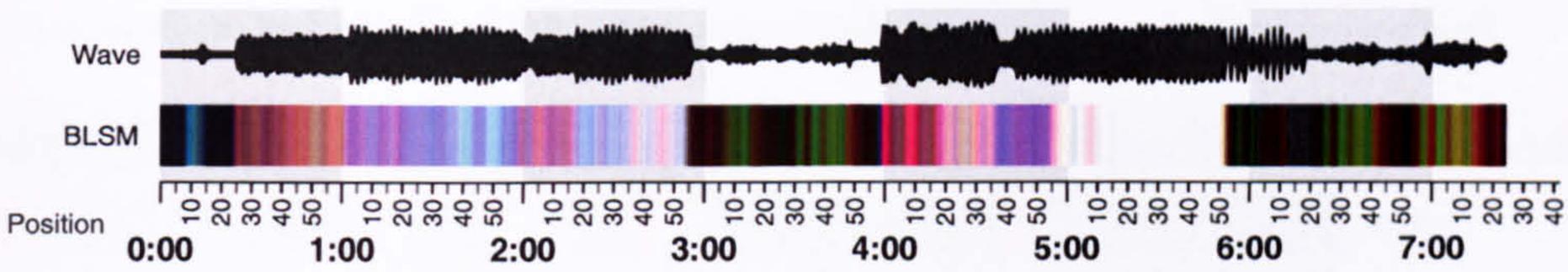


Figure 3.16: *Clubbed to Death (Kurayamino Mix)* by *Rob D* visualised as a basic wave (Wave) and with bandwise loudness smoothed magnitude(BLSM).

Figure 3.16 shows the BLSM image for *Clubbed to Death*. The piano parts are clear to see as two portions of green. The blue at the very beginning marks the Elgar's "Enigma" Theme used as the introduction. The red parts which fade into purple show the portion of percussion (red) fading to pale blue as other voices are added (taking over the mid and high-ranges and eventually reducing the apparency of the percussion). The large shift in loudness from a multitude of instrumentation to a single piano is clear at 2:55 and again at 5:52.

Banding between pink and gray is visible at 3.58 which continues to 4:17. This represents the addition of a parametric EQ filter (most likely a bandpass or notch), sweeping through the frequency spectrum and changing the timbre of the various sounds.

Without Colour

Here I will compare the main method to the basic non-bandwise version. The only difference is that the magnitude of the entire psychoacoustic spectrum is taken, rather than splitting it into three parts. We may define this formally as $\mathcal{P}_{LSM}$:

$$\mathcal{P}_{LSM}(c, \vec{\mathbf{b}}) \equiv \|\overline{L(\vec{\mathbf{b}})}\| \tag{3.19}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively), and $\vec{\mathbf{b}}$ is a vector of three seconds of spectra.

The well known jazz track *Green Onions* is shown in figure 3.17. The imprint of the funk guitar between 1:10 and 1:50 is visible on both images as a bright spot. Half-way through (at 1:30), the funk guitar steps up a key; this is clearly visible on the SBL image as

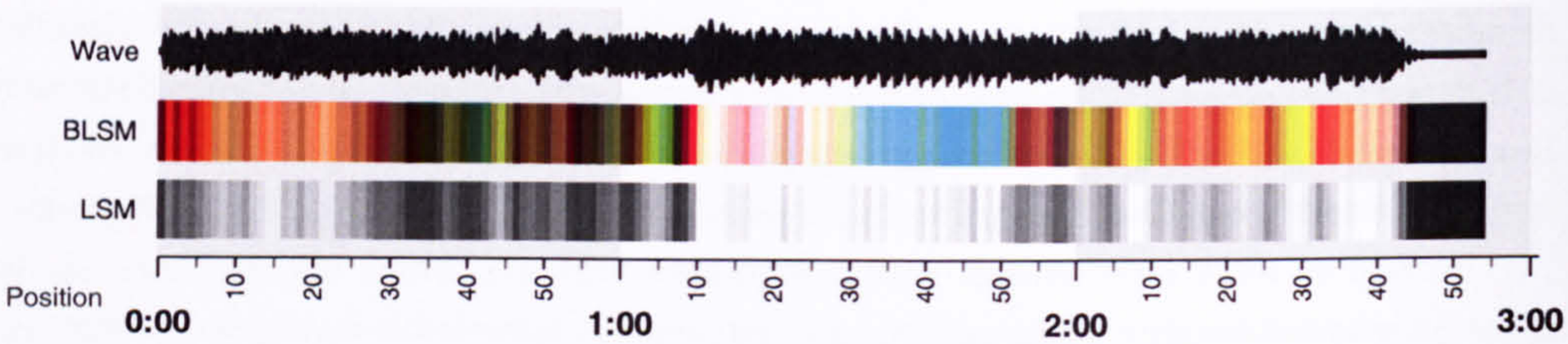


Figure 3.17: The track *Green Onions* by *Booker T. and the MG's* displayed with bandwise loudness smoothed magnitude (BLSM) compared to loudness smoothed magnitude (LSM).

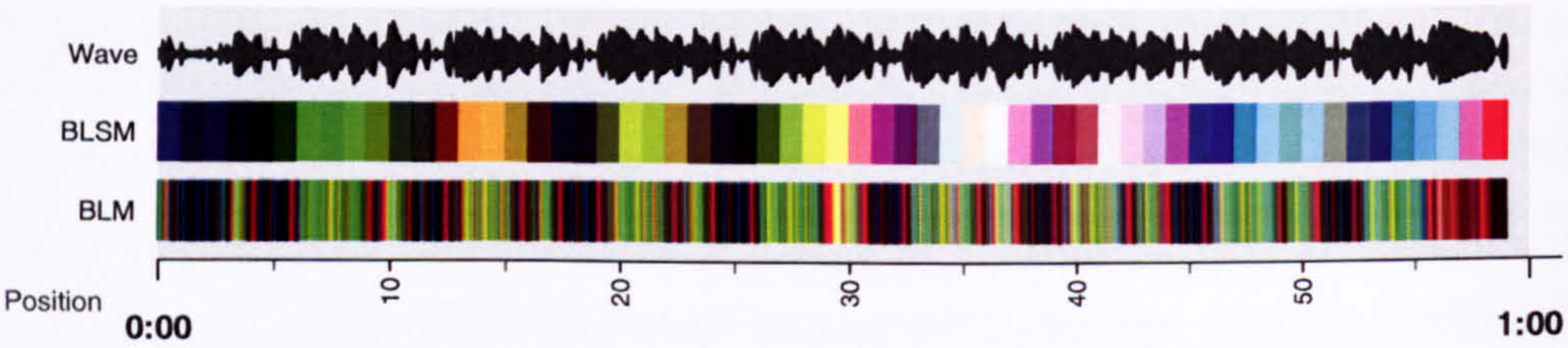


Figure 3.18: *Tak1* by *Plaid* displayed with bandwise loudness smoothed magnitude (BLSM) compared to bandwise loudness magnitude (BLM).

a change of hue from pink/yellow (denote mid-pitch sounds) to blue (denoting high-pitch sounds); however it appears as a constant white tone on the SL image.

Without Smoothing

We now compare the main method to the unsmoothed version. More formally, the bandwise loudness magnitude, $\mathcal{P}_{BLM}$ is defined as:

$$\mathcal{P}_{BLM}(c, \mathbf{b}) \equiv \|L(\mathbf{b})_{8c \ldots 8c+8}\| \tag{3.20}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively) and $\mathbf{b}$ is a spectrum.

Tak 1 is a short track by electronic music artists *Plaid*, which is visualised in figure 3.18. It contains several short repetitive figures, played on an organ-like instrument that progress into a lower overall key throughout, with a chaotic drum beat in the background. In the fourth figure, another, louder, organ-like voice is introduced with a higher key that gets higher-pitched in the final two figures.

It is a good example of where reduction of short-term features (such as individual onsets) makes medium-term progression much clearer. In the unsmoothed version, the 'shape' of the individual figures can be seen in terms of note onsets. The smoothed version does not have this, but better shows the changes between the figures as they progress

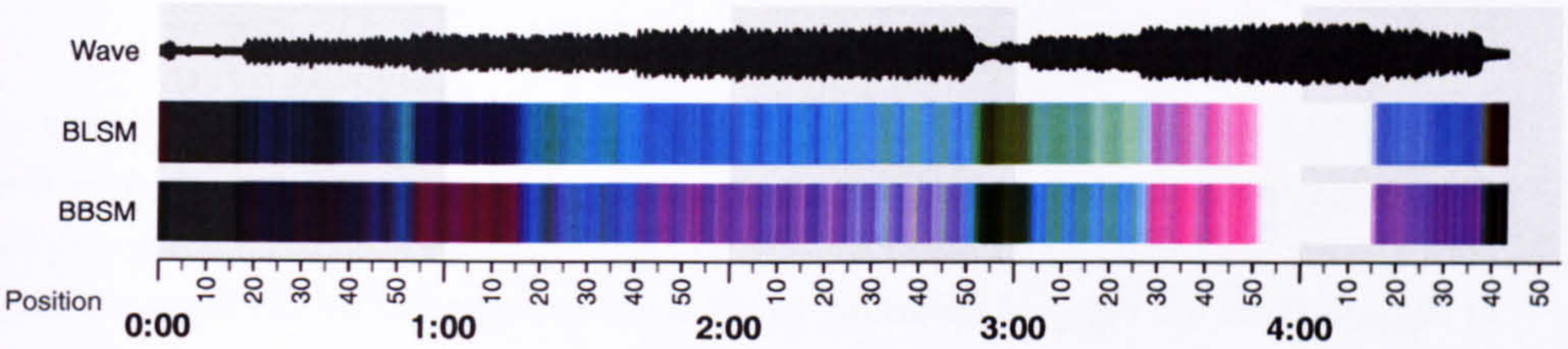


Figure 3.19: *Zala* by *Plaid* displayed with bandwise loudness smoothed magnitude (BLSM) compared to bandwise Bark smoothed magnitude (BBSM).

through the track, with colour slowly changing throughout to denote the progress and the introduction of instruments.

Without Psychoacoustics

If we remove the psychoacoustic loudness scaling from the method, using just the basic magnitude from the spectrum, we see, crucially for visualisation, the relative dynamic ranges of the three channels decrease. This gives a smaller range of hues than before. We formally define this variant as $\mathcal{P}_{BBSM}$:

$$\mathcal{P}_{BBSM}(c, \vec{\mathbf{b}}) \equiv \|\overline{Bark(\vec{\mathbf{b}})}_{8c\ldots8c+8}\| \tag{3.21}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively), and $\vec{\mathbf{b}}$ is a vector of three seconds of spectra.

From the addition of the psychoacoustic loudness scaling, a significantly greater role is taken by the blue channel takes in defining the artefacts in the visualisations. Without psychoacoustic scaling, blue tends to stay at a roughly constant level throughout, or be covariant with the red channel. The frequency-dependent loudness scale appears to distribute the channels values far more evenly. To illustrate this, we look at a piece of electronic music by the abstract electronic artist, *Plaid*.

Both images look remarkably similar, and both identify roughly the same aspects in a similar way. However in the BBSM image, the red and blue components of the image are largely covariant, resulting in an image dominated by purple. Whereas in the BLSM image there is a slightly better use of hue with portions of light green, blue, pink and dark indigo. A far more important criticism, however, is the lack of differentiation between the portion of 3:30-3:52 and much of the first half, due to the sharing the hue. This is in marked contrast to the BLSM image, which colours the two sections entirely differently (pink versus varying shaded of blue). This turns out to be musically important. The latter portion has a clear bass component, and there is no instrumentation in the higher key of the first half.

Comparison to Mel-Frequency Cepstrum Version

Another common method of psychoacoustic audio preprocessing is the extraction of the cepstrum co-efficients from the mel-scaled spectrum. Mel-scaling, as proposed by Stevens and Volkmann (1940), is similar in concept to the Bark scale, whereby the log-amplitudes of the spectrum are mapped according to empirical experiments on the human perception of tone change. The cepstrum co-efficients (of which only approximately the first twenty are taken), are simply the first amplitudes of the discrete cosine transform, when mel-scaled spectrum is treated as a discrete signal.

This was visualised as before, except with the substitution of the critical banding and psychoacoustic loudness scaling for the first 24 mel-frequency cepstrum coefficients (MFCC). As with the 24 critical bands, these were split into three subsets, the magnitude was taken as the Euclidean distance from zero, and the three colour channels were valued accordingly. Formally, the colour projection function is $\mathcal{P}_{BMSM}$ where:

$$\mathcal{P}_{BMSM}(c, \vec{\mathbf{b}}) \equiv \|\overline{MFCC_{0\ldots23}(\vec{\mathbf{b}})}_{8c\ldots8c+8}\| \tag{3.22}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively), and $\vec{\mathbf{b}}$ is a vector of three seconds of spectra.

Generally, the use of three subsets of the MFCCs seems to give a reasonably useful visualisation, with images noticeably matching those from the spectral version of the algorithm. However, there are two key points that seem to burden the MFCC variant, making it on the whole less useful for visualisation of music: Firstly, while the brightness of the signal represents the loudness, it is not quite as detached from the hue as it is in the spectral visualisation. Secondly, the distribution of hues and their brightnesses implied from the MFCC data results in a less informative visualisation, due to fewer artefacts.

To illustrate the first point, we use the rock track *Moving* by British indie rock band *Supergrass*. The track features a fairly typical *AAB AAB A* verse/chorus structure. However the sets of verses ('A's) get increasingly louder, with increasing instrumentation and more emphatic vocals. Importantly, this loudening is barely noticeable over the general theme.

Figure 3.20 shows the smoothed bandwise loudness and smoothed bandwise cepstral magnitude, both with amplitude outlines. In both, the choruses are instantly recognisable as the two bright white blocks in the middle. In both (though less obvious in the MFCC version), each verse is noticeably a single theme repeated. In the smoothed bandwise loudness visualisation it is, however, quite clear to see that the second pair of verses are repetitions of the first pair, but only louder (brighter, but same orange-green-black-red pattern), the third verse (the outro?) being of the same quality, but perhaps louder still. In the MFCC variant, with a far more complex set of colours to denote the verse, and with the colours apparently far more noisy from set to set, it is very difficult to extract

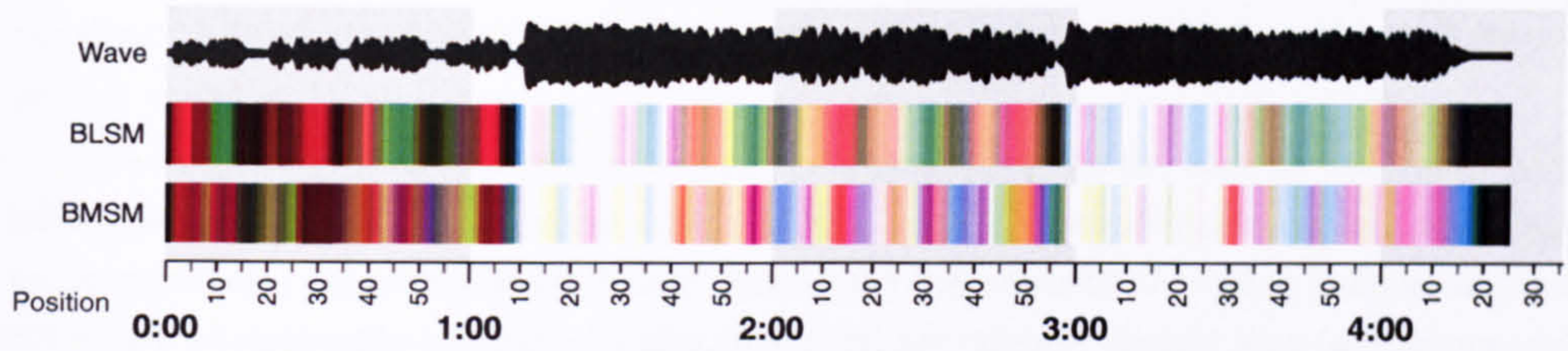


Figure 3.20: *Moving* by *Supergrass* displayed with bandwise loudness smoothed magnitude (BLSM) compared to bandwise mel-frequency cepstral smoothed magnitude (BMSM).

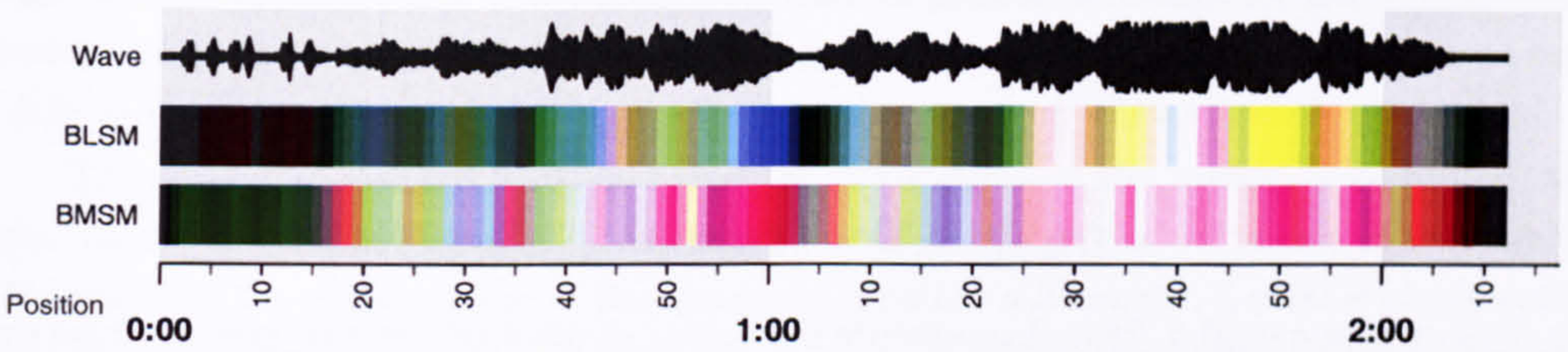


Figure 3.21: *Gabriel's Oboe* by *Ennio Morricone* displayed with bandwise loudness smoothed magnitude (BLSM) compared to bandwise mel-frequency cepstral smoothed magnitude (BMSM).

the same information.

We can see the visualisations of *Gabriel's Oboe* by *Ennio Morricone* in figure 3.21. The structure of the song is essentially *AB-AB-A'B'*, with the *A'* having an extra somewhat bassy instrument present compared to the *A*s, and the *B'* being slightly augmented from the *B*s to form an outro. The MFCC visualisation presents this well, though the different *A'* from *A* goes entirely ignored. Very little textural information within the *B* portions of the song is apparent, with them being mostly white.

Conversely, the loudness version clearly shows the *B* portions being comprised of two distinct smaller parts, corresponding to two distinct themes, the second including a mandolin-like instrument. The stripes between red and yellow in the first part of the *B*s correspond to the playing and silence of the oboe. However, where the loudness variant's visualisation becomes somewhat questionable is in the third *A*, running from 3:27-3:58. Due to the extra spectral content from the extra instrument, it becomes brown rather than green followed by red.

3.3.5 Bandwise Loudness Rhythm Magnitude

Definition

The rhythm magnitude is a novel technique to deliver the 'rhythmicity' of audio at a particular point. It is calculated by using the rhythm spectrum (also known as beat spectrum) as a vector, and taking its magnitude. We use the algorithm by Foote (1999a) for calculating the rhythm spectra, which involves populating a self-similarity matrix and summing across the super-diagonals. Section 3.2.2 gives a full definition of the technique.

As before, this is an extension to the standard rhythm magnitude technique done to provide colour. The output of the critical banding is split into three subspectra, a rhythm magnitude for each one is found. Each are normalised individually, and used as their corresponding red/green/blue component in the final colour. The technique may be more formally defined as the projection function $\mathcal{P}_{BLRM}$:

$$\mathcal{P}_{BLRM}(c, \vec{\mathbf{b}}) \equiv \|Rhy(L(\vec{\mathbf{b}}))_{tc\ldots tc+t}\|, \quad t \equiv \frac{s}{6} \tag{3.23}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively) and $\mathbf{b}$ is a vector of 128 spectra (approximately 1.5 seconds).

$$Rhy(\vec{\mathbf{x}})_l \equiv \sum_{k=0}^{s-l} \mathrm{M}_{\vec{\mathbf{x}}}(k, k+l), \quad 0 \leq l \leq \frac{s}{2} \tag{3.24}$$

where s is the size of the self-similarity matrix M and $M_{\vec{\mathbf{x}}}$ is determined from the series of spectra $\vec{\mathbf{x}}$. Figure 3.22 shows the process as a dataflow pipeline.

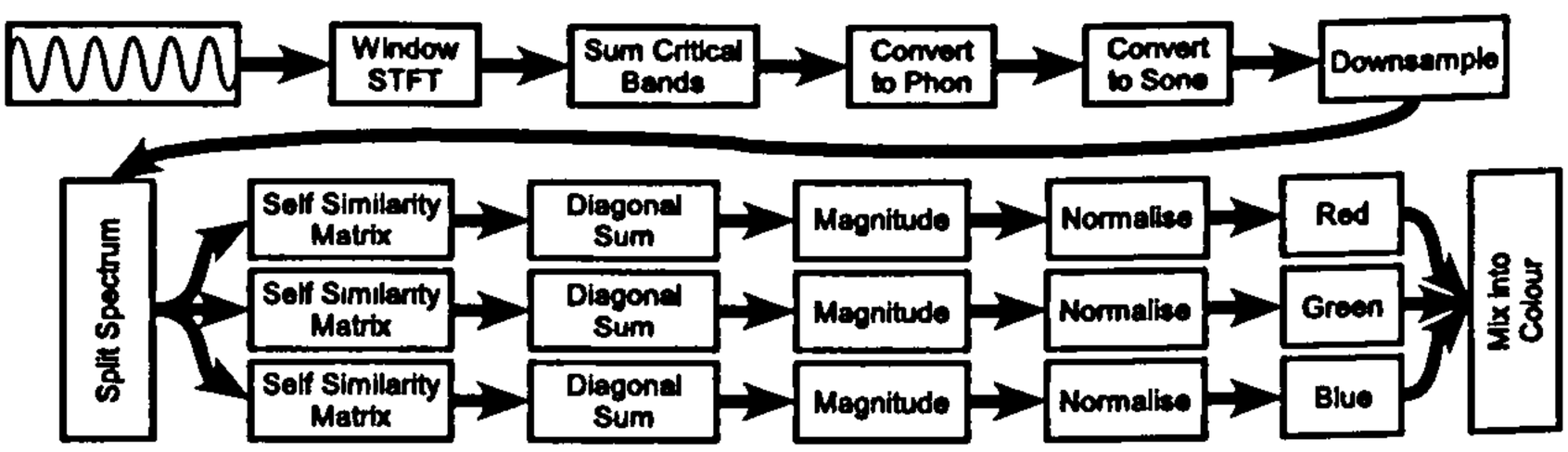


Figure 3.22: An activity flow chart of the bandwise loudness rhythm magnitude (BLRM) technique.

Expectations and Discussion

This is a rhythm-timbre based refinement; it should benefit music whose short-term autocorrelative properties change with regard to different voices over time. As such, more rhythmic music should benefit, such as hip-hop and dance, whose timbre may stay roughly

constant throughout, but whose rhythmic properties (perhaps with regard to different voices) changes.

The brightness of the colour relates to the strength of the rhythm at that point (i.e. how self-similar the signal is over a short period of time), whereas the hue describes where in the spectrum that simplicity lies. If there is little correlation, or if it is compromised between two successive and unique rhythms, however, then it should have a lower overall power and thus be darker in shade.

If the rhythmicity is mostly in voices in the upper part of the frequency spectrum, the hue will be cooler (blue/cyan/green). If in the lower part, the hue will be warmer (yellow/orange/red). Hues would therefore change whenever there is a shift in the relative lag-correlation of the three sub-spectra. As different portions of the total spectrum change in their self-similarity (perhaps by introduction, removal or interruption of voices), a greater shift in hue would be expected.

The number of spectra used, and thus the imprecision of the metric, is equivalent to the number of bands of the rhythm spectrum (or the cardinality of the vector we measure). Determining the optimum size of the spectrum is rather a black art; a smaller size results in better time precision and less processing. A larger size gives less noise and allows higher-level features to be captured; I settled on a window of around 1.5 seconds, which gave a generally reasonable output.

Demonstration

Rounds, an album by Kieran Hebden (released under the moniker *Four Tet*), is considered (e.g. by Clarke, 2003) as being a prime and largely seminal example of a genre called 'folk-tronica'. This is a fusion between the abstract electronic and folk music genres. *And They All Look Broken Hearted* ('*Broken*') is a track from this album which focuses on chaotic drumming and a repetitive harp melody, underlined with slow bass chord progressions, with other relatively soft 'instrumentation' progressing through the track. *Unspoken* is another track from the album, again featuring a repetitive melody (this time piano) and a clear progression, though with more esoteric sounds creeping in. Figure 3.23 shows the tracks under the BLRM visualisation.

In *Broken*, we can see that despite the instrumentation and loudness being relatively similar, the visualisation clearly differentiates the repetitive melody of the harp (1:25-2:30) from its 'chorus' theme (2:30-2:55). It also disambiguates between the solo harp and the harp with bass and drums which transitions at 1:45. The repetition of the section 1:45-2:55 at 3:17-4:27 is also clear to see, despite the addition of an extra, rather loud, bass drum.

Unspoken, like *Broken* focuses largely on repetition. It progresses throughout by adding and removing instruments[8]. Particularly notable in this visualisation is how the addition

[8]'instruments' is used loosely here; many of the sounds not only are not recognisable instruments, but

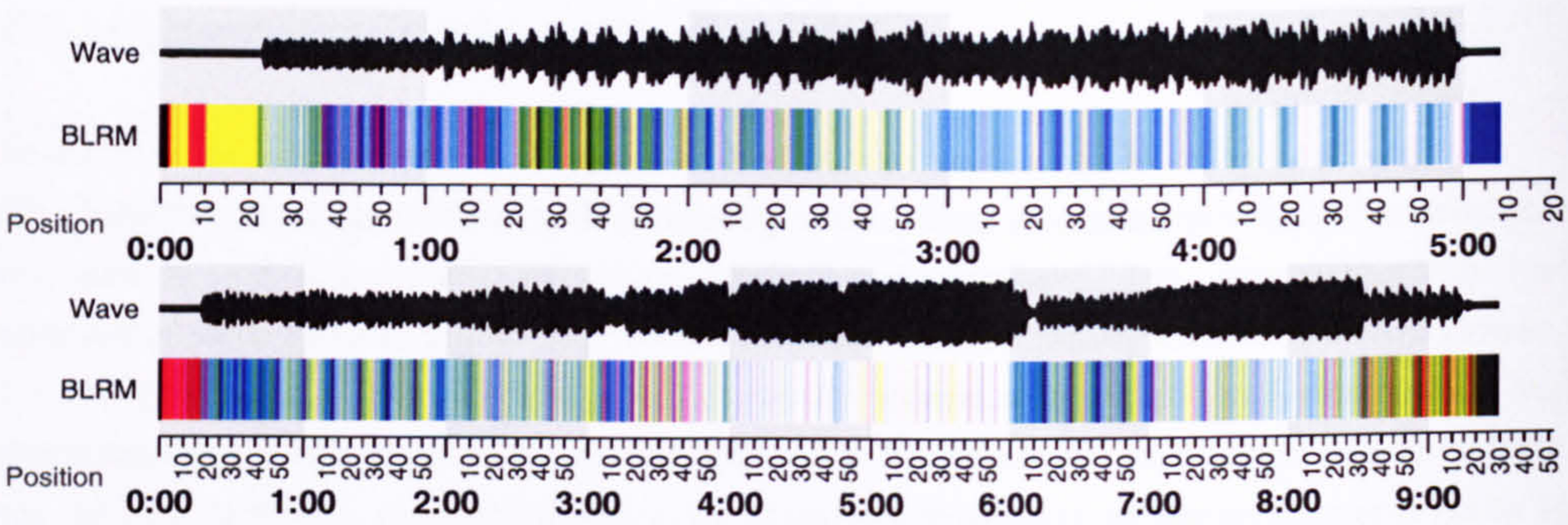


Figure 3.23: *Four Tet's And They All Look Broken Hearted* (top) and *Unspoken* (bottom) visualised as a basic wave (Wave) and with bandwise loudness rhythm magnitude (BLRM).

of a loud drum (6:11) to the much quieter bass, and later (6:14) the piano, makes very little difference to the blue stripes of the visualisation. This is despite changing the dynamics significantly. What the visualisation clearly shows is the addition of chaotic sounds (5:00), similar to a feedback loop one would expect to hear from a microphone being moved close to a loud speaker amplifying its signal. The change from blue to a pink/white hue at 3:40 seems to be caused by the 'dirtying' of the rhythm, with the addition of an extra bass sound and subtle string-like theme. Interestingly the removal of the primary melodic instrument, the piano, at 3:06 causes no significant change on the colour.

With these two tracks, the effects of visualising through rhythm are relatively noticable; despite changes in timbre, the visuals remain largely unaffected due to the extreme similarity of rhythm.

Without Colour

A basic version of the the rhythm magnitude ignores the per-channel aspect of the original method, and merely notes the lag-correlation evident in the entire spectrum. The formal definition is given by $\mathcal{P}_{LRM}$:

$$\mathcal{P}_{LRM}(c, \vec{\mathbf{b}}) \equiv \|Rhy(L(\vec{\mathbf{b}}))\|, \quad t \equiv \frac{s}{6} \tag{3.25}$$

Figure 3.24 depicts the classical track *Sabre Dance*, a highly rhythmic fast-paced orchestral track whose rhythm is maintained by strings throughout. Woodwind and brass give the main melodic content, whose playing affects the rhythm magnitude by disrupting it and by reducing the brightness of some primary colours.

The basic LRM visualisation does not clearly denote the point where the loud brass instrumentation exits the foreground at 0:50; the BLRM in contrast changes hue from pink

---

are not even immediately recognisable as being tonal at all

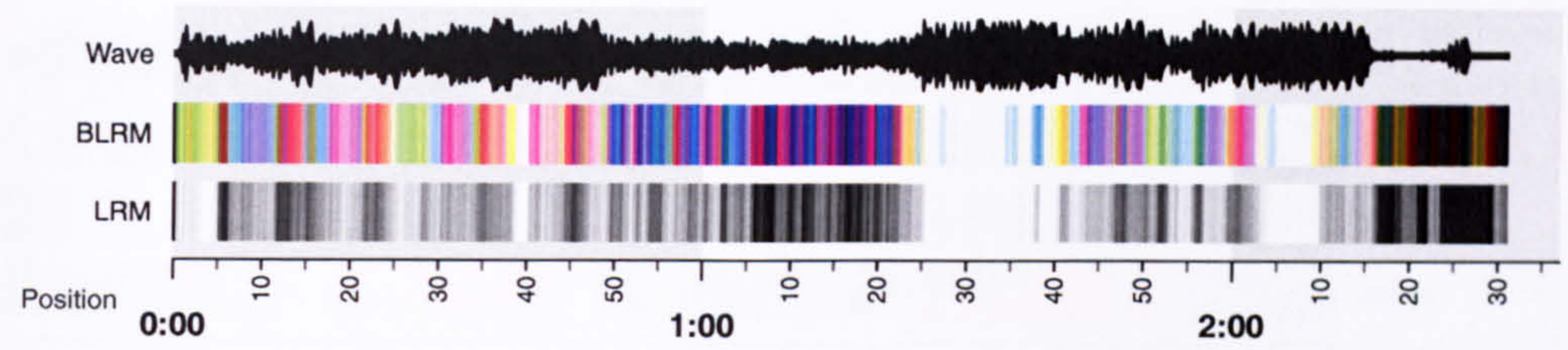


Figure 3.24: The track *Sabre Dance* by *Aram Khachaturian* visualised with bandwise loudness rhythm magnitude (BLRM) and loudness rhythm magnitude (LRM).

to cyan/yellow. The point at which the woodwind melody is introduced at 1:07 is also clearly marked with a change of hue, again to blue/red, denoting disruption to mid-range's overall rhythm strength. This change, in the context of the entire spectrum, is subtle and difficult to identify clearly.

Without Psychoacoustics

Removing the psychoacoustic loudness scaling from the method has the effect of reducing the relative dynamic ranges of the three channels, giving a smaller range of hues. The formal definition is given by $\mathcal{P}_{BBRM}$:

$$\mathcal{P}_{BBRM}(c, \vec{\mathbf{b}}) \equiv \|Rhy(Bark(\vec{\mathbf{b}}))_{tc\ldots tc+t}\|, \quad t \equiv \frac{s}{6} \tag{3.26}$$

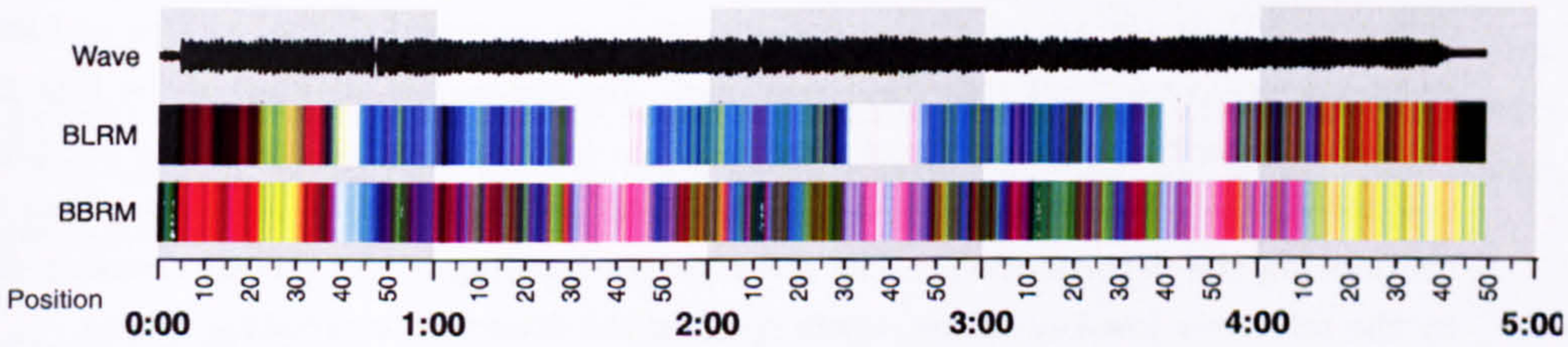


Figure 3.25: The track *Without Me* by *Eminem* visualised with bandwise loudness rhythm magnitude (BLRM) and bandwise Bark rhythm magnitude (BBRM).

To illustrate this, we use a rap track by Marshall Mathers, also known by the name *Eminem*. Figure 3.25 shows the track *Without Me*. In the BLRM image, we can see it is far clearer to find the intro, verse and chorus changes; the outro is also clearly visible. Blue appears due to Mathers' vocals being the rhythmic element which changes throughout the track; other rhythmic voices contributing to the spectral loudness, such as the drums, tend to be constant throughout, thereby having no effect on the colour balance. The white appears due to the self-similarity of the whole sound; the spectrum is far 'fuller' from more instruments being introduced, all with a simple repetitive theme.

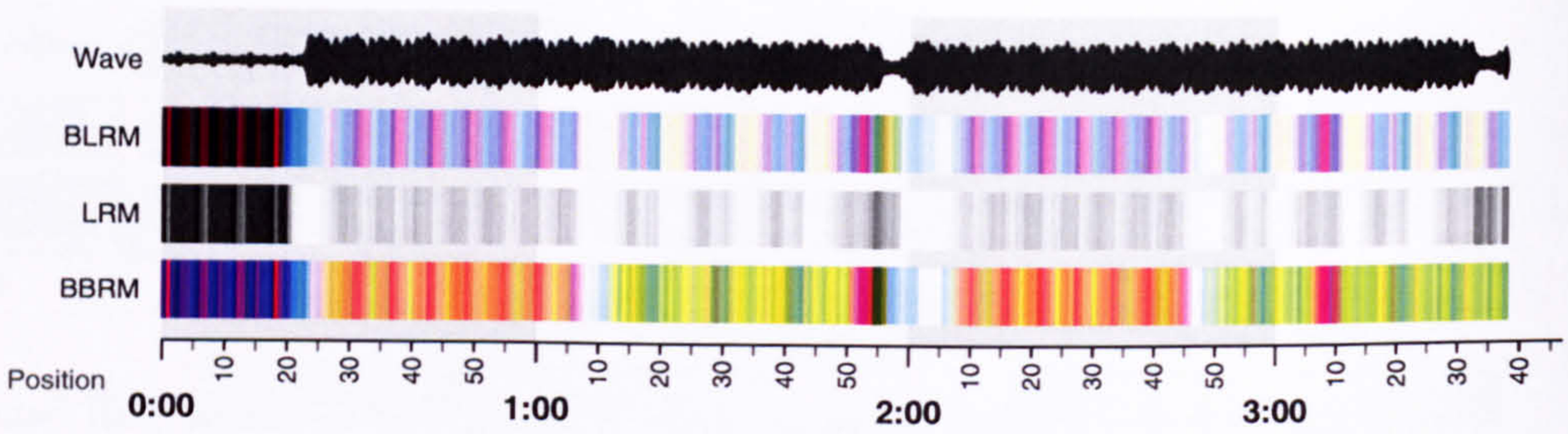


Figure 3.26: The track *Time is the Enemy* by *Quantic* visualised with bandwise loudness rhythm magnitude (BLRM) and bandwise Bark rhythm magnitude (BBRM).

As a poignant counter-example to the usage of psychoacoustics here, we can look at the excellent visualisation afforded by the Bark-based rhythm spectrum in figure 3.26. The track is *Time is the Enemy* by Will Holland (released under the moniker *Quantic*), a multi-talented musician who often plays lead duty on guitars, bass, double bass, piano, organ saxophone and percussion. In the figure, the waveform makes visible only the two large sections to the track, with the gap at 1:56 to 2:01. Aside from the start and finish, little else is visible.

In the BBRM image, we can see the very clear sectioning of the track. The introduction is given by the blue/red stripes at the beginning, the first repeating theme with red-yellow-red striping, and the second repeating theme with green-yellow-green striping. The gaps with loud noisy guitar notes played are white. The first theme is just a repeating high-note arpeggio on the piano, the second incorporates a second theme, as well as having a lower-key and very quiet repeating arpeggio. The red-yellow striping of the first theme is produced by the difference between the piano with the drums (red since the piano is not immediately lag-correlative), and yellow with only the drums (since the piano is not introducing the decorrelation). The green-yellow banding of the second theme is caused by the relatively low-key theme, again spoiling the otherwise correlating drums.

The BLRM image, by contrast, is fairly nondescript and barely better than the included (non-bandwise) LRM image, showing that each section is in fact two subsections. In particular, the stressed purple marks at 1:53 and 3:17, each of which represents a loud and unexpected guitar riff, is far clearer in the BBRM image than in either of the others.

3.3.6 Novelty Score

Definition

The novelty score was introduced by Foote (1999a). It provides a value determined by the cross dissimilarity of the portions of signal both before and after the moment in time. Like the rhythm spectrum, it relies upon a prior abstraction of the signal known as a

self-similarity matrix, which is calculated simply by evaluating the similarity of the signal to itself at varying intervals (given by $x - y$). Section 3.2.2 gives a full defintion of the technique. Formally, we define the colour-projection function $\mathcal{P}_{Novelty}$

$$\mathcal{P}_{Novelty}(c, \vec{\mathbf{b}}) \equiv |N(L(\vec{\mathbf{b}}))| \tag{3.27}$$

where $c \in \{0, 1, 2\}$ and represents the colour channel (either red, green or blue respectively), and $\mathbf{b}$ is a vector of spectra preprocessed from the signal. Futhermore, we define the function N:

$$N(\vec{\mathbf{x}}) \equiv \sum_{i=0}^{i<s} \sum_{j=0}^{j<s} \mathbf{M}_{\vec{\mathbf{x}}}(i, j)\mathbf{K}(i, j) \tag{3.28}$$

$\mathbf{M}_{\vec{\mathbf{x}}}$ is the self-similarity matrix determined from the series of spectra $\vec{\mathbf{x}}$ (both of size s), and $\mathbf{K}$ is the Gaussian kernel function given in section 3.2.2. Figure 3.27 shows the process as a pipeline in a canonical fashion.

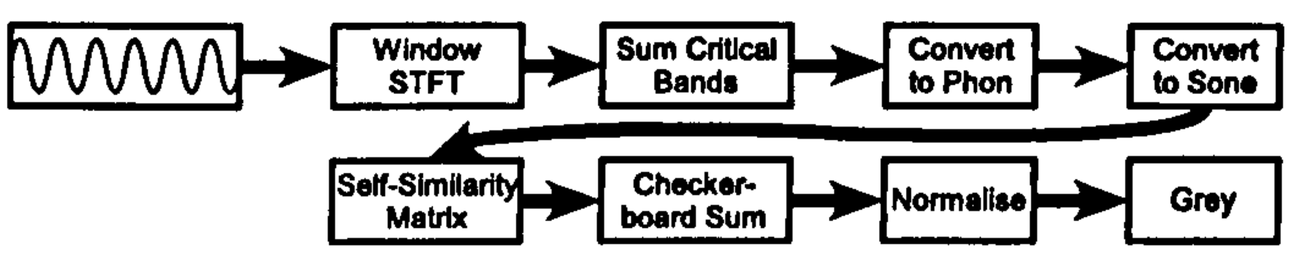


Figure 3.27: An activity flow chart of the bandwise loudness rhythm magnitude (Novelty) technique.

The size of the self-similarity matrix and accompanying checkerboard kernel were experimented with and qualitatively evaluated. I found that a value of around 128 spectra (1.49 seconds) provided a good balance between time precision and larger scale feature presentation.

Demonstration

Figure 3.28 shows a visualisation of the well-known James Bond theme, which will suffice to illustrate the form of visualisation that a novelty score produces. The novelty output is visibly different to the spectral magnitude, since it is one level of indirection away. Rather than showing the track directly, and allowing the user to determine when the metric changes enough to denote a feature, it instead shows the changes directly, essentially providing a differential view.

It is clear to see the main orchestral figure plays four times between 0:40 and 1:00. This is despite slightly different sets of instruments and slight variations on the figure. The repetition of the guitar theme, twice at the beginning and twice at the end, is also visible as several gray lines followed by a block of black. The main orchestral climax after

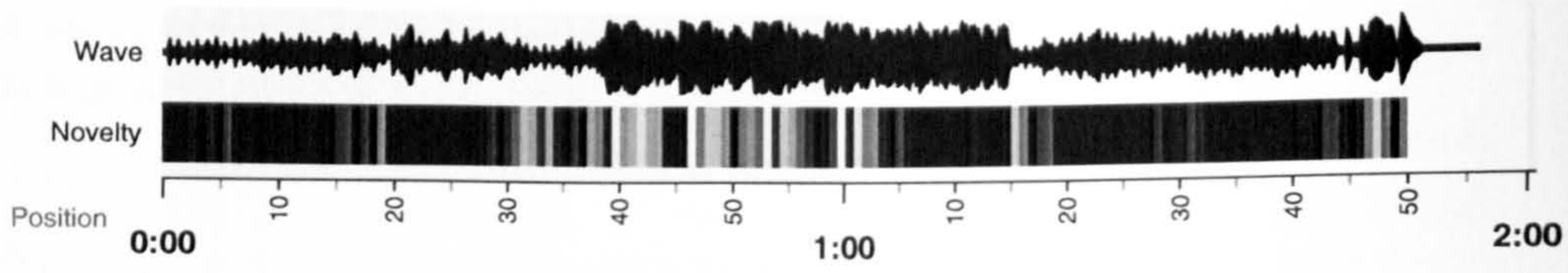


Figure 3.28: *James Bond Theme* by *John Barry* visualised as a basic wave (Wave) and with the novelty score (Novelty).

the figures, which runs from 1:06 to 1:16 is not discernable as being any different from the guitar themes after it. This illustrates the main drawback of this method; the refinement is so high that too little information can be made from visual inspection of the output. We will nevertheless consider it, since it may yet show musical properties that other methods are unable to show.

3.4 Discussion of Methods

We will now discuss the strengths of weaknesses of each of the techniques proposed. We will do so by means of a side-by-side comparison of these techniques, over several pieces of music from a broad range of genres, including jazz, classical, rock and rap. We will, where possible, base the critique on musical aspects of the audio such as long-term structure, variation and rhythm. The techniques for comparison will be the three previously proposed:

- Bandwise loudness smoothed magnitude (BLSM).
- Bandwise mel-frequency cepstral magnitude (BMSM).
- Bandwise loudness rhythm magnitude (BLRM).
- Novelty score (Novelty).

These will be compared alongside two variations without colour:

- Loudness smoothed magnitude (LSM).
- Loudness rhythm magnitude (RM).

We will also provide the amplitude projected as height, with a one second moving average (Wave). I consider this type of visualisation of musical audio as canonical and a reasonable baseline.

The specific tracks used do affect the content and conclusions of the critique, even within reasonably well-bounded genre categories. The selection criteria used for a track were two. Firstly, the track should be representative of the genre. Secondly it should

properly represent how the genre's facets manifest themselves within the visualisations. The selection given in this section is small, but is enough to illustrate the main differences between genres in terms of a particular visualisation method.

3.4.1 Test Tone

We begin by comparing visualisations with a manually-generated test tone; a series of synthetic 'plucks' with a simple ADSR (attack-decay-sustain-release) envelope modulating a pure tone. The plucks are at a constant frequency of 880 Hz but change in tempo in three blocks (70 bpm, 105bpm, 158bpm). A further amplitude envelope is applied over the audio in order to keep perceived loudness constant. Figure 3.29 shows the generated images of this audio.

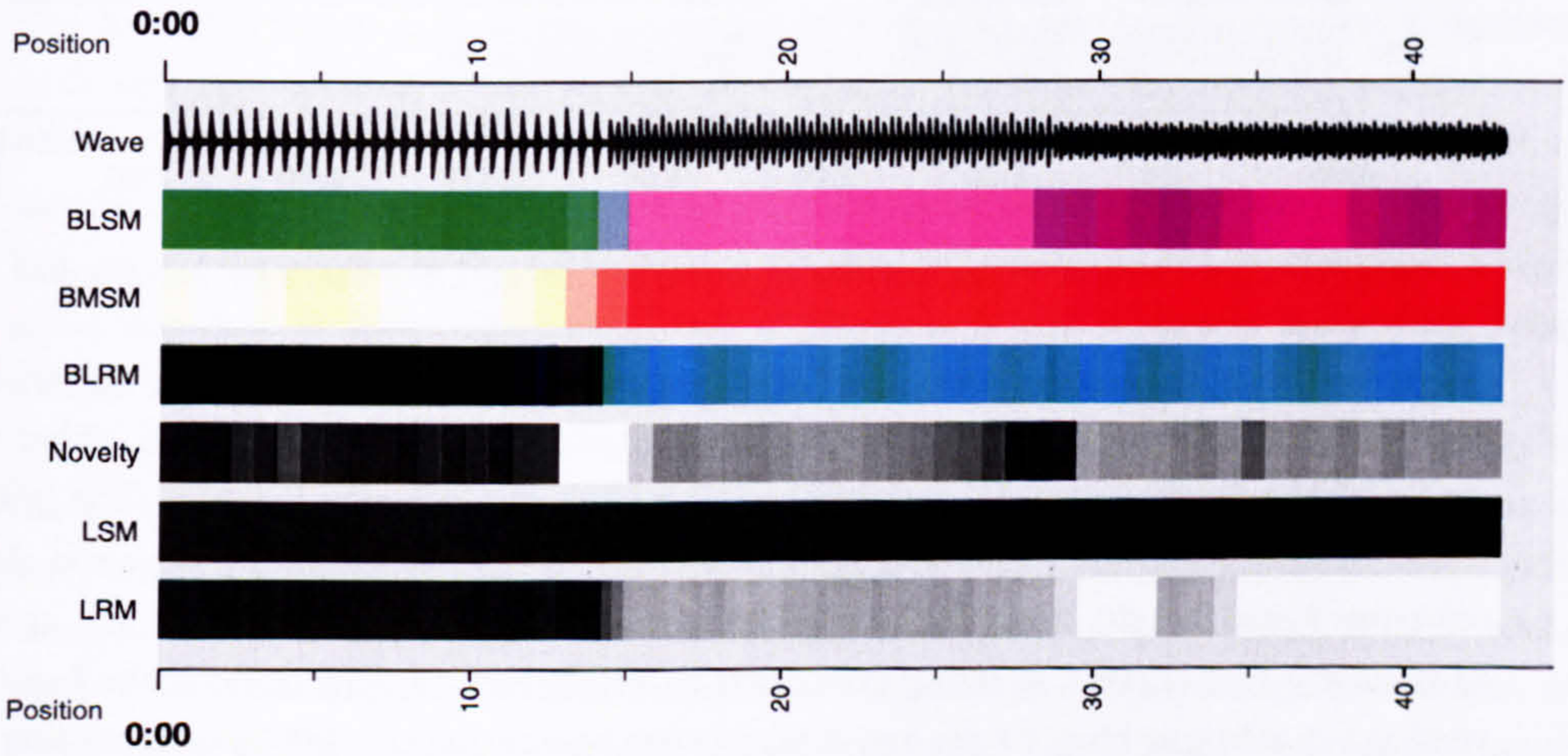


Figure 3.29: Several visualisations of the test tone 'plucks'.

The first thing to notice is the lack of a visualisation for the loudness spectral magnitude; the loudness stayed so constant throughout, that the normalisation stage reduced the image to complete darkness! This contrasts to the bandwise variant (BLSM), which, through purely incidental interference effects, differentiates the tempos by frequency bands. Of course the colours it produces are quite arbitrary. The MFCC based visualisation apparently benefited in the same manner through the interference.

The self-similarity matrix based visualisations (Novelty, LRM & BLRM), also distinguish the first far better than the latter two. The novelty shows the boundaries fairly well with the tempos being largely indistinguishable.

3.4.2 Trip-Hop

The discussion of actual musical audio begins with the trip-hop track by Josh Levis *Stem/Long Stem*. This is tremendously rhythm and timbre based, with both a great amount and a broad range of samples.

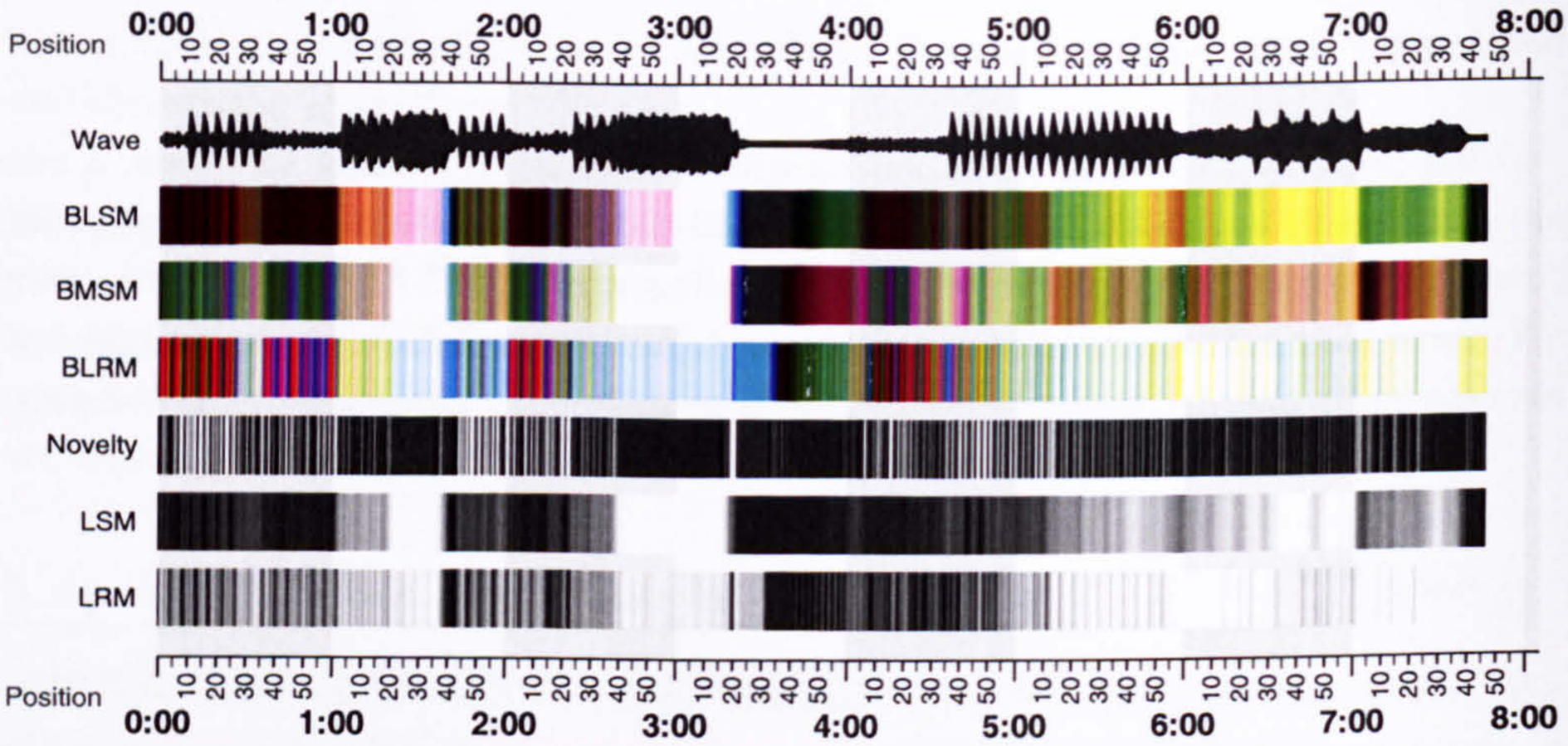


Figure 3.30: Several visualisations of the track *Stem/Long Stem* by *DJ Shadow*

In the first half of the track, dominated with a foreground melody of a guitar, each of the spectral magnitude based versions denote the thematic variations well, either with contrasting intensities (LSM), patterns (wave) or hue and intensity (BLSM). The latter is particularly effective with the fairly subtle thematic variations of this track; we can clearly see transitions at 1:20, 1:40, 2:40 and 3:00 corresponding to extra instruments, changes in intensity and speed. A string crescendo is visible around 3:19 in both the bandwise versions as a bright blue stripe, and a pipe organ sample at 3:40-4:05 is recognisable as a green-blue patch of increasing brightness.

A large brass-dominated section (5:14 onwards) in the second half is coloured green-yellow in BLSM and is simply brighter in LSM. Notably, however, in the wave image, the same portion is not noticeably different from the textures found elsewhere in the track, though the textures can clearly be seen to change. The change at 5:58 represents the end of a guitar part, and the texture introduced at 6:26 represents the return of the guitar melody although at a lower tempo. At 7:06 the guitar is removed permanently, leaving only the brass.

The BMSM image proves around as useful as the other spectral magnitude versions for the first half, alternating around green, gray and white for the various different variations. Colours are representative of the overall timbre of the music; e.g. green represents the playing of a string instrument in pizzicato. The red block fading in at 4:08 represents the

aforementioned pipe organ. The second half of the track is less clear overall; the only clear artifact is the ending of the guitar at 7:06.

The LRM image discriminates well at 1:02 with the onset of drums, and again at 1:20 with a more regular and frequent percussion. The break and subsequent theme repeat are visualised appropriately as a black bar and with a texture similar to the initial portion of the track. The loud and regular percussion portion between 2:40 and 3:20 is visible as a large bright patch. The break following the crescendo at 3:23 has only a single black line to disambiguate it from the silence which follows. The lag-correlation of the silence is apparently similar to that of the percussion. In the second half of the track, the main feature that is noted is the introduction of the occasional brass note in the background at 5:12. The following texture finishes with the end of the infrequent (but regular) percussion at 5:57. The further melodic progression and instrumentation goes largely unrepresented.

The BLRM image denotes significantly more information than RM; the silence following the crescendo of 3:23 is now covered cyan. This a clear if non-obvious clue to the content. The following organ fade-in is represented in a similar way to the spectral magnitude methods. Two portions of the track at the beginning are separated and coloured distinctly differently (orange until 0:28, purple until 1:01); the musical difference being the lack of bass in the second portion. This goes largely unnoticed on all other methods. The purple hue is used once more in the LRM version, where similar instruments and no bass is once again apparent at 2:02. The second half of the song is still relatively featureless, being mostly bright white.

The novelty method is once again somewhat difficult to decipher. The build up to the percussion at 1:03 is made visible, though the proper introduction of the percussion itself is not represented. The removal of percussion at 1:44 is represented as a change from black to white, it flips again when the percussion restarts at 2:41, before a large white band denotes the climax. The black block after this (3:23 to 4:05), encompasses silence, an organ fading in, and the onset of a high-pitched xylophone-like sample. The texture change and reduction in intensity corresponds to the removal of a (non-self-similar) sample of a man complaining. Subsequent changes in the track, both in terms of melody, instruments and speed, go without representation.

In conclusion, either the BLSM or the simple wave are the best overall visualisations; the wave providing more textual information, especially with regard to the end of the track and the BLSM providing a clearer overall picture with its use of hue.

3.4.3 Rap

A piece of rap music, by artist Andrew Turner, named *The Force* is now for consideration. Figure 3.31 depicts the visualisations of it.

The constant intensity and rhythm of this rap track is immediately noticeable in the

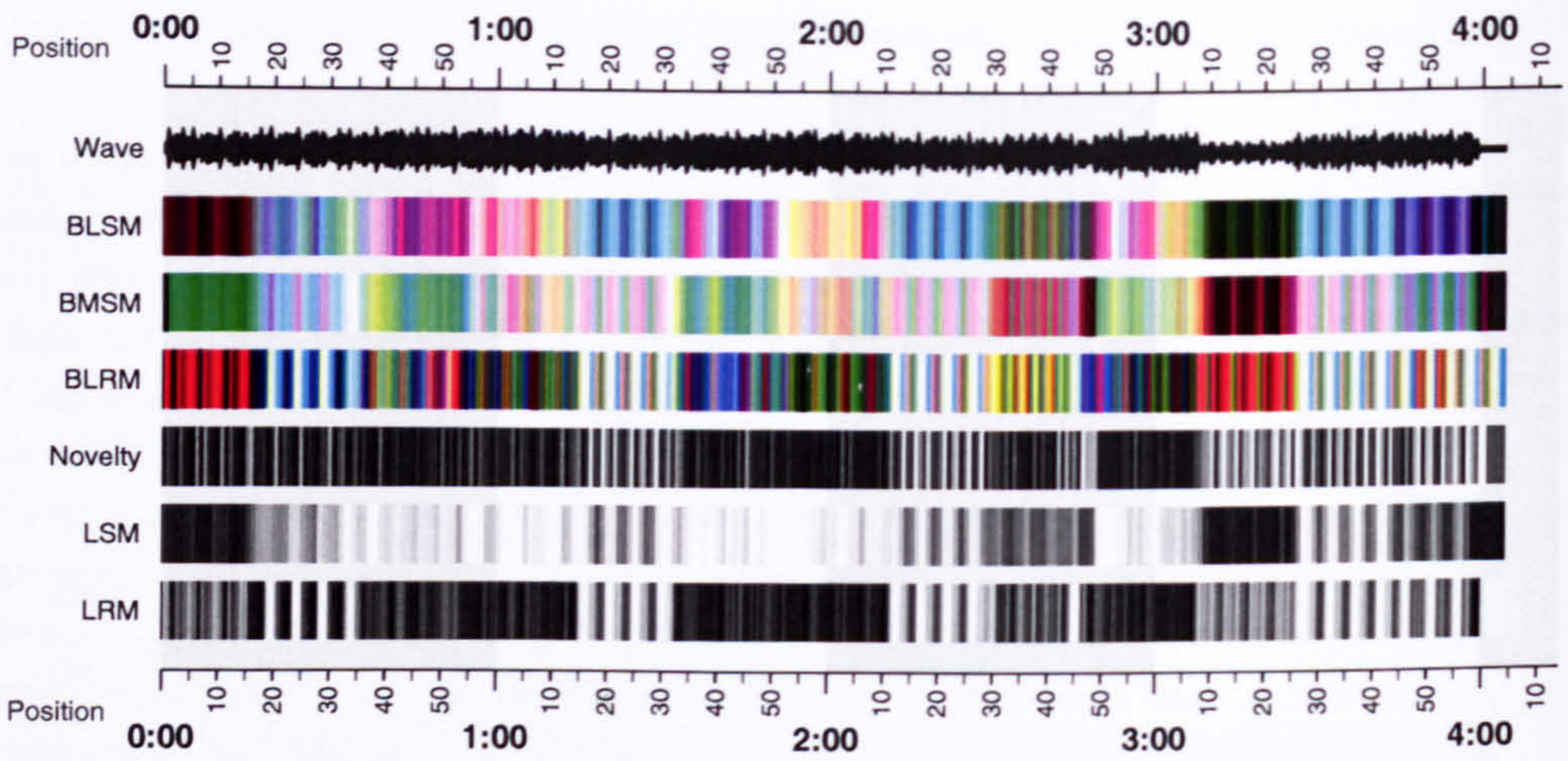


Figure 3.31: Several visualisations of the rap track *The Force* by *Aim* (featuring *Q'n'C*).

basic wave display, which gives little away about the musical content.

With the more advanced psychoacoustic preprocessing of LSM, the three choruses are immediately more recognisable at 1:15, 2:12 and 3:28. The inclusion of colour with BLSM gives a slightly more informative view; choruses are blocks of four light and dark-blue bands. Verses of pink and orange striped blocks represent the voices of the first and second rapper respectively. Each of the other visible combinations of colour similarly represent an instrument or voice. The BMSM method presents similar information to BLSM, though somewhat less clear, with the intensity distribution seeming to be heavily biased to the bright end.

The three self-similarity based methods each make an impressive visualisation on this track, though the novelty is still a last place, with cues for structure taking the form of "more white lines" in the case of choruses. The LRM image is probably the clearest of all methods, and certainly the clearest of the monochrome methods. The general form of the track is immediately visible as a set of roughly equal length blocks, each containing four repetitions of a basic figure. Each of the blocks containing different voices are visibly differently shaded.

The addition of colour in the BLRM image allows the recognition of the piano at 2:30 with the yellow stripes, and of the rappers' voices as blue and green. However the initial onset of rapping at 0:38 is not represented especially well, with confusing pink stripes introduced and a highly variant colour. The red in the break at 3:09 does not denote the same as the red at the beginning of the track; both have a large bass content, though the red of the break does actually have some (rather minimal) green content representing four wails. Though the colour is slightly more instructive to the content, it seems in terms of

a clear expression that the non-bandwise variant, RM, performs better.

3.4.4 Jazz

The jazz track *Take Five*, performed here by the *Dave Brubeck Quartet* will be considered now.

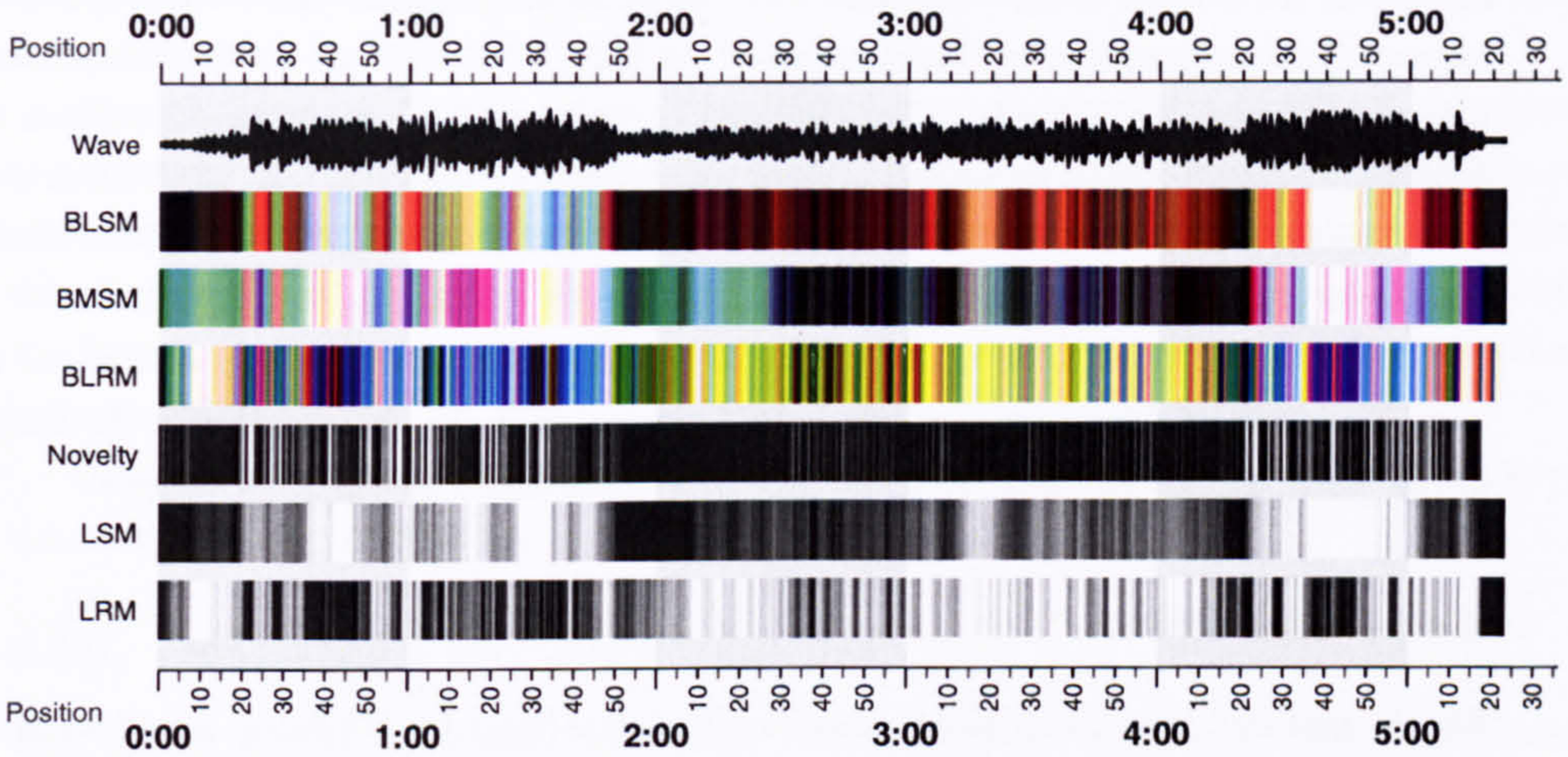


Figure 3.32: Several visualisations of the jazz track *Take Five* by *Dave Brubeck*.

The Novelty visualisation is once again quite noisy, the areas of lightness that it does have correspond to the saxophone playing.

BLRM has a yellow portion for the part of the track without saxophone, and notes the small counter-melody with the clear and dark red and blue stripes at 0:36 to 0:49, and again with the lighter stripes at 4:37 to 4:50. The drum and piano duet denoted by the yellow hue has a significant amount of texture change on it; whiter parts correspond rarely to visible a more regular and frequent drumming style; darker and stripey parts correspond to infrequent drumming and, specifically, in the quick runs of molto crescendo.

The primary melody (made up of a repetition of another smaller melody) is reasonably clear in the LRM image as a double white bar at 0:23, 0:50, 4:25 and 4:54.

The BMSM clearly shows the end of the first saxophone portion, with the change to green at 1:52. The change to a lighter green denotes a somewhat small change of the ride cymbal being used less frequently; the change to black is representative of the end of the ride cymbal to dictate rhythm. The subtle light portion around 3:19 represents the poco crescendo of the piano. The BMSM does not make clear any of the changes of melodies, and for the most part the representation is noisy and difficult to discern.

The wave gives a basic overview to the song's structure but little else is discernible; slightly louder (larger) portions are visible at the beginning and end, but other than that

little is clear.

The LSM, with the psychoacoustic loudness measure, clearly marks the aforementioned piano poco crescendo (3:16 to 3:24) and drumming introduction (2:10), with clear changes in brightness. The saxophone playing is clearly denoted with the bright regions at the beginning and end of the track. Its repetitive nature, however, is not made clear. With colour, in the BLSM image, the repetition becomes the clearest yet, with a green band followed by a longer pink/red band repeated (at 0:23, 0:50, 4:25 and 4:54). This shows the nature of the smaller melody being repeated to make a single main melody, which itself is repeated four times. The period of minimal drumming between 1:49 and 2:10 is clearly marked as a dark patch; less clear is where the cymbal stops. As with the LSM method, the piano crescendo is a clear light patch.

On the whole, the BLSM is the clear winner, depicting the musical aspects of the track more clearly than the others. I would note, however, that the visualisation of the variation of the small counter-melody with the BLRM is unmatched in the other images.

3.4.5 Classical

For the classical track, we defer to the *Nachtmusik Allegro* by *Wolfgang Amadeus Mozart.*

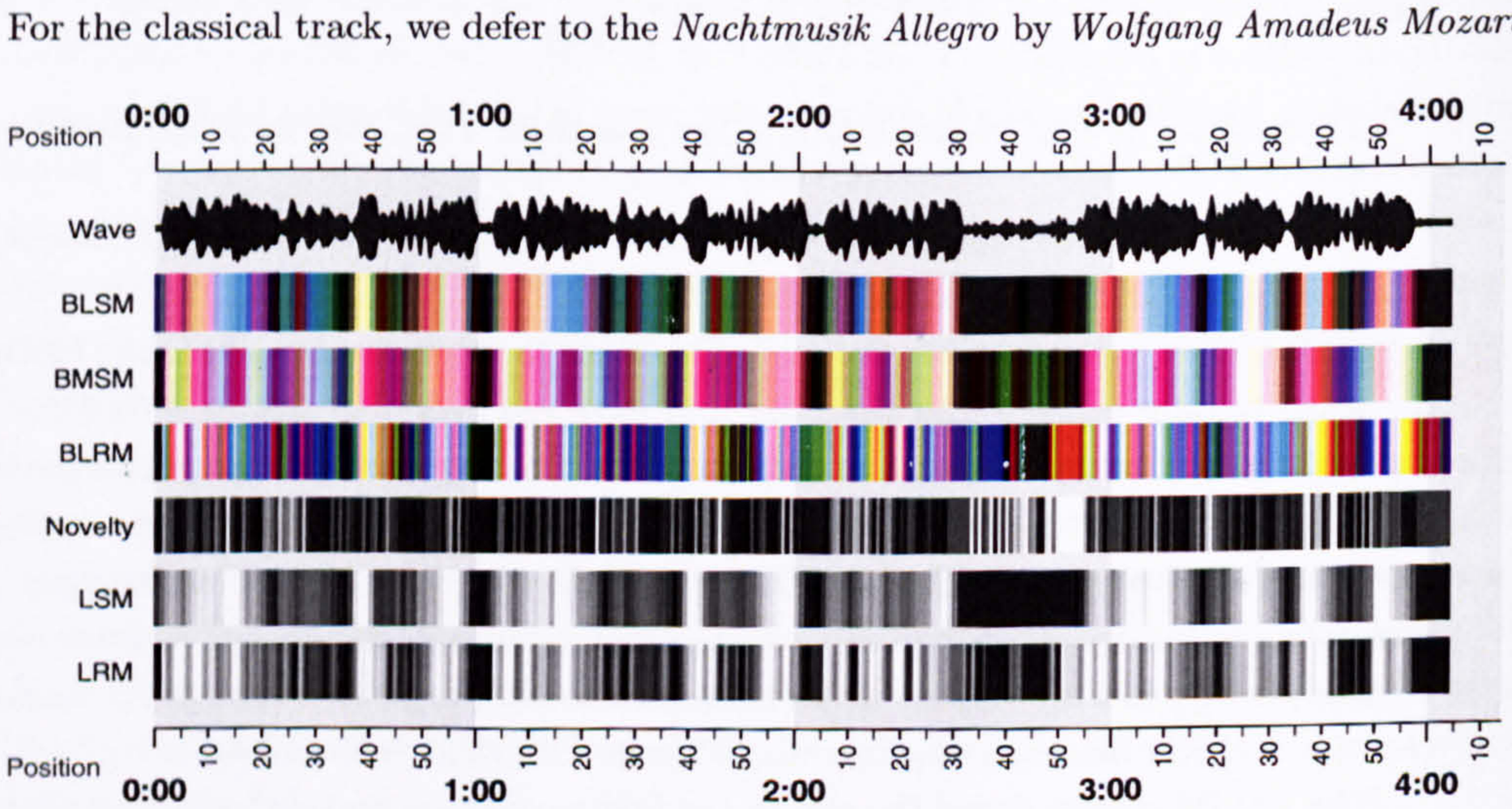


Figure 3.33: Several visualisations of the classical track *Nachtmusik Allegro* by *Wolfgang Amadeus Mozart.*

With this track, and indeed much classical music, what is immediately noticeable is the difficulty of seeing overall structure. I see this as most likely caused by the greater tendency of classical music's structure to stem from "higher level" musical aspects, such as harmony and (counter-)melody, rather than "low-level" aspects, readily analysable from

the signal of the performance (e.g. timbre)[9]. This would happen naturally through its tendency to separate composition from performance.

The overall view of structure is improved somewhat by the techniques that use medium-term windowing (BLSM, LSM & BMSM). The wave depiction suffers from clutter due to the large amount of short-term dynamics. With the exception of the particularly cluttered BLRM method, the slower, quieter portion of strings between 2:30 to 2:56 is readily recognisable.

Though having a limited palette, the BMSM method seems to give the best overall presentation in terms of information. Repetitions are relatively clear and unambiguous, with the first minute largely resembling the second and the start of the third. Aside from the aforementioned portion of strings, colours seem to have little obvious connection with the figures of the theme, apparently being caused by occasional notes reaching a critical band threshold.

Of all genres, classical is the one which these direct signal-based visualisations have the most difficulty presenting.

3.4.6 Downtempo/Electronic

The Dining Rooms, an Italian downtempo electro-acoustic jazz band provide the next track, *Occhi Neri*.

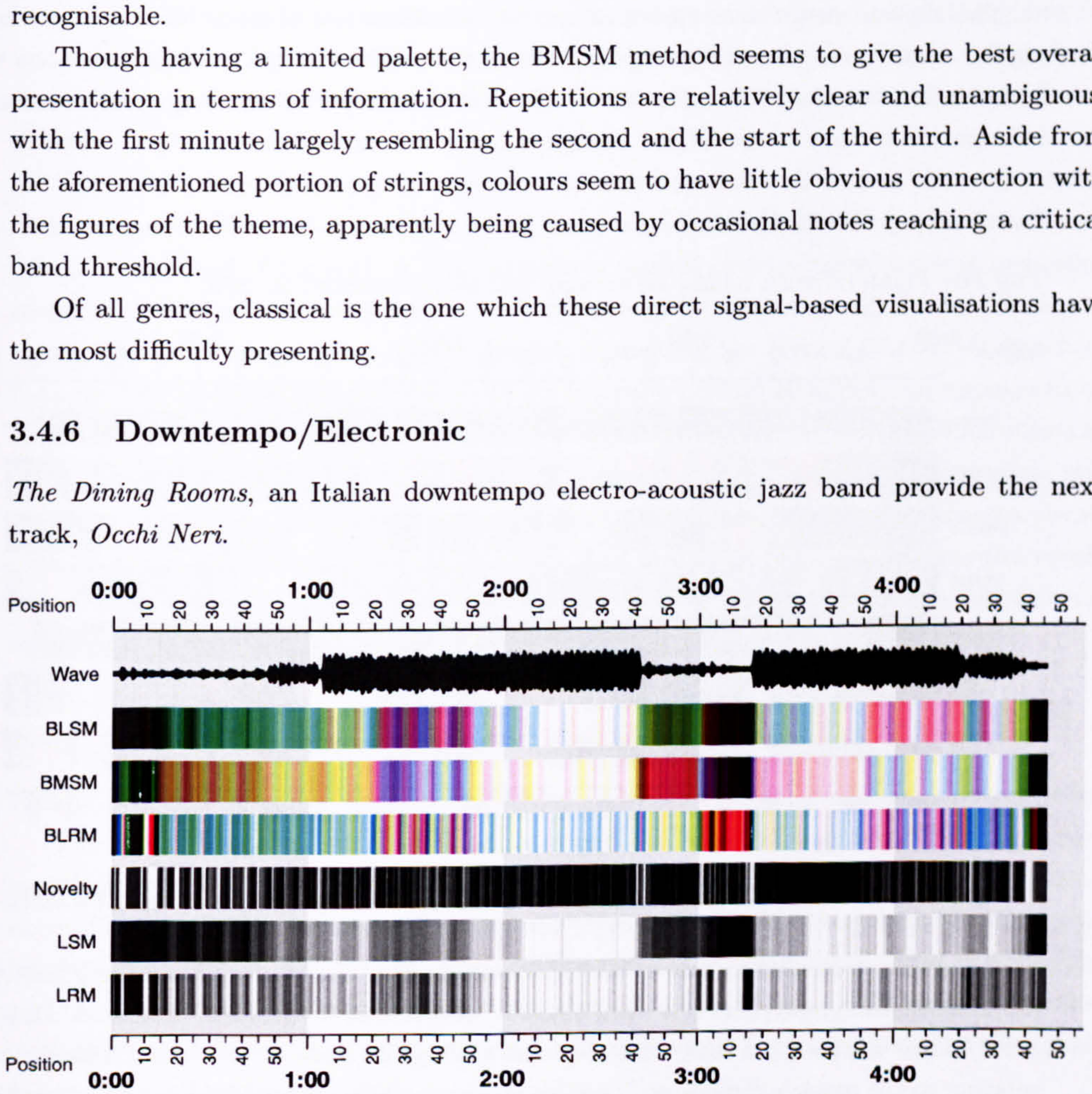


Figure 3.34: Several visualisations of the downtempo track *Occhi Neri* by *The Dining Rooms*.

With downtempo music the visualisations are usually fairly easy to spot; tracks are typically clearly segmented and use a range of colour. The regular texture of the segments

[9]Middleton (1990) makes a similar observation

also tends to be visible.

The structure and content of the music is easily visible with each of the basic magnitude methods. There is clear break from 2:45 to 3:20, and the track gets progressively louder until that break. With the bandwise methods, we can see the change in colour resulting from the loss of a mid-range voice (a background guitar); this manifests itself as pale-green-yellow to purple in the BLSM image and yellow to pale-purple in the BMSM image.

The rhythm magnitude measures appear somewhat less obvious; the LRM method's regular texture represents the regular beat in the music, though the brightness has very little obvious musical relevance here. The colour of the BLRM image provides similar information to that of the BLSM though is paler and more faded.

3.4.7 Pop/Rock/Metal

The *Foo Fighters*' rock ballad *Generator* will be commented on now.

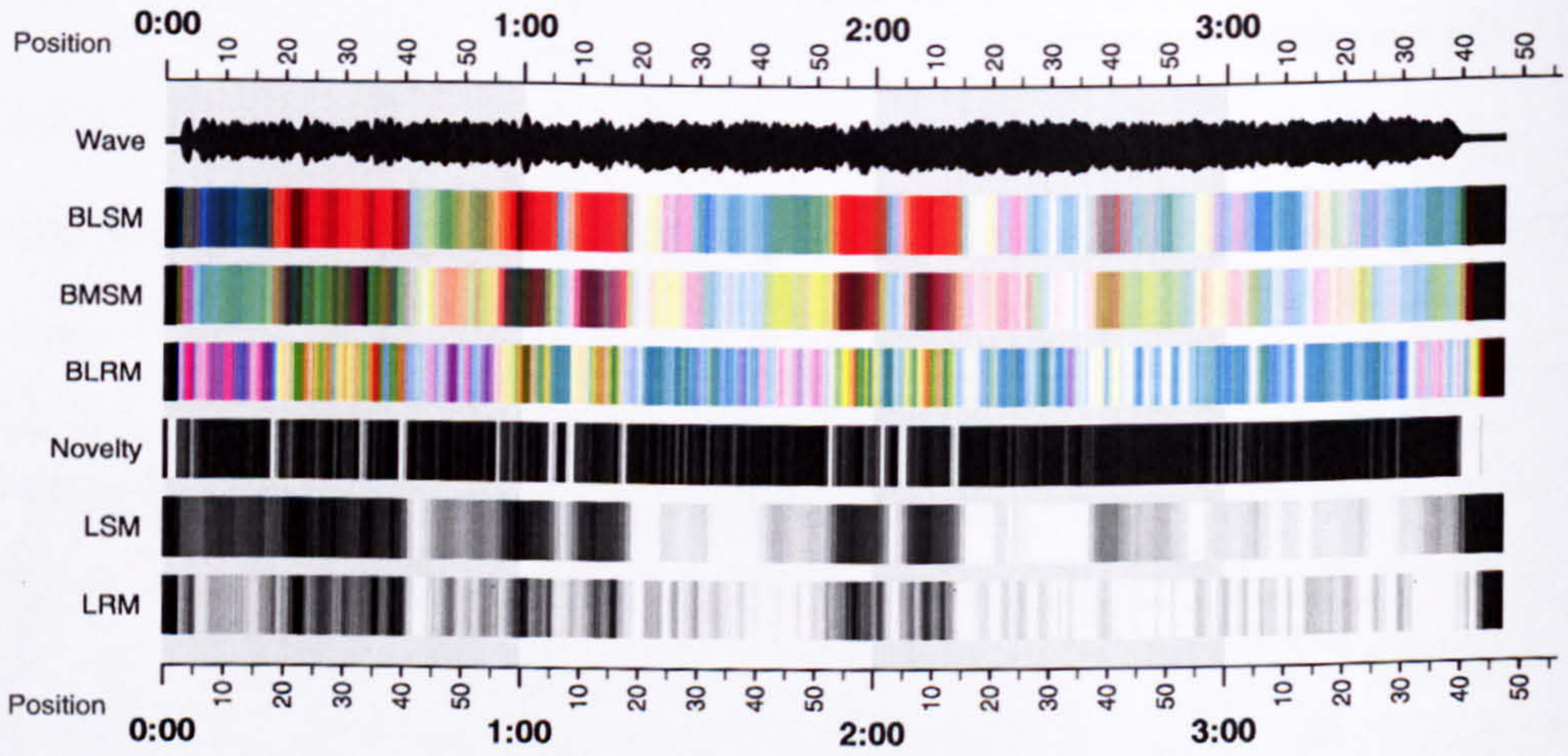


Figure 3.35: Several visualisations of the rock ballad *Generator* by *Foo Fighters*.

In a manner similar to that of the rap track by *Aim*, *Generator* has a largely consistent amplitude throughout, giving a wave image largely devoid of artefacts. Contrast this to the LSM image which, from its better loudness scaling, presents us with information readily relating to the music's dynamics. The two repeating choruses are visible as two dark blocks separated with a bright band. Left unclear are the relations between the rest of the track; in fact it follows a roughly A-B-A'-B'-C-B'-C-D-C-C structure, with the A' being similar to A but without vocals and B' being similar to B but with a clear guitar riff in the middle as well as at the end.

The BLSM image has bright and contrasting cues which depict the structure clearly; the two B' choruses are clear, this time as red blocks with blue in the middle. A (start to

0:19) could perhaps be identified as a faded and slightly dark version of A' (0:44 to 0:56), and we can see that vocal parts are denoted with red, guitar as light blue/cyan, and both as pink/gray. The initial guitar solo is deep blue; the vocal dominated verses are a clear red, with a cream band for the distinctive guitar riff. The small guitar solo (D) sandwiched between choruses (C), is also visible on close inspection between 2:38 and 2:55. Repetition of the choruses (C) is also visible as gradients from white to cyan.

The BMSM image is no better; vocals are denoted by darker faded green; the guitar riff is a faded purple. The rest of the track is largely featureless, the few faint artefacts have no musical meaning. Unlike with BLSM, the combination of vocals and guitar appears no different to guitars alone.

The novelty visualisation gives a reasonable monochromatic display of the track; verses with the guitar riff are visible as four white lines, with the centre two being closer together (the start and end of the riff). A faint pattern of several closely spaced lines of increasing intensity at 2:58 and again at 3:26, are neither especially musically significant nor repetitions. The LRM image does a reasonable job of showing the central B'-C-B'-C structure between 1:00 and 2:38. Quiet vocal parts are typically darker, though there appears to be little more clear information. This is improved again by the BLRM image, which seems to show the same structural information as BLSM except with more noise, less precision, and vocals not clearly corresponding to any particular hue. Pale yellow-green denotes vocals with no major guitar proportion around 0:25, 1:04 and 2:00, but cyan denotes the vocals when mixed with guitars throughout the choruses at 1:24, 2:27 and 3:05. The entire guitar solo and the starts and end of the various features are not clearly marked either.

3.4.8 Dance/Classical Fusion

Finally, the high-dynamic range track *Clubbed to Death* by *Rob Dougan* will be reviewed.

The main structure of the track changes between strings (to 0:23), percussion/bass/samples/strings (to 2:53), piano (to 4:00), percussion/bass/strings with a sweep filter (to 4:38), percussion/bass/samples/piano (to 5:53), bass/piano (to 6:20) and finally piano/strings. This is noticeable on the wave image through the small (percussion) and large (classical) patches, with the transitions of 4:38, 5:53 and 6:20 being somewhat more subtle. Aside from this only one other artifact is clear; a darker patch at 1:59 signals the end of the foreground strings.

The LSM image gives a far clearer image of the track with the superior relative intensities throughout, noticable when compared to the waveform image. Despite being monochrome, not only are all aforementioned sections of the track visible, but so, too, is the repetition of the strings in the background of the classical regions (3:28 to 3:58, 6:20 to end) as four lighter bands.

The colour of the BLSM reinforces the above structure to the eye. the strings at the

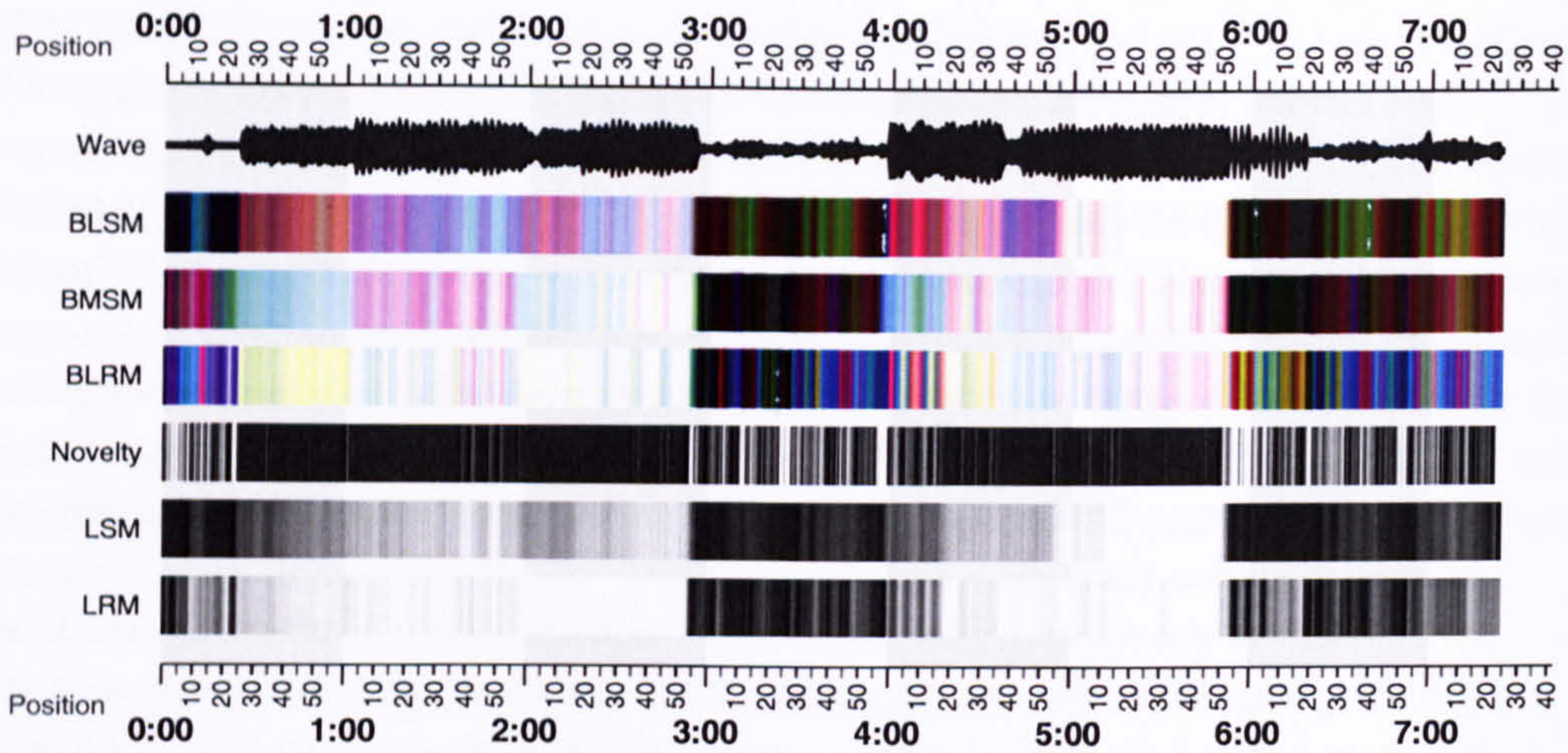


Figure 3.36: Several visualisations of the dance/classical track *Clubbed to Death* by *Rob D.*

beginning are noticeably blue, though throughout the track a blue tint and strings do not seem to correspond robustly. the bass (as always with BLSM) is marked as a deep and dark red, most evident in between the green of the piano figures at 5:53 to 6:20. Overall, the subtle changes in colours in the percussion portions largely represent voices being added and removed well. As the hue changes from a red tint to a cyan tint, more mid-range voices such as the samples, strings and piano take precedent in the sound over the bass. The marcato of the piano during the classical portion, between 2:53 and 4:00, is distinguishable as the flecks of green getting brighter. Furthermore, the use of sweep filters to change the tone of the sound is reasonably visible before the strings get added, between 4:00 and 4:19, as a transition between red and gray. This effect becomes subtler, representing the sound to some degree, in the 20 seconds following when the strings are added.

Using MFCCs for the preprocessing gives a somewhat less informative image; the classical sections are noisy and overall texture is difficult to make out. Texture and hue changes in the percussion portions do give rise to musical changes; the change from cyan into pink at 1:02 denotes the introduction of a foreground sample; the important introduction of strings at 1:20 goes without visual artifact. Movement of strings from foreground to background and removal of the sampled voice is denoted by a change to white. In all, the visual changes are subtle, non-exhaustive, and not immediately representative.

The self-similarity based measures perform, on the whole, poorly. Novelty notes many useful changes in the timbre space of the track with a single, precisely located, bright white line. In particular, the transition points at 0:24, 2:55, 3:28, 4:00 and 5:53. However, points within sections are either entirely without feature (generally in the percussive portions), or

laden with details that are of no immediate musical importance (in the classical portions). The LRM image suffers in much the same way, though the transition between bass/piano to piano/strings at 6:20 is perhaps a little clearer. Use of colour in BLRM has less clarity in the classical portions with a generally noisy use of all colours. On closer inspection, the repetition of the theme at 3:27 again at 6:20 is denoted through the use of the same (complicated) pattern of colours; as a nice touch the same theme played once more at 6:58 in marcato is denoted as the same pattern once more but in a lighter shade.

3.4.9 Summary

Each of the proposed methods of visualising music audio signals perform sensibly with certain types of music, and typically outperform the trivial wave representation, despite being comparable in terms of screen space required. Notable points include:

- Generally, bandwise loudness magnitude performs either the best or not much worse than the best.
- For rap music and in some manners jazz too, the bandwise rhythm-spectrum method typically performs best.
- Classical music tends to be for more difficult to visualise than other genres for these methods.
- With few exceptions, the addition of colour through the bandwise step improves the visualisation.

3.5 Conclusions

I have provided a thorough review of literature concerning the visualisation of musical audio signals. I have proposed two novel and efficient methods for visualising musical audio. Each of the proposed methods are single-stage audio-analysis methods not employing complex machine learning algorithms or multi-stage pipelines. They produce a simple visualisation of form *S-3* making them good for usage on a popular navigation system.

I have discussed and provided examples for each non-obvious step in creation of the visualisation methods. I then discussed the aspects of music that each visualisation method depicted over a number of tracks from a board range of genres. I finished by summarising the findings from that informal inspection.

In the next chapter I will consider one particular path for improving the simplicity and legibility of the image; the use of more advanced dimensionality reduction techniques in order to project an audio block into a colour. In particular, principal component analysis

(PCA) and the self-organising map (SOM) will be utilised as methods to project an audio block onto a chromaticity plane.

Chapter 4

Chromaticity Plane Trajectories

"The painting has a life of its own. I try to let it come through."

—Jackson Pollock (1912–1956)

4.1 Introduction

In this chapter I propose and discuss a novel music audio visualisation technique, designed to utilise the topology-preserving dimensionality-reduction capabilities of the Self-Organising Map.

4.1.1 Chapter Summary

I begin by reviewing work related to that presented here; dimensionality reduction methods and examples of their usage in music and musical audio signal processing. I then detail the proposed visualisation technique able to utilise linear and non-linear dimensionality reduction methods; first giving an overview and then detailing and defining each stage. This is followed by demonstrations of the technique in action. Finally I review each of the proposed variations on the technique, using the same pieces of musical audio in the previous chapter.

4.1.2 Contributions

- A formalisation of a dimensionality-reduction-based core technique, drawing upon the earlier visual creation technique, together with two novel concrete visualisation methods.

- Discussion of these techniques, their advantages and problems, and their relations to existing techniques.

4.2 Related Work

The self-organising map together with principal component analysis are two common methods for visualising high-dimensional data. This is essentially the problem which the chroma-projection function must solve.

4.2.1 The Self-Organising Topographic Map

Introduced by Kohonen (1982), and later described by Kohonen (1990), there is a large amount of literature describing the Self-Organising Map (SOM) and its various applications. Oja et al. (2003) compiled a bibliography of all its various uses, fittingly illustrated by a SOM visualisation of the types of subjects covered. Notably, music is one of the few areas without a large amount of attention. It is regarded as a general-purpose unsupervised neural network, and as such is covered in overview and reference texts including those by Skapura (1996), Fausett (1994), and Callan (1999). 'Neural networks' are a set of biologically-inspired methods and techniques for recognising patterns, typically through a connectionist paradigm.

There are several variations on the SOM algorithm. Kohonen (2007a) gives a good starting point. The underlying concept in all of them is to have a set of model vectors, each of which represents a point in input data space. These model vectors are then organised in some form of topology, typically a quadrangle, though various alternatives to planar and indeed Euclidean geometry have their uses (Ritter, 1999 goes into more detail about this). The topology is important to the algorithm in one way only; it helps define the neighbourhood function. This determines how the reinforcement of one node will affect each of the other nodes. Of course this topology plays an important part when the SOM is inspected, but that is not part of the algorithm per se. Once properly trained, nodes that lie close to each other on the grid have relatively similar model vectors, and thus represent similar data.

Perhaps the most archetypal use for the SOM is the classification of Finnish phonemes. A map is trained to decode the low-level audio features of various Finnish phonemes into points on its planar surface. Figure 4.1 illustrates the trained map together with the phoneme represented. Spoken audio may then be visualised as trajectories in the SOM space.

Definition

Formally, we may define the SOM as a topologically defined set of model vectors $\mathbf{m}$, as Kohonen (1990) does. Each model vector defines a point in high-dimensional (input) space and has an implicit position in the lower-dimension (output) space. It thus acts as a point-to-point mapping between spaces. An input point may therefore be mapped to a

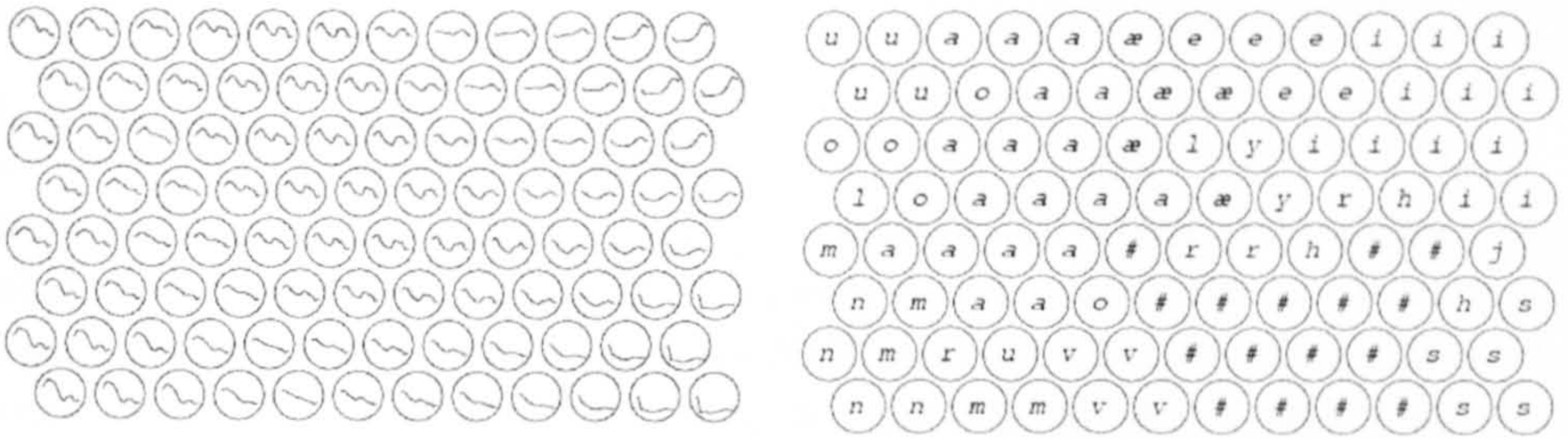


Figure 4.1: An illustration of a fully trained SOM on Finnish phonemes. Left are the low-level spectral data, right are the phonemes represented by the data. *Reproduced from article on Scholarpedia by Kohonen (2007a), according to the licence. Copyright remains with the original author.*

corresponding output point by determining the model vector that most closely matches it. Formally, the point in output space of input vector $\mathbf{x}$ is equivalent to the implicit position of node i where:

$$\min_i \{\|\mathbf{x} - \mathbf{m}_i\|\} \tag{4.1}$$

The SOM may be trained by an iterative learning process, where each step of time t may be denoted by:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{ci}(t)[\mathbf{x}(t) - \mathbf{m}_i] \tag{4.2}$$

h_{ci} is a neighbourhood function to determine the size of structure enforced; this decreases to zero throughout training and is typically implemented as a Gaussian-centered distribution:

$$h_{ci} = \alpha(t) e^{-\|\mathbf{r}_i - \mathbf{r}_c\|^2/\sigma(t)^2} \tag{4.3}$$

where α and σ are decreasing functions over time.

The initialisation of the model vectors is of lesser importance to the definition of the SOM, and is discussed fully in the description of our particular algorithm (section 4.3.4).

4.2.2 The SOM with Musical Audio

The SOM has had several applications in the field of musical audio information retrieval. The three typical uses are as a means of a similarity 'proxy' of musical audio, as a method in itself for visualisation of data concerning music and as a classifier.

For Visualisation

The SOM has been instrumental in a wide range of visualisation tools, and is by no means limited to music. However in the various fields concerning music it has seen use as a preprocessor to a self-similarity matrix of tonal content by Toiviainen (2005); a method of visualizing melodic data from a symbolic format for comparative analysis of folk music by Toiviainen and Eerola (2001). It is also heavily used as a visualisation aid in music collections; Lubbers (2005) use it for combining visualisation with "auralization" for a multi-media collection browser. Morchen et al. (2005) uses it to visualise collections according to perceptual distance, in a similar manner to the original *Islands of Music* by Pampalk (2001); Rauber et al. (2002); Rauber and Frühwirth (2001); Pampalk et al. (2002):

Islands of Music is a project spanning several research avenues. It is centred around the utilisation of a SOM to navigate around a collection. It works by appealing to a visualisation representing clusters of music on a plane as islands and mountains. Several concepts are trialed, including feature vectors for SOMs (which we use in the present work), and utilising SOMs as a dimensionality reduction method for a large audio feature vector to generate 'meta-features' (an idea adapted for use in the present chapter).

For a Similarity Proxy

The SOM has also seen considerable use in the field of music analysis algorithms as what we will term a 'similarity proxy'. We use this term to mean 'a spacial mapping using the topology preservation characteristics in order to give a more efficient, and perhaps effective, measure of similarity between two feature vectors'. Vembu and Baumann (2004) use tags generated from online descriptions, (in this case from the retailer *amazon.com*), to train the SOM. Artists can then be recommended according to where they fall on the resultant map, rather than comparing the given tags directly. Aside from being more efficient, this can make queries more effective, since 'neighbourhoods' of artists can form where several tags are shared between artists. This allows artists to be grouped together that do not necessarily share the same tags, but have close associates that do.

Dittenbach et al. (2003) presents *PlaySOM* which, stemming in part from the Islands of Music project, allows an interface for hierarchical navigation ('drilling down' in data-mining slang), and area selection for playlist generation. This again relies on the topology preserving characteristics of the feature space, and on the feature space being reasonably representative of the perceived qualities of the music. Toiviainen and Eerola (2002) use the SOM to determine melodic similarity; trained on many high-level vectors modelling melodies, the SOM functions as an effective alternative to string-matching.

Finally Wood and O'Keefe (2003) describe some basic work attempting to utilise the SOM as a measure of similarity for musical pieces. They attempt to predict whether two

pieces share the same album by their distribution of transformed points on the map.

For Classification of Musical Audio

SOMs have had uses in musical audio feature classification. With genre classification being a popular challenge in the field of music information retrieval, SOMs have been used as a pattern classification tool. Knees et al. (2006) utilise the SOM to project music onto a 2-dimension 'map' of genres. Reminiscent of the Islands of Music work, the various genres of music may be visualised as portions of the map segregated by a learned viewer, or as in this instance, a web-based artist-tag retrieval system. Some 66,000 different features were extracted for this task, being reduced to the 20 that most accurately described genre-splits with Pareto-Density Estimation. Ponce de León and Iñesta (2002, 2003) also attempt to identify musical style but in a more limited sense; to discriminate between jazz and classical. The SOM is utilised as a feature space partitioning tool, the features here being derived from symbolic data.

Rather than classifying music from symbolic data, Cemgil and Gürgen (1997) attempt to use the SOM to classify instruments from audio data. They discuss several types of neural network with respect to the problem of instrument sound classification. They find that SOM did not perform as well as the other networks. However they found that the SOM proved more useful for being able to see what the network was doing and how it was ordering the data internally.

4.2.3 Principle Components Analysis

Principle Components Analysis (PCA) is a technique used in statistics typically for simplifying a dataset, reducing it to a lower dimensionality (perhaps for visualisation). This may be done by ignoring high-order components in favour of lower-order, thus retaining as much of the variance of the original dataset as possible, with as few dimensions as possible. Indeed, PCA is the optimal linear transformation to determine the subspace of the largest variance.

Introduced by Pearson (1901), and described in depth by Jolliffe (2002), the principle component of a set of n-dimensional vectors is the n-dimensional vector which describes the projection giving the greatest variance over the dataset. Formally, we can describe the principle component on a zero-mean dataset $\mathbf{x}$ as $\mathbf{w}_1$ where:

$$\mathbf{w}_1 = \arg\max_{\|\mathbf{w}\|=1} \text{var}\{\mathbf{w}^T\mathbf{x}\} \tag{4.4}$$

If the dataset is projected by this vector (i.e. the variance is removed), the principle component of the new dataset is the second principal component of the original dataset. Each principle component is naturally orthogonal to all others. Thus the full set of n

principal components can said to be an n-dimensional rotation, which, when applied to the dataset, orders the dimensions in terms of maximum covariance. We may call this the result of Principal Components Analysis. Formally, we can describe the k-th component as $\mathbf{w}_k$ where:

$$\mathbf{w}_k = \arg\max_{\|\mathbf{w}\|=1} \text{var}\{\mathbf{w}^T \hat{\mathbf{x}}_{k-1}\} \tag{4.5}$$

where

$$\hat{\mathbf{x}}_{k-1} = \mathbf{x} - \sum_{i=1}^{k-1} \mathbf{w}_i \mathbf{w}_i^T \mathbf{x} \tag{4.6}$$

PCA is typically implemented by:

1. Subtracting the mean from the dataset.
2. Determining the covariance matrix of the dataset.
3. Determining the eigenvectors and eigenvalues of this matrix.
4. Sorting the eigenvectors in terms of corresponding eigenvalues.

The eigenvectors represent the principle components of the dataset.

In Musical Audio IR

Being a conceptually simple, effective, well-understood and accessible technique, PCA is used in a wide variety of situations throughout numerical processing, to reduce the amount of data by discarding that which is statistically most redundant. An example of its use in the processing of musical audio, which serves us well since it was directly adapted for use in the present work, would be that of Pampalk et al. (2004, 2003) who utilised PCA to reduce two high-dimensionality feature vectors (around 1000-dimension) to just tens of dimensions, with very little loss of quality in the resulting visualisation.

4.3 Proposed Visualisation Methods

In this section, I discuss a visualisation method which we term *chromaticity-plane trajectories*. This method builds on the core visual construction method in the last chapter. In particular, this method reduces a block of the musical audio to a two-dimensional 'position' representative of its content. A particular two-dimensional colour-space is used to decode this position into a colour. This colour is considered representative of the musical audio. Therefore, as the positions determined by the audio blocks change, a trajectory forms though the colour space.

This technique can be considered in three distinct parts; (1) perceptual-audio feature extraction, (2) vector projection and (3) perceptual colour encoding. Since no contextual information of the colour is available to part (2), the performance of this technique (is partly) contingent on users not caring about an absolute colour-feature mapping, but only about the inter-relationship of colours. This is unlike many of the visualisations in the previous chapter, which mapped fairly absolute elements of the audio (e.g. loudness and spectral-ratio) to colours.

The two proposed methods share in common most of the technique as a whole; the only difference being the 'projection' method. Figure 4.2 shows a diagram of the three transforms that the data goes through. A large feature vector of the song is created. It is then projected down with a dimensionality reduction technique. This portion of the method is based largely on the *Islands of Music* work. After a low-dimensionality position has been attained, it is transformed by way of a colour-space into a single colour. This technique therefore fits well into the *audio-colour projection* framework devised in section 3.3.1.

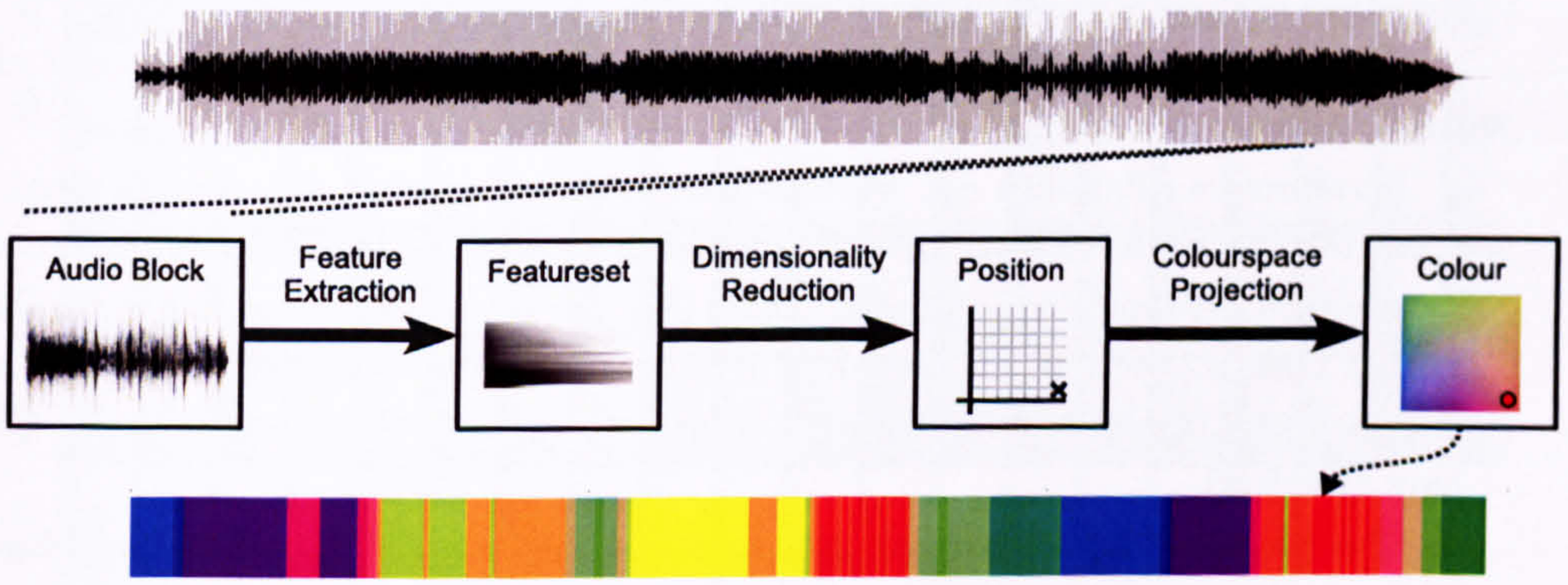


Figure 4.2: The stages and intermediate data types required for the general CPT colour-projection method.

In particular, perceptual differences in audio should be well represented in the first refinement, and perceptual differences in colour should be approximately linear from the latter stage. Thus by assuming a linear mapping in the middle stage, the results is a mapping between audio and colour that to some degree preserves perceptually-proportionality. These attributes are discussed in more detail in the relevant sections.

We can not directly map a high-dimensionality perceptual audio feature vector to a planar surface trivially, so we propose two methods. Firstly, we preserve linearity at the cost of discarding some of the information, by discarding low-variance dimensions in a variance-ordering rotation of the input space with PCA. Secondly, we propose sacrificing proper linearity with a non-linear but topologically-correct mapping, in order to represent

all the information. The latter method also has the advantage of providing a natural method for content-based (i.e. not necessarily linear) quantisation of the output.

Initially I describe the feature vector preprocessing method. Secondly I discuss colour spaces. I finish with two forms of dimensionality reduction; a linear method using PCA and a non-linear method using the SOM.

4.3.1 Feature Vector Processing

Following the work of Pampalk et al. (2004), I used the *spectrum histogram* featureset (*SP*) to model the perceptually motivated audio feature space. SP, although it does not model aspects of timbre such as instrument onsets, has been found by Pampalk et al. (2003) to significantly outperform other feature extraction techniques in grouping tasks. When compared to several other more complicated and computationally intense methods, it outperformed each of them at distancing musical audio of differing style, artist, genre and album. Though this is no guarantee that it will deliver a useful space for disambiguating musical features in an individual piece, it is encouraging. Using Bark 'critical band' summation as well as specific loudness scaling, SP is a psychoacoustically robust preprocessing method.

Furthermore SP, unlike e.g. rhythm-based techniques, takes only a small amount of signal to collate a feature vector. The amount of signal to collate a robust rhythm-based feature vector is typically significantly more (e.g. the periodicity histogram rhythm-based feature takes 12-seconds to collate). Furthermore, it is not clear how a rhythm-based feature vector would describe the sorts of aspects of music that users typically wish to have visualised.

The two other techniques evaluated as performing worse at disambiguating objective music attributes by Pampalk et al. (2003). These, named after their proposers, Aucouturier and Pachet (AP) and Logan and Saloman (LS) notably used mel-frequency cepstrum-based signal preprocessing (i.e. MFCC features). Although this has never been clearly shown as being a worse method of preprocessing music signals than the Bark/Sone preprocessing, this work would support that theory. For all these reasons, I adopted the spectral histogram as the algorithm, with no major changes from that proposed originally.

The Algorithm

Figure 4.4 shows the process as a dataflow pipeline. More formally, the basic form of the feature extraction method is:

1. Take a window of STFT spectra; I used three seconds worth of spectra which appeared to deliver good results.

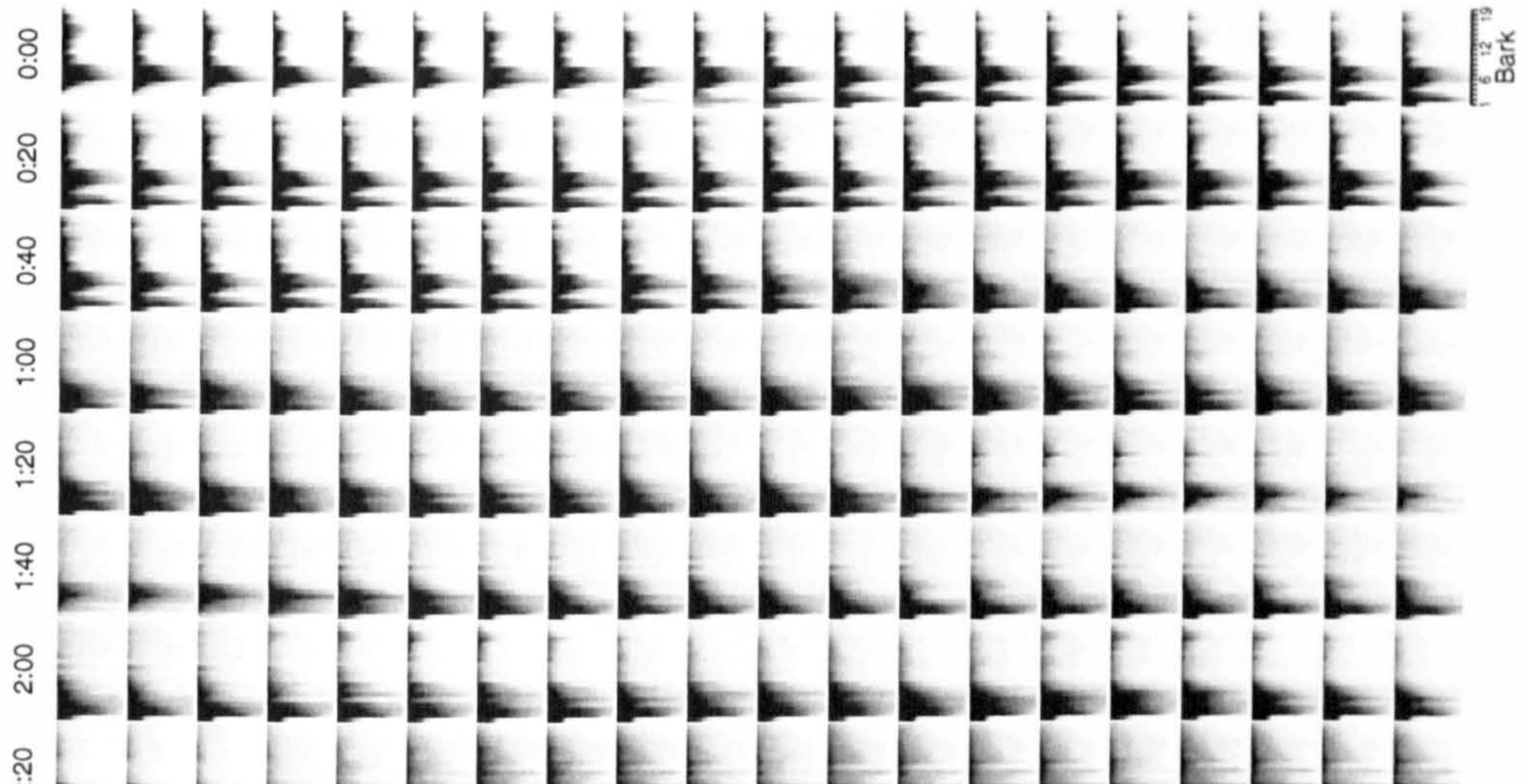


Figure 4.3: The full set of histograms for the track *Lebanese Blonde* by *Thievery Corporation.*

2. Modify each spectrum to a critical-banded Sone-scaled psychoacoustic spectrum, as described in section 3.3.3:

   (a) Rescale axes logarithmically to dB-SPL.

   (b) Sum into first 20 critical bands according to the Bark scale.

   (c) Scale each frequency band to phon units by interpolating equal loudness curves (phon).

   (d) Scale each band into specific loudness sone units (sone).

3. Rescale so maximum loudness *over the piece* equals 1 sone.

4. Sum a spectrum histogram matrix $\mathbf{H}$ with each spectrum $\mathbf{s}$:

$$\mathbf{H}_{f,l} \equiv \sum_{\mathbf{s}} \begin{cases} 1, & \mathbf{s}_f > \frac{l}{50} \\ 0, & \text{otherwise} \end{cases} \tag{4.7}$$

5. Scale the histogram so that the maximum value contained is 1.

$$\mathbf{H}' \equiv \mathbf{H} \div \max_{\forall f,l}(\mathbf{H}_{f,l}) \tag{4.8}$$

This gives a 1000-dimension perceptually-based audio 'timbre' feature-vector, which I use as a model for human perception of the musical audio. Figure 4.3 illustrates the track *Lebanese Blonde* as 280 distinct histograms.

The eigenvalues may be inspected to reveal the internal dimensionality in figure 4.5. In almost all cases, 90% of the total variation of the datasets is captured in just the first two principle components. This serves as an initial validation of the argument to project the data onto a 2-dimensional plane. In the next section, I will consider the perceptually-opposite notion of turning such a 2-dimensional coordinate into a colour. After this, I will discuss two methods of reducing the high-dimension feature space in order to 'project' the audio features into a colour.

4.3.2 Colour Space

In this section, I will discuss colour spaces, in particular the mapping from a position to a colour.

In most colour spaces, chrominance (*chroma*[1]) subspace, (which I will define here as being the aspect of colour which does not change with the amount of light received by the viewer, i.e. non-luminosity dependent), are naturally representable with 2 dimensions. Brightness is ignored for two main reasons; firstly over concern that it would be of greater apparency over chroma. As such, it may mislead the viewer to identify it with some particular aspect of the sound (e.g. spectral brightness, loudness). Secondly, at the extremities, luminosity naturally folds all colours down to one (i.e. white or black), making the other dimensions indistinguishable.

[1]chroma actually has at least two meanings; together with being a shortened form of the word *chrominance*, it has a more traditional meaning, which I will not use in the present work, that makes it synonymous with saturation or purity of a tone.

Figure 4.4: An activity flow chart of the CPT preprocessing technique.

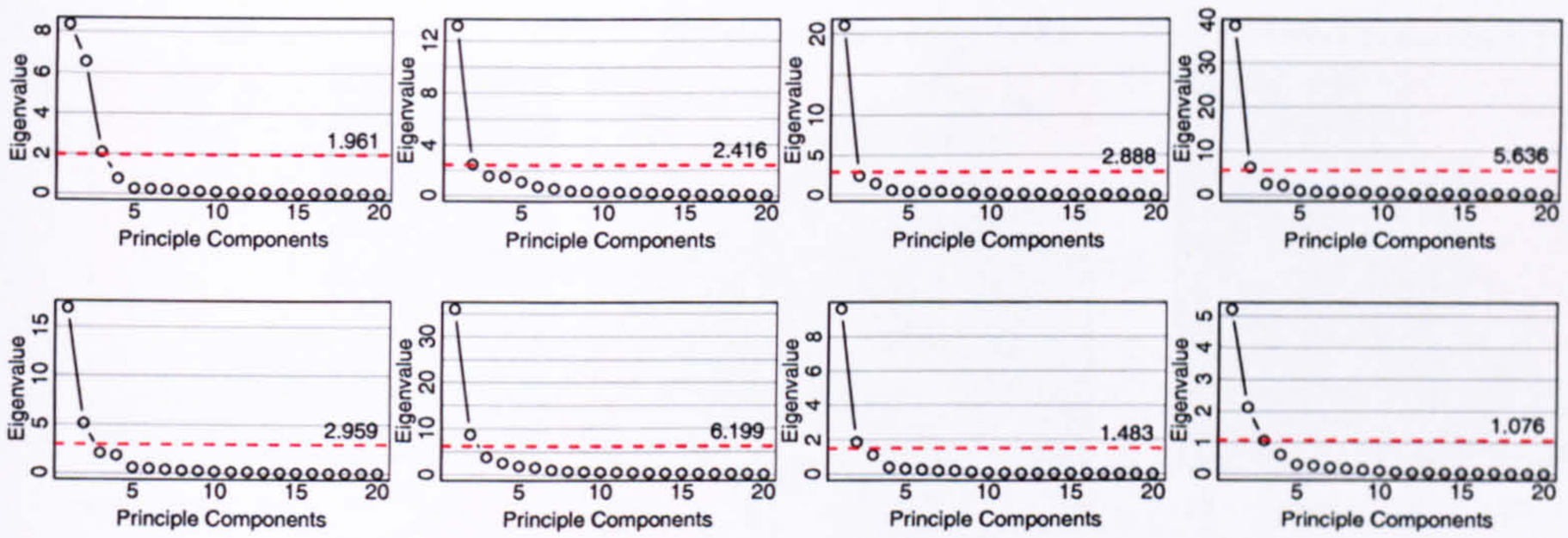


Figure 4.5: Each of 8 tracks' eigenvalues (Lebanese Blonde, classical, rock, fusion, downtempo, trip-hop, jazz, rap). The dashed red line denotes the 10% cutoff for the sum of the eigenvalues.

An advantageous quality of a colour space would be to use as much colour (i.e. as wide a gamut) as possible, in order to better aid an onlooker in distinguishing features. A further related quality would be to capitalise, where possible, on the primary colours of the colour model RGB, since they will naturally be the best defined on the display system used.[2]

Three planar chroma subspaces are depicted in figure 4.7.

Perceptual Similarity of Colour

In addition to the consideration of gamut, there is a second question of distribution by perceptual similarity. This was addressed with an experiment by MacAdam (1942) on the subjective similarity of colours.

In a manner analogous to the Bark critical banding of frequencies (see section 3.2.1), a large number of human experiments were carried out to form a general view of how people perceive colour and changes therein. MacAdam experimented by asking people to match a pure spectral colour, (i.e. one that can be denoted perfectly by only a single frequency of light), to a composite colour of which they had control over two chroma parameters; x and y (the luminosity was fixed). He found that the values chosen for x and y varied between subjects, but that over the course of the experiment formed an ellipsoid distribution on the x-y colour plane. These ellipses (one for each colour tested) became known as the MacAdam Ellipses, each denoting a 'region of indifference' in human perception of colour.

It is useful to define a (theoretical) space where differences in the position are proportional to differences in the perceived colour. Formally, I define a *perceptually euclidean colour space* as function S which maps a colour to a point. The apparent change in colour

[2]Unfortunately for this document, this is at ends with the ink-based printing system.

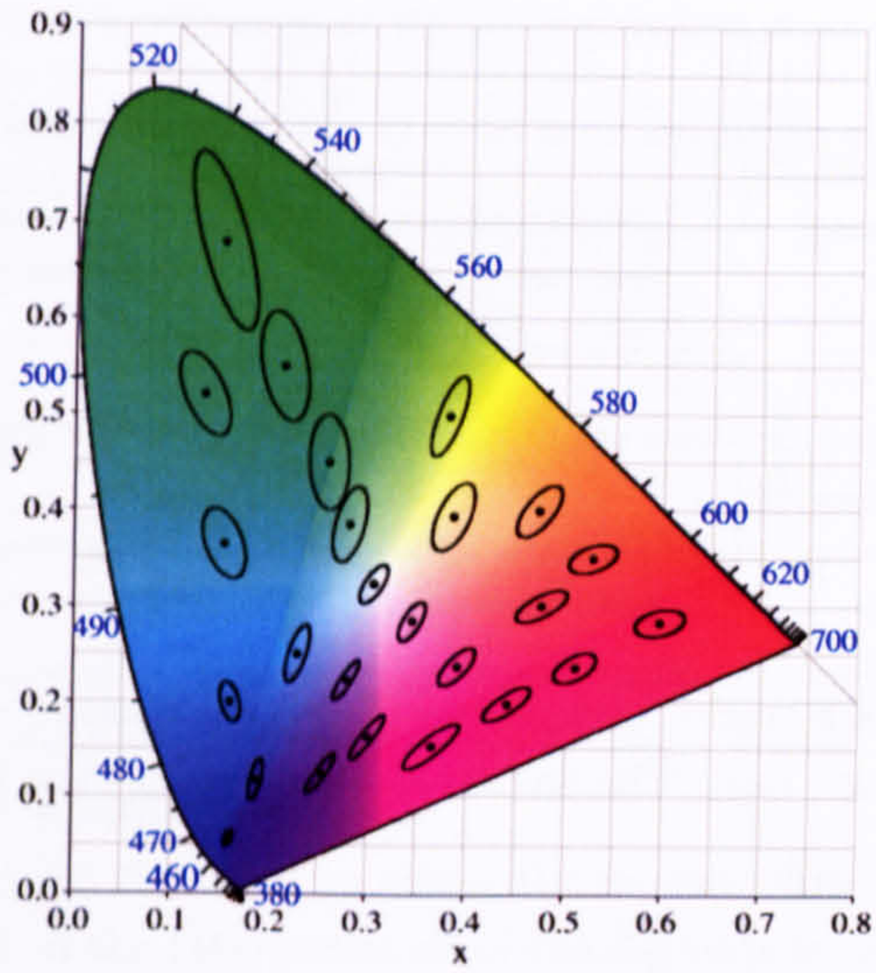


Figure 4.6: The CIE 1931 xy chromaticity diagram, with the human-visible gamut enclosed in the pure spectral colours (the 'tongue'-shaped curve) and the line of extraspectral purples. The ellipses correspond to the MacAdam Ellipses but are reproduced at ten-times the actual size.

is proportional to the euclidean distance between endpoints of the change:

$$P(c_1, c_2) \propto \|S(c_1) - S(c_2)\| \tag{4.9}$$

where P is a function defining perceived linear difference between two colours.

Simple Spaces

One simple colour space is based around the hue and saturation portions of the HSV colour model. Since the output is the same for all hues at zero saturation, I use a polar coordinate system for the space rather than a Cartesian system. We define the space thus:

$$C(x, y) \equiv C_{HSV}(\frac{(tan^{-1}(\frac{y-0.5}{x-0.5}) + \pi + 0.5}{2\pi}, \sqrt{(x-0.5)^2 + (y-0.5)^2}, 1) \tag{4.10}$$

On inspection, it is clear to see three aspects of non-linearity in the perceptual colour distribution; firstly, the center provides far too small a jump in quality when compared to the edges. Secondly, the changes in hue are far greater around the secondary colours (i.e. yellow, magenta and cyan) than around the primaries (red, green and blue). Finally, as is often the case, the space given to green hues has far fewer changes in colour to that of other hues.

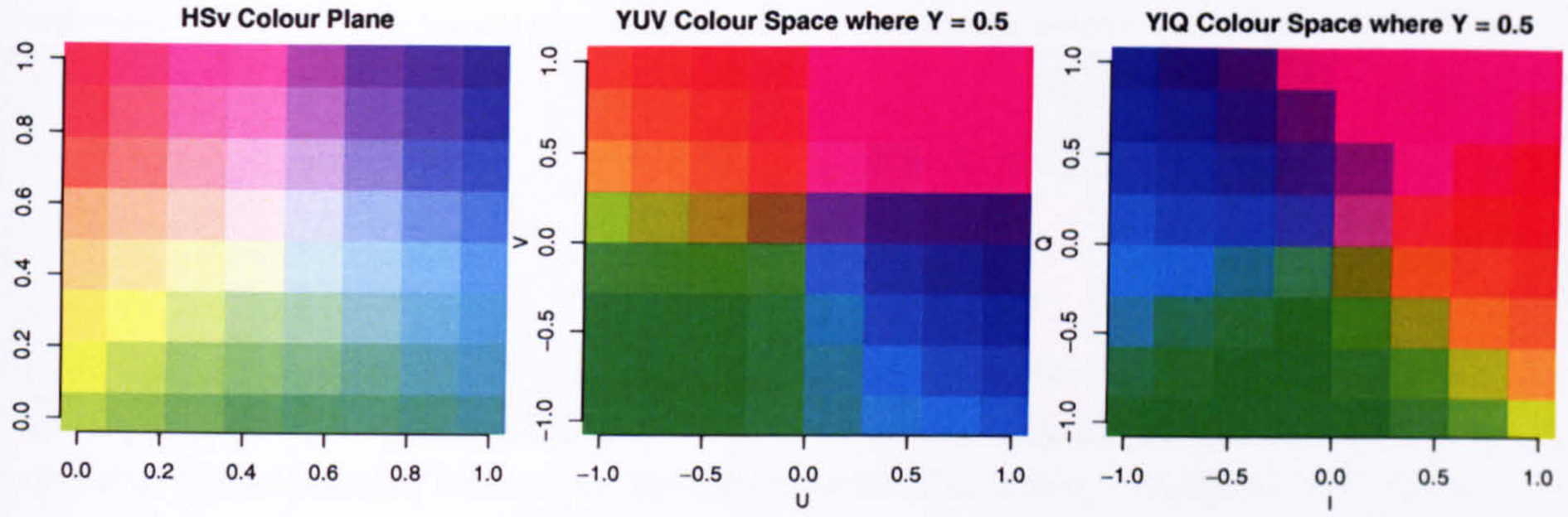


Figure 4.7: Three simple chrominance colour planes. Note: The colour reproduction is almost certainly inaccurate and should be taken as a demonstration and not a reference.

YUV and YIQ (top middle and right of figure 4.7) are both used for the transmission of analogue television signals; they are designed to map colours in such a way as to minimise obvious degradation when noise is introduced. They represent exactly the same colour space except with a rotation of 30°. Though interesting since they encode the chroma-information explicitly as 2-dimensions, neither of these proved a particularly good mapping space due to excessive concentration of the same colour over a wide area.

CIE Spaces

As described by Kasson and Plouffe (1992), CIE (*Commission Internationale d'Eclairage*) XYZ colour space (CIEXYZ, bottom left of figure 4.8 with a fixed luminosity Y) was one of the first mathematically defined colour spaces. It is particularly special since it is based on perceptual experiments done on the human eye by Wright (1929); figure 4.6 depicts the famous chromaticity diagram derived from his experiments. As such, CIEXYZ is designed to encompass all distinguishable chromaticities in the human gamut, each as a single point[3]. As such, it forms the basis for many colour spaces which encompass all visible colours.

CIEXYZ can be translated into the sRGB colour space by:

$$\begin{bmatrix} R_{linear} \\ G_{linear} \\ B_{linear} \end{bmatrix} \equiv \begin{bmatrix} 3.2410 & -1.5374 & -0.4986 \\ -0.9692 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0570 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{4.11}$$

[3]doing so without guaranteeing each colour mapped to only one point is trivial; one simply uses a histogram of the visible portion of the EM spectrum

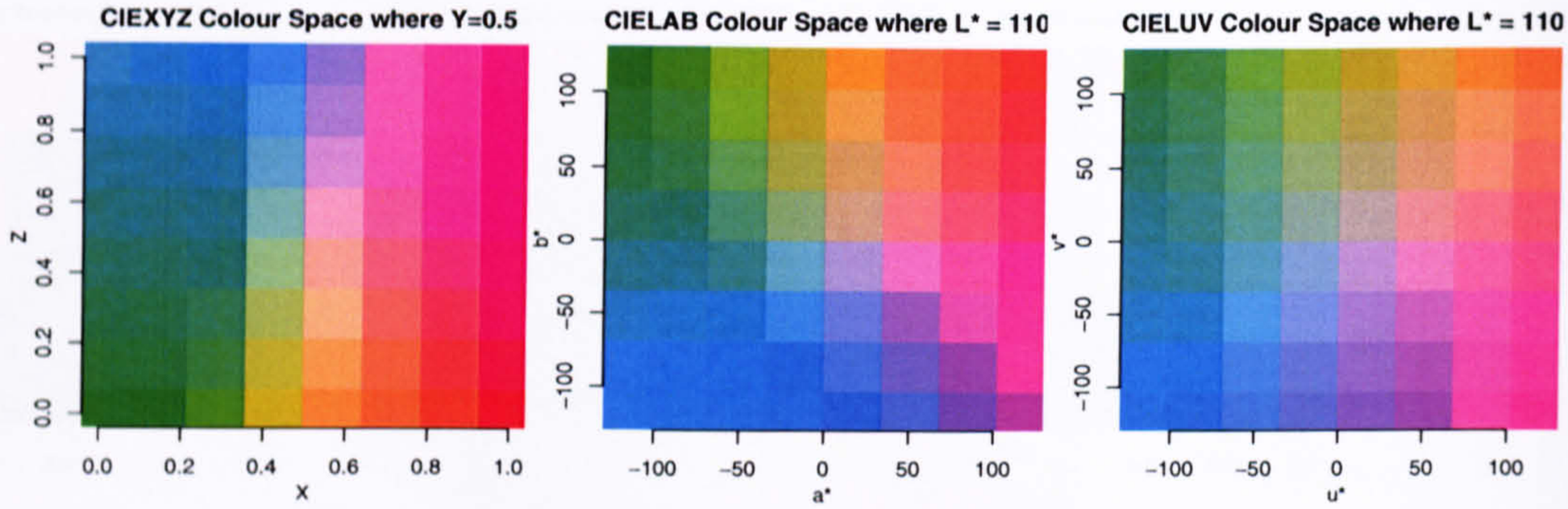


Figure 4.8: Three CIE-specified chrominance colour planes. Note: The colour reproduction is almost certainly inaccurate and should be taken as a demonstration and not a reference.

$$C_{sRGB} \equiv \begin{cases} 12.92C_{linear}, & C_{linear} <= 0.0031308 \\ (1+a)C_{linear}^{1/2.4} - a, & C_{linear} > 0.0031308 \end{cases} \tag{4.12}$$

where $a = 0.055$ and C_{linear} takes the values of R_{linear}, G_{linear}, B_{linear} in turn to make the triplet C_{sRGB}.

For the RGB space, I assume sRGB, a widely adopted standard, and common on mainstream computer systems. sRGB is a practical colour space; most light imaging systems form colours from the three primaries red, green and blue, and as such it is advantageous for software to store colours in such terms. These however do not accurately model human vision, which for a perceptual system is desirable.

The CIE 1976 L*a*b* (CIELAB) colour space, depicted middle of figure 4.8, is a perceptually motivated colour space. Based on CIEXYZ, it is well-defined and encompasses all colours in the human gamut. Its three parameters, like most other colour spaces, correspond to a single luminosity value ('L*') and two chroma values ('a*', 'b*'). It is considered the most complete colour model used conventionally to describe all the colours visible to the human eye, and furthermore was designed largely as a perceptually linear colour space. It may be converted to CIEXYZ space by:

$$f_y \equiv (L^* + 16)/116 \tag{4.13}$$

$$f_x \equiv f_y + a^*/500 \tag{4.14}$$

$$f_z \equiv f_y - b^*/200 \tag{4.15}$$

$$Y \equiv \begin{cases} Y_n f_y^3, & f_y > \delta \\ (f_y - 16/116)3\delta^2 Y_n & \text{otherwise} \end{cases} \tag{4.16}$$

$$X \equiv \begin{cases} X_n f_x^3, & f_x > \delta \\ (f_x - 16/116)3\delta^2 X_n & \text{otherwise} \end{cases} \tag{4.17}$$

$$Z \equiv \begin{cases} Z_n f_z^3, & f_z > \delta \\ (f_z - 16/116)3\delta^2 Z_n & \text{otherwise} \end{cases} \tag{4.18}$$

The systematic perceptual linearity of CIELAB is determined from the MacAdam ellipses (see section 3.2.9). The CIELAB colour space uses these to warp the CIEXYZ space in order to better preserve human perceptual linearity. The space is designed so that the euclidean distances between points in it are proportional to those given in the Munsell book of colour, a seminal text on colour notation by Munsell (1912). CIE 1976 L*u*v* (CIELUV) is a similarly perceptually motivated system, but whose colour differences present different values, smaller than those in the Munsell book of colour. Of the two, I found visually better results from CIELAB.

CIELAB has, in some manners, been superseded by the CIECAM97c, and later CIECAM02. These spaces however have several drawbacks for this situation. Firstly, unlike the a*b* and u*v* subspaces of CIELAB/CIELUV, they do not provide a clear planar subspace. Furthermore, they are high-dimensional colour spaces; CIECAM02 uses seven parameters (lightness, brightness, colourfulness, saturation, hue quadrature, chroma and hue), not including whitepoint adjustment and surround correction.

Proposed Colour Plane

On the video equipment available, I found the CIE space simply far too non-linear. Because of this, a colour plane was devised manually which can be seen as a modified red-green colour plane; a blue component is added which takes the value of the maximum of the red and green value subtracted from unity. Formally, I define C_{RGb}, a function which maps positions on an x/y plane to colours:

$$C_{RGb}(x, y) \equiv C_{sRGB}(x^{0.3+\gamma}, y^{0.7+\gamma}, (1 - max(x, y))^{0.4+\gamma}) \tag{4.19}$$

where

$$max(a, b) \equiv \begin{cases} a, & a > b \\ b, & \text{otherwise} \end{cases} \tag{4.20}$$

I found γ had to be varied between 0 and 0.2 depending on the characteristics of the visual display unit.

From inspection of figure 4.9, one may see that the plane is reasonably uniform in terms of perceived changes of colours. Although not perfect (e.g. the outer perimeter generally

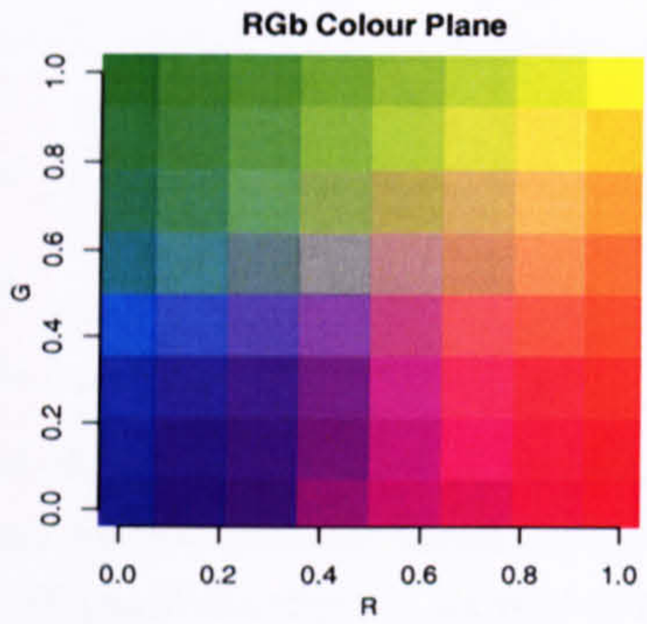


Figure 4.9: The proposed RGb colour plane. Note: The colour reproduction is almost certainly inaccurate and should be taken as a demonstration and not a reference.

has a higher degree of change than the internals), it is at least as perceptually uniform on typical commodity video equipment as any other spaces discussed. Furthermore, it makes good use of all primaries, as well as supporting a wide range of hues.

4.3.3 PCA Projection

In this section, I will describe and illustrate a basic linear approach for reducing the audio feature vectors to the 2-dimensional position vector for mapping to a colour. The process will be illustrated with examples of the final visualisation, together with demonstrations of how the final visualisation is determined.

Definition

The approach can be summarised by taking the two greatest principle components of the dataset. All other information is discarded; thus although this is a linear approach to projection, a small amount of information is ignored entirely. Initially, the two eigenvectors are found which have the highest eigenvalues of all the data in the dataset x. To transform a given feature vector x_i into a position vector $\mathbf{p}_i$ it should be projected into the subspace defined by those two eigenvectors. After projection, the position vector should be normalised by mean-subtraction and σ-division, calculated from the distribution of all position vectors for the track. Formally, a position projection function p' is defined such that $p'(\mathbf{x}_i) \equiv p_i$ where:

$$\mathbf{p}_i \equiv \frac{(\mathbf{x}_i - \overline{\mathbf{x}})\mathbf{V} - \overline{p}}{\sigma(\mathbf{p})} \tag{4.21}$$

where $\overline{\mathbf{p}}$ and $\sigma(\mathbf{p})$ are respectively the mean and standard deviation of position vectors of the track, c is a clamping function and $\mathbf{V}$ is the 1000x2 matrix containing the two

eigenvectors ($\mathbf{e}$) of highest corresponding eigenvalues (e):

$$\mathbf{V} \equiv \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix}, \quad \forall n, 0 < n < 1000 : e_n > e_{n+1} \tag{4.22}$$

As has been shown in section 4.2.3, the two principle components account for around 90% of the total variance of the dataset, thus less than 10% of the statistically significant information content is discarded. It should be noted, however, that the clamping does discard more information.

The visualisation is finalised by correcting the position and mapping it into the chromaticity plane to get a colour:

$$\mathcal{P}_{PCA}(\mathbf{x}_i) \equiv C_{RGb}(0.5 + 0.5 l_{-1,1}(\mathbf{p}'(\mathbf{x}_i))), \quad l_{a,b}(x) \equiv \begin{cases} a, & x < a \\ b, & x > b \\ x, & \text{otherwise} \end{cases} \tag{4.23}$$

Illustration

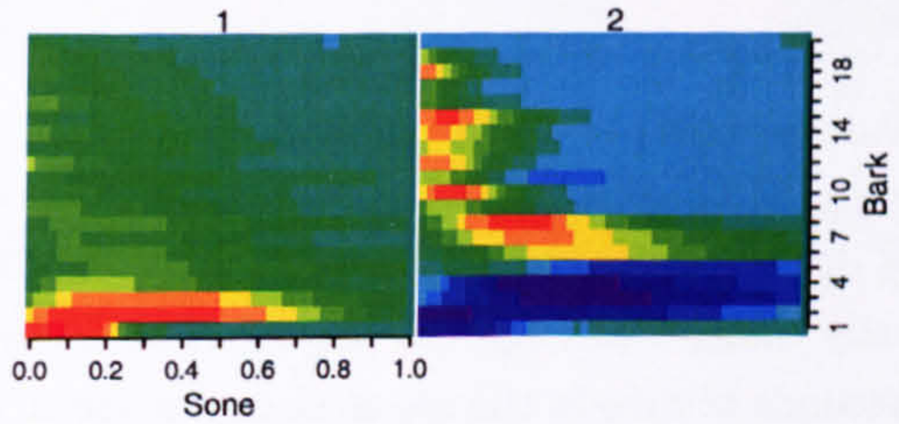
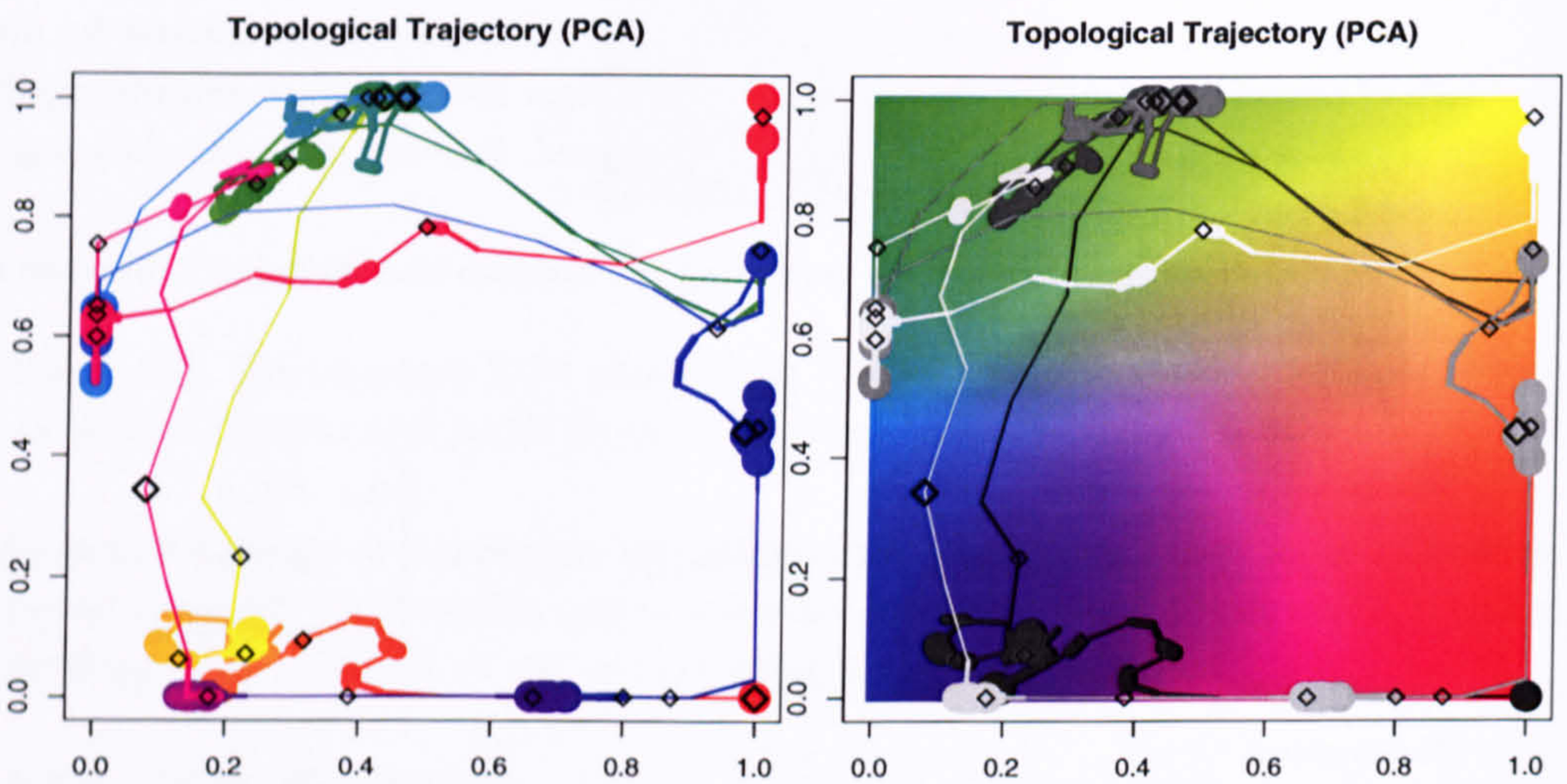


Figure 4.10: The first two principle components of the spectral histograms for *Lebanese Blonde* by *Thievery Corporation*.

Figure 4.10 illustrates the two most significant principle components of *Lebanese Blonde*; it is the features represented by these two components and only these which are accounted for by the visualisation. Since principle components are vectors in the data space they can be negative; cyan is to be considered zero and thus the component being agnostic to that particular dimension. The first component can be seen to cover what is essentially the DC component, in that it has only a positive effect on the loudness. Notably, the low and mid-levels of loudness of the bass are weighted more than the rest of the track.

The second component represents the weighting of the higher-frequencies over the bass; while the bass (especially the mid-levels of loudness) is excessively negative, much of

the the higher frequencies are positive, especially mid-levels of loudness around the mid-range frequencies, and quieter levels in the treble. This component would therefore likely represent the presence of speech in the music, or lack thereof.



(a) Red through yellow, green, blue and back to red.(b) Black to white, with the chromaticity plane in the background.

Figure 4.11: A demonstration of the trajectories followed by *Lebanese Blonde* according to the PCA mapping; the beginning of the track is at the bottom right, with the ending mapped to the top right. The line's width is increased for smaller distances moved, and the colour monotonically changes through the playing time of the track. Small diamonds are plotted every 15 seconds of time in the music along the locus; each minute is denoted by a larger diamond. For ease of viewing, the trajectories are smoothed by a four-sample-wide moving average, and a small amount of Gaussian noise is added.

Figure 4.11(a) shows the trajectory mapped for the downtempo track *Lebanese Blonde* by *Thievery Corporation*. The time of the track is mapped as a trajectory around the plane. The line is made thinner when the points are spaced farther apart with respect to the playing time of the track. This gives an image not unlike that of a can of paint being held directly above a flat canvas, with the paint drizzled according to the timbre playing. If the track stays with a timbre for an extended period, a large amount of paint is dispersed; if the timbre changes abruptly and often, it moves around the canvas accordingly, forming thin streaks of paint.

Time is denoted in two ways here; firstly by cycling the colour of the proverbial 'paint' through the hues of the visible spectrum, starting with red and ending similarly. Secondly, diamonds are drawn at every 15 seconds through the track, whereby a larger one represents

the passing of a minute. The rainbow colours are changed to levels of gray (starting at black) and RGb chromaticity plane added as a background, one may visualise how colours are picked for the creation of the navigation aid. Figure 4.11(b) illustrates this.

One may now imagine a streak of colours on a linear time scale which correspond exactly to where the paint has fallen on the chromaticity plane.

Demonstration

Figure 4.12 shows the principle-components-based chroma-projection (known henceforth as *PCP*) of *Lebanese Blonde*. The colours may (with some effort) be matched to those under the trajectory of figure 4.11(b). Clearly, the PCP visualisation is far more information-rich than that of the wave-based method. The fluctuating texture found throughout the track is a typical trait of PCP. A fairly clear 'top-level' segmentation of the track can be seen as it starts red/purple/green/orange/green, then blue/orange/purple/violet finishing green/blue/gray/yellow. Fluctuation patterns stemming from rhythmic variance of the timbre are also visible.

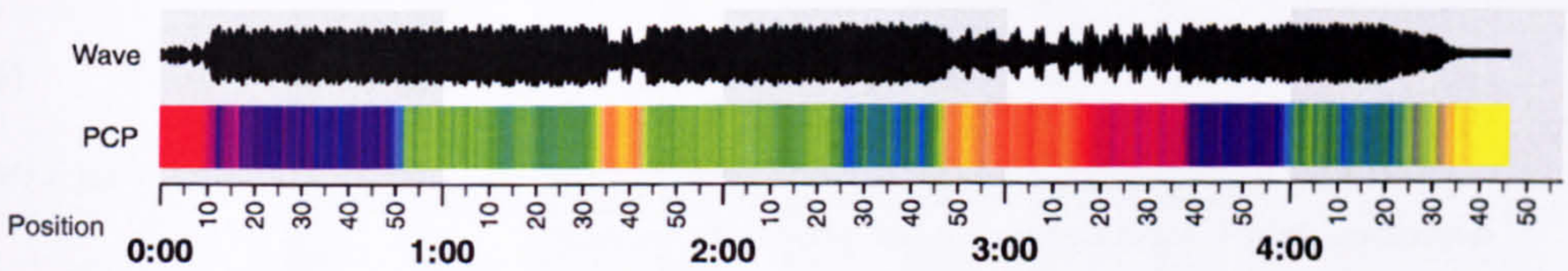


Figure 4.12: *Thievery Corporation*'s *Lebanese Blonde* visualised as a basic wave (Wave) and with PCA chroma-projection.

With vibrant, heavy use of colours, no underlying structure seems especially apparent. The variance-preserving projection misses the mean values, being expanded to the edges before clamping. This is not a trivially fixable trait; informal experimentation showed that e.g. normalising to within 2σ made the majority of the visual appear faded and difficult to interpret; only relatively unimportant outliers had vibrant colours. In the next section, we will propose and discuss a non-linear projection method, utilising the self-organising map in an attempt to overcome this issue.

4.3.4 SOM Projection

In this section, I will introduce the method I constructed for a non-linear feature-to-2D vector mapping. The SOM algorithm used and the initialisation method will be described, together with a discussion and illustration of the map size, the training process and the generated output.

Description of Algorithm

The set of model vectors m are defined to have their positions on a square grid of size mxn. For direct visualisation, a hexagonal grid is typically preferable, however as the SOM is being utilised in order to map directly onto a square plane, the slightly simpler square grid is used. The initialisation phase (discussed and described in the following section) is completed before training in earnest.

The *Batch Map* variant of the SOM algorithm, described by Kohonen (1999), was used which was found to have similar results to the traditional incremental-learning method, but taking under half the time (with the same number of 'iterations'[4]) for the equivalent training. The Batch Map learning algorithm used may be defined in terms of the set of model vectors $\mathbf{m}_p$ (p is a pair), the dataset $\mathbf{x}$, and parameter σ by the following equation:

$$\forall p : \mathbf{m}_p^* \equiv \frac{\sum_k h(c(k), p)\mathbf{x}_k}{\sum_k h(c(k), p)} \tag{4.24}$$

where $c(k)$ denotes the position (as a pair) of the model vector which most closely represents the input vector k in the mapping:

$$c(k) = \arg\min_p \|\mathbf{x}_k - \mathbf{m}_p\|^2 \tag{4.25}$$

and $h(a, b)$ defines the neighbourhood function on two map coordinates, which can be described as a Gaussian-modulated proximity measure:

$$h(a, b) \equiv \alpha e^{-\frac{\|a-b\|^2}{\sigma^2}} \equiv \alpha e^{-\frac{(a_i-b_i)^2+(a_j-b_j)^2}{\sigma^2}} \tag{4.26}$$

I implemented this by first calculating the sum of all vectors in a Voronoi[5] set $\mathbf{V}_j$, then using them to calculate the weighted average, thus:

$$\mathbf{V}_j = \{\mathbf{x}_i | c(i) = j\} \tag{4.27}$$

$$s_j \equiv \sum_{\mathbf{x}_i \in \mathbf{V}_j} \mathbf{x}_i \tag{4.28}$$

$$\forall p : \mathbf{m}_p^* \equiv \frac{\sum_k h(c(k), p)s_p}{\sum_k h(c(k), p)|\mathbf{V}_k|} \tag{4.29}$$

A further parameter, k denotes how the former parameter σ should be changed for each iteration:

[4]Technically, the Batch SOM's iterations are in groups spanning the entire training data, so iterations are only comparable if one considers an entire round of training data in the incremental method a single iteration.

[5]for an overview see the Voronoi reference by Okabe et al. (2000)

$$\sigma^* \equiv \sigma e^{-k} \tag{4.30}$$

As such there are three parameters to determine; the number of nodes s^2, the learning rate k and the initial neighbourhood radius σ. Determining the latter two parameters is difficult; with extensive experimentation I had reliably good results from $k = 0.01$ and $\sigma = s/4$. The training halts on the first iteration where the model vectors have no more significant changes applied, which was determined by:

$$\arg\max_p \|\mathbf{m}_p^* - \mathbf{m}_p\| < 0.001 \tag{4.31}$$

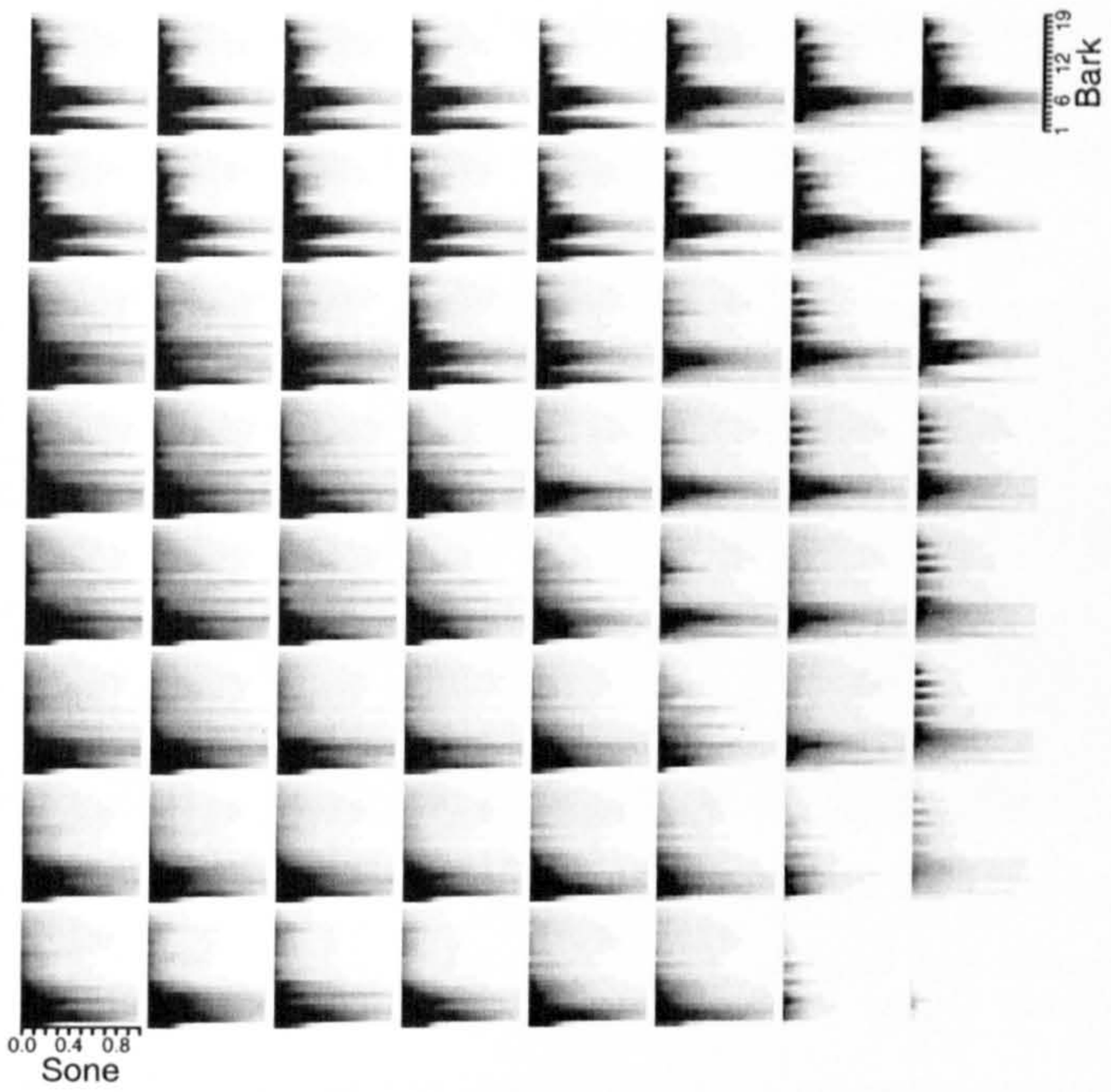


Figure 4.13: An example of an 8x8 SOM trained on the track *Lebanese Blonde*.

Figure 4.13 illustrates a trained SOM. Notice how similar feature vectors are grouped near to each other, but also the non-linearity of the map. The model vectors representing a consistently high degree of low bass and mid-range are found along the top of the map. Vectors representing quiet portions are found towards the lower right, with silence found at the bottom right hand corner. On the right hand side, vectors sporting a heavily stippled effect throughout the mid-high range bands; this represents the high-pitched strings at the start of the fourth minute of the track. Vectors on the bottom and middle left hand

sides represent temporally inconsistent, but generally loud, parts of the track through most critical bands; musically, these correspond to periodic notes from the brass instuments.

Initialisation

It has been suggested by Attik et al. (2005); Kohonen (2007b) that the training of the SOM may be optimised by initialising the nodes' vectors as a plane through the subspace of the first two principle components. More formally, the initialisation may be defined:

$$\mathbf{m}_{i,j} \equiv (\frac{2i}{m} - 1)\mathbf{e}_0 e_0 + (\frac{2j}{n} - 1)\mathbf{e}_1 e_1 + \overline{\mathbf{x}} \tag{4.32}$$

where the map $\mathbf{m}$ is of size mxn and $\overline{\mathbf{x}}$ is the mean of all input vectors over the dataset. $\mathbf{e}_n$ and e_n are respectively the nth unit eigenvector and value ordered with decreasing eigenvalues, i.e. satisfying:

$$e_n > e_{n+1} \tag{4.33}$$

$$\forall n : \|\mathbf{e}_n\| = 1 \tag{4.34}$$

Despite extensive attempts, I could not find a set of training parameters which provided a significantly faster training mechanism with such an initialisation than without. This method was therefore used for initialisation only when generating different sizes of SOM to compare against each other (since the final mapping is more likely to be similar). An initialiser where all dimensions of all vectors were set to a pseudo-random was used for the general case:

$$\forall d : \mathbf{m}_{i,j}(d) \equiv r, 0 <= r <= 1 \tag{4.35}$$

where $v(d)$ is the dth element of the vector v, r is a value chosen at random.

Input Vector PCA

PCA (discussed in section 4.3.3) is a method of calculating a linear projection such that a number of dimensions may be omitted from the data with minimal loss in total variance of the dataset. This can therefore be used as a data compression mechanism, reducing the high dimensionality of the input vector space.

Although conducting PCA on such a high dimensionality of data is computationally intensive and omits some details from the dataset, its advantages are significant. It has a dramatic effect on speeding the training for maps with a large number of nodes, and algorithms that rely on random values for clustering are more likely to have consistent outcomes over multiple runs, since the number of possible solutions and variables that need to be determined are reduced significantly.

Inspection of the output revealed no apparent differences between training with the original 1000 dimension vectors and only the 20 most principle components (typically containing around 99% of the dataset's total variation). As such, input vectors put through PCA are prefered on two conditions:

1. PCA has already been conducted anyway. In the case of many of my experiments I compared the results of PCA and the SOM, thus using PCA came at no computational cost.

2. The training time for the SOM at 1000 dimension input vectors is greater than the PCA computation time. This is generally the case at a map size of $n^2 = 25$.

Figure 4.14 illustrates the first 20 principle components of the track *Lebanese Blonde*. As in the previous section, cyan is to be considered agnostic to that particular dimension. One may see that the first component represents a varying bassline together with a level of general loudness in the music. The fourth conversely represents the varying of a signifcant amount of loudness centred around the 600 Hz bands.

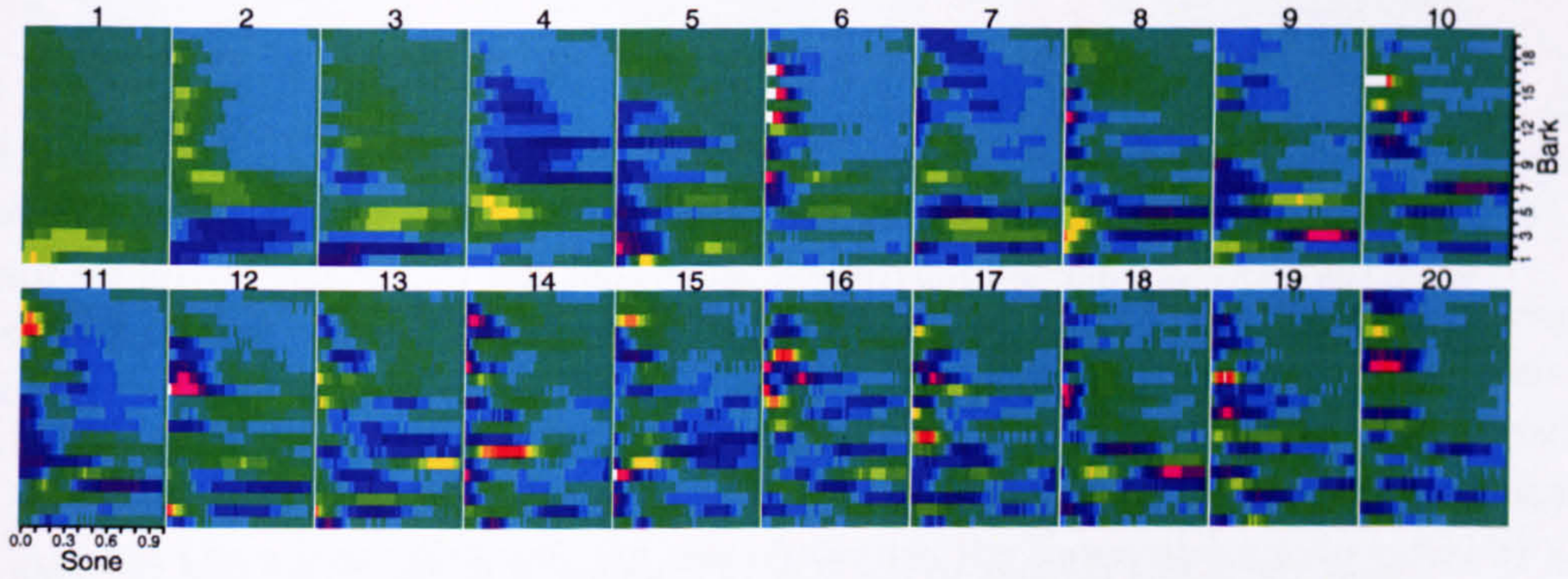


Figure 4.14: The first 20 principle components of the spectral histograms for *Lebanese Blonde*.

Size of SOM

The topology of the SOM is the final parameter to determine prior to training. The number of nodes used varies the degree of quantisation in the lower-dimensionality space. A lower level of quantisation allows subtler differences in the high-level dimensionality to be denoted by changes in colour, however this comes at a significant cost of computation. The SOM algorithm scales approximately with $O(n^2)$ where n is the number of nodes, thus the difference in training time when the size of a map is doubled from a 4x4 map to an 8x8 map is around a factor of 16.

As we have a relatively small quantity of data, one concern is that the SOM might overfit data, causing a uniform distribution of points over colour plane. Instead all the large-scale features would be captured (with their distance on the map corresponding to their perceptual distance), but intra-cluster distances would be left smaller. This would have the effect of keeping a reasonable mapping between perceptual audio difference and the perceptual chromatic difference. Clearly, this cannot happen when the number of nodes is significantly smaller than the quantity of data, so from this point of view the 4x4 SOM is favoured.

A higher-level of quantisation gives a simpler visual with a more consistent mapping of low-dimensionality variation to high-dimensionality variation, since reducing the number of possible states 'flattens' the degree of possible separation between those states. Generally, there should be fewer nodes than items in the dataset. Since there is one datum per second of audio in a track, and many popular music tracks are between three and five minutes long, this would suggest an upper limit of the map size of 12x12; in this section I discuss the results of 2x2, 4x4, 8x8 and 16x16 maps. The visualisations are called SOM2, SOM4, SOM8, SOM16 respectively.

Demonstration

Figure 4.15 shows the trajectory of *Lebanese Blonde* by *Thievery Corporation* through the RGb colour plane, as projected by four sizes of SOM. Since the line width is increased when the trajectory moves a relatively small amount over the running time of the track, it may be identified as paint dribbling from a brush held aloft. Notably, the image of the 2x2 SOM demonstrates that all transitions are between neighbouring nodes. This suggests a track without relatively abrupt changes in timbre; something not wholly unexpected from a downtempo ('chillout') track.

From the SOM4 through to the SOM16 mapping, it is clear to see that the main form of the initialisation from the two principle components has been preserved through the training process. This suggests that the main advantages to using the SOM should be that of having a natural quantization technique (in the lower map sizes), and that of finding and better expressing substructure (in the higher map sizes).

Figure 4.16 shows the final SOM-based visualisations for the same track. The varying sizes of SOM demonstrate the varying simplicity of the visualisations; the 4-node SOM2 representation essentially segments the track into four portions. On inspection, these segments correspond to:

Blue Tabla playing only.

Red Tabla with drums/bass.

Yellow Drums/bass with vocals/keyboards.

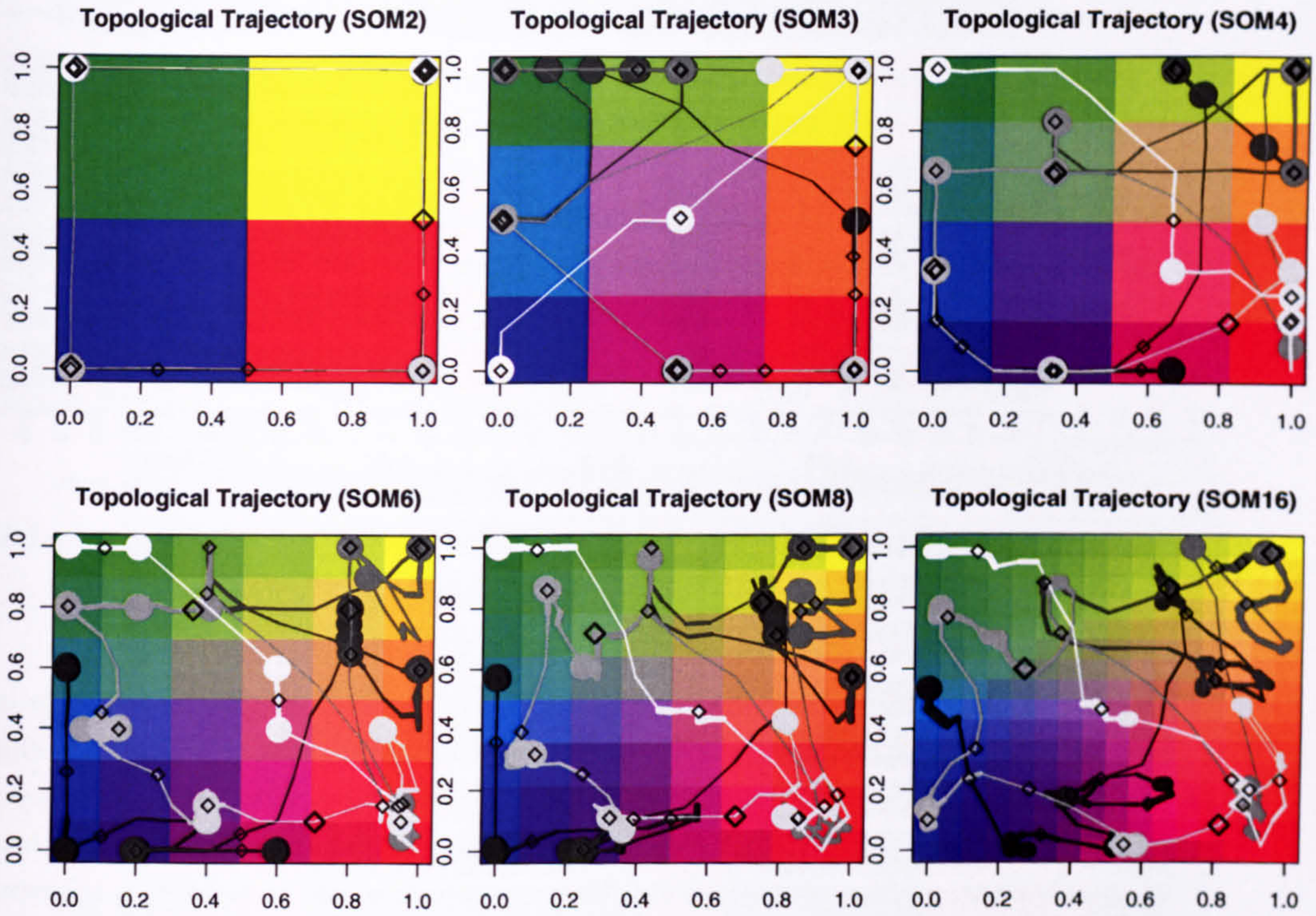


Figure 4.15: A demonstration of the trajectories followed by *Lebanese Blonde* according to six sizes of SOM. The line's width is increased for smaller distances moved, and monotonically changes from black to white throughout the playing time of the track. Small diamonds are plotted every 15 seconds of time in the music along the locus; each minute is denoted by a larger diamond. For ease of viewing, the trajectories are smoothed by a four-sample-wide moving average, and a small amount of Gaussian noise is added.

Green Drums only or with quiet keyboards.

The onset of brass instruments go unrepresented in the yellow sections, as does the break in drums in the green section.

The 3x3 and 4x4 SOM representations have an overall structure similar to that of the 2x2 SOM, although they increasingly add detail to the monotonic blocks. The change from scatting to singing at 1:12 is hinted at with the change in the shades of green. The introduction of a brass accompaniment is denoted by the change from green to yellow at 2:24. Both of these changes are also visible in the higher sizes of SOMs. Flecks in the visualisation, visible as thin 'dribbles' on the paint drizzle, begin to show; the green/yellow bar at 4:09 corresponds to a quiet tabla note in the background.

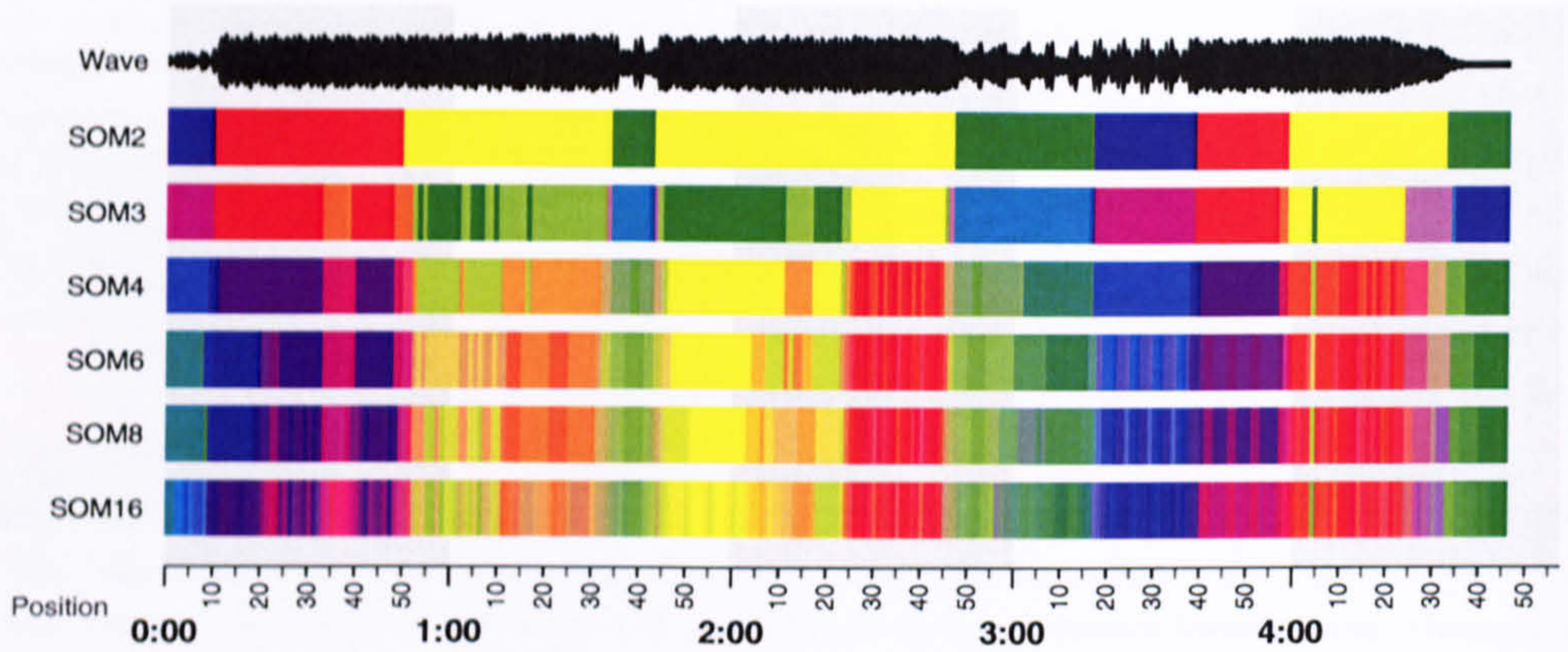


Figure 4.16: *Thievery Corporation*'s *Lebanese Blonde* visualised as a basic wave (Wave) and with a 2x2 SOM, 4x4 SOM, 8x8 SOM and 16x16 SOM.

As the SOM's size increases to 6x6 and 8x8, an increasingly textured and noisy image may be seen, but with further details shown about the track; in particular the introduction of the drums over the bass at 0:21 becomes readily visible as a change from blue to purple. Texturing added to the first yellow block of the SOM2 visualisation makes the internal structure clear, while still maintaining the overall orange hue to denote its cohesiveness and dissimilarity with surroundings. The change from scatting to vocals is again clear, but the two rhyming lines of vocals are also clear as a pair of dark/light orange blocks:

> (1:13 [dark orange]) Too low to find my way,
> (1:16 [light orange]) too high to wonder why. / I've seen this place before, / summer in another time.
> (1:23 [dark orange]) Now I can hear the sound,
> (1:26 [light orange]) o'cars drifting through the blinds. / I have a million thoughts, / all flowing through my mind.

Finally, in all but the two lower sizes of SOM, it is clear to see four blocks of colour between around 3:05 and 3:55. This colour changes a small amount from block to block; e.g. in SOM4, it starts as gray, changes to faded blue-green, then to blue, and finally to purple. These iterative changes are reflected in the music as the timbre makeup changes by a perceptually small but significant amount. In this case, the first transition is of the main drums stopping, leaving strings and bongos, the second is the introduction of the tabla and the third, the re-introduction of the drums and removal of the strings.

Effects of Training

The effects of training are quite pronounced; the SOMs are initialised with the two principle components of the dataset (which itself has been reduced through PCA to 20 dimensions). The visualisation performed on *Lebanese Blonde* by the 4x4 SOM with this set of model vectors is given by i=0 in figure 4.17. With each iteration of the Batch-Map SOM training algorithm, one may see that the map gets increasingly compressed, better fitting the data, as indicated by the colours becoming more vibrant.

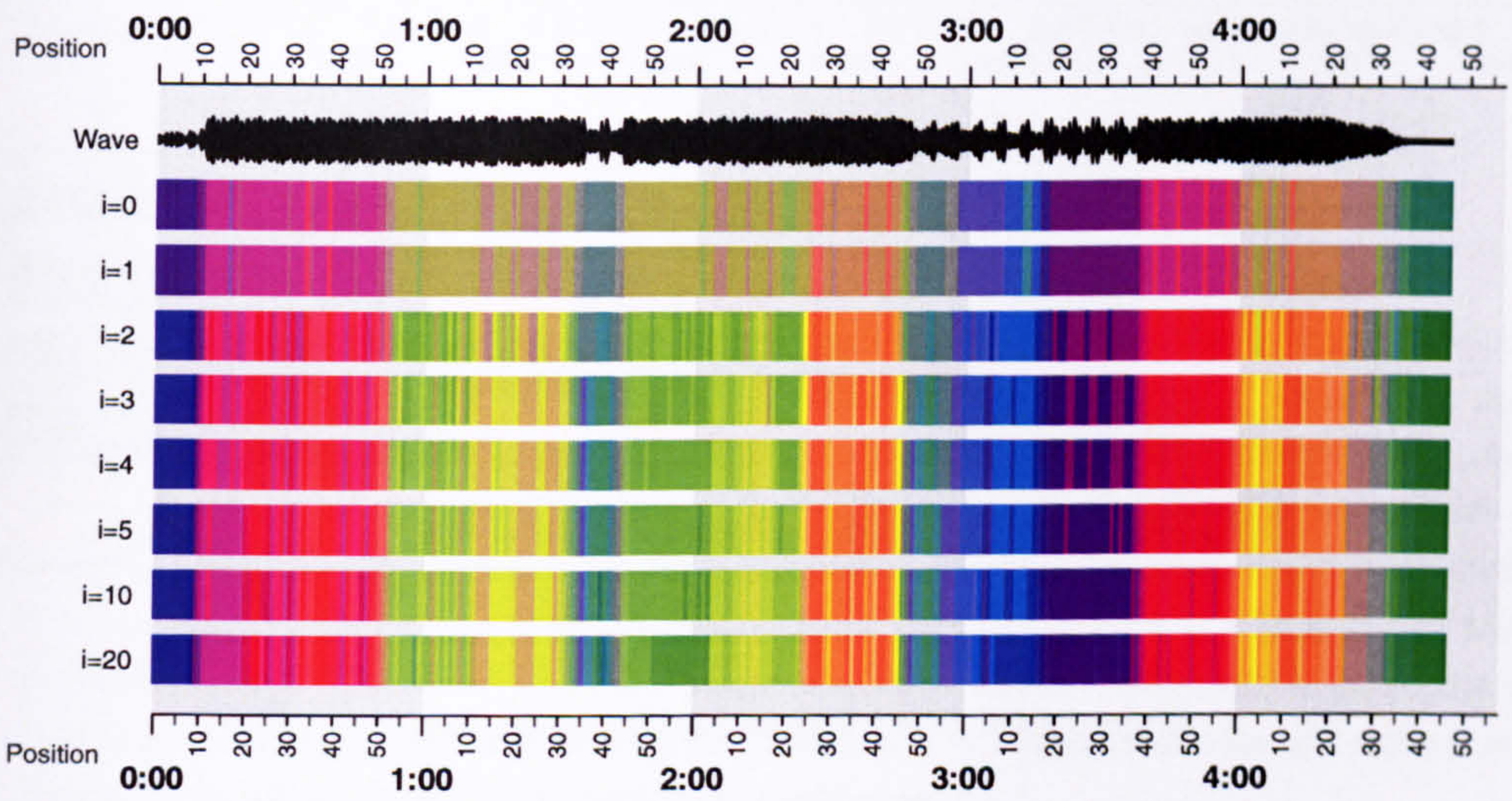


Figure 4.17: *Thievery Corporation*'s *Lebanese Blonde* in the various stages of training an 8x8 SOM. **i** denotes the number of iterations of the learning phase.

In particular, the musically important break in the drums and keyboards becomes increasingly evident at 2:05. A one-off, quiet, but nonetheless perceptible, collection of bongo beats in the background at 2:50 also becomes increasingly noticeable through the training algorithm. Furthermore, the previously mentioned break from scatting to singing at around 1:10 becomes clear.

4.4 Discussion of Methods

I will now conduct a brief qualitative evaluation of four of the variants of visualisations proposed in this chapter. I will do so by discussing their generated visuals for the same test tone and seven pieces of music from the discussion of the previous chapter.

4.4.1 Test Tone

The test tone consisting of several plucks in three blocks of differing tempos is visualised in figure 4.18. There is little of significance to discuss here; each of the methods presents a fairly good visual interpretation of the audio. SOM2 is clearly the best, which is somewhat to be expected, since the states of the audio in this simplistic example may be directly represented by the model vectors of the map. Notably, with PCP, the colours used to describe the three states are the secondary colours, found not at the extremities of the colour-space but in the centre of the edges. In general, however, it performs about as well as the other two SOMs.

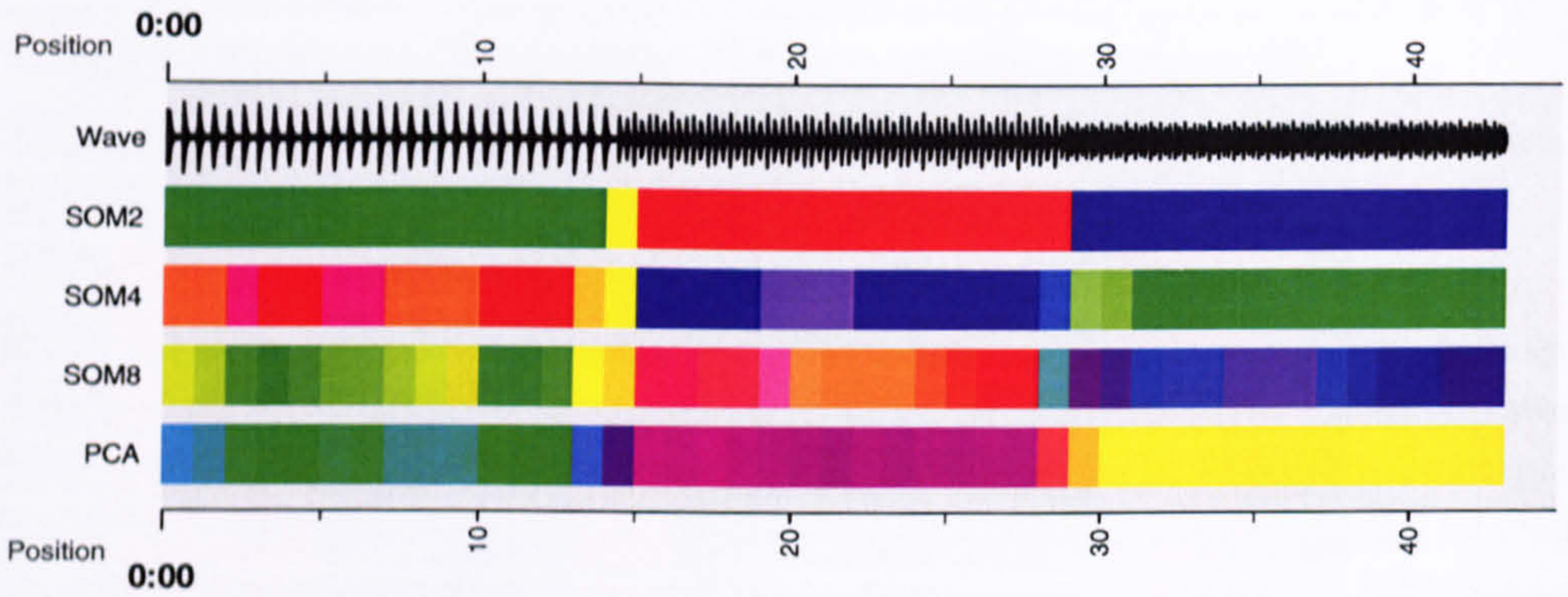


Figure 4.18: Several visualisations of the test tone 'plucks'.

The higher sizes of SOM become warped slightly as they attempt to give some sub-structure to the interference effects of the audio windowing. This perhaps highlights an edge-case of the preprocessing method made worse by the SOMs' attempts to account for it.

4.4.2 Trip-Hop

I begin the music evaluation with the trip-hop track *Stem/Long Stem*. This is a fairly complex, irregular (i.e. with no repetitive structure), subtle (changes are often gradual) and diverse piece of music. The track has two main crescendos preceded by similar progressions, the first, smaller one at 1:40, the second at 3:23. After this it changes style somewhat, removing the originally quite important percussion altogether and changing the ensemble.

The SOM2's depiction of the overall structure of the track is mediocre; crescendos are represented clearly in blue, and red denotes dominant samples of brass instruments in the foreground. However, the SOM2 misrepresents the change in style with a yellow block at 3:24, suggesting a repeat of the start of the track. In general, the yellow represents simpler, quieter music of the same basic style to that of green. Unfortunately, applying

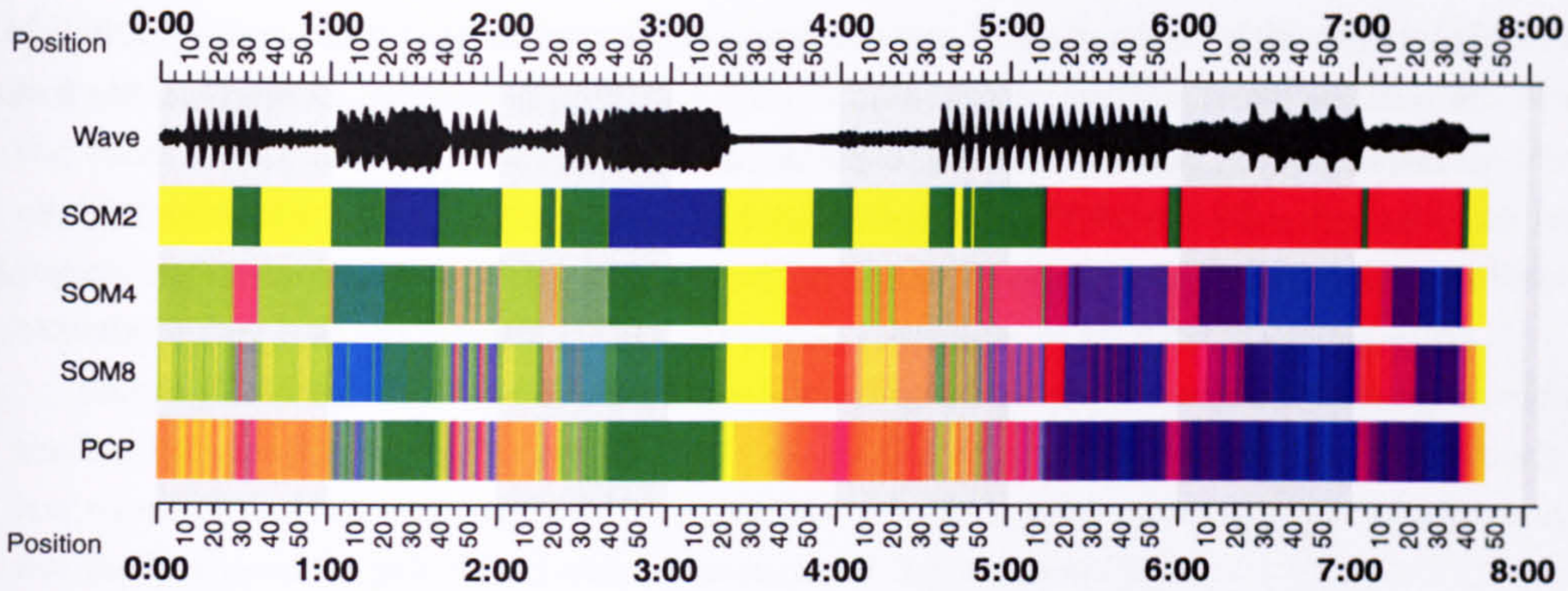


Figure 4.19: Several visualisations of the track *Stem/Long Stem* by *DJ Shadow*

such a boundary does not adequately represent its diverse content.

The SOM4 utilises a significantly greater palette to construct its image, with the SOM8 presenting an image with more detail, noise, and only a few significant changes. The rose block at the beginning (around 0:30) represents a note from a string instrument, most likely a double bass, in the background. The change to dark green represents the onset of erratic drums, and light green to sustained percussion and the crescendo. With the SOM2, the following change to gray does not clearly show that the content is a variation of that at the beginning. The SOM4 and SOM8 represent the variation reasonably well, depicting it as faded yellow/yellow/dark faded green, and finally a longer crescendo of bright green (rather than yellow/rose/yellow/dark green/light green).

The brightness of green denoting the veracity of percussion is also a reasonable analogue. However like the SOM2, SOM4 and SOM8 suffer from reusing yellow in the second half (at 3:23), which, aside from being on the whole quieter, does not share much in common with the beginning. The use of orange (at 3:43) for the organ is reasonable, however it is depicted as a clear block in the SOM4, and a gradient from yellow in the SOM8; the latter is a better representation of the gradual fade-in of the organ.

Between 4:08 and 5:12, the track becomes relatively similar to the beginning (yellow blocks), although speech apparently sampled from a film as well as some other instruments are mixed over the top; as such, the mild orange and pink hue changes may be considered warranted. Following this, the two higher SOMs improve on the large red block of the SOM2 by introducing some substructure; loud brass notes are consistently represented as red and purple blocks; red being for the higher notes (there are only three main notes).

On initial inspection, the PCP image is visually pleasing. In particular, the gradient between 3:24 and 4:05 is largely reflective of the approaching organ. However, like the other methods, it suffers from being the same general hue as other portions of the track (not least that which follows it directly) which are musically different. The end portion of

the track with blue and red hues are somewhat less clear than the banding of SOM4 and SOM8. With the PCP, with the blue-purple striping are barely distinguishable from the blue; the brass notes in the music are readily so.

In general, the PCP provides no more information that the SOM4 or SOM8, both of which are clearer images. The SOM2 seems unable to represent the diversity of the track properly with its minimalist palette; the clarity lost by the SOM8 over the SOM4 is arguably too great a price to pay for the small amount of extra information.

4.4.3 Rap

Figure 4.20 *The Force*, *Aim*'s rap track. All four of the visualisation methods appear to work excellently on this timbre-dominated track of roughly constant intensity.

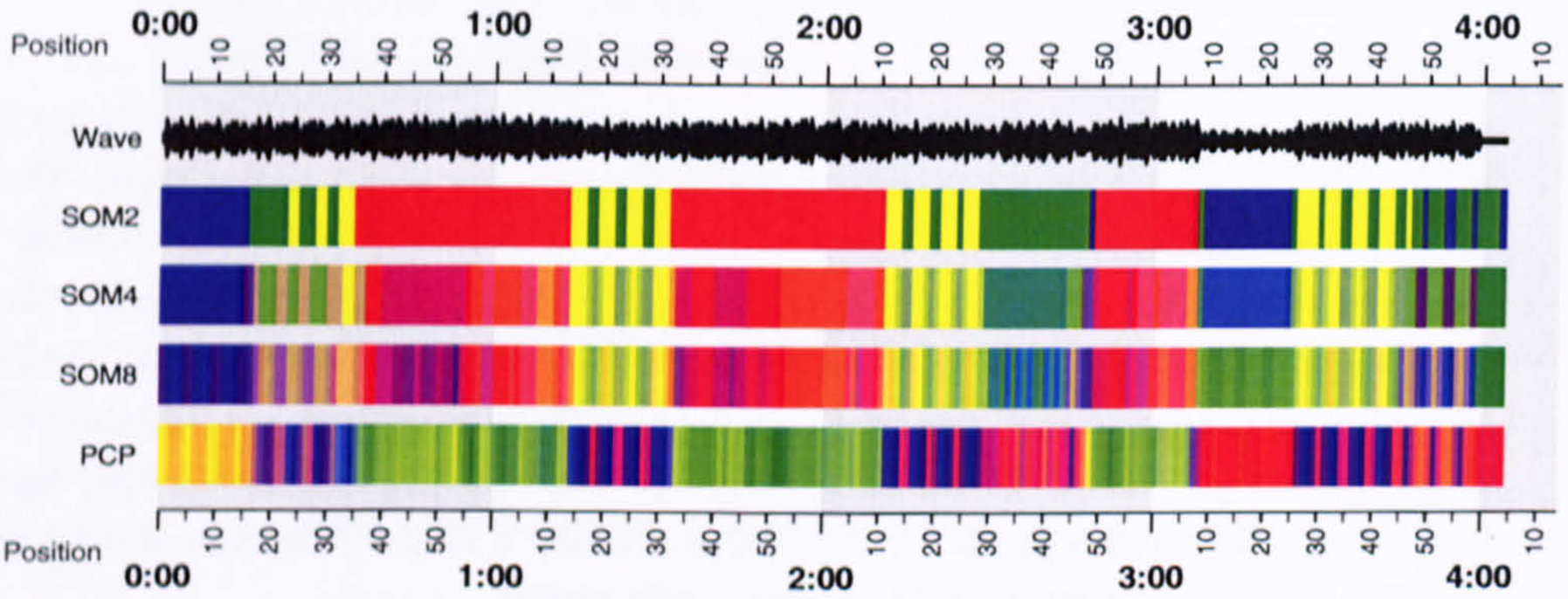


Figure 4.20: Several visualisations of the rap track *The Force* by *Aim* (featuring *Q'n'C*).

Both the PCP and the SOM2 do a very reasonable top-level visual representation of the track; the other two introduce a little more substructure. An introduction is visible (blue in the SOMs and yellow in the PCP), as are three repetitions of the chorus (yellow/green in the SOMs and blue/red in the PCP), and two repetitions of the verse (red in the SOMs and green in the PCP). SOM4 and SOM8 clearly show that the initial chorus is a variation on the following versions, with a slightly different coloured section (green/cream in SOM4 and gray/cream in SOM8). All four visualisations represent the chorus substructure as four repetitions of nondescript vocals.

SOM2 misrepresents the section starting 3:11 as being identical to the first section; although both are quiet the first is purely percussion and the latter purely nondescript vocals. However, aside from that, SOM2's scheme is generally robust. Green represents percussion, with yellow and red representing the vocals of the female and male respectively. The striped portions denote broken vocals over a background of percussion.

The other SOM visualisations largely build on the simplicity of SOM2, with SOM8

removing some advantageous quantisation of colour. The main contribution of SOM4 is to split the vocals between each vocalist into two clear sections, one purple-red and the other a perceptually similar—but noticeably different—orange. SOM8 removes the misrepresentation of the apparent repetition of the first section at 3:11, but does so at the cost of a significant amount of noise added throughout, and a less clear demarcation between vocals and non-vocals.

Although close, SOM2 and SOM4 arguably perform best overall. PCP tends to provide similar information to SOM2 as does SOM8 to SOM4, but in both cases it is simply less clear. Selecting between SOM2 and SOM4 would depend upon whether clarity and simplicity should be prioritised over substructural information.

4.4.4 Jazz

Figure 4.21 depicts *Dave Brubeck*'s jazz classic *Take Five*. The complexity of each image is immediately noticeable.

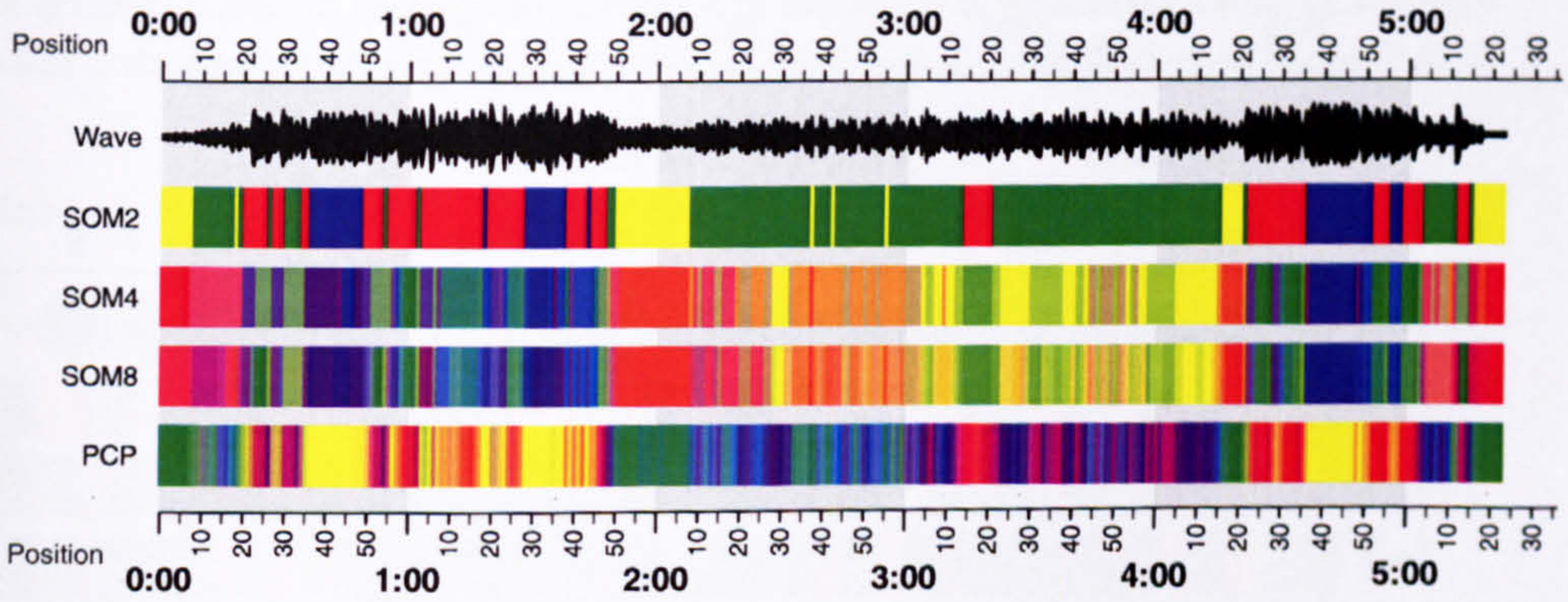


Figure 4.21: Several visualisations of the jazz track *Take Five* by *Dave Brubeck*.

In terms of a top-level overview, the track has three main portions; an initial theme with each of the instruments (until 1:52), a central percussion-dominated section (until 4:23), and the reprise of the theme in the last section. This structure is broadly noticeable in each visualisation; SOM2 segregates the track into red/blue and green/yellow, the other SOMs into cooler colours green/blue and the warmer red/orange/yellow, while the PCP separates similarly but with the opposite meaning.

Events in the track are also depicted commonly; each of the SOMs makes visible the consistent and relatively quiet percussion and piano between 1:52 and 2:10 when the percussion becomes dominant. This is not quite so clear in the PCP, but is discernible with the faint green tinting. Notably, both PCP and SOM2 misrepresent a section of loud percussion around 3:15 as a block pertaining to the main theme; presumably this is an

unaccounted-for outlier of the distribution.

In terms of substructure, the two SOMs reveal the gradual crescendo achieved by the drums by a similarly gradual change in hue: From orange to pale orange through grey, yellow with pale green, yellow with grey specks (representing loud and irregular drumming) and finally unbroken yellow. The PCP and SOM2 are less instructive here, however in terms of repetition and variation of the main figure of the saxophone, they perform significantly better than the other methods. The theme is clearly visible in the PCP as two bold violet bars on a yellow backing repeated; the reprise of the theme at 4:24 is clearly recognisable, despite the bars being red denoting a slightly louder accompaniment of instruments. The two larger SOMs do show this information (but as double green/grey bars), and as such it is entirely possible that the clearer visuals is simply due to an unfortunate rotation of the colour space.

4.4.5 Classical

Figure 4.22 shows various representations of *Mozart*'s *Nachtmusik Allegro*, a classical string track; immediately noticeable is the difficulty of extracting structure from such music with purely timbrel information.

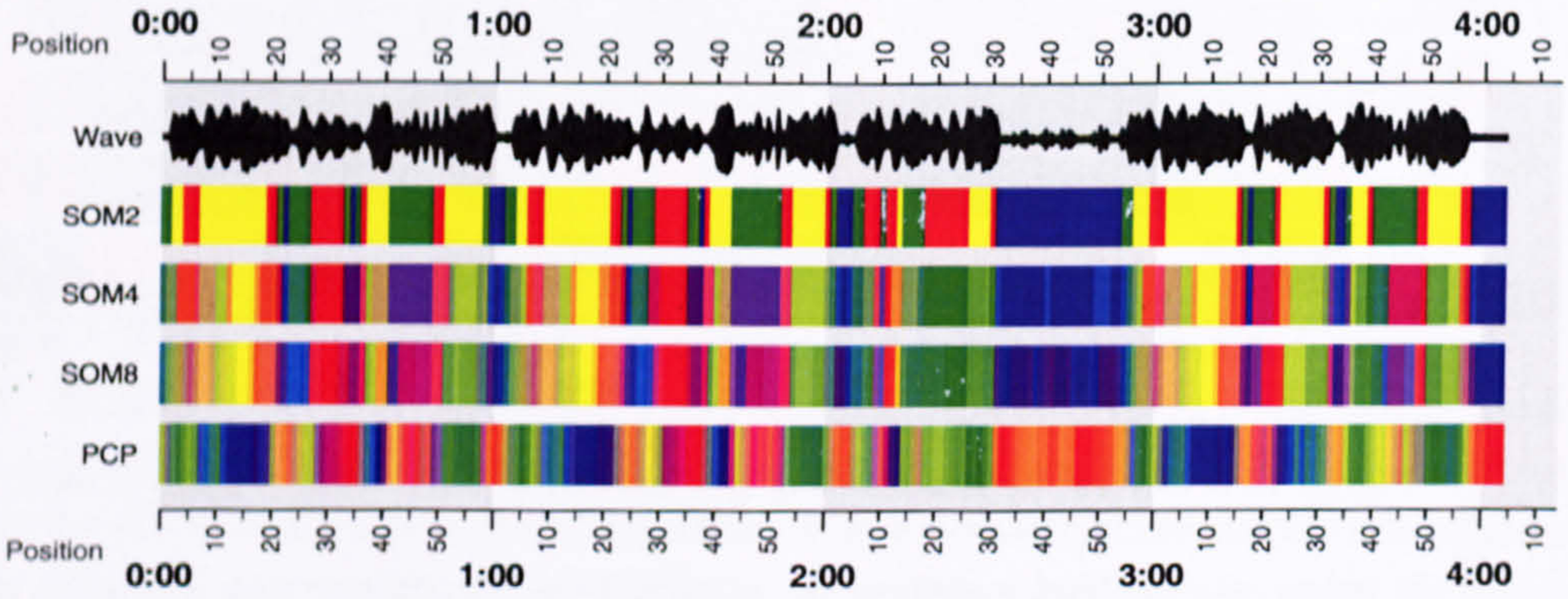


Figure 4.22: Several visualisations of the classical track *Nachtmusik Allegro* by *Wolfgang Amadeus Mozart*.

The complex series of primary colour stripes of the SOM2 visual does not appear to be especially instructive as to the content of the track; the large blue portion at around 2:30 denotes the quiet theme, performed without a second set of strings in the background, giving the track its fast-paced rhythm. On close inspection, we can see that the track, prior to the blue portion, comprises a single theme played twice at the beginning and again just after the first minute. It may be seen that a variation of this theme is played once more, directly after the blue portion, at 2:56.

With a slightly better packed gamut, the SOM4 and SOM8 visualisations make the above facts slightly easier to spot, especially as the non-repeated theme before the blue section at 2:06 is of a different colour. However, this is not reflective of the music, which is, at that point, a similar motif played in a lower key. Despite considerable detail, no substructure is apparent.

The PCP visualisation makes a similar amount of information apparent as the first two, although the quiet break between 2:30 and 2:55 is not as apparent, with oranges and reds used, which are flecks elsewhere in the track.

Although none performed especially well, the SOM4 visualisation seemed the best suited to provide as clear a picture of the track as possible under the circumstances.

4.4.6 Downtempo/Electronic

Figure 4.23 shows the visuals generated for *Occhi Neri*, an electronic downtempo track by *The Dining Rooms*.

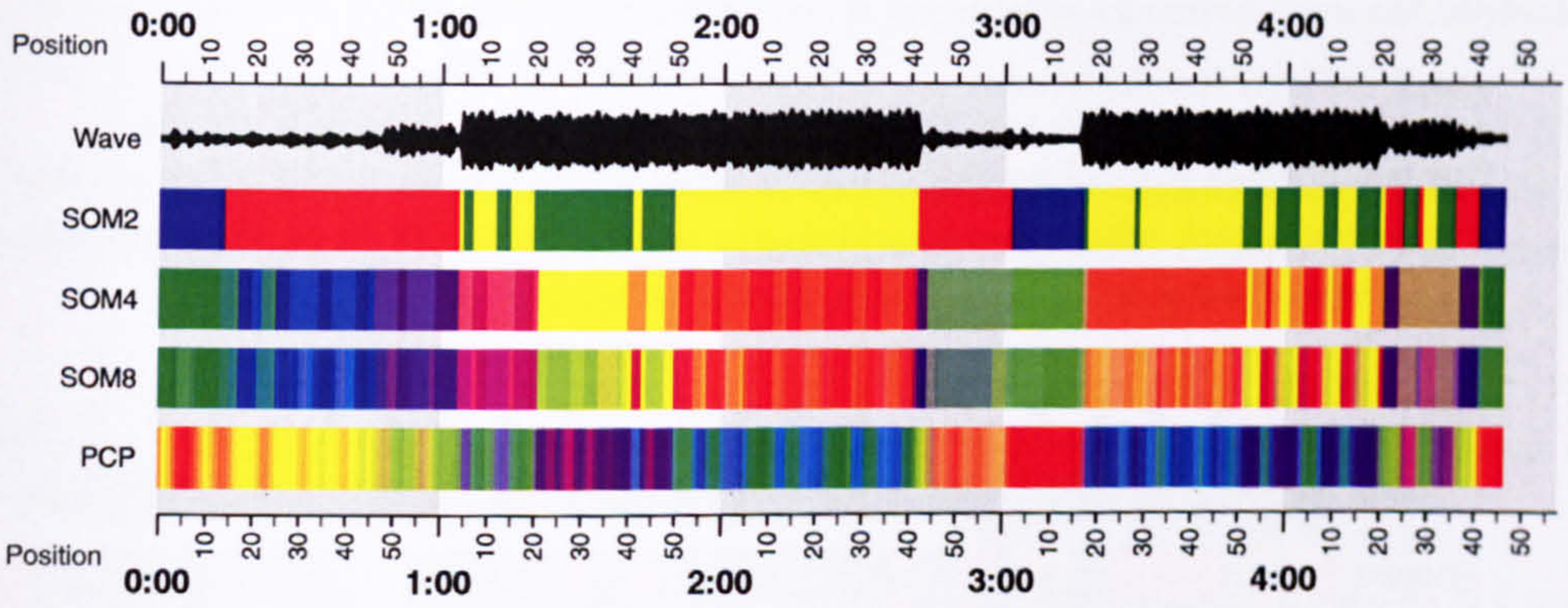


Figure 4.23: Several visualisations of the downtempo track *Occhi Neri* by *The Dining Rooms*.

Completely unclear on both the PCP and Wave visualisations is that the music does not start until 0:15. All three SOM visualisations are sensitive to this fact, with clear and accurately abrupt colour changes. A further aspect largely missed in both the SOM2 and PCP visuals is the very noticeable start of percussion at 0:48. Although there is a slight change of tint in the PCP image around this time, it is undeservingly subtle and barely noticeable. The quite important introduction of bass at around 1:05 is well represented in each SOM image, but once again unclear with the PCP.

For the rest of the track, each of the visualisations performs roughly equivalently, each presenting some sub-structure between 3:50 and 4:20. The PCP image delivers the most subtle substructure, giving very blurred stripes between blue-purple and purple. Arguably,

this could be considered this the most accurate—the music at this point is a stable run of percussion with samples merged over the top. Conversely, however, the track is very similar to this throughout; as such there is an argument for 'normalising' actually small but *relatively* large differences in the music to equivalently large visual features. Furthermore although the stripes may not denote particularly significant timbrel features, they do represent well the repetition of a figure, and so may be considered as musically important.

The end of the track is a run of percussion ending at around 4:40 (clear on the SOM4 and SOM8), with fairly quiet samples of chatter mixed. This appears to disrupt the SOM2 and PCP algorithms, which result in an unclear combination of colours for the last 25 seconds.

SOM4 and SOM8, except for the smoothing, are especially similar in this track; SOM8 provides a subtle cue to the equally subtle change in the loudness and the addition of an extra periodic sample at 3:35. For this track, the SOM2 representation underperformed slightly with its harsh segmentation noting points of lesser significance in the music, and presenting a slightly confused image in the latter half.

4.4.7 Pop/Rock/Metal

Our penultimate piece of music to consider is the rock ballad *Generator*. This is a relatively simple rock track with a verse/bridge/chorus structure, and is well-represented on an overall structural level by each image, shown in figure 4.24.

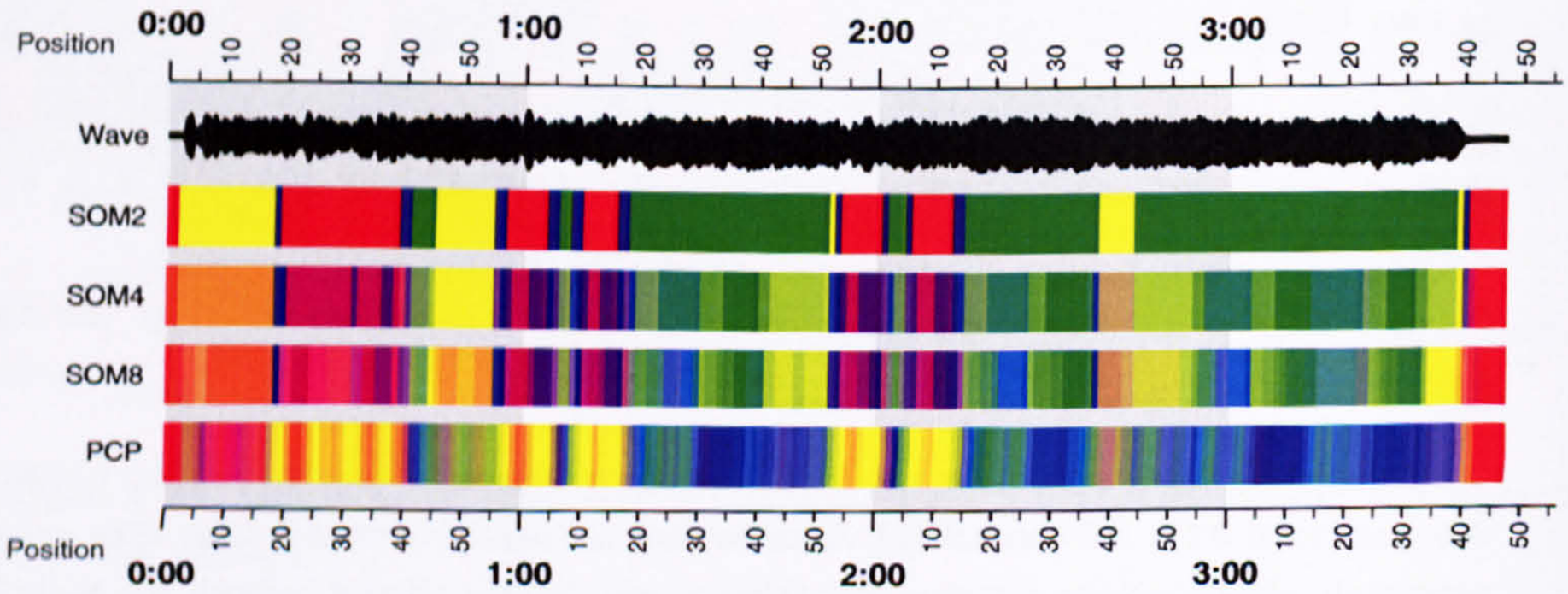


Figure 4.24: Several visualisations of the rock ballad *Generator* by the *Foo Fighters*.

The clear segmentation afforded by SOM2 gives the viewer an instant idea of the structure of the track. After the introduction, the three verses are clearly depicted in red, with the abrupt guitar riff of the latter two depicted with an abrupt green stripe in the middle. Blue appears to be used essentially as a waypoint between red and green here, occurring at boundaries between the two with no apparent equivalent in the music. Unlike

SOM4, which clearly defines the three keys of the chorus (dark green/grey, light green and yellow), the lack of colours available to the SOM2 algorithm prevents any such substructure from becoming apparent. This is particularly noticeable after the bridge between 2:38 and 2:57, after which the SOM2 remains green until the outro.

The SOM4 and SOM8 once again build upon the SOM2 representation, but blur it slightly (especially in the red-dominated verses), reducing the clarity. Small amounts of extra substructure is introduced with the SOM8, such as the repetition of a six-second guitar riff at 0:44 and 0:50, visible as a pair of yellow/pale orange stripes. On the whole, the extra structure added with the SOM8, while perhaps following the music perceptually (which changes timbre fairly frequently), does reduce its utility in terms of recognisable features.

The PCP provides a more subtle view overall. Although visible, the chorus sections of yellow are not quite as clear as in the other images, especially the musically quite clear separation at 0:56. In the blue-dominated choruses, the PCP provides a subtle cue to the change in key, as it changes from blue-green to blue and then to blue-purple. The bridge at 2:38 fading into a murky green-blue, however, is not clearly separated from the obviously different vocal section after it at 2:56.

In general, the simple structure of this track is captured well with the SOM4 image. The PCP image does not provide a clear guide to the contents; emphasising a large difference on the overall structure, it gives too slight a difference on reasonably important substructure.

4.4.8 Dance/Classical Fusion

Figure 4.25 shows the visuals generated for the dance/classical fusion track *Clubbed to Death*.

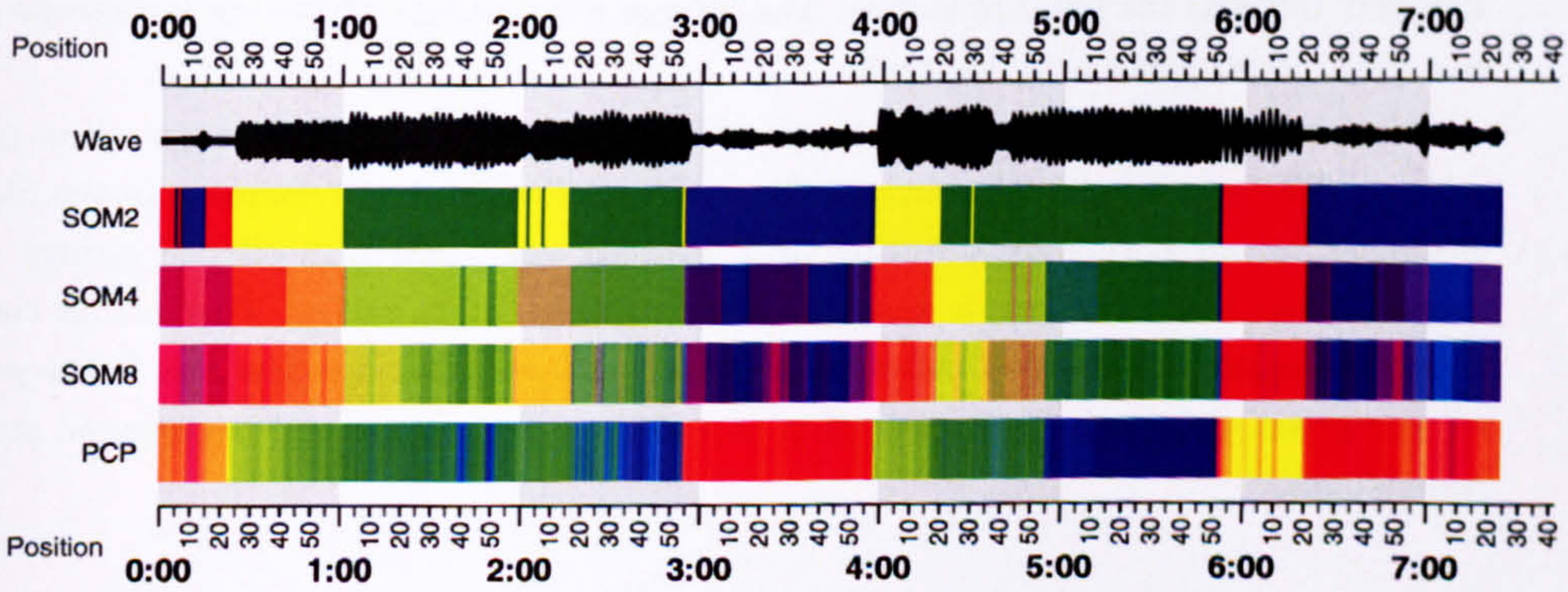


Figure 4.25: Several visualisations of the dance/classical track *Clubbed to Death* by *Rob D.*

The PCP visual shows a dubious separation of the classical portions of the track from

the percussion; yellow is used to denote the quiet bass and piano section around 5:56, with the perceptually similar yellow-green used to denote the onset of heavy drums and bass around 0:24 and 4:00. The bold red and blue blocks do well at separating the extreme timbres present in the track, but the simplistic dimension mapping does not seem to reflect the multi-timbre nature of the first two and a half minutes, most of it being a (difficult to interpret) mix of yellow/green/blue colours with no clear boundaries (which are present in the music as changes in key and voice ensemble). This track illustrates the problems of relying upon a linear map which provides too few degrees of freedom in a long and timbrely-complex track.

The SOM2 method once again conducts a reasonably clean segmentation. Blue represents the purely classical portions of the track that *Rob D* himself composed and played including both piano and strings. Red represents quieter strings and piano. The red/blue striping at the beginning represents the excerpt of strings taken from the *Main Theme* of *Elgar*'s '*Enigma Variations*'. Thus the change from red to blue at approximately 6:25 represents the introduction of strings and a loudening of the piano. Yellow represents percussion with a small, if any, string accompaniment, whereas green represents louder percussion with more voices (sampling, strings, piano, bass) in the background. The SOM2 functions here as a simple method that draws attention to the main changes and that acts essentially as a top-level segmenter.

The SOM4 and SOM8 methods build upon the SOM2 representation considerably; classical portions of the track are again represented by the red-blue portion of colour-space, but the use of light and dark purple helps represent substructure. Blue represents a raised key in the piano solo. The pinks at the beginning represent the violins; notably, after they are introduced around 6:55, the intense blue and purple become less so, becoming 'dragged' towards the pinks of violins. The intense red portion denotes the low-tempo bass and quiet piano portion of the track.

The percussion portions of the track are represented well in the orange-yellow-green colours. The colours change subtly as the medley of instruments changes; in the SOM8 visualisation, an abrupt substitution of a sampled voice and high-pitched strings with slower strings in a lower key, is denoted by a change to bright yellow. This is a far clearer change compared to the two before it of orange-yellow to faded-yellow and faded-yellow to pale green, denoting the introduction of a single extra sample and background strings respectively.

4.5 Conclusions

I have detailed and discussed a novel audio visualisation technique, and evaluated it in terms of being useful as musical audio navigation aid. I consider the results of the discussion

of the images to be encouraging. Usage of the self-organising map as a dimensionality reduction method appears to be largely vindicated over simpler techniques such as PCA, due to the amount and clarity of information perceived by the viewer. The linearity of the PCA seems well suited to depict overall structure clearly and attractively, however subtler and transient extreme changes are often ignored or misrepresented. The natural usage of the SOM as a quantisation and clustering technique also seems reasonable, with the 4x4 map often presenting the majority of information of that given by the 8x8, but with increased clarity.

I have now examined a range of simple musical audio visualisation techniques that are appropriate for augmenting a navigation aid. In the next chapter I detail a quantitative task-oriented evaluation of the better methods of visualisation that have so far been identified.

Chapter 5

Evaluation of Navigation Aids

"The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' (I found it!) but 'That's funny ...' "

—Isaac Azimov (1920-1992)

5.1 Introduction

My aim is to determine whether automatic content-based visual aids actually do help people navigate. A common method for visualising sound is the loudness waveform, which is likely to be easy for people to understand and use. I use this as a baseline to determine whether other visualisations are more effective. To test the theories presented over usage improvement in chapter 2 I determine what learning effects there are, if any, over time, and how this changes between those who use the visualisation and those who do not. To determine whether the theory concerning irrelevance of absolute information is founded together with the theory that the SOM presents information better than the PCA methods, I test whether users utilising the SOM or PCA methods can perform as well as with the original methods.

In general, I actively look for evidence to support or refute these theories. I test these hypotheses by constructing a null hypothesis which I may attempt to refute, statistically, to some degree of certainty. In all the statements I make in this chapter the statistically accepted probability of their truth is at least 95%, and considerably more in many instances.

5.1.1 Chapter Summary

I begin by detailing work related to the evaluation of visual navigation aids, of which there is very little. I continue with the first two studies, which, though essentially simple pilot studies, resulted in enough information to make some statistically significant statements. After these, I move on to the major study, where, with realistic use cases, I determine

usage patterns and comparative performances over a variety of what I consider the best proposals of visual aid. I then detail the final study carried out to investigate the learning effects over several weeks of usage (prior to this all studies had assumed a minimum of training).

5.1.2 Our Contributions

With the studies made here, I present, with statistical significance, evidence supporting the theory that content-based visualisations can indeed guide users around the track, contributing to their performance over a variety of tasks. I show that both the basic bandwise-loudness and SOM-based visualisations aid users in determining answers to a variety of questions posed on the content of pieces of music, over simpler methods such as the widely used waveform. Furthermore, I present evidence suggesting the reasons for this are a pronounced reduction on the number of seek operations around the music and an increased accuracy of seek operations. I also make a number of other significant observations about individual usage and methods.

5.2 Notes

Statistical Testing

For determining general relations and making categorical statements concerning these results, I defer to the *Tukey Honest Significant Difference* (Tukey HSD) method of inference of p-values (hypothesis certainty) as described by Yandell (1997). This is more conservative (e.g. than a basic pairwise two-way t-test), since it takes into account experiment-wide variation of mean, and increases p-values accordingly.

Problems with Testing

The nature of tests means that finding statistically significant results is difficult, due to wide ranges of people's skill and experience. This could be mitigated though conducting significantly more trials. However the resources to do this were not available. In total, 106 individual trials were conducted. 82 individuals took part in the studies, and, with several tasks in each trial, I collected 2686 individual trial results.

Software

In augmenting the software with which to do the trials, the principle of least surprise was followed and the playback interface was changed as little as possible. The media player that I set about to augment, *amaroK*, already provided a highly intuitive interface with the now-standard navigation bar. In amaroK's case, a triangular pointer scrolls across

Figure 5.1: The amaroK music player with the BLSM visualisation operational.

the top of the bar, denoting the current position of the player through the track. The only change I made to the interface was to have the internal portion of the bar coloured according to the visualisation algorithm. As before, the user was free to click anywhere on the bar, to warp the player to the corresponding position in the song.

5.3 Boundary-Finding Tests

The first two performance examinations in this chapter are based around the concept of boundaries in music. By this I mean distinguishing points in the music which someone would want to remember or know in advance. These might include onsets or removals of instruments and voices, chorus/verse/bridge changes and crescendos. I leave exactly what constitutes a boundary as a decision for the user, but conduct a preliminary study to verify that there is some agreement between the opinions of multiple users over such boundaries.

5.3.1 Objectivity of Boundaries

I initially conduct a small study to test the hypothesis that multiple people do not disagree considerably about where they consider major boundaries to fall in a given piece of music. The validity of this hypothesis will be assumed in interpreting the following experiments. Furthermore, I expect that there will be more agreement on boundaries in popular music, owing to the presumption of typical and familiar structure (i.e. verse/bridge/chorus) that participants may utilise to determine prototype boundaries.

There is prior work by Tzanetakis and Cook (2000b), who concluded in favour of this hypothesis. The results showed that around three quarters (0.73, 0.76 or 0.79 depending on the configuration of participants) of the total number of boundaries reported would typically be agreed upon by at least 50% of participants. This was a study of 20 participants (twice ours), where they were given a special segmenting audio editor which automatically suggested to them potential boundaries.

Although the order of audio tracks given to the participants was randomised in each case, the measured learning curve for the audio editor was steep, suggesting that there may be a significant amount of (albeit fair) noise in the results. Participants were being timed, potentially biasing them into selecting at speed rather than accuracy and precision. Audio was not discriminated; music together with spoken word was used. This experiment is a further test of the hypothesis under slightly different conditions (entirely music, specific genres), and thus largely complementary. Since the music used in this test also happens to be that used in two of the later experiments, it also helps give some direct validation for the interpreting of their results.

Method

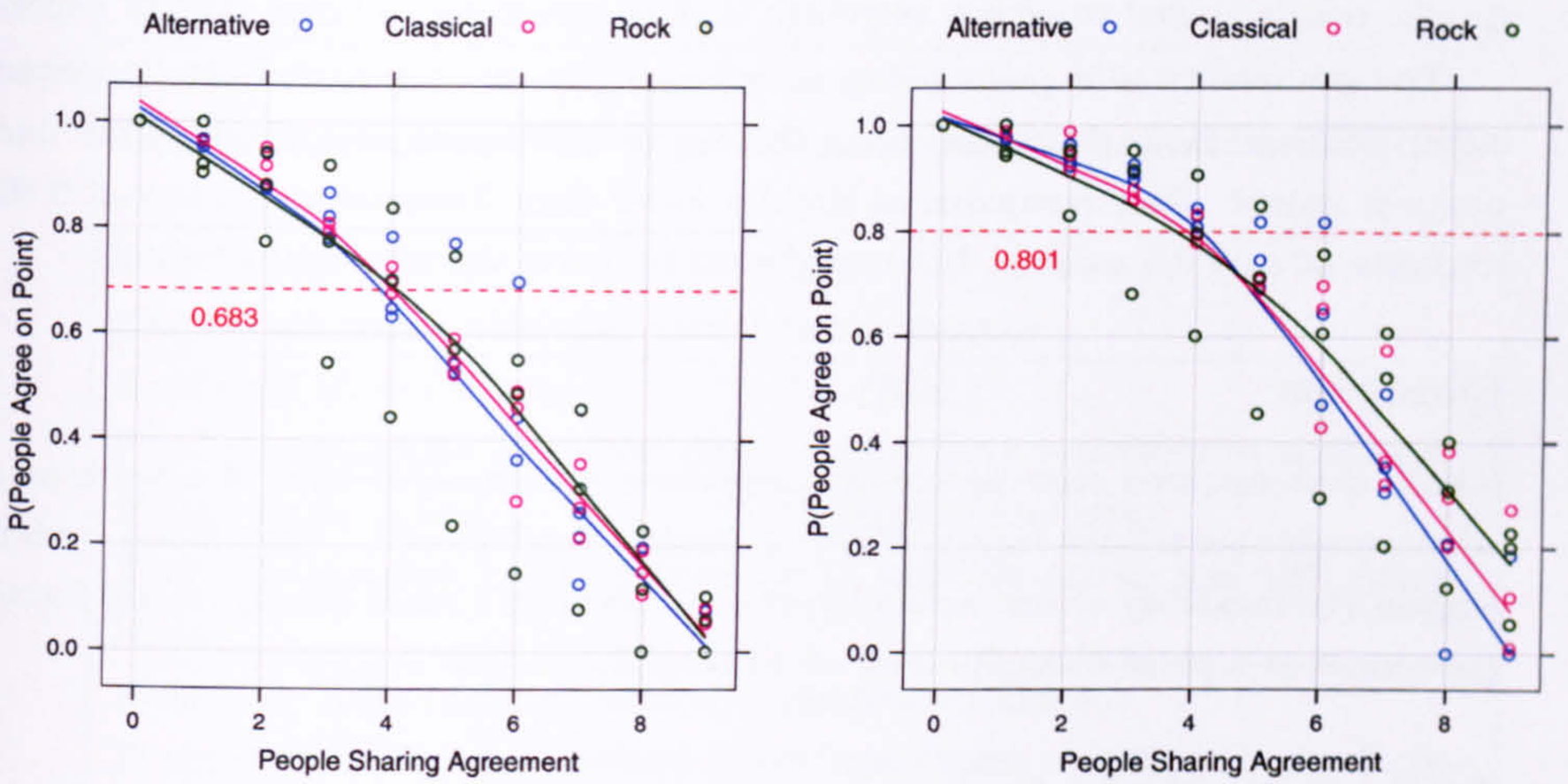
Ten participants were given nine pieces of music each, to determine the boundaries in. For the music, three tracks were used from each of the following genres; classical, rock/pop and alternative electronica. The tracks used are given in table 5.3.1.

Name	Genre
Air on a G-String (*J. S. Bach*)	Classical
Siren Song (*Devics*)	Pop/Rock
Elios Therepia (*Blue States*)	Alternative Electronica
Country House (*Blur*)	Pop/Rock
Minuet in A (*Luigi Boccherini*)	Classical
Lebanese Blonde (*Thievery Corporation*)	Alternative Electronica
Paranoid Android (*Radiohead*)	Pop/Rock
They're Hanging Me Tonight (*Red Snapper*)	Alternative Electronica
Barber's Adagio for Strings (*T. G. Albinoni*)	Classical

Table 5.1: Tracks used for determining agreement upon boundary positions.

Each of the participants listened to the tracks, free from a time limit, on their own music equipment. They noted down 8 ± 2 most important boundary points in the music. Exactly what constituted a boundary was left for them to decide, although as a guideline they were given the same directions for selecting them as those taking part in the future experiments. Notably, each of the participants listened to Western popular music for at least one hour each day, suggesting that they may have a relatively shared appraisal of what would constitute a boundary.

Once collected, the points were collated and cross-referenced to test for agreement.



(a) 1 second disparity for agreement

(b) 2 second disparity for agreement

Figure 5.2: The approximate probability of any given boundary point being agreed upon between a number of people.

Results

Having tabulated the results, the change in approximate probability of finding a degree of agreement of a given boundary point with the number of other persons' opinions sought can be graphed. Figure 5.2(b) shows the graph, requiring a disparity between points of at most two seconds for them to be considered in agreement. Figure 5.2(a) shows the same data with a stricter disparity of one second.

An analysis of variance may be conducted, to check how each factor apparently affects the result. Table 5.3.1 shows the results of the analysis, confirming the significance of the curve. Notably, for the stricter appraisal of agreement, the small differences in agreement between genres appears statistically significant. A post-hoc Tukey HSD test reveals that we can be 95% certain that classical music has a **greater** amount of agreement (albeit only by a mean difference of 0.068).

Degrees of Freedom		**P-value**	
		1-second	2-second
People	1	≈ 0	≈ 0
Genre	2	0.01465	0.1721

From the figures, it is clear to see the difference of overall curve of the graphs, with the two second disparity yielding a shallower curve overall. This shows how, among 2-3

people, points tended to have a very high level of agreement, if being slightly imprecise.

The probability of a point being agreed upon by at least half of the participants is either 0.68 and 0.80, depending upon the degree of forgiveness applied for the duration between points. This compares as slightly lower than Tzanetakis's measured 0.76 at a disparity of only 0.5 seconds, but nonetheless supports the primary hypothesis.

Conclusion

I have demonstrated that people typically have a reasonable level of agreement about what constitutes a boundary in Western music. Furthermore, I have found evidence to suggest the invalidity of the second hypothesis, showing a small but significant increase in agreement at a point-disparity level of ± 1 second.

5.3.2 Basic Methods

The first study made was essentially a pilot study to test the hypothesis that casual music navigation can be aided by provision of an image. Five visualisations from those proposed in chapter 3 were integrated into the navigation bar of the popular Free-software music player, *amaroK* (see The Amarok Team, 2005 for more information). I aim to either refute or strengthen the hypothesis by finding that at least one navigation aid does 'help' people.

In this experiment, 'help' is defined with the task of finding boundaries in music, similar to that of the previous section; here, however, a strict time limit is imposed for each track. Their performance is determined by comparing their given boundaries to a ground truth formulated by the combination of several persons' boundary delimiting.

Method

Initially, five tracks were selected from a range of music. The tracks were selected to give a good range of different types of music and of different difficulties of problem. Tracks were chosen for their interesting features that would best test the systems. Mood changes (both subtle and blunt), instrument changes, vocal changes and rhythm diversity are among the features I attempted to utilise in order to best examine the systems. Table 5.2 describes the tracks that were chosen.

For this study, I use the ground truth provided by two people who delimited the music in question manually, with no aids. In this early experiment, it was supposed that this would be enough to get a good idea of the outcome. Further experiments use a significantly greater corpus of ground truth. 18 subjects were then each given five trials—one for each of the five tracks. For each track, each of the five analysis techniques given in table 5.3 together with a 'blind' control with no annotation were rotated through. This study therefore assumes that any cross-learning effects between the visualisation algorithms are

Track	Genre
Walk in the Night (*D. McMurray*)	Jazz
Reasonable, consistent beat structure and loudness. Minimally defined transitions.	
Plug In Baby (*Muse*)	Rock
Simple rock ballad with clear verse/chorus structure.	
Goodnight Moon (*Shivaree*)	Pop/Folk
Fluid pop song with little beat structure and hazy verse/chorus structure.	
Prague Radio (*Plaid*)	Electronic/Abstract
Timbrally complex, highly dynamic with multiple moods and well defined beats.	
Keep Hope Alive (*Crystal Method*)	Electronic/Dance
Timbrally simple, well defined beats/transitions, consistently loud, few moods.	

Table 5.2: The five tracks chosen for the user study.

minimal, something of which we cannot necessarily be sure; this is circumvented in later studies.

Giving multiple tracks and algorithms to the same user means that there is no chance of an individual learning the tracks, nor how to use the algorithms during the experiment. It also means that a relatively large amount of data can be collected (five sets of times for each of 18 people). However the issue of learning effects between different algorithms (countered slightly with change of order) goes unanswered. Furthermore, using different people means that there are no matched pairs, thus the results are far more difficult to model statistically. As different people are likely to perform very differently at this task, the variance is likely to be large. This is exacerbated by having different people with a different permutation of tracks/algorithms.

Name	Abbrev.
Spectral Magnitude	SM
Bark Bandwise Magnitude	BBM
Rhythm Magnitude	RM
Bark Bandwise Rhythm Magnitude	BBRM
Novelty	Novelty

Table 5.3: The five visualisation algorithms chosen for the user study.

Each subject was given an initial period of training (some required more than others), until they felt familiar with the controls of the player. Aside from getting to grips with

the "look and feel" of the application, they were given no specific information on the visualisation algorithms. For each trial, the subject was given 60 seconds with the player, incorporating the given analysis technique with the track loaded. They were allowed to utilise the random access of the navigation bar. During this time, they had to determine as many boundary points in the music as they could. They were told that additional boundary points would not be penalised.

Participants had to manually write down the times of the boundaries, thus limiting their time even more. As with other studies, participants came from the undergraduate/postgraduate student body, as well as postdoctoral staff. Following the trials, a short interview was conducted, calling for comments on each of the algorithms.

User Commentary

The overall feeling of those interviewed was that they preferred SM/BBM visualisations over any of the others. Having utilised all five of the methods, many also indicated that they intuitively related the intensity of a point with the loudness at that point. Some went on to suggest that they would then intuitively relate the colour of a point with any instruments playing at that point. Only one candidate suggested that brighter parts in the visualisation might mean increased dynamics and otherwise less constancy.

The standard RM measure as well as the novelty measure were generally disliked. Specific comments were "daunting" and "less predictable". The general feeling was that differential measurement (i.e. novelty) was unsuitable for intuitive learning; people expected to see "chunks" of similar sounding portions of time, rather than specific points at which the music changed. Those who commented felt that the rhythm magnitude measure simply looked overly populated and excessively contrasting, and thus determined it to be too "daunting" for general use.

Cosmetically, almost everyone preferred colour over monochromatic visualisations (one even went so far as to say it was "pretty"). The majority of those suggested that they also found colour to be the better visualisation in respect to usability also. They found it "easier to distinguish", and "more informative". The opinion of colour in the BRM was somewhat more divided. While nobody made it out to be worse that the mono-chrome variant, most favoured the look of the BBM, finding the RM less well defined.

As for usability and comfort, the participants were quite polarised in their opinions as to whether adding colour was more helpful in the experiment. While some decided that it gave them more information and thus was more useful, others felt that the addition of colour increased the learning curve too much. Most went on to suggest that perhaps, given enough time to learn, the colour might eventually be better anyway.

One participant suggested that the three colour components used to create the colour from the low, mid and high portions of the spectrum should be switched. Apparently they

expected bluer hues to relate to "warmer" (i.e. bassier) sounds with redder hues related to harsher, sharper (presumably higher) sounds.

Results

For each of the five trials per person, the times are collated given against the ground truth, creating a single 'score'. A two second linear kernel is used on the distance from the nearest ground truth point to determine the score of any single given point; the final trial score is the sum of all such points. Formally, to score a point $\mathbf{t}_i$, given the closest point to that time in all of the collated ground truth for that track $\mathbf{r}$:

$$s(\mathbf{t},\mathbf{r}) = \sum_{\mathbf{t}} \begin{cases} 3 - |t_i - r_j|, & |t_i - r_j| \leq 2 \\ 0, & \text{otherwise} \end{cases}, \quad j = \arg\min_j(|r_j - t_i|) \tag{5.1}$$

The (perhaps slightly lenient) two seconds of allowable deviation was chosen due to the nature of the experiment. Not only were the participants heavily rushed, but the player's time reading is accurate only to within ± 0.5 seconds, and they had to write the time as well as inspect it and listen to the music. Once collated, the distributions over the genres may be plotted. Figure 5.3(a) shows the distributions as a box-and-whisker plot. The box represents the lower and upper quartiles, the inner line is the median and the outer lines are the minimum and maximum[1] each genre. It is clear to see that BSM and SM visualisation algorithms appear to have helped more than the others. I make a standard analysis of variance test on the findings to check if they can be analysed further. This is presented in table 5.4.

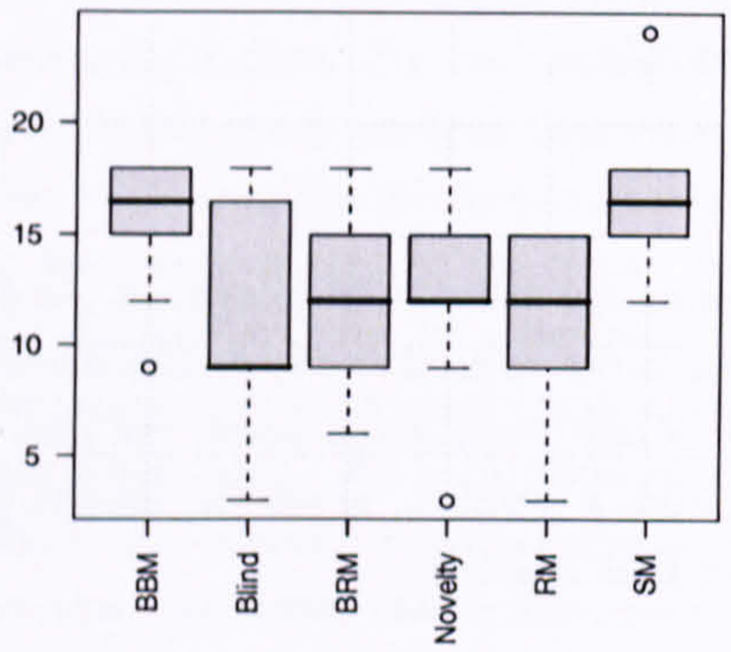
	P-value
Algorithm	0.0006228
Person	0.0019703
Track	0.1041948

Table 5.4: Analysis of variance of the factors of the experiment.

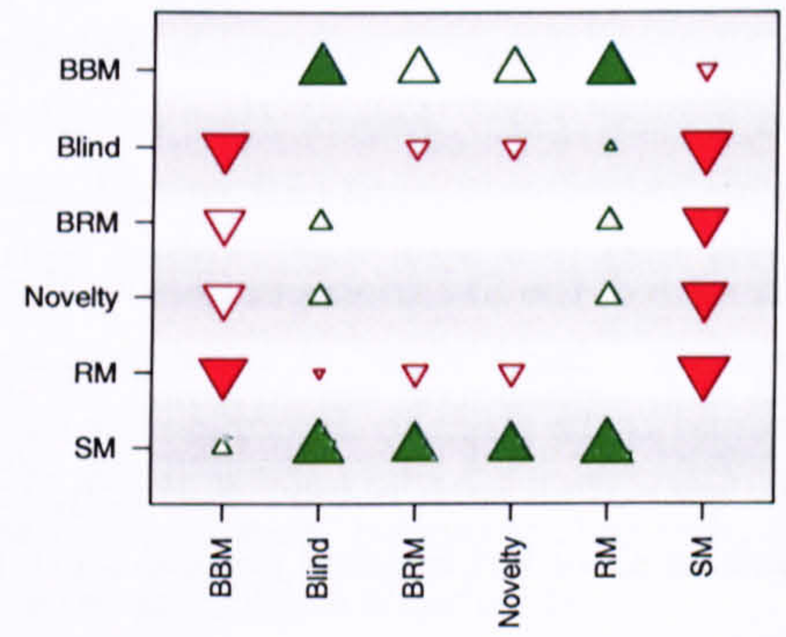
This shows that the algorithm is highly significant at a level of $> 99\%$, and, to a lesser degree, the individual in question (less fortunately). Performing a Tukey HSD pairwise comparison gives the comparison matrix of figure 5.3(b). Two statements may be made immediately from the results concerning the tracks:

- SM and BSM are better than RM, Blind.
- SM is better than Novelty, BRM also.

[1] or the upper quartile plus one-and-a half the interquartile range



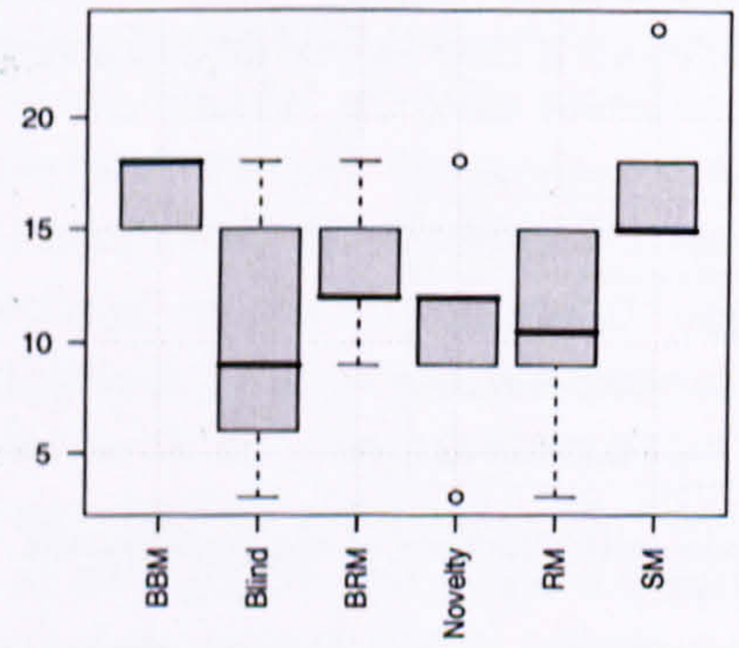
(a) Box-and-whisker plot of distributions of genre scores.



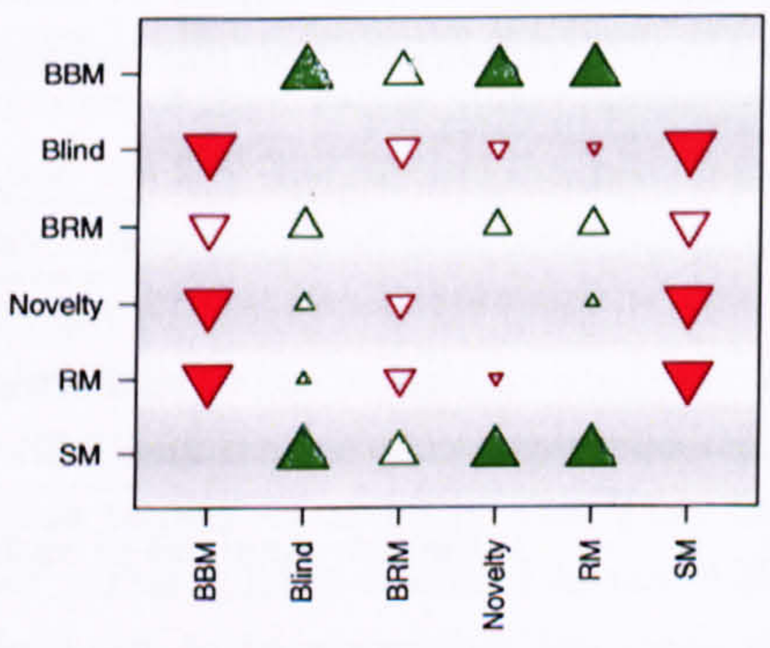
(b) Tukey HSD pairwise mean score difference matrix. Filled triangles are significant to 95%.

Figure 5.3: The box plot and score difference matrix for all music over the experiment.

If the results data is limited to only that of the electronic music, there is still reasonable statistical significance from the ANOVA test (p=0.003898 for algorithms). This is not the case for pop which has p=0.1924. The distributions may again be graphed as box plots and the comparison matrix; this is shown in figure 5.4.



(a) Box-and-whisker plot of distributions of genre scores.



(b) Tukey HSD pairwise mean score difference matrix. Filled triangles are significant to 95%.

Figure 5.4: The box plot and score difference matrix for electronic music over the experiment.

The results of the electronic music suggests similar findings to the others, with the extra finding of BBM being significantly better than Novelty, and SM being not so significantly better than BRM.

Conclusion

I have shown that basic music visualisations can help people navigate around a track for basic boundary-finding tasks when given a tight time limit. In particular, the two spectral-magnitude based visualisations significantly outperformed all others. There is evidence to suggest that the BBM may be more helpful for the timbrally-clearer electronic music than for e.g. rock and pop.

I have further presented evidence suggesting that people may themselves find such aids both aesthetically pleasing (especially in the case of BBM) and helpful. These are crucial points for a system that is to succeed as a popular navigation aid.

5.3.3 CPT Projection

This experiment is to determine if, under a similar test framework to the previous experiment, we can show any differences in the performance of the methods proposed in chapter 4. In particular, I would like to test the hypothesis that the SOM-based navigation aids do actually provide help over navigation bars without such visuals; I construct a null hypothesis that they do not and attempt to refute it. Furthermore, I wish to find evidence to refute the hypothesis that timbrely-well-defined electronic music is more likely to be susceptible for help through visual aid than other types of music. One other null-hypothesis that I wish to test is that the direct loudness visualisation is at least as helpful as SOM visualisations (i.e. I believe that the SOM visualisations are better).

The music used for this experiment is different to that of the previous; firstly to test the latter hypothesis, I wish to use as much electronic music as possible to minimise the possibility of the tested tracks being grossly unrepresentative. Secondly, I wish to have a slightly narrower distribution, refocusing on three key genres; classical, rock and electronic. Thirdly, since these tracks are a subset of those used for the previous study on boundary agreement, there is a good ground truth to work with.

Method

The experimental method is generally similar to that of the previous experiment. As mentioned before, six tracks are used, detailed in table 5.5. 24 participants were used in this experiment; each given all six of the tracks in a randomised order with the six visualisation agorithms. The order of the algorithms was changed for each participant, to reduce the bias from learning. The visualisation algorithms used were the four from chapter 4, together with the spectral magnitude (SM) measure, which is akin to the front-runners from the previous experiment, and a 'blind' algorithm (i.e. no visualisation at all).

The advantages and disadvantages with using this experiment outline are highlighted

Track	Genre
Country House (*Blur*)	Pop/Rock
Simple rock ballad with clear verse/chorus structure.	
Paranoid Android (*Radiohead*)	Pop/Rock
Log, progressive rock with multiple instrumental sections. Well defined transitions.	
Minuet in A (*Luigi Boccherini*)	Classical
Timbrely narrow strings-based track. Repeating and varying figures arise in unclear transitions.	
Barber's Adagio for Strings (*T. G. Albinoni*)	Classical
Long strings/organ track with pronounced dynamics. Several timbrelly-contrasting regions give clear transitions.	
Lebanese Blonde (*Thievery Corporation*)	Alternative Electronica
Fluid rhythm-dominated downtempo track with some timbre-contrast to define boundaries well.	
They're Hanging Me Tonight (*Red Snapper*)	Alternative Electronica
Log, complex and dynamic track with many interwoven timbres.	

Table 5.5: The six tracks chosen for this user study.

in the previous chapter so, I will not repeat them here. The only significant change to the method was to implement a feature in the software allowing participants to press the space bar on the keyboard to denote a boundary, rather than having to break focus and write it manually. They were allowed to delete the boundaries after pressing the space bar by simply right-clicking on the boundary represented by a small rectangle in the bar.

Results

The results are collated in a similar way to those of the previous experiment; scores are calculated with the same two-second linear kernel on the closest matching ground truth for each set of ground truths (10 in total for this experiment). The box-and-whisker plots depict the distributions over all genres in figure 5.5(a). There is an approximately normal distribution of the points, with the trials of the SOM-based aids having higher but broader scores on average. A standard ANOVA test is used to attempt to refute the null hypothesis on the data that scores across all algorithms are on average the same. The test reports a significantly low probability of any of the factors leading to a consistent mean, leading us to refute it:

The post-hoc Tukey HSD test is conducted to check for any significant comparative differences in the data; the results of this are depicted in figure 5.5(b). For this, we

	P-value
Algorithm	0.002400
Person	0.001736
Track	1.363×10^{-11}

can see that scores for each of the trials of SOM-based navigation aids are significantly higher than those with no such help. There was no significant difference between any other algorithms, although the means (i.e. size of triangles) are suggestive that the SOM methods do typically result in a higher score then either of the PCA and SM algorithms.

This is continued by splitting the data into three genre-sets. Figures 5.5(c), 5.5(d) and 5.5(e) show the distributions of data for the classical, rock and electronic music respectively. Using a similar method to that of the above, the probability of the differences in score means being significant may be found with an analysis of variance test:

Degrees of Freedom		P-value		
		Classical	Rock	Electronica
Algorithm	5	0.1256	0.776173	0.0007852
Person	20	0.4007	0.002981	0.0052872
Track	1	0.2406	0.473618	0.7357833

The only genre with a significant difference in means is that of the electronica, which has a particularly high probability of significance ($> 99.9\%$[2]). Post-hoc tests on the electronica corpus of trials give us figure 5.5(f).

One may see from the post-hoc tests that restricting the corpus to only electronica results in exactly the same significant conclusions as those of all genres.

User Feedback

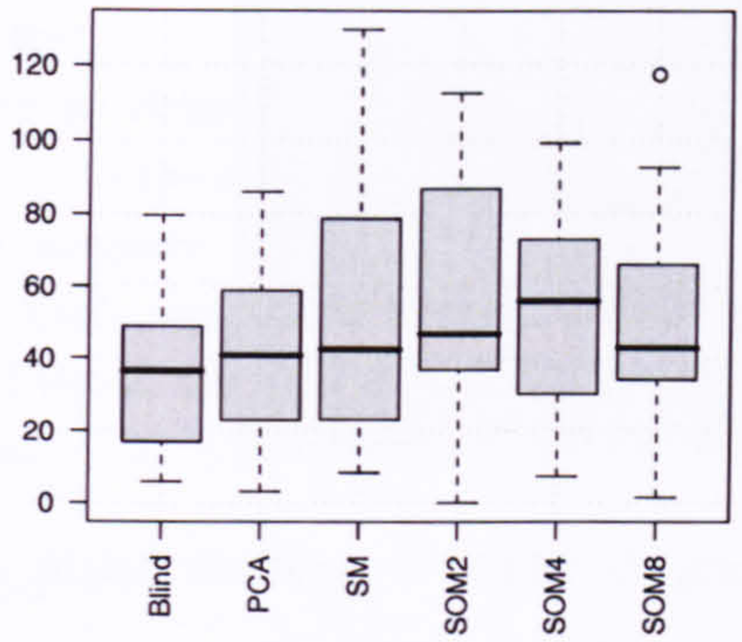
Short informal interviews were conducted following the trials, to solicit any major opinions the participant had after their small period of usage. Few found the visuals detrimental to the experience of audio software; those that did found no utility or aesthetic value in their combination of tracks and visuals. Of the visual aids, the basic spectral magnitude aid was invariably placed last; opinions over the chromatic plane trajectory methods were quite diverse, but on the whole the SOM4 was the most popular, with a common criticism of the SOM2 being too little information, and PCA of being too difficult to read though its complexity. As the participants had so little training time, these opinions are indicative of only the initial reaction.

[2]This P value is only accurate if this subset is considered as independent from the other two. I continue anyway since it is small and the post-hoc analysis gives only supportive results.

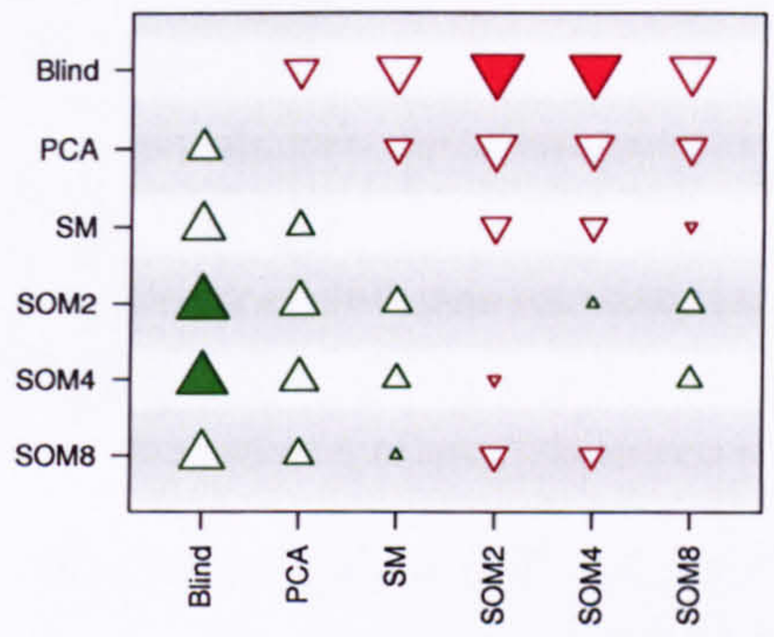
Conclusions

I have shown statistically significant evidence to support the hypothesis that SOM-based navigation aids help in the task of boundary selection under a time-limit. Although I have no statistically significant evidence to refute the claim that more basic methods such as spectral magnitude are at least as good as the SOM-based methods I have, at least, very suggestive circumstantial evidence. Furthermore, the experiment suggested that the PCA-based aid may not perform as well as those of the SOM.

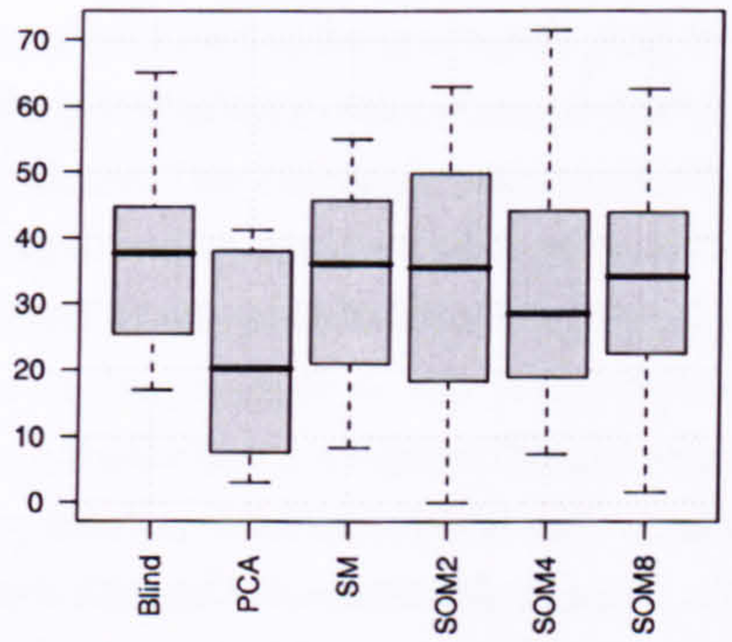
In this task, there is no significant nor apparent difference in the performance of each of the SOM variants. In terms of feedback however, users typically preferred the SOM4 over all others.



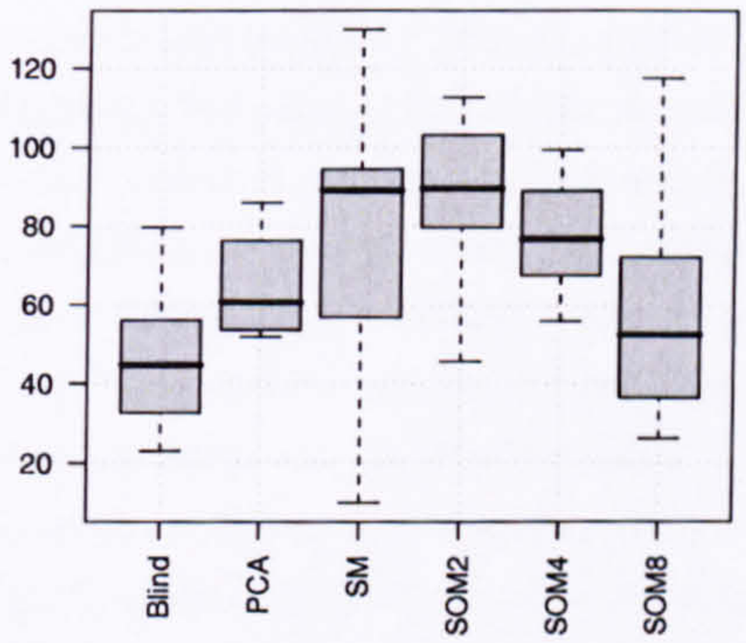
(a) Box-and-whisker plot of all scores.



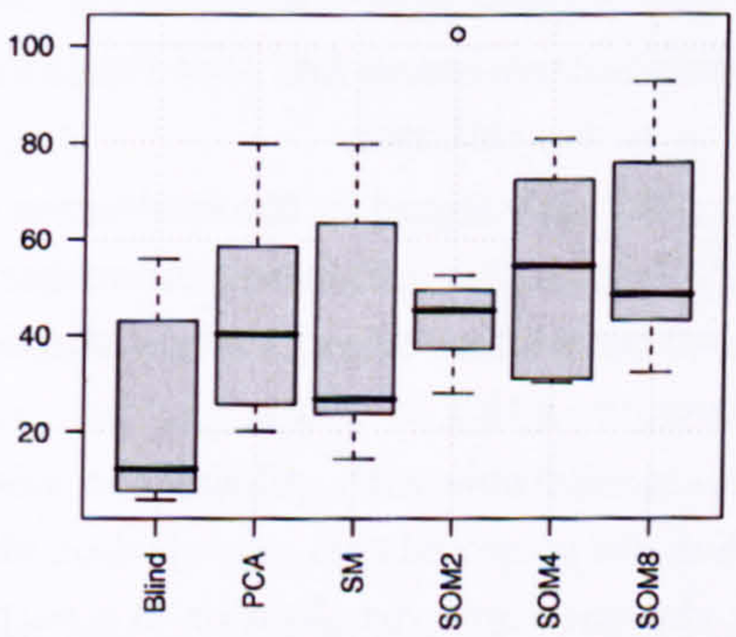
(b) Tukey HSD pairwise mean score diff. matrix.



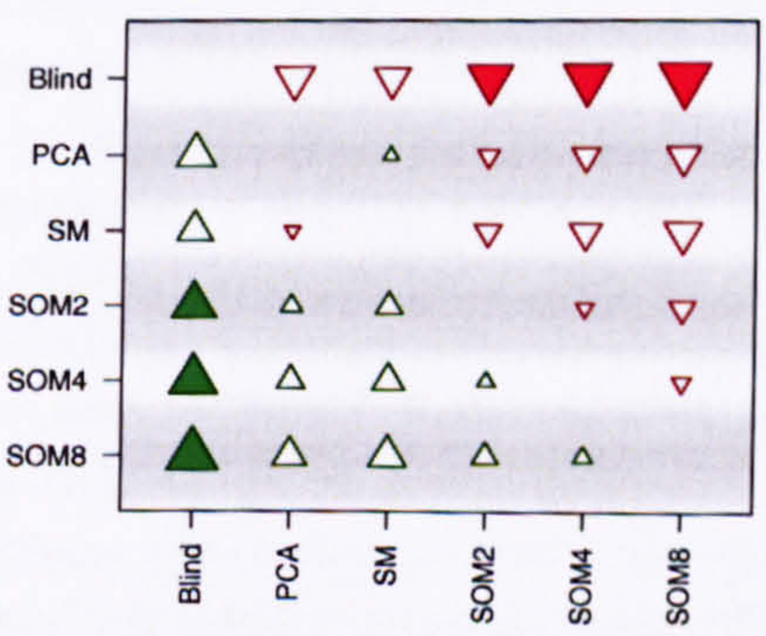
(c) Box-and-whisker plot of classical scores.



(d) Box-and-whisker plot of rock scores.



(e) Box-and-whisker plot of electronica scores.



(f) Pairwise mean electronic score diff. matrix.

Figure 5.5: The experiment and genre-wise box plots and score difference matrix for electronica and in general.

Figure 5.6: The amaroK music player with the BLSM & Loudness Height visualisation operational.

5.3.4 Learning Rate

The final boundary-times based experiment was one designed to test the hypotheses concerning the learning curve of navigational aids. In particular, I collect empirical evidence on how well people perform the time-limited boundary-finding task, over the course of a number of weeks' usage. A selection of navigation aids proposed throughout the present work were tested in three groups; no visual aid, basic aids of the variety tested so far and aids that varied the height as a function of loudness.

I expect little or no learning curve for the 'blind' navigation aid with no visualisation, since participants should be typically comfortable with the concept of random-access navigation and are thus unlikely to improve in their performance of the task over the course of three weeks. As such, I expect this curve to denote a control that shows the degree of task-learning that happens with increased familiarity with the media. With no reason to suspect otherwise, I imagine the other two methods to yield similar curves, thus hypothesising that the learning curve for the algorithms is either so fast that the initial period of training given is enough, or so shallow as to be insignificant.

The aids that vary the height of the graphic with regard to the loudness are essentially an integration of the classical 'waveform' visualisation (discussed in section 3.2.1) with the proposed visualisation methods. They form a silhouette of the visualisation with the loudness waveform. Figure 5.6 gives an example of this in use.

I expect users to be on the whole more comfortable with utilising the height loudness aspects of the visualisation more readily than the proposed colour-projection visualisations. This is because, as mentioned previously, the waveform visualisation is a particularly traditional one, being found in all manner of professional and semi-professional software[3]. As such, if there is a detectable learning curve, I expect the curve to be less pronounced (shallower, since part of the learning should already be complete). I also expect performance to

[3] an example being the open-source Audacity audio editor

be generally better with the loudness silhouette, since more information is made available to the user.

Four participants were used in this study; although this may seem a low number compared to the other studies, it is important to note that for testing the hypotheses this proved adequate. Each participant had a total of nine sessions over the course of four weeks. Each session comprised nine trials on varying genres of music, and thus a total of 324 individual trial scores were collected. Participants did not use the software between sessions.

Method

Prior to starting the first session, participants were given around 10 minutes training on the software with each visualisation used in the trials (but not the music). A total of eight visualisation algorithms were used in addition to the blank navigation bar ('Blind'). Four included the loudness silhouette, four did not. The four chosen were SOM2, SOM4, BMFM and BLMS, and were assumed a diverse but reasonably performing selection from the proposals. The different methods of visualisation were chosen to help prevent learning effects from multiple occurrences of the same combination of track/trial, and to maximise the number of trials per session in order to maximise throughput of data.

In a similar manner to the previous two experiments, this does not take into account learning effects through trials. To minimise this effect, the order of music was randomised for every session conducted. As before, participants were given only 60 seconds per trial, and had to identify boundary points. Unlike in previous experiments, participants were given the music prior to the study. This was an attempt to mitigate the effects of learning the music through the course of the study. For this experiment, the time read-out on the software was disabled to prevent users from learning the individual boundary point times in the track. The space-bar functionality described in the previous section made it redundant.

Scores were made in exactly the same way as those of the previous two studies. The music tracks used were from the study of boundary agreement, detailed in section 5.3.1; three each of classical, rock and alternative electronica. The ground truth found from that study is used here also.

Results

Once the scores are collated, an analysis of variance on the results is conducted to check the probabilities of significance of difference of means, which, as table 5.3.4 shows, is reasonably certain.

Following this, the learning curve proper is plotted. Figure 5.7 shows this as each trial's point together with the approximate regression lines to show the trend. One may

	P-value
Run	1.019×10^{-05}
Algorithm	1.622×10^{-05}
Person	9.731×10^{-07}
Track	≈ 0

see that over the period of initial learning (until around day 13) each of the three methods has largely parallel learning curves, suggesting this is the time required for participants to learn how to do the tasks generally (in terms of becoming efficient at random access as well as learning the music).

In the latter half of the experiment, however, the curves split, with the 'blind' and 'both' (proposed visualisations with waveform) level ling off and the 'colour' (i.e. proposed visualisations) alone continuing to rise to meet the level of 'both'.

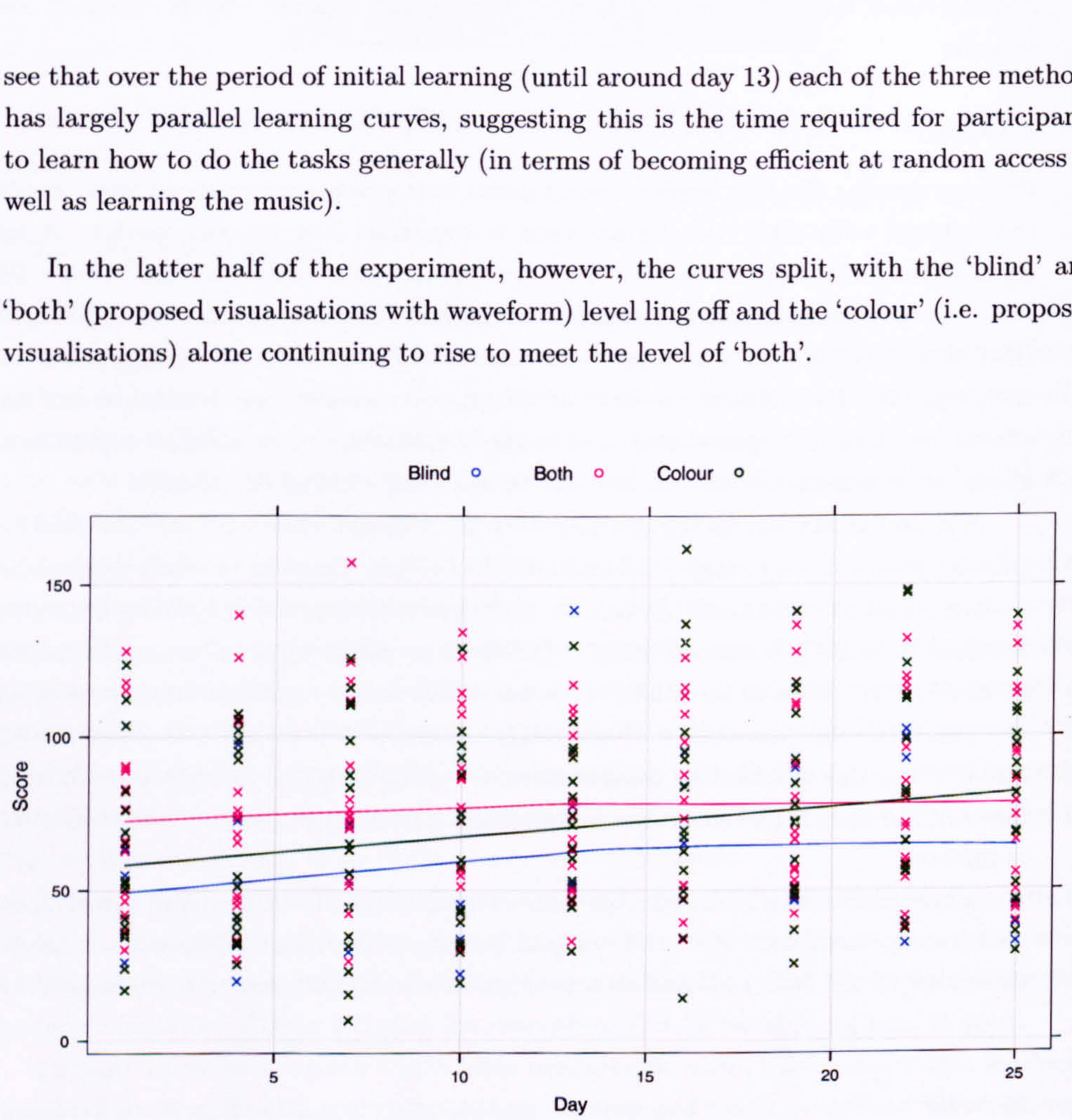


Figure 5.7: Scores of every one of the 324 trials conducted. The x-axis denotes the point throughout the study that the trial was conducted. The different navigation aids are denoted by different colours.

User Feedback

Interviews were conducted following the trials; participants were asked how useful they considered each of the visualisations, what uses they could see for it, and whether they thought these would be an overall improvement or detrimental to the interface. Candidates generally considered the visualisation to be less useful for classical than for the other genres. In terms of learning, they generally considered that the curve ended after half the number of sessions, also suggesting that their increasing knowledge of the song played a big role.

The loudness 'silhouette' was generally accepted as being useful initially, though again less so for the classical music. The high degree of changeability was also commented on, suggesting that this 'noise' was too dominant. BLMS was generally considered one of the best, although like the loudness silhouette, the high frequency variation was pointed out as a defect.

The 2x2 SOM-based visualisation received both considerable praise and criticism. It was considered one of the better representations of the classical tracks, and was typically considered to "do the right thing" in terms of representing "tonal variations". However precision, aesthetics and accuracy were issues. Considered with the loudness silhouette, it overcame many of these issues and was typically considered very useful. In particular having the duality of colour for tone and height for loudness was considered advantageous.

Notably, there was disagreement as to the efficacy of the SOM4-based aid. Some considered it a "more cluttered version" of the SOM2-aid, despite considerable improvement from the silhouette. Others considered the SOM4 a significant improvement, suggesting that the previously problematic greater levels of colour were a specific advantage. There was similar disparity in the opinions as to which was more aesthetically pleasing of the two, though both were generally considered the two 'best looking', and there was little doubt over their improvement by the silhouette.

The MFCC was typically considered the weakest visualisation, being a particularly "unintuitive representation". One participant suggested he was less likely to use it due to the excess "clutter and general ugliness".

In terms of the usefulness of the aids, most were generally impressed; as one said it is "easier to jump around to a certain bit". One thought it would work well as a good iconic representation of a track on a newly bought album; perhaps as a visual thumbnail. Exactly when it would be useful was particularly polarised. Some felt that it would be most useful before a track was heard, and thus in terms of 'jisting' over 'bookmarking'. Others were of the opposite conviction, suggesting that it would be largely useless initially, but be good for getting to know a piece having heard it once already, and further increase in usefulness with knowledge of the piece.

Conclusion

In general, the original hypothesis is supported. Each of the curves roughly resemble that of the 'blind' curve. The curve 'both', denoting the combined waveform and colour-projection visualisation, gives a similar but shallower curve. Interestingly, the 'colour' actually meets 'both' during the time frame chosen for this experiment. I take this as suggestive that any extra information provided by 'both' becomes redundant after a user is familiar enough with the colour-projection, and that this level of familiarity can be attained within three weeks of intermittent usage.

5.4 General Task Tests

We arrive at the final and more significant experiment conducted as part of the present work. In all of the previous studies, the utility of a navigation aid was judged by how much it improved a user's ability to determine boundary points that best agree with some ground truth. While this is an interesting use that no doubt requires navigation to succeed, it is a slightly indirect method of measuring, whose results, as we have seen, tend to be quite noisy.

This study was conducted under a different design; an apparent 'score' will not be measured within a set time limit, but rather the time taken for a user to complete a particular task is measured. These tasks not only require use of navigation, but are representative of the various use cases for navigation, and are the same tasks used for the reference in chapter 2.

I have shown how the proposed visualisations, and in particular the SOM4 and BLMS, represent the music visually. I have further provided evidence that users are able to benefit in terms of their navigation from having a visualisation.

I therefore submit the following proposals:

- To be a useful navigation aid, the visualisation need not provide absolute information regarding the content but rather relative information.
- A single-level visual segmentation of the track will not aid so much as one focusing instead on relative-distance information.
- A superior navigation aid will reduce the amount of time to complete a task by increasing the accuracy of individual navigation seek operations, and reducing the amount required.

As such, I predict that computer literate people will be able to complete a set of high-level music information retrieval tasks faster when given an audio-derived visualisation as

a navigation aid than those not. This assumes the users are given a short self-training period, and that similar accuracy requirements are placed upon the answers given.

5.4.1 Method

50 participants are taken and split them into six groups of eight people and one pair. Each participant is given a set of instructions and left alone. This document covers basic disclaimers (withdrawal, anonymity) and notes that the participants are being timed, but that it is the system being tested, not them. It relates the usage of the audio player application, in particular that of the timeline bar.

When ready, each person is allowed five minutes of training time to familiarise themselves with the navigation system on a track unrelated to the tasks. This is timed automatically by the system; they are left on their own throughout the session. They are given a ten second warning before the first question is given. After completing the tasks they fill out a questionnaire concerning themselves and their thoughts on the software before leaving.

Each group of eight people is associated with one of six conditions; the pair is associated with a seventh condition. Each condition generally relates to a specific visualisation given as a navigation aid. Aside from this unique condition, each participant is subjected to exactly the same script, detailed below. Throughout the trial, neither the participant nor the supervisor (the question script) know of which condition they are under and the experimenter is not present, thereby giving, in this context, a triple-blind experiment. The visualisations associated with the conditions are given in table 5.6:

Condition Name	Abbrev.
Sequential; no random access	(Seq)
Blank	Blind
Bandwise Loudness Magnitude Smoothed	BLMS
4x4 SOM Chromatic Plane Trajectory	SOM4
Loudness Waveform	LMS*
Sandler/Levey Segmentation	Seg
As BLMS but for a different track	Bad

Table 5.6: The seven conditions in the user study.

A few conditions need a little more explanation; '(Seq)' refers to a baseline made where users were not only given no visualisation, but did not use the random-access navigation facilities of the software. They listened to the music without seeking, until they decided upon the answer for the task and then moved on to the next. This, therefore, represents a sensible lower-bound for the average duration of tasks. Only two people were subjected to

this condition due to the lack of variance in the possible approaches that could be taken.

Blank or 'blind' is simply where no visualisation is given, as with the other studies. Loudness Waveform 'LMS*' is where an untextured silhouette of the loudness is depicted. This was used as a good benchmark to measure the performance of our methods. I expect users to perform well with this method generally, since it is the traditional and widely used form of representing audio, and in a fairly acute test such as this study, I expect the most obvious representation to fair the best.

Sandler/Levy Segmentation or 'Seg', is a visualisation derived from the segmentation method recently reported by Levy et al. (2006). It can be considered a state-of-the-art method of performing a high-level musical feature-based segmentation. The basic segmentation information comes in the form of contiguous chunks, with start/end times and an arbitrary index according to its 'type'. With this data, a visualisation is constructed in the same manner as is exemplified in their publication, distributing hues to 'types' evenly and without preference to their content.

Finally, the 'Bad' visualisation is one which misinforms and misleads the user deliberately. An unrelated track's BLMS visualisation is provided. For the initial minutes of training, users received a valid BLMS visualisation.

The tasks take place at a normal computer terminal. Questions appear at a console, and users must navigate through the track (played automatically) to determine the answer. The entire navigation trace, including the duration between and length of seeks is recorded silently. Once an answer has been entered, the duration taken is recorded silently and the next question is given. Answers are not checked for accuracy at this stage.

5.4.2 Results

The results are inspected in three sections; firstly, the mean time taken which is the primary objective, quantitative metric for evaluating the performance of the visualisations. Secondly, the seeking behaviour is analysed and finally the questionnaire is discussed.

Importantly, before any results are analysed we discarded all incorrect answers. Although the analysis could be conducted on the general behaviour, I consider using only the data for correct answers far more telling of 'proper' usage. As an aside, I informally analysed the incorrect answers and found no significant differences between that corpus and this; even the differences in proportion of correct to incorrect were statistically insignificant between conditions.

Time Taken

Having collated the durations taken for each of the 26 tasks by each of our 50 participants, the distributions of time taken for each of the conditions is depicted (see figure 5.8(a)).

Seeing the relatively well-formed distributions, a three-way analysis of variance is conducted to check for significant differences in the means; this is given in table 5.7.

	P-value
Algorithm	4.309×10^{-12}
Task	2.2×10^{-16}
Person	1.065×10^{-15}

Table 5.7: ANOVA of task based study's collated durations.

Not only the algorithm is highly significant, but both of the other dimensions of the data (participant and task). This suggests that it may be difficult to make all inferences from our experiment. Nevertheless, a set of visualisations for the data may now be constructed, including a conditionwise matrix of comparison with the Tukey HSD. Figure 5.8(b) shows these visualisations.

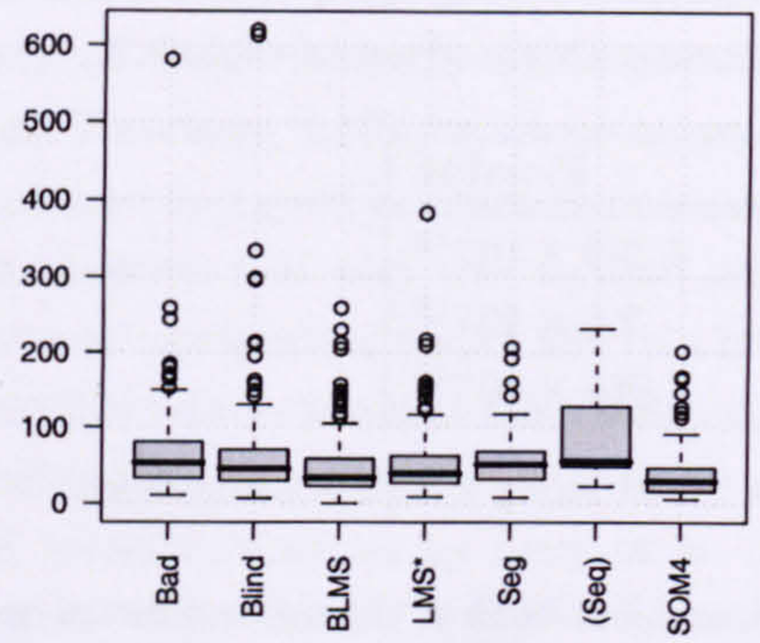
The results show a statistically significant improvement (i.e. reduction) in time taken to complete a task, between the direct-visualisation of the loudness waveform and the SOM4 visualisation. Indeed the SOM4 clearly outperforms all visualisations. The only one that is not statistically significant is BLMS, although the suggestion is there nonetheless. The results are further suggestive of the BLMS visualisation being better than the loudness waveform, but, as figure 5.8(c) shows, it is not quite at the 95% confidence level. The loudness waveform outperforms the blind at 95% confidence levels, though the margin is relatively small with a mean reduction in time of around 17%.

One may see from figure 5.8(d) that is takes on average around 40% longer to get a correct answer with the LMS* condition than the SOM4, and an extra 65% with no visualisation at all.

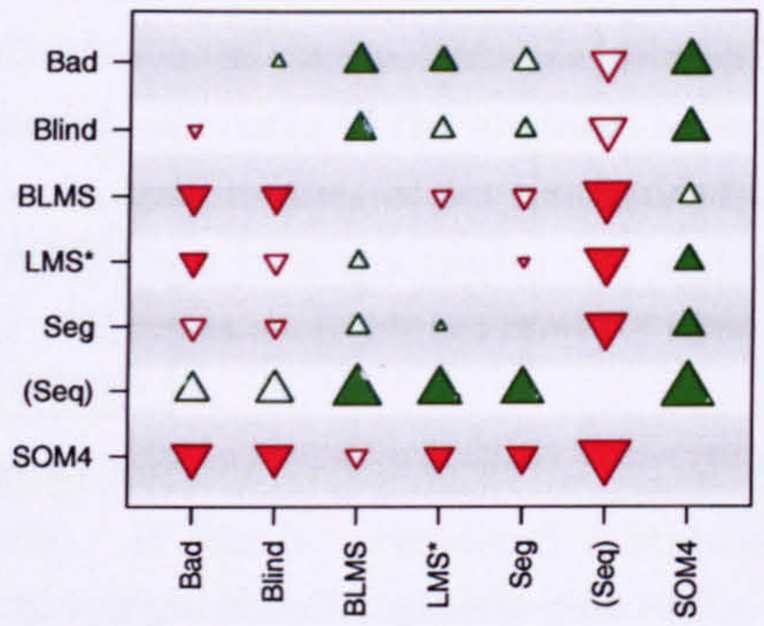
Seek Behaviour

With the seek behaviour, two metrics are used; the number of individual seeks (i.e. clicks on the navigation bar) used to determine a correct answer, and the mean distance travelled in a seek (i.e. the difference between the playback positions before and after the seek). The theory is that a more expressive visualisation (i.e. one that conveys the information it contains better), should effect an increase in the average length of seeks as users learn to utilise the graphic to direct larger-scale navigation on faith. A generally more accurate visualisation (i.e. one that corresponds well to the music) should effect a decrease in total number of seeks required, as they navigate to the desired point in the music without blindly skipping through the track as we saw in chapter 2.

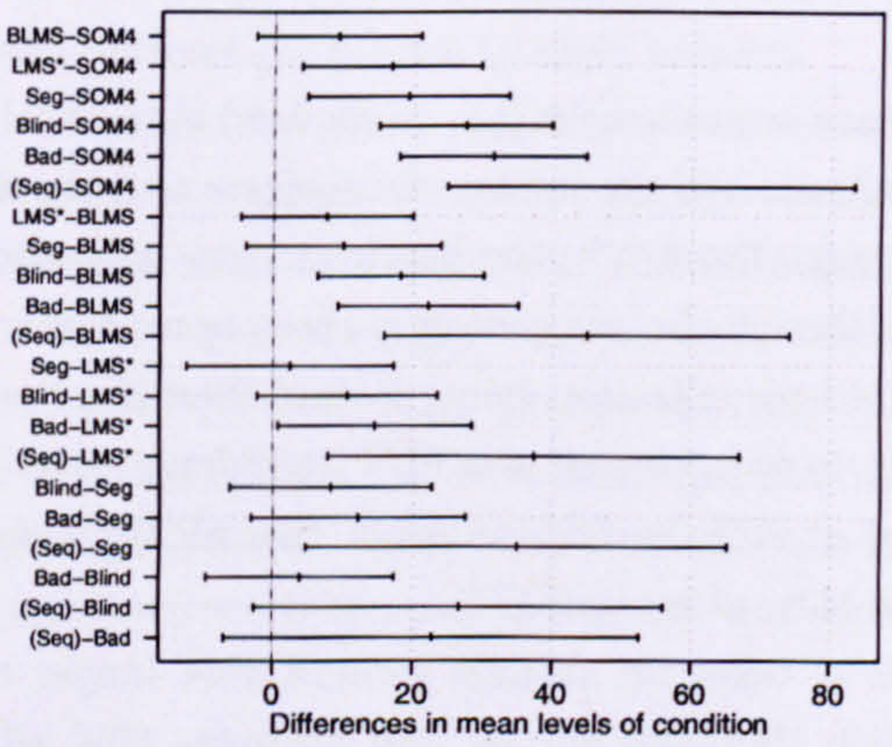
As with the durations after plotting the graph, a three-way ANOVA is used to check for relative amounts of variance in the results, and find the means are very significant



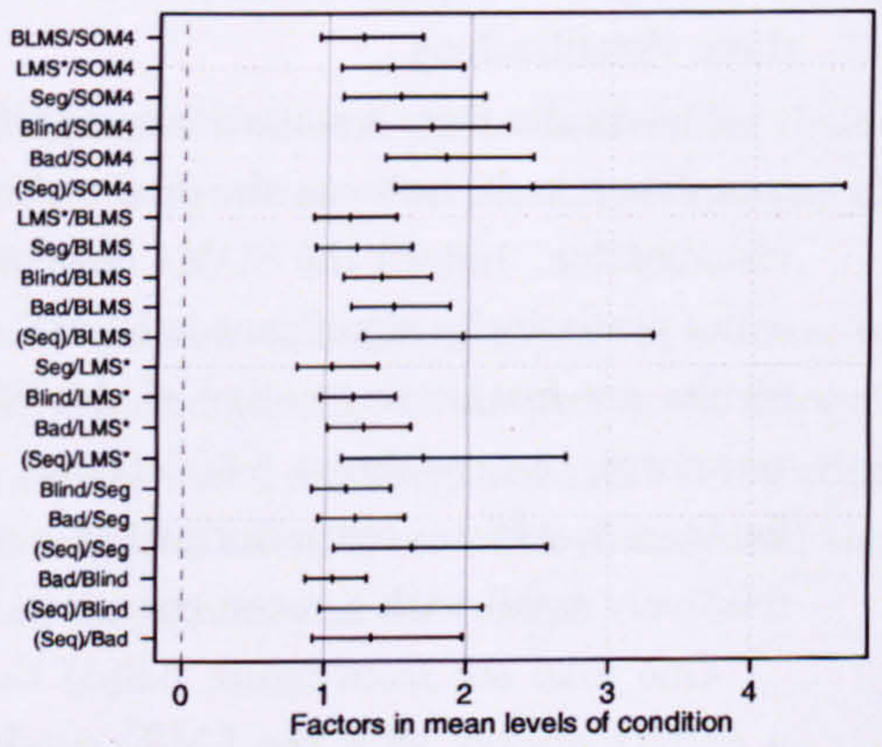
(a) Box-and-whisker plot of all durations.



(b) Tukey HSD pairwise mean duration diff. matrix.



(c) Mean difference 95% confidence intervals.



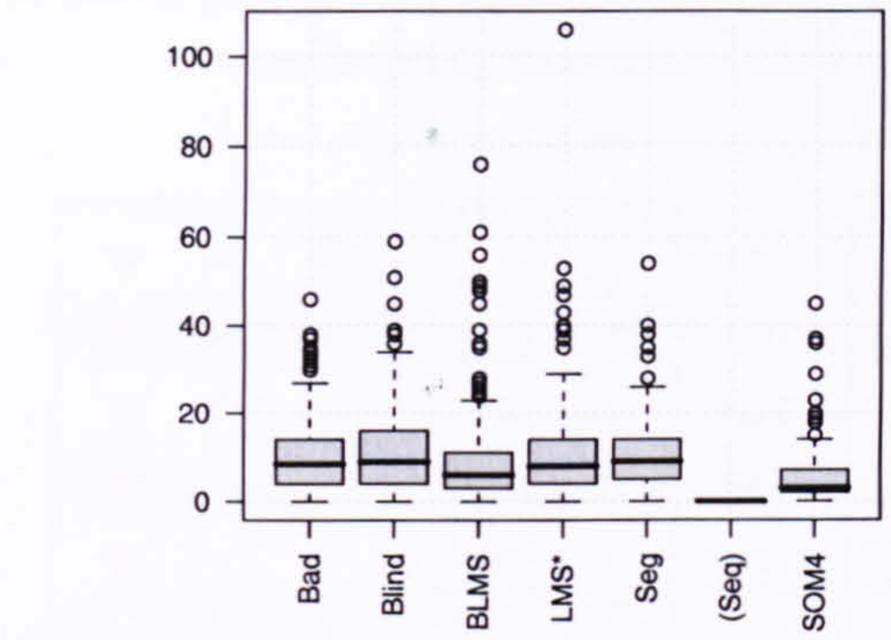
(d) Mean factor 95% confidence intervals.

Figure 5.8: The collated duration results of the task-based study.

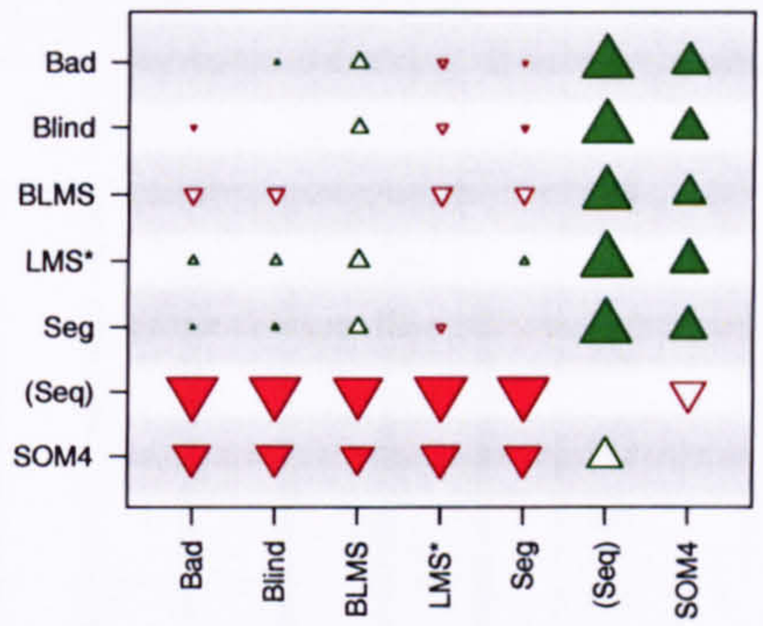
indeed. The results are plotted in figures 5.9 and 5.10.

	P-value	
	Seeks	**Seek Lengths**
Algorithm	2.612×10^{-15}	2.779×10^{-16}
Task	2.2×10^{-16}	2.2×10^{-16}
Person	5.077×10^{-06}	6.210×10^{-09}

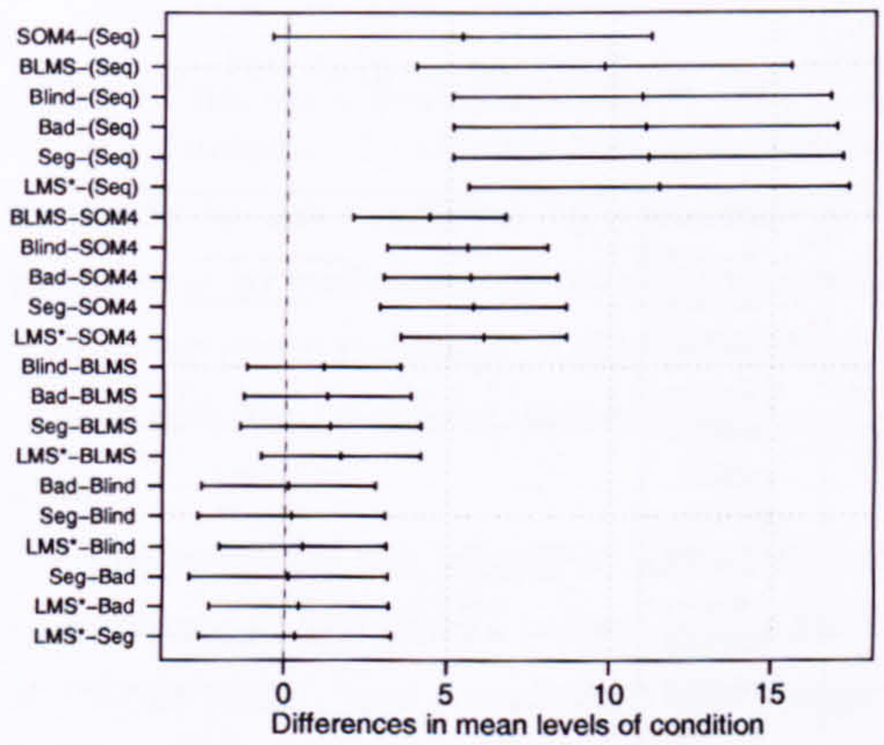
Using the Tukey HSD post-hoc test, results indicate that we can be 95% certain that LMS* requires, on average, more than twice as many seek operations as the SOM4 for determining correct answers. The SOM4 generally requires significantly fewer seeks for determining correct answers than any other of the tested navigation aids, including the



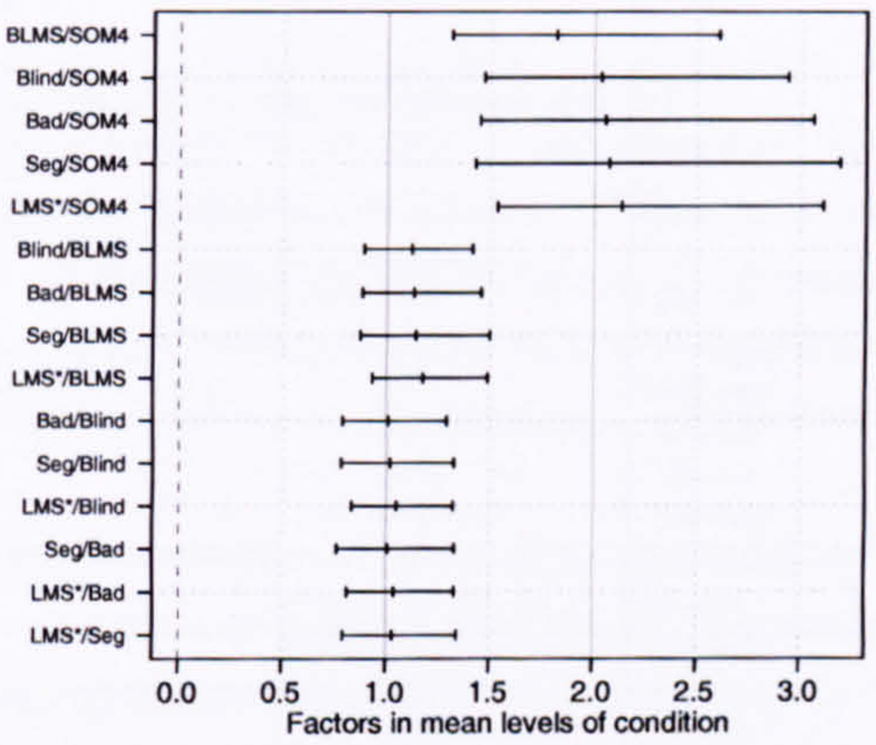
(a) Box-and-whisker plot of all seek counts.



(b) Pairwise mean seek count diff. matrix.



(c) Mean difference 95% confidence intervals.



(d) Mean factor 95% confidence intervals.

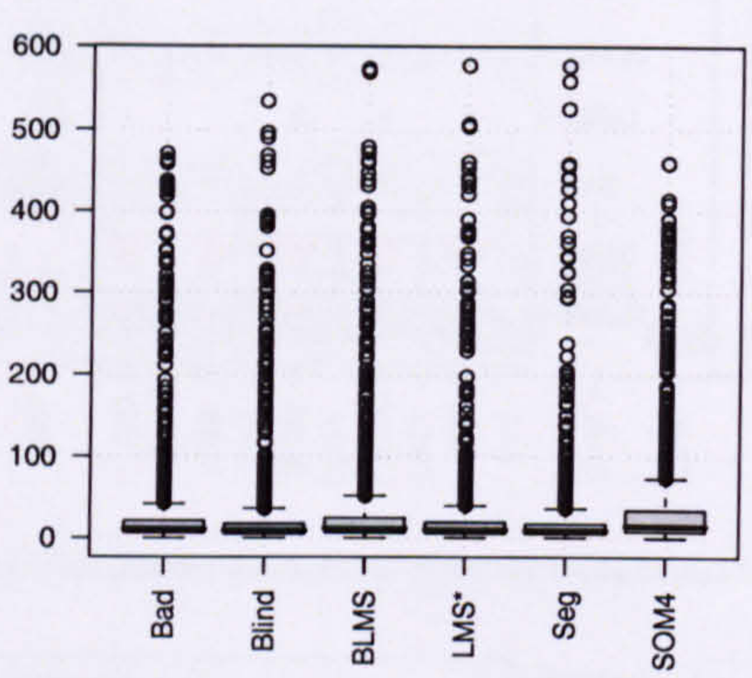
Figure 5.9: The collated seek count results of the task-based study.

BLMS, which itself requires fewer seeks than the segmentation-based visualisation.

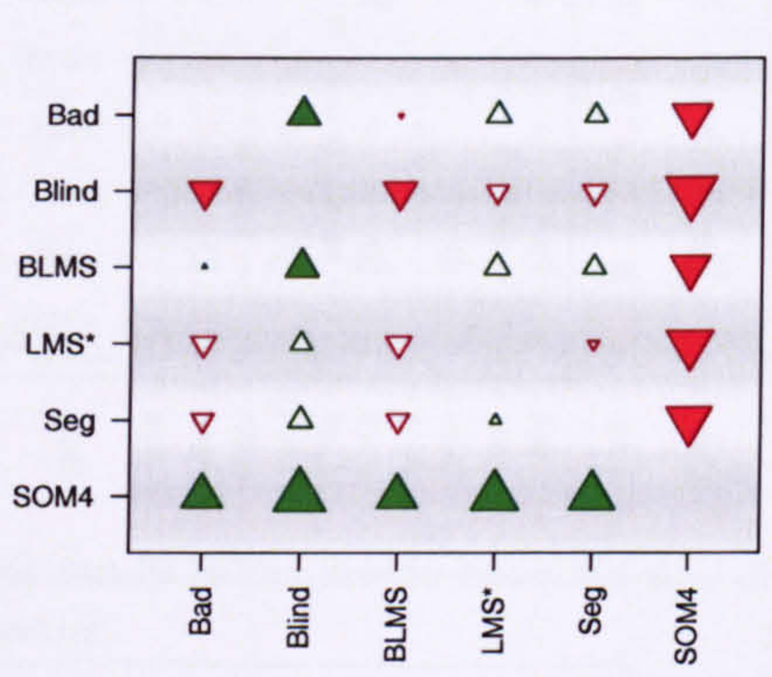
In terms of the lengths of seek operation, a distribution with an apparently long tail can be seen. Further investigation shows that the apparently large number of samples falling outside the plot in fact constitute very few ($\lesssim 5\%$) of the total number, and I therefore consider the distributions to be approximately normal. The '(Seq)' navigation system is ignored here, since there are no seek operations to analyse. One may see quite clearly that the SOM4 navigation aid has a pronounced effect on the mean length of seeks; significantly increasing them over all others.

From these results, one may say with 95% confidence that for determining a correct answer, the SOM4 navigation aid leads users to seek around $85 \pm 40\%$ further than with no visual navigation aid, and around $75 \pm 40\%$ than with only the traditional waveform visualisation. Perhaps more interesting is how both the 'bad' BLMS navigation aid and

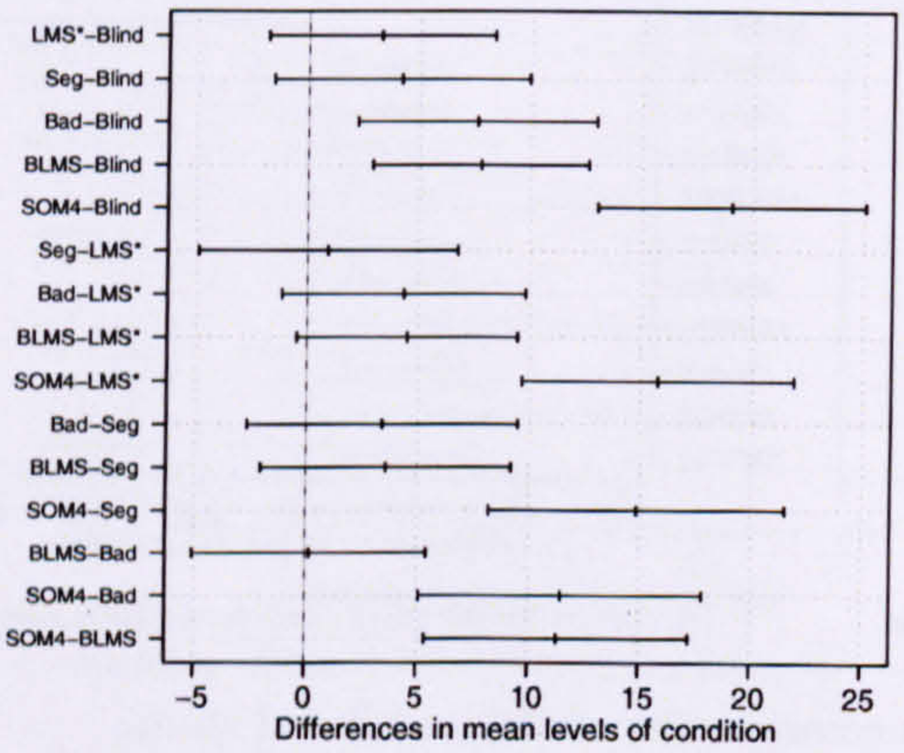
the legitimate BLMS aid resulted in significantly larger jumps than having no aid at all. Indeed, according to the confidence intervals they have near identical means.



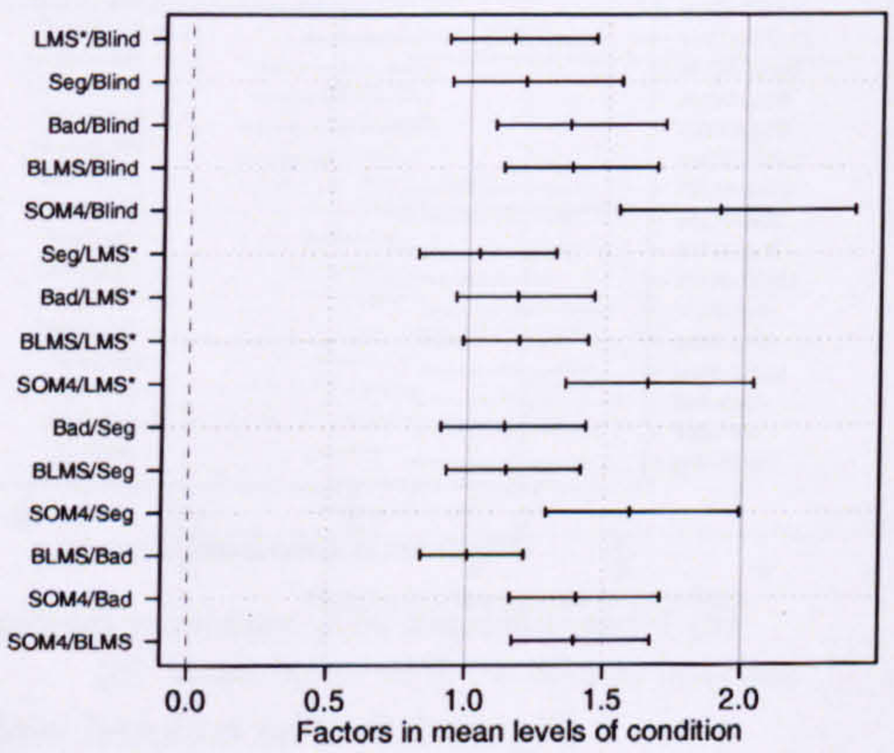
(a) Box-and-whisker plot of all seek lengths.



(b) Pairwise mean seek count diff. matrix.



(c) Mean difference 95% confidence intervals.



(d) Mean factor 95% confidence intervals.

Figure 5.10: The collated seek length results of the task-based study.

Questionnaire

The questionnaire following the session was filled out by the participant alone. It can be found in appendix D; all questions were multiple choice, most had five ordered responses. The answers given from the questionnaire were collated in order to determine any statistically significant findings. This was done for two reasons; for the objective questions about the participants' background, I want to attempt to refute the groups being approximately i.i.d.[4], a precondition to the previous findings being significant. Secondly, for opinion

[4]Independent & Identically Distributed

Question	Mean	Median	P-value
listen	3.6	4	0.4405
musician	3.0	3	0.9812
computer	4.7	5	0.1016
software	1.1	1	0.5053
useful	3.6	4	0.001829
training	1.5	1	0.3101
representative	N/A	3	0.7333
know	1.2	1	0.4245

Table 5.8: The averages and ANOVA probabilities of non-equal means over each question in the questionnaire.

questions, this is done to check for any statistically significant trends in perceptions.

An analysis of variance test was conducted on each of the sets of the questions as a response to the condition as a predictor; table 5.4.2 shows the P-values. The question pertaining to usefulness of navigation aid has an exceptionally low chance of a consistent mean over each condition. Aside from this, none qualify as having a particularly significant chance of differing means, and certainly not enough to refute the assumption of approximate i.i.d..

The averages are largely in agreement over each question. People typically had heard at least one of the tracks before, found the five minutes of training time either 'too much' or 'about right', and found the tasks generally representative of how they use navigation facilities; we plot the latter in figure 5.11. 'Never navigate' was for people who said they either did not listen to audio, or at least never needed to navigate in audio. 'Not these' was for people who did not consider the tasks given as being representative of their navigation needs. This accounted for one in three participants. The rest, around 50%, considered the tasks representative of their navigation needs.

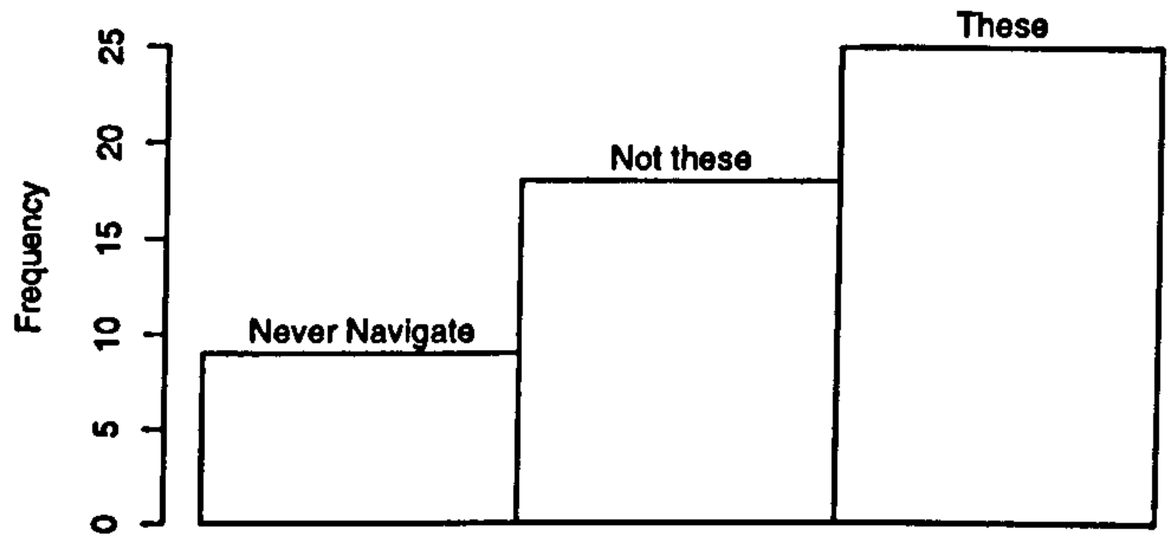
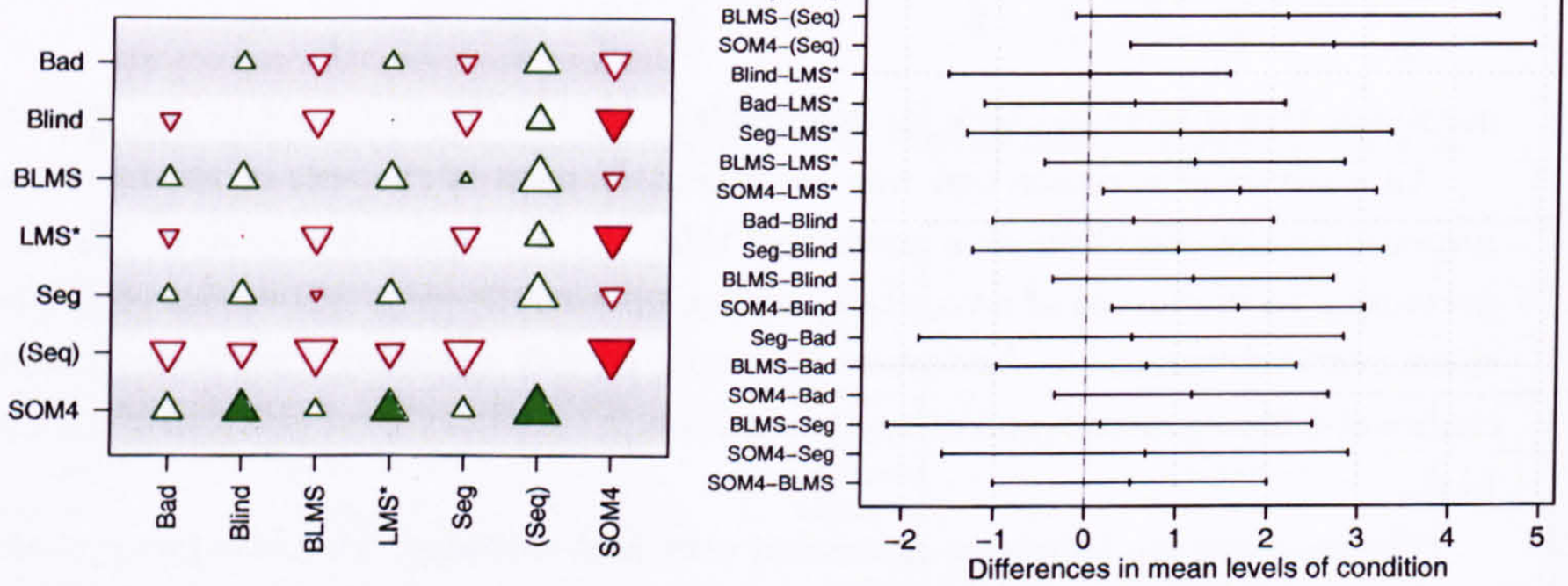


Figure 5.11: The histogram of opinions as to the degree of representation of the tasks.

Finally, the 'useful' question is investigated; namely how people perceived the helpfulness of the navigation aid. The Tukey HSD significant difference matrix and 95% confidence ratios for difference are given in figure 5.12. On the whole people that had no navigation capabilities whatsoever considered the 'aid' less than useful. People using the SOM4 and to a less significant degree the BLMS and Seg, typically considered them more useful than people using other visual aids. In particular, people using the SOM4 visualisation rated it on average higher (at 95% levels of confidence) than those using Blind and the LMS* visuals.



(a) Pairwise mean useful diff. matrix.

(b) Mean difference 95% confidence intervals.

Figure 5.12: The collated 'useful' question on the questionnaire for the task-based study.

5.5 Conclusions

The initial assumption that the music information retrieval tasks are, on the whole, representative of navigation tasks carried out by people is supported. Objective boundary points do exist in music that can be agreed upon by many people.

The (null) hypothesis that the traditional waveform method of visualising audio is as at least as good as a SOM-based chromatic-plane trajectory method of visualisation for conducting tasks is refuted, and I instead advance the theory that the SOM4-based visualisation will substantially aid the casual navigation of musical audio.

The current standard of supplying no visualisation at all is significantly worse than even basic visualisations for certain tasks. There is evidence leading us to refute the theory that the PCA method of dimensionality reduction for chromatic plane projection is as good as the SOM methods for projection.

A simple visualisation is a reasonable augmentation for a popular music player in that the learning curve is minimal with my proposed methods, that it is both objectively measurable to be helpful and generally perceived as being so too. A visualisation is helpful for navigation because it provides enough of a cue to direct individual seeks accurately to the desired point in the music, thus decreasing the number of seeks required and increasing the mean seek distance.

Chapter 6

Conclusions

"The whole problem can be stated quite simply by asking, 'Is there a meaning to music?' My answer would be, 'Yes.' And 'Can you state in so many words what the meaning is?' My answer would be, 'No.' "

—Aaron Copland (1900–1990)

6.1 Discussion

In concluding a work such as this, it is best to review the original aim. At the beginning, I wished to learn more about navigation aids, and in particular how useful they might be in the context of popular music playback software. There was reason to believe that a depiction of the waveform might be useful through its ubiquity among audio applications. However, I sought to explore further, to determine if other methods might be better suited in the context.

To determine if, scientifically, there was any truth in these thoughts, I first conducted initial studies concerning navigation in popular music playback software. Largely because of this, I hypothesised that content-based visual navigation aids based around a projection from audio to colour would prove helpful for common music playback tasks due to providing a visual analogy, allowing better directed random-access seek operations. I went further, hypothesising that even more useful would be a non-linear psychoacoustic transformation, whose mapping of colour to sound was arbitrary, but which rather attempted to preserve relationships between colours, while maximising the shades used. I thought that this would better express the variety of sound, the inter-relations and the constancy of individual portions.

After comparing my proposed visualisations over a range of music, and conducting several user evaluations involving over 100 participants, I have a body of evidence to conclude the following:

- For my chosen context, the core method of augmenting the navigation system with a content-based visualisation is a success. It does have a significant effect on how the navigation facilities are used over a multitude of users and on a multitude of tasks. Unsurprisingly, this changes dependent upon what algorithm is used to generate the image.

- With the notable exception of classical music, I found basic spectral-surface-based visualisations to be adequate for the visualisation of most important aspects. I did not test many other more 'musically-inspired' types of feature extraction, but it may be reasonable to expect that a combined approach may yield superior results. The two non-spectral-surface-based feature extractions I proposed proved themselves less effective than more basic methods.

- Despite mapping an entirely arbitrary colour to any particular moment of music when considered individually, I found the chromaticity-plane projection through dimensionality reduction techniques to be, on the whole, at least as good as any other technique. In particular, this turned out to be such an unimportant property of visualisation that it allowed the 4x4 SOM-based aid to perform statistically significantly better than almost all others. This means that the same technique could be used with a different chroma-plane to provide a visual that better fits the colour scheme of a particular player with no loss in performance (assuming the plane had a similarly good evenness of perceptual colour distribution).

- The best method, the 4x4 SOM, appeared to agree with my original hypothesis concerning improvement of navigation through visual means. The evidence shows that a better performance in terms of time taken to complete a task correctly arose from fewer—but better directed—random-access seek operations.

To summarise, the main contributions of this work are:

- The proposal and implementation of a general signal-processing metamodel which can be used to model a number of structures difficult to express in other graphical metamodels of computation.

- Empirical data concerning how people navigate around tracks in order to complete appropriate tasks along with an analysis of the data.

- A comprehensive review of literature concerning the content-based visualisation of musical audio.

- Several distinct novel methods for generating visualisations of music.

- The implementation of these methods, together with a comparison and discussion of their output over several styles of music.

- Empirical data from user studies conducted to determine the performance of the visuals as navigation aids, together with a thorough statistical analysis of the data.

6.2 Future Directions

The present work, although making some important contributions, has only scratched the surface of this interesting topic. I will now discuss the future directions in which this work could be taken.

Alternative Visualisation Methods

The feature set used for the SOM-based visualisations, while proving its mettle in this study, can almost certainly be improved upon. The low-level features used model little other than the timbre, and therefore do not model time-varying changes in music. A range of musically significant features, e.g. rhythm, melody, key and harmony, could prove to be a useful addition to the feature set.

I have made few efforts to analyse the effect of combining the waveform visualisation with the proposed colour-projection visualisations. Further work is necessary to better verify my tentative hypothesis that the addition of the loudness silhouette is not useful after the brief period of learning has finished.

A second visualisation route, based around segmentation technology, could attain a visualisation similar to the 4x4 SOM by hierarchical segmentation. Tracks could be segmented as per the literature, but then each segment could be further 'subsegmented'. Colour could be based upon the model feature represented by the segment (unlike the SOM method where colours are chosen arbitrarily). But the colours used in subsegments would be a blend of mostly their parent segment and the other segments to which the subsegment model leans (if it is not exactly equal to the parent segment, that is).

In so far as the bandwise mechanism (for introducing colour) was implemented, our study conducted only a simple 3-way fair division of the critical bands. Informal experimentation suggests that uneven division could significantly improve the fidelity of the resultant visualisation. Two points through the critical band spectrum could, for example, be chosen such that the three subspectra minimised their cross variance.

Alternative Navigation Aids

In the present work, a static image is given on the navigation bar. Another approach would be to determine the image of the navigation bar by the current point playing. This

might be a single corresponding row of the self-similarity matrix, for instance.

One further avenue of investigation would be in the translation of abstract pseudo-continuous linear audio data into a discretely annotated format similar, perhaps, to that presented by Couprie (2004). The translation represents a far easier problem than that of generalised music transcription, since the representations are similarly low-level and imprecise, dealing with such concepts as speed, brightness and dynamics.

Miscellaneous

One of the most exciting aspects (indirectly) opened up in this project (as far as I'm concerned) is the prospect of a 'unified audio visualisation architecture'. Such an architecture would help develop visualisation components that are able to actually deliver what they promise. Through either a pre-analysis stage or in a real-time context (our metamodel implementation supports both), the musical audio could have specific features extracted and correlated to the rest of the track, in order to deliver a useful and consistent visualisation. Because the visualisation can then be programmed separately to the analysis framework, the visualisation author could concentrate on the task at hand, and the feature extraction author would be able to improve the visualisation, precision, and accuracy, without needing their co-operation.

The information could be presented to the visualisation module by way of an opaque interface, and the format could be in either a continuous form (as I have already demonstrated), or a higher-level, discrete form as discussed in the preceding section.

The annotation information could be in the digital music file itself, as technologies already exist to encapsulate metadata in a track. This would allow both portable devices (such as Apple's iPod) to utilise the information accordingly and, with proper standardisation, could be used to "jist" music tracks whilst browsing in music stores. Such a "fingerprint" may describe music adequately enough to allow a potential purchaser to determine their interest in a record.

Appendix A

Tools Used

Document Preparation

- **L^{A}T$_{E}$X** Typesetting software.
- **Bibtex** A bibliography administration and translation tool.
- **KBibtex** A bibliography collection manager by Thomas Fischer.
- **Kile** A Latex document editing environment.
- **Winefish** A Latex document editing environment.
- **OpenOffice.org** An office suite featuring figure-authoring software.
- **GNUplot** A data plotting tool.
- **GNU R** A statistical analysis and visualisation language.

Software Creation and Analysis

- **KDevelop** A multi-language integrated development environment.
- **Qt** A cross-platform GUI toolkit by Trolltech of Norway.
- **GNU Compiler Collection** A cross-platform cross-language compiler.
- **GNU Debugger** A run-time debugger.
- **Perl** The Practical Extraction and Report Language interpreter.
- **SLOCcount** A source code analysis tool by David A. Wheeler.

Experimentation

- **K Desktop Environment** The Unix desktop environment.
- **amaroK** The Unix music player.
- **PostgreSQL** A database management system.

All software used for this project is Free software, as defined by the Free Software Foundation.

Appendix B

Navigation Tasks

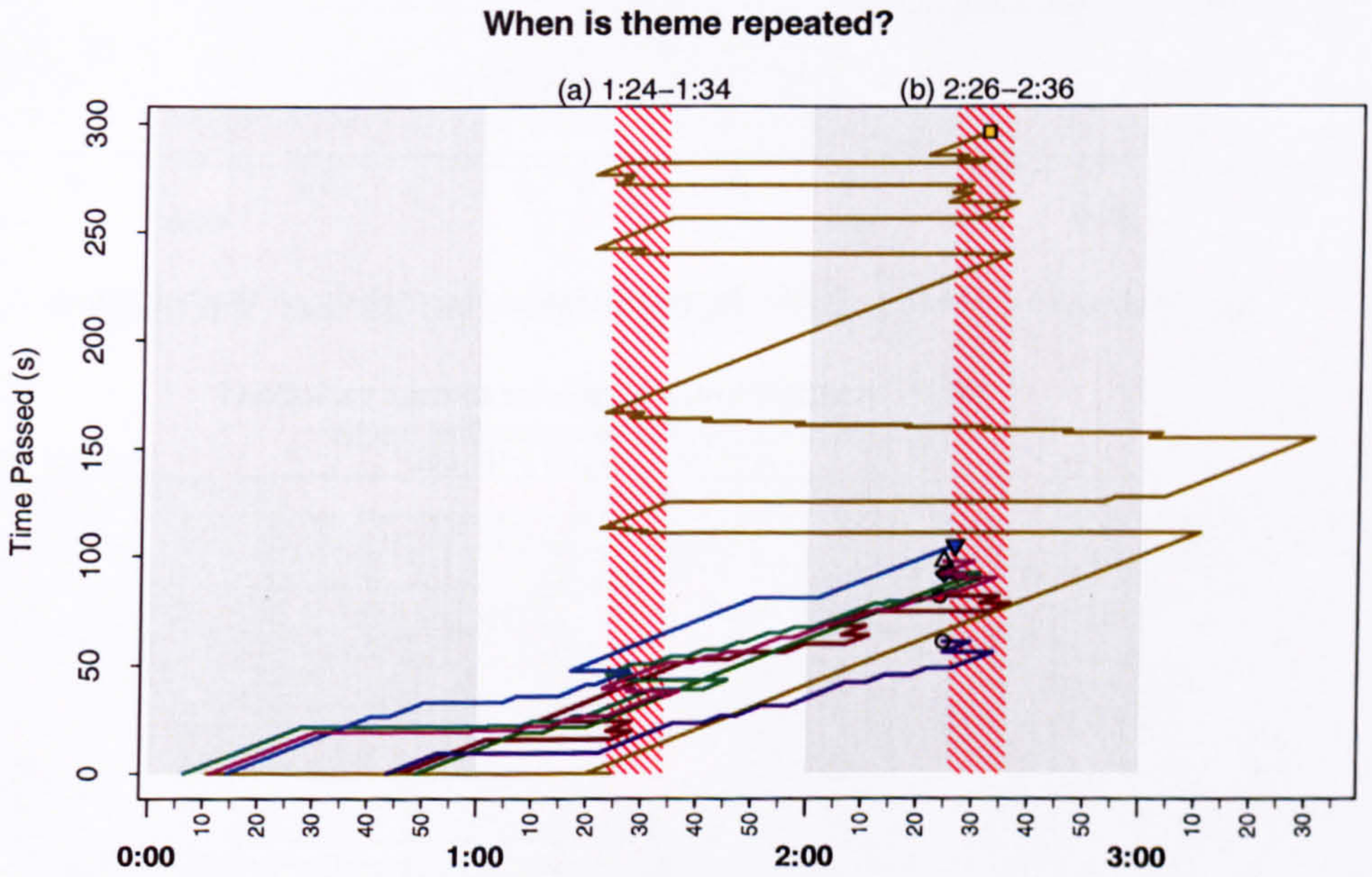


Figure B.1: Task 7: (a) is the original theme, (b) is the repeat.

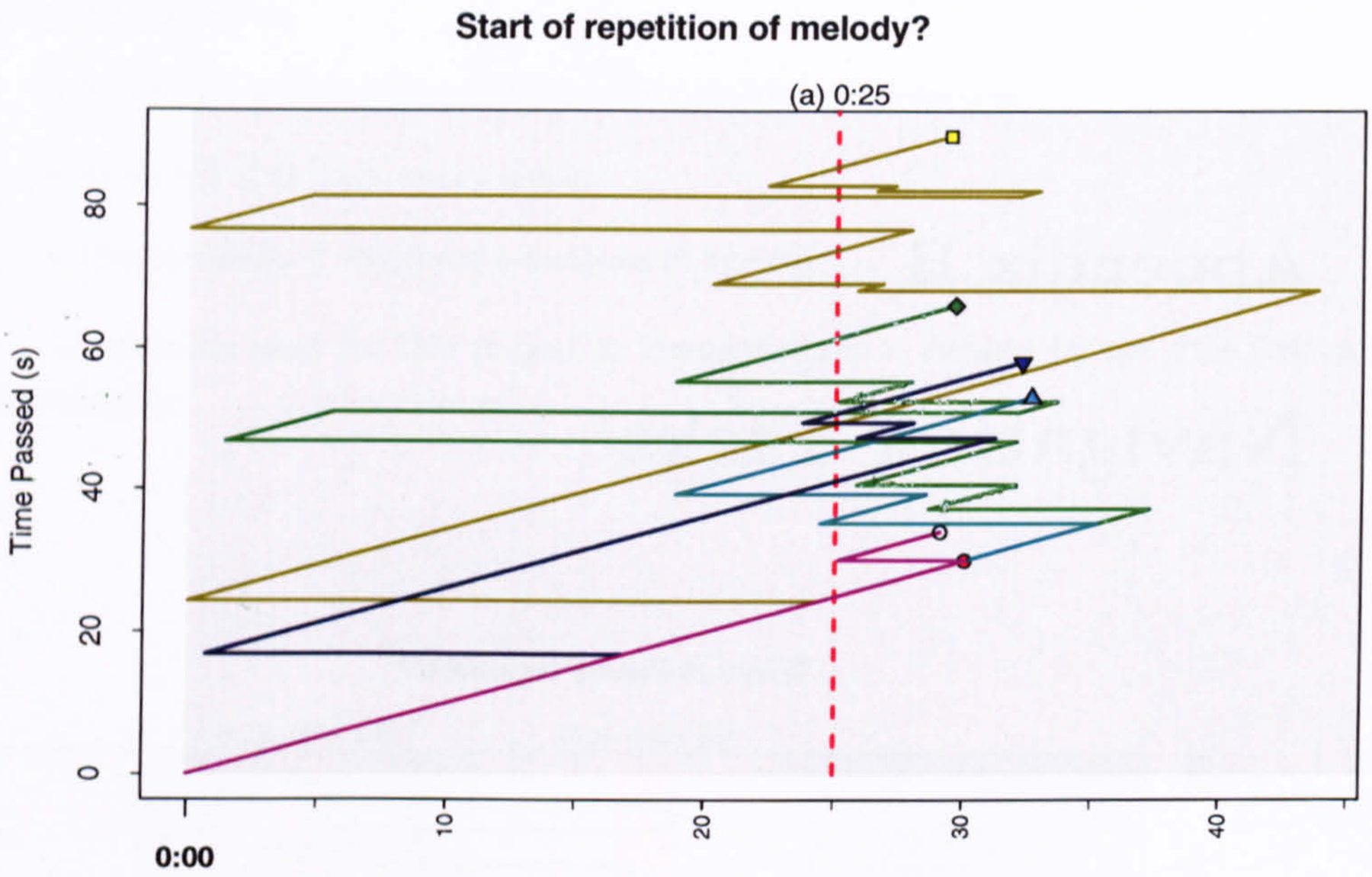


Figure B.2: Task 8: (a) is the repetition of the melody at the beginning.

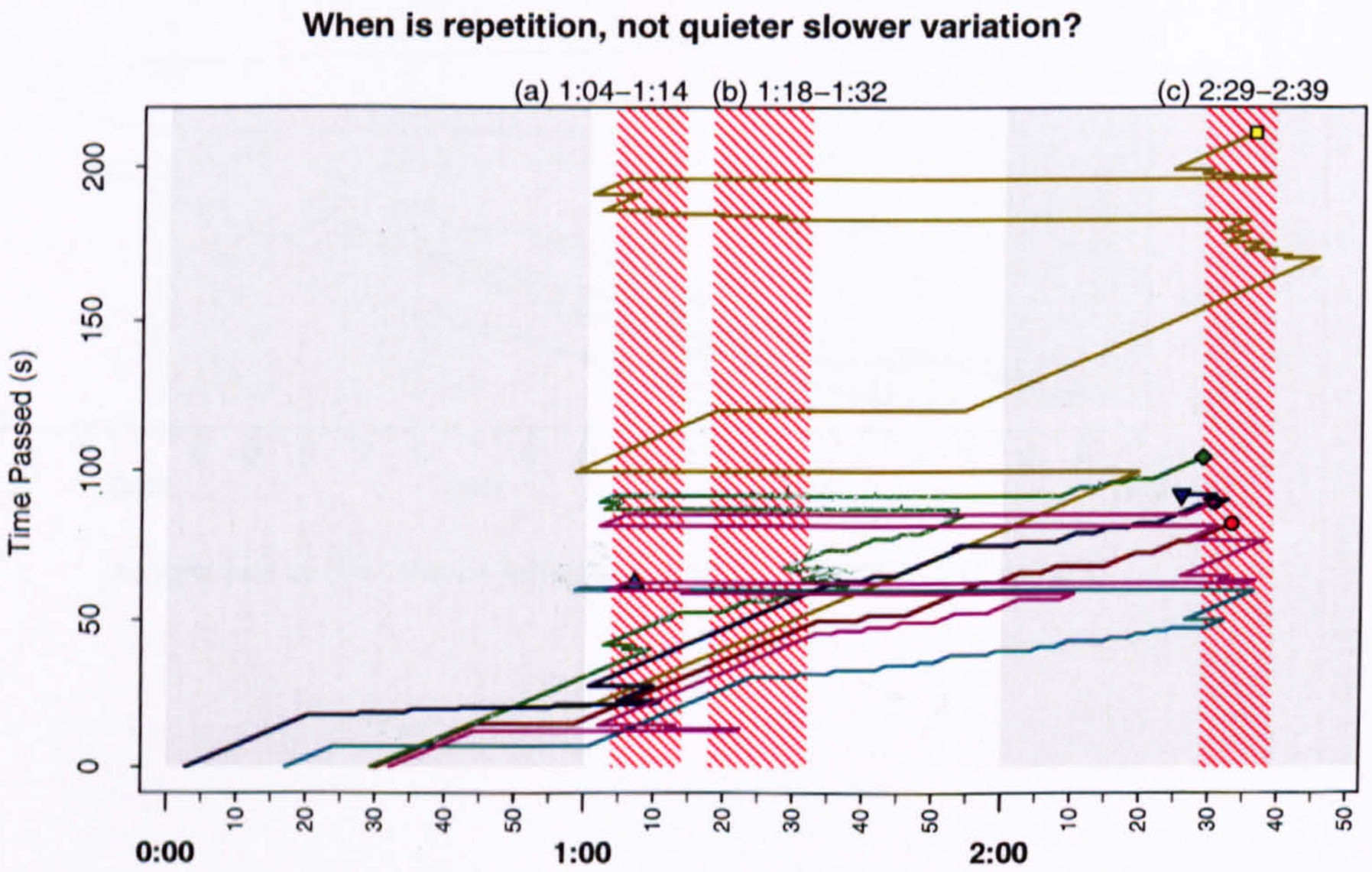


Figure B.3: Task 9: (a) is the original theme, (b) is a quieter/slower variation, (c) is the repetition.

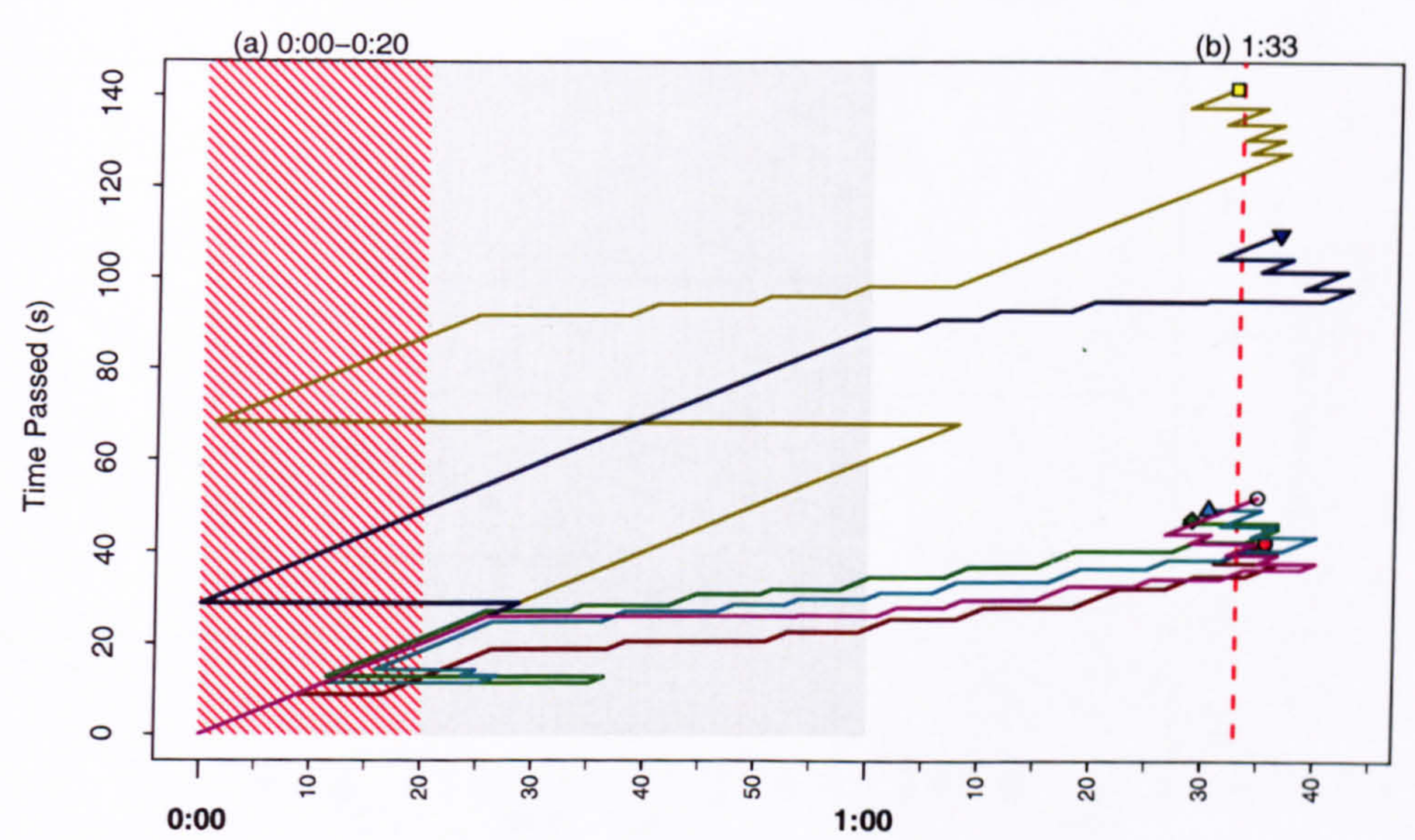


Figure B.4: Task 10: (a) is original period of playing, (b) is the second onset.

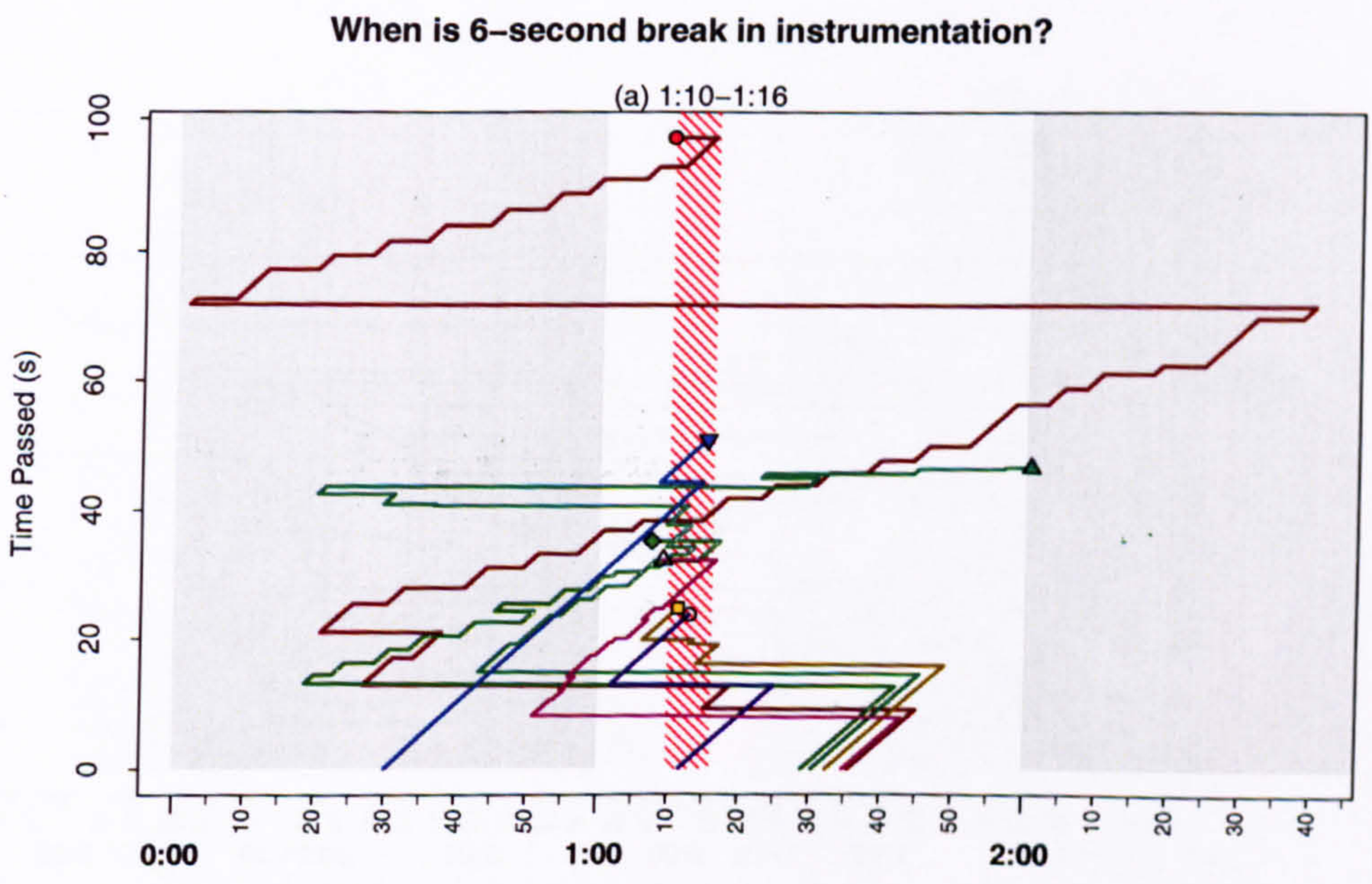


Figure B.5: Task 11: (a) is the 6-second break.

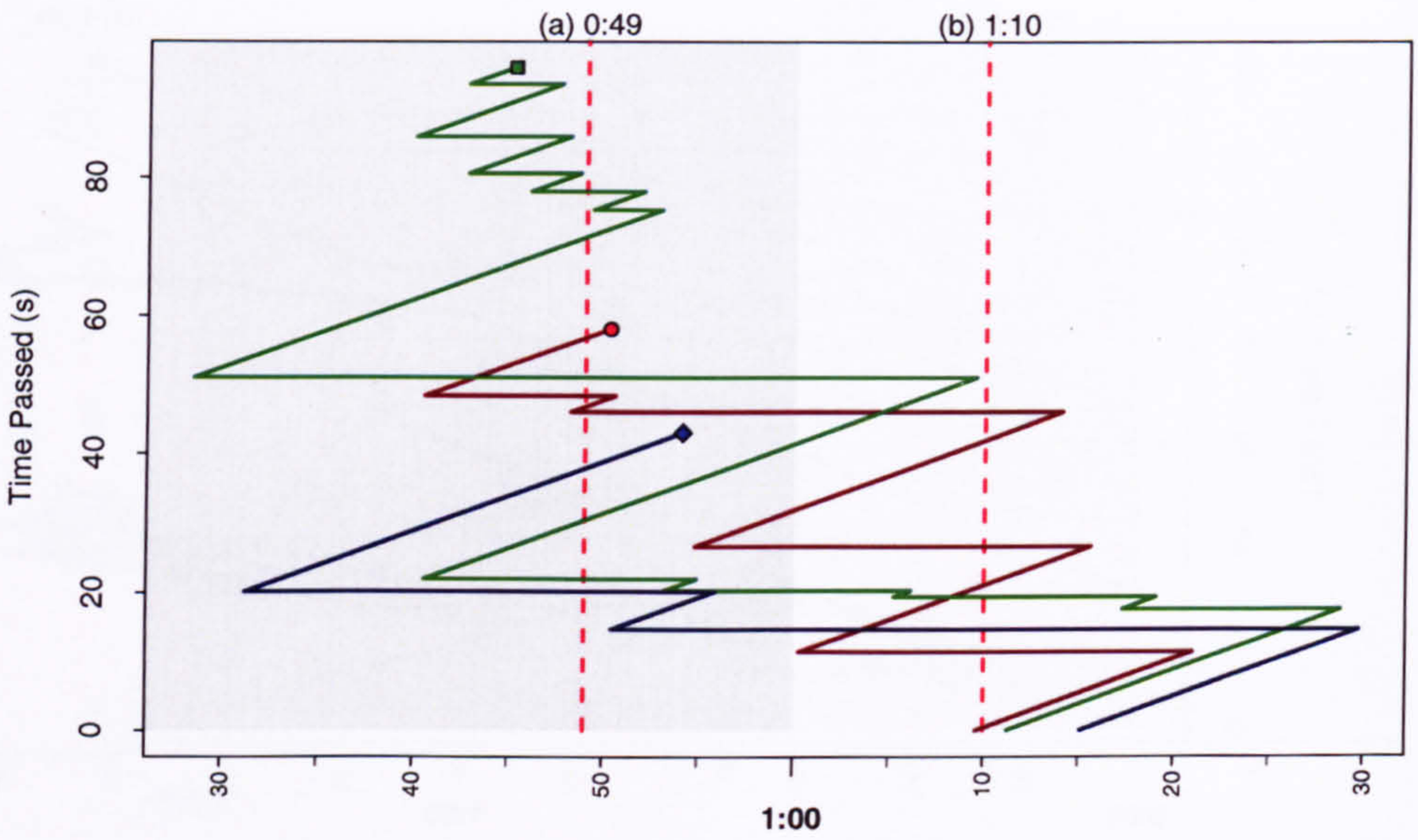


Figure B.6: Task 12: (a) is the onset of the instrument, (b) marks 1:10.

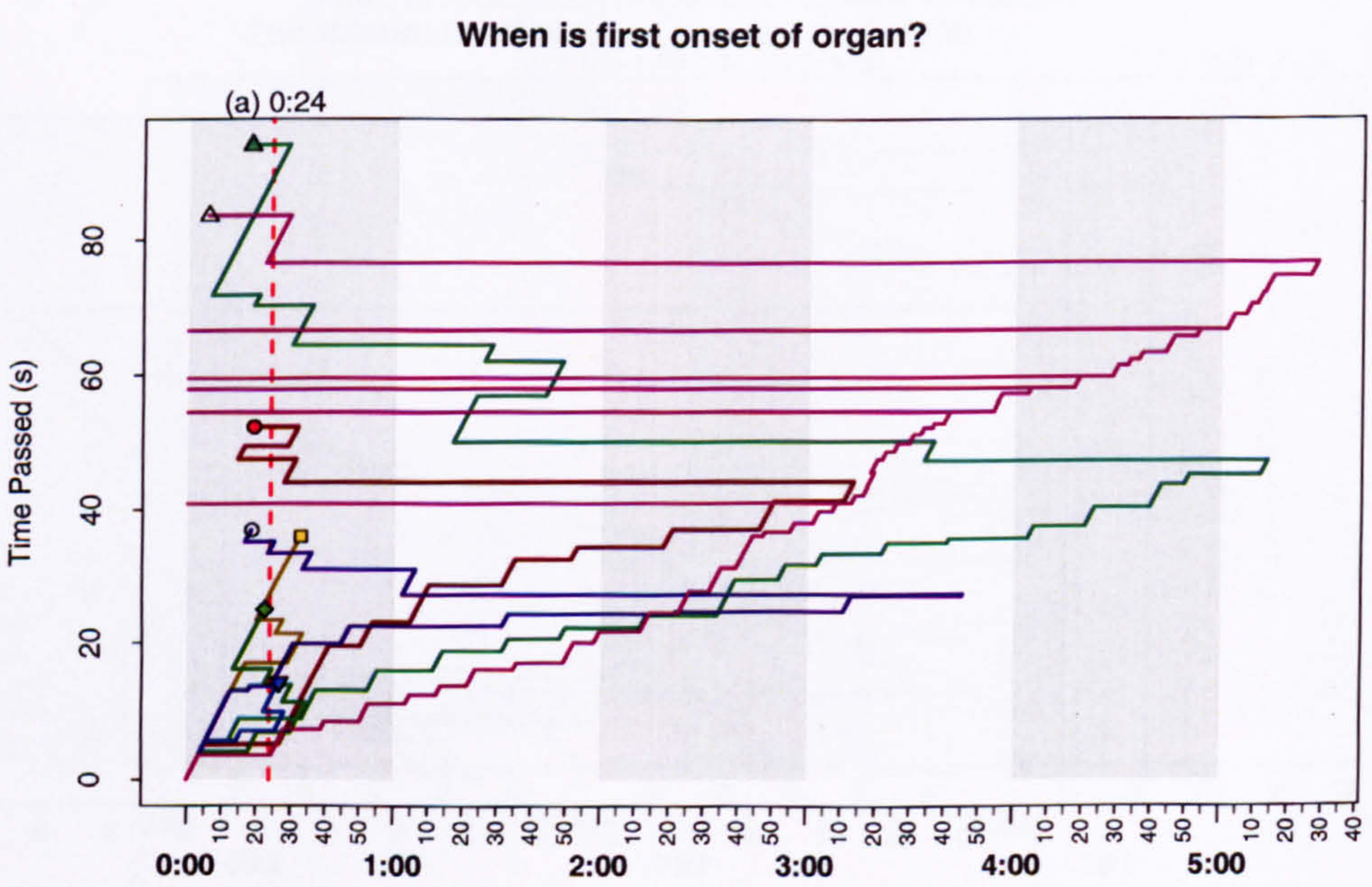


Figure B.7: Task 13: (a) is the onset of the organ.

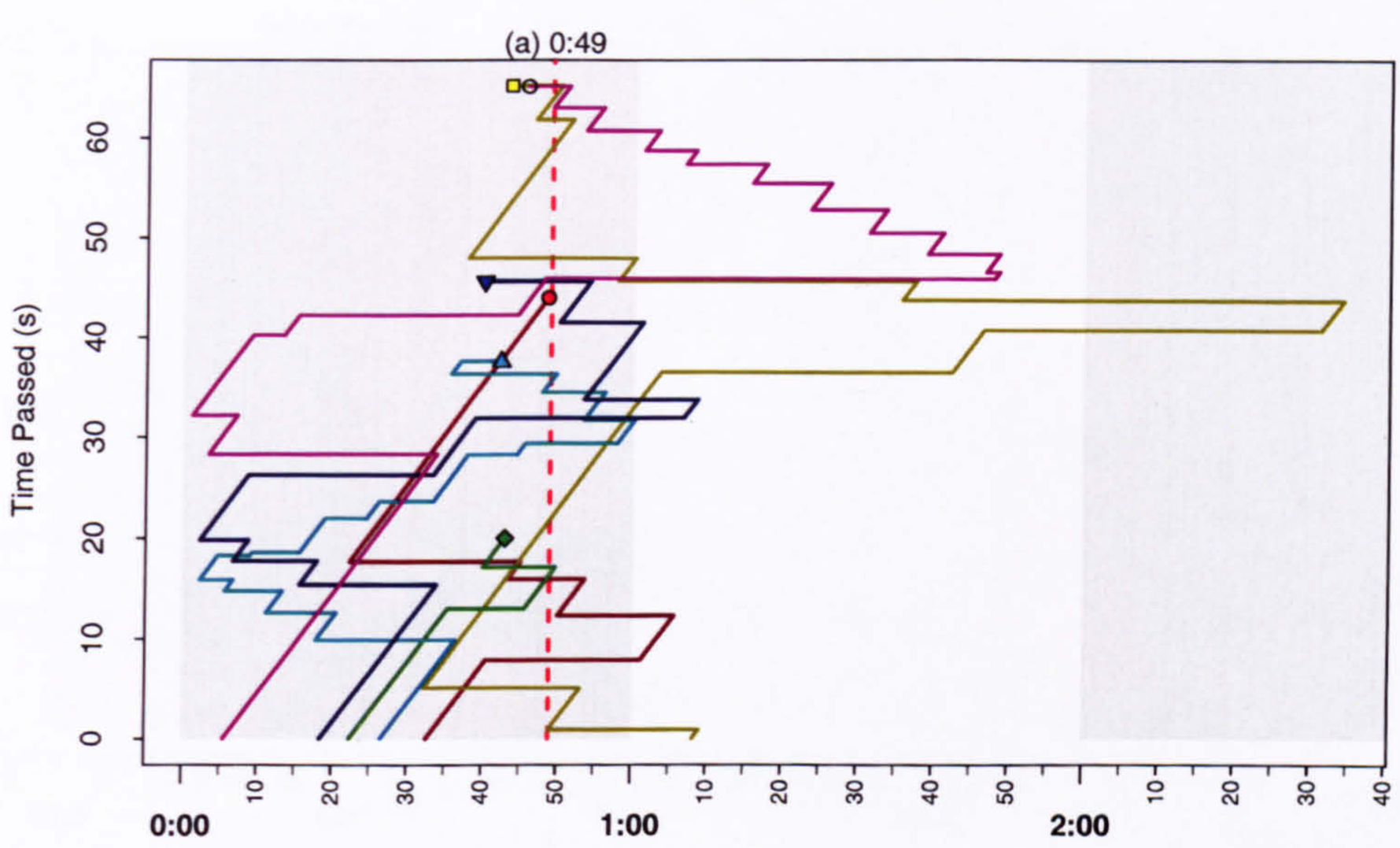


Figure B.8: Task 14: (a) is the onset of the drums.

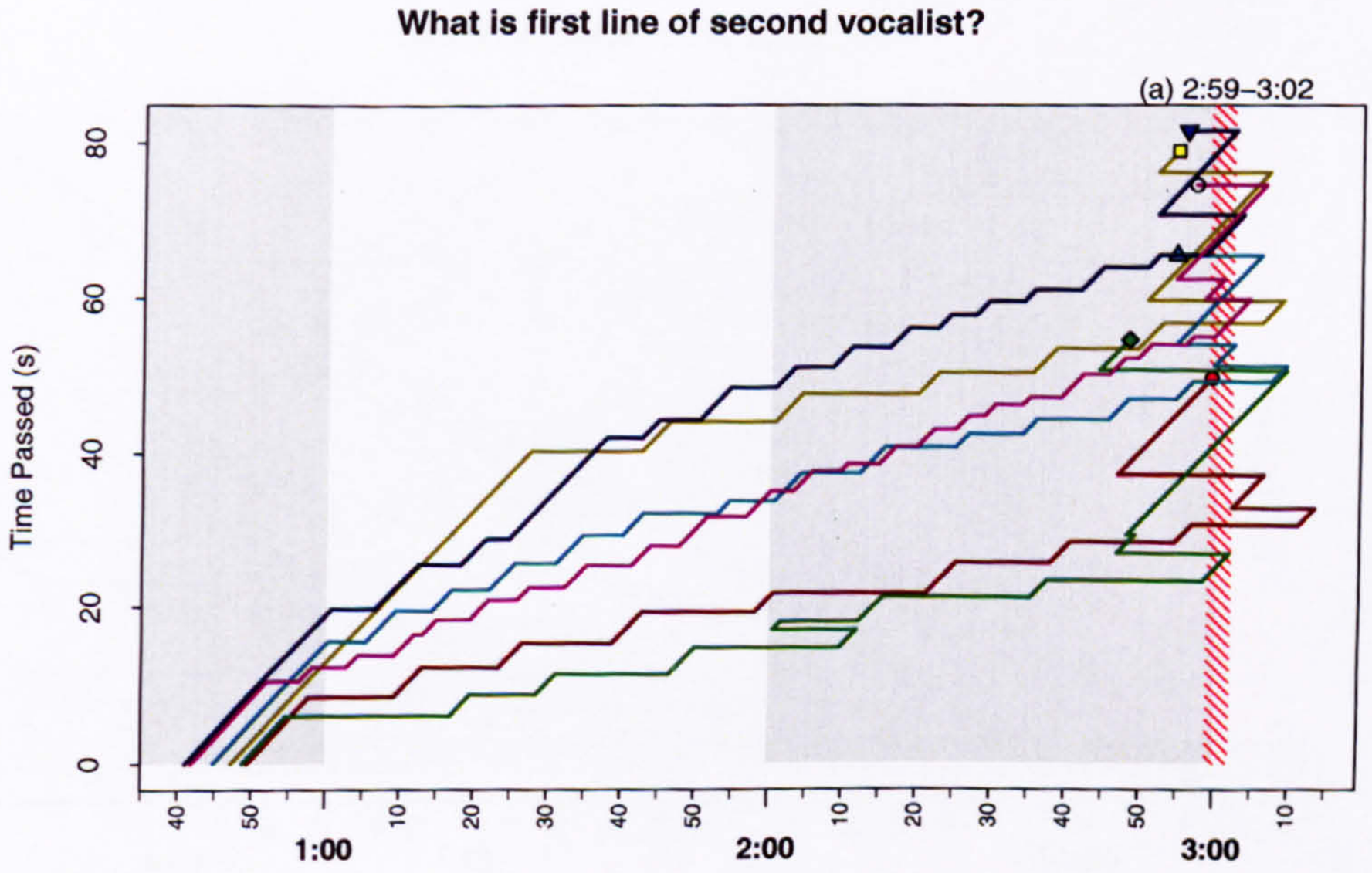


Figure B.9: Task 17: (a) is the first line of the second vocalist.

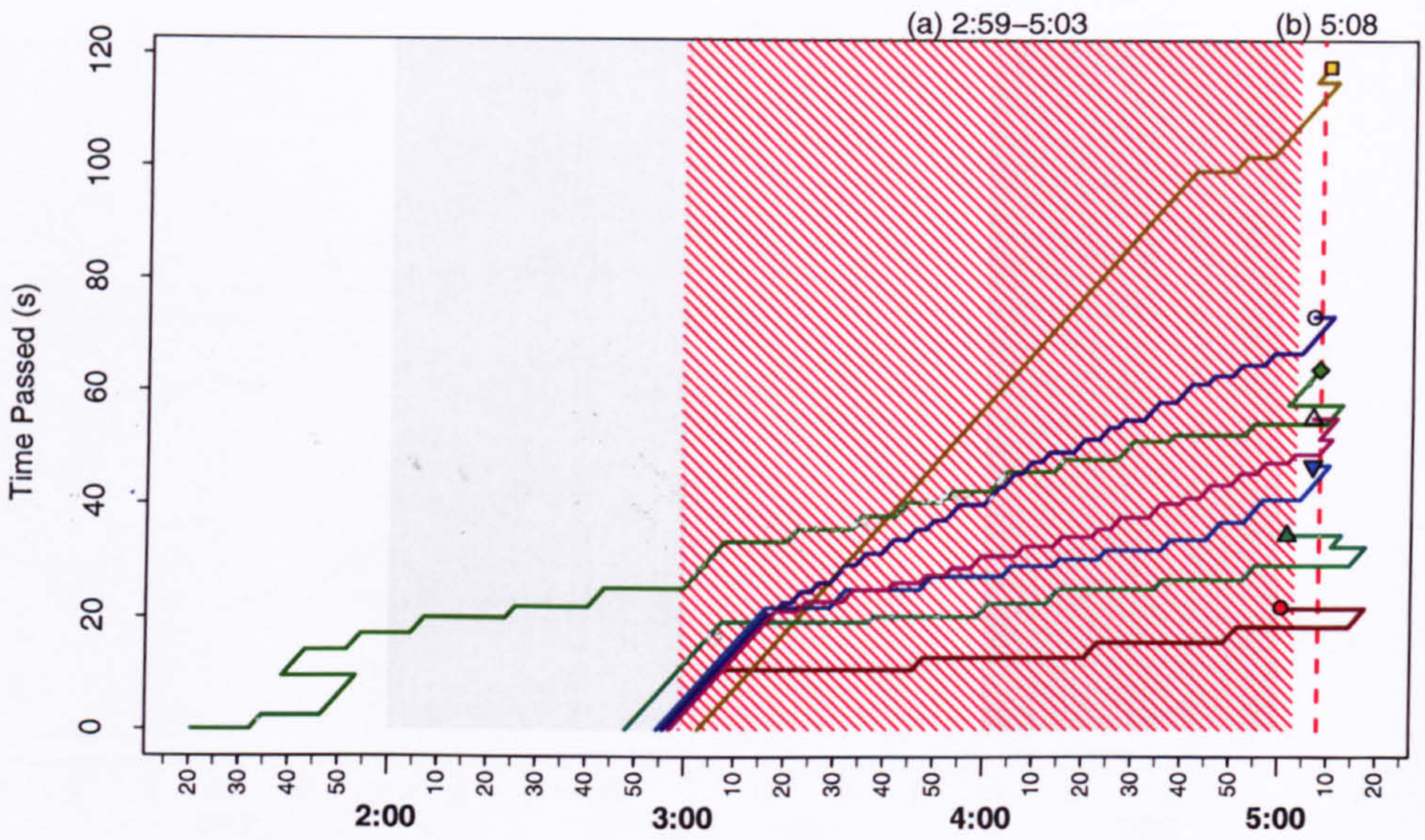


Figure B.10: Task 18: (a) is the period of the second vocalist, (b) is the onset of the first vocalist.

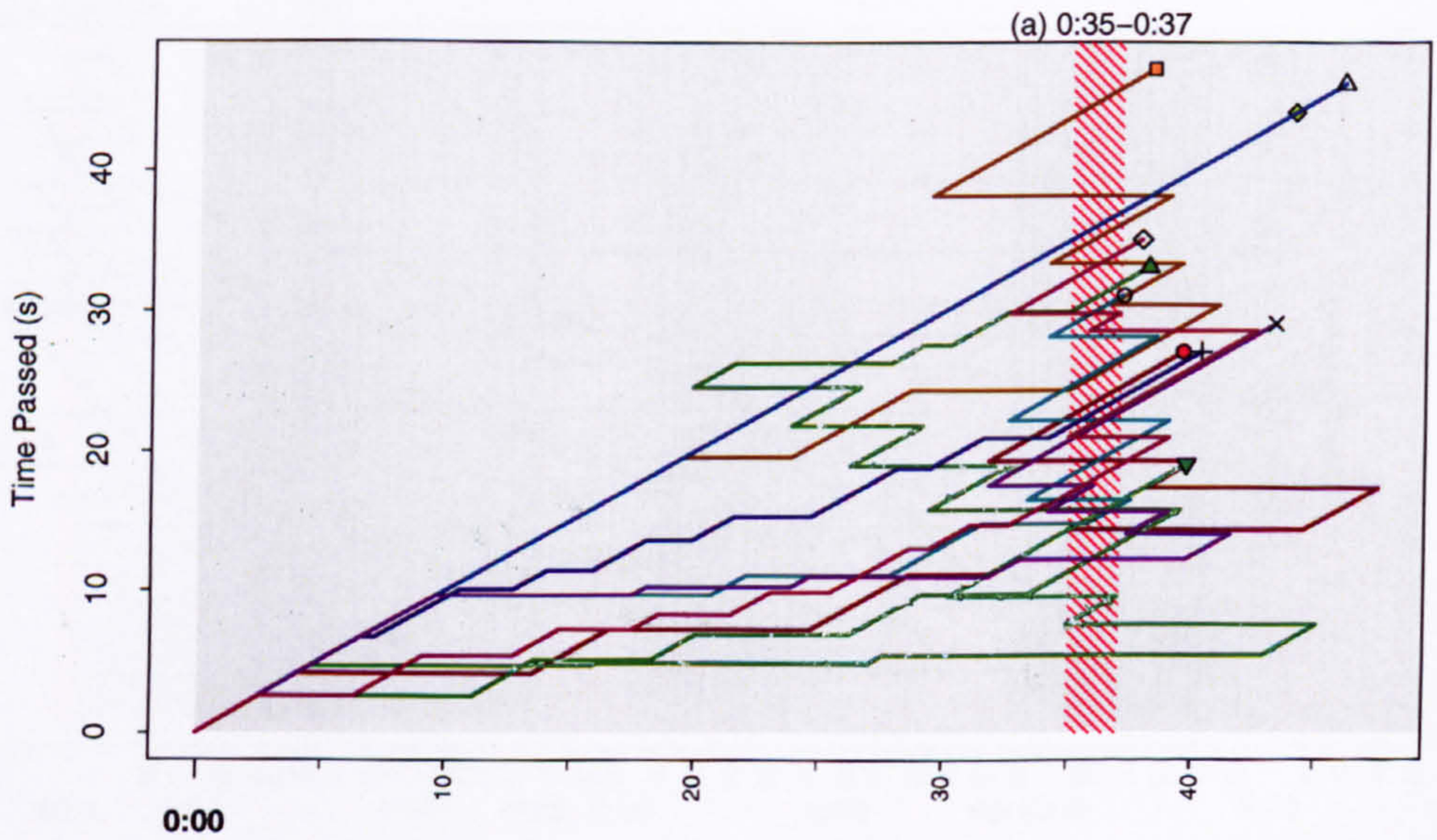


Figure B.11: Task 20: (a) is the period over which the first words are sang.

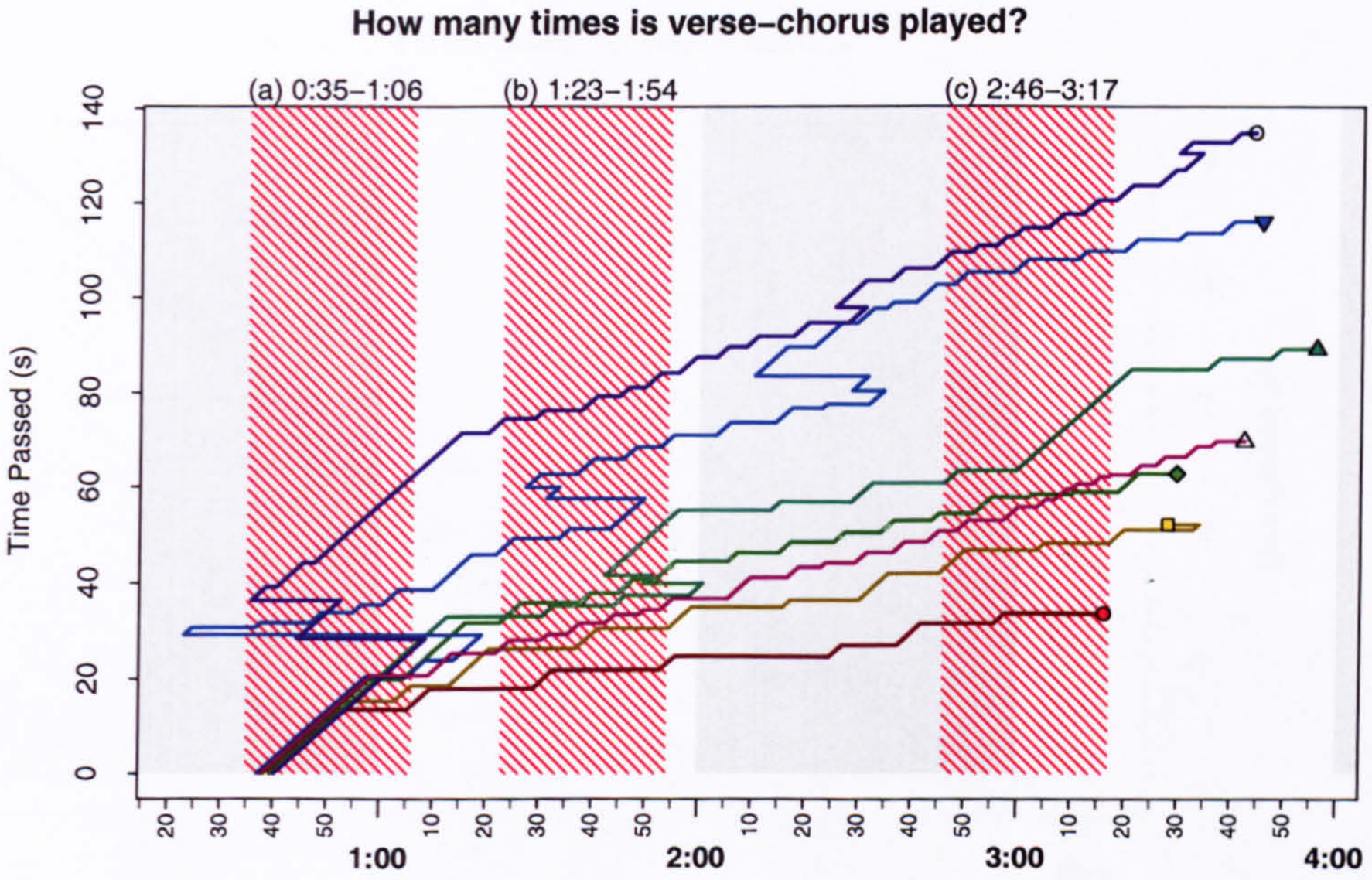


Figure B.12: Task 21: (a), (b) and (c) all mark the verses. There is a bridge between (b) and (c).

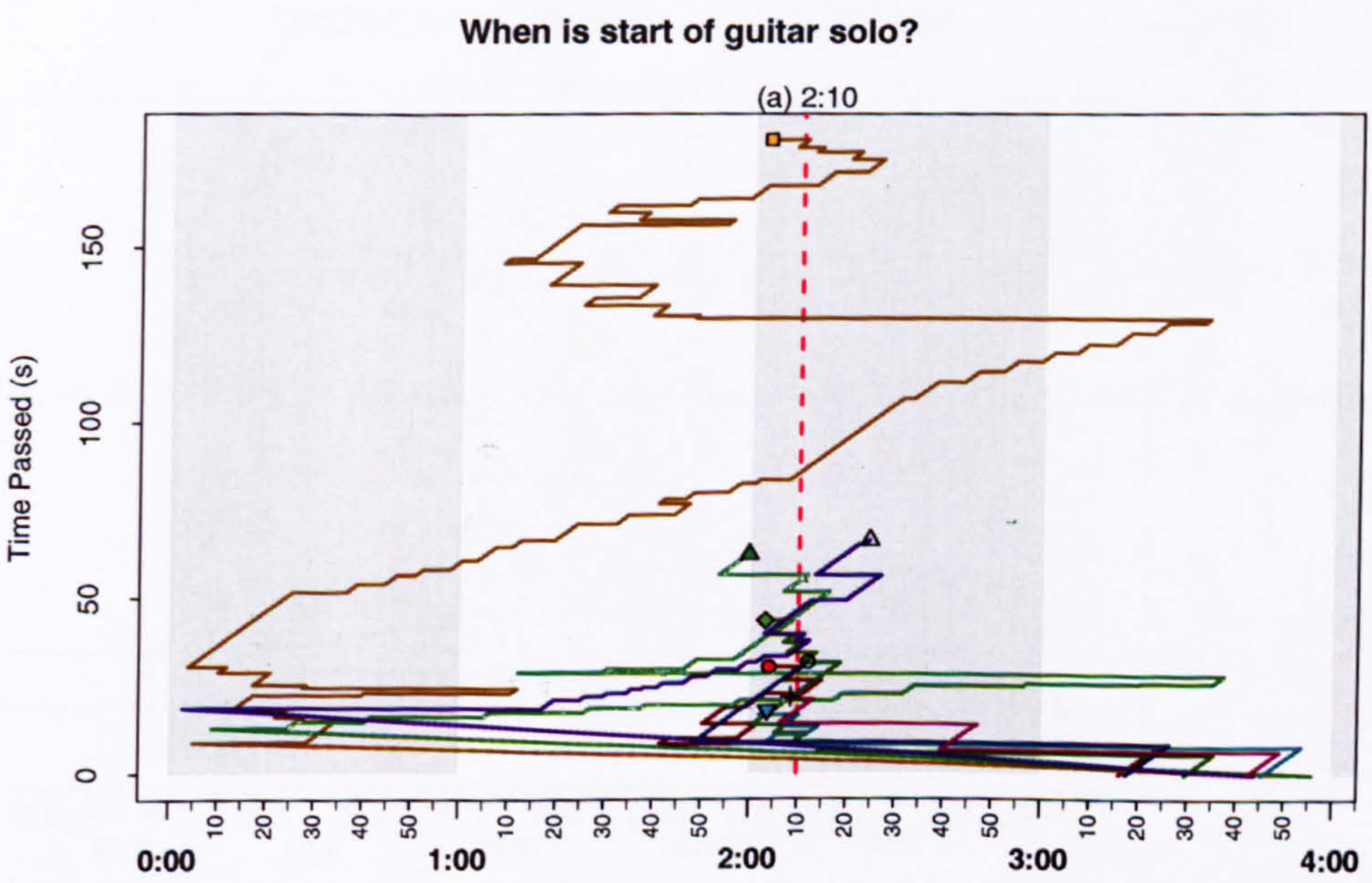


Figure B.13: Task 22: (a) marks the start of the guitar solo.

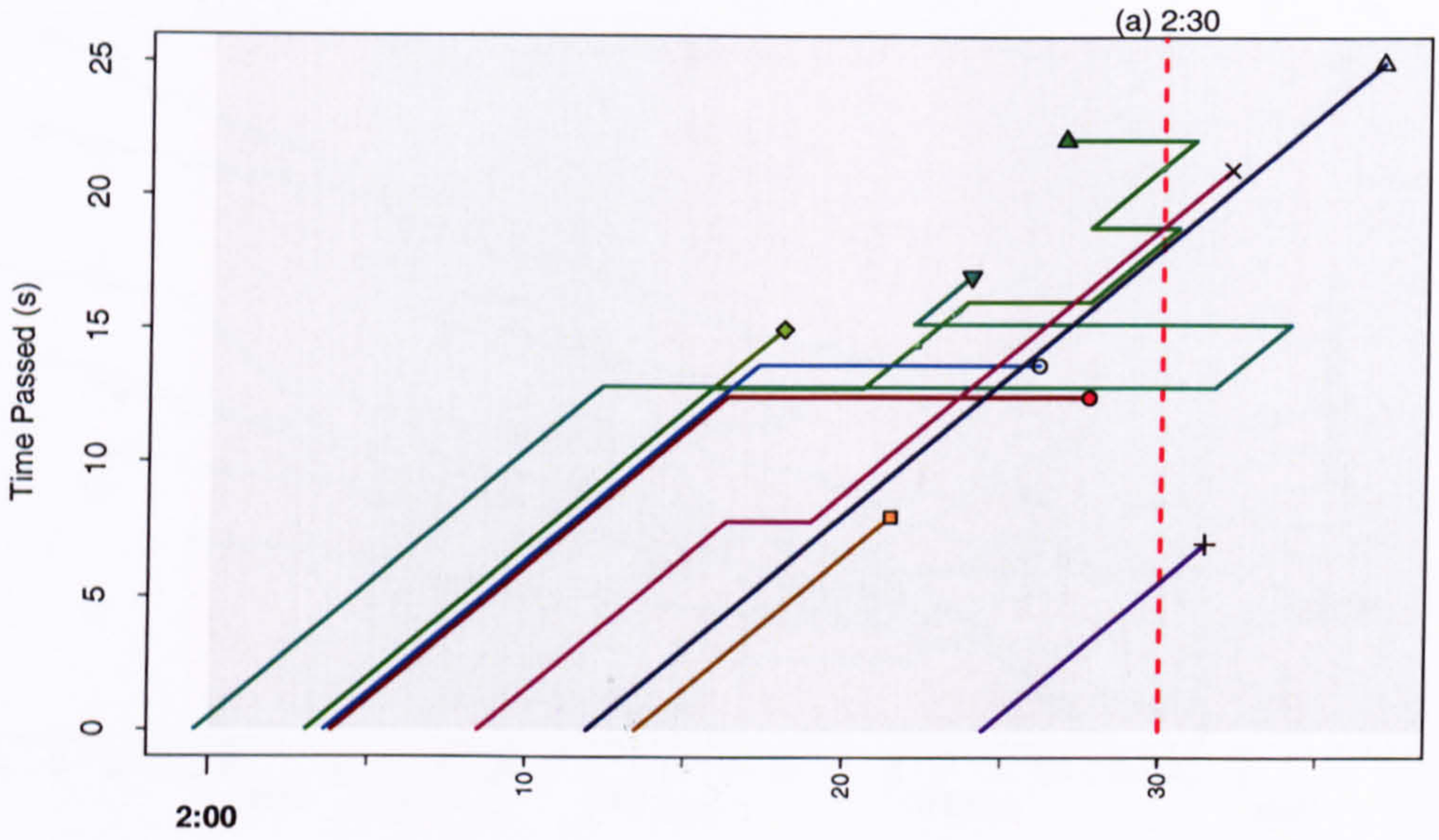


Figure B.14: Task 23: (a) marks the end of the guitar solo.

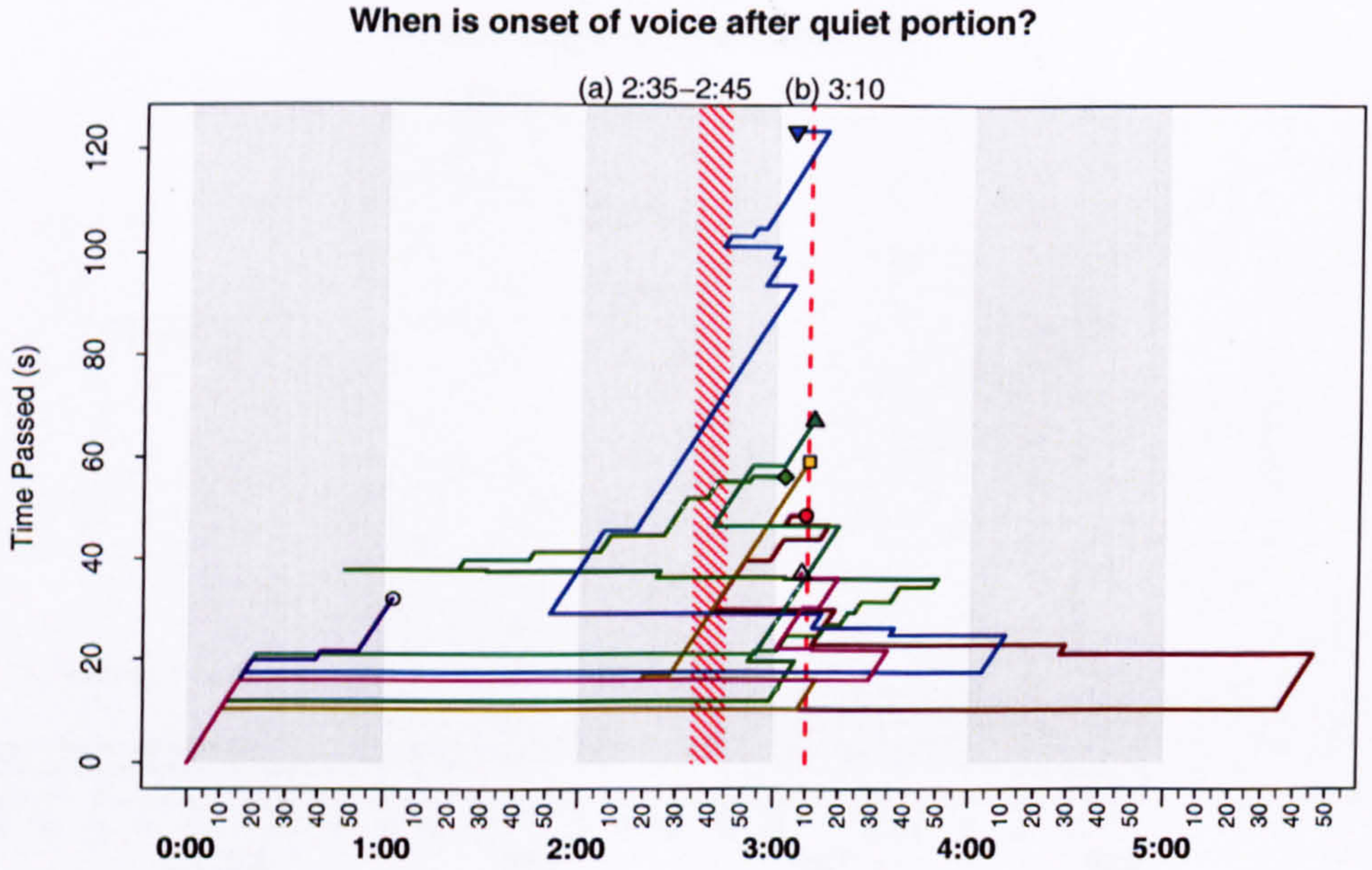


Figure B.15: Task 24: (a) marks the quiet portion, (b) marks the onset of the voice.

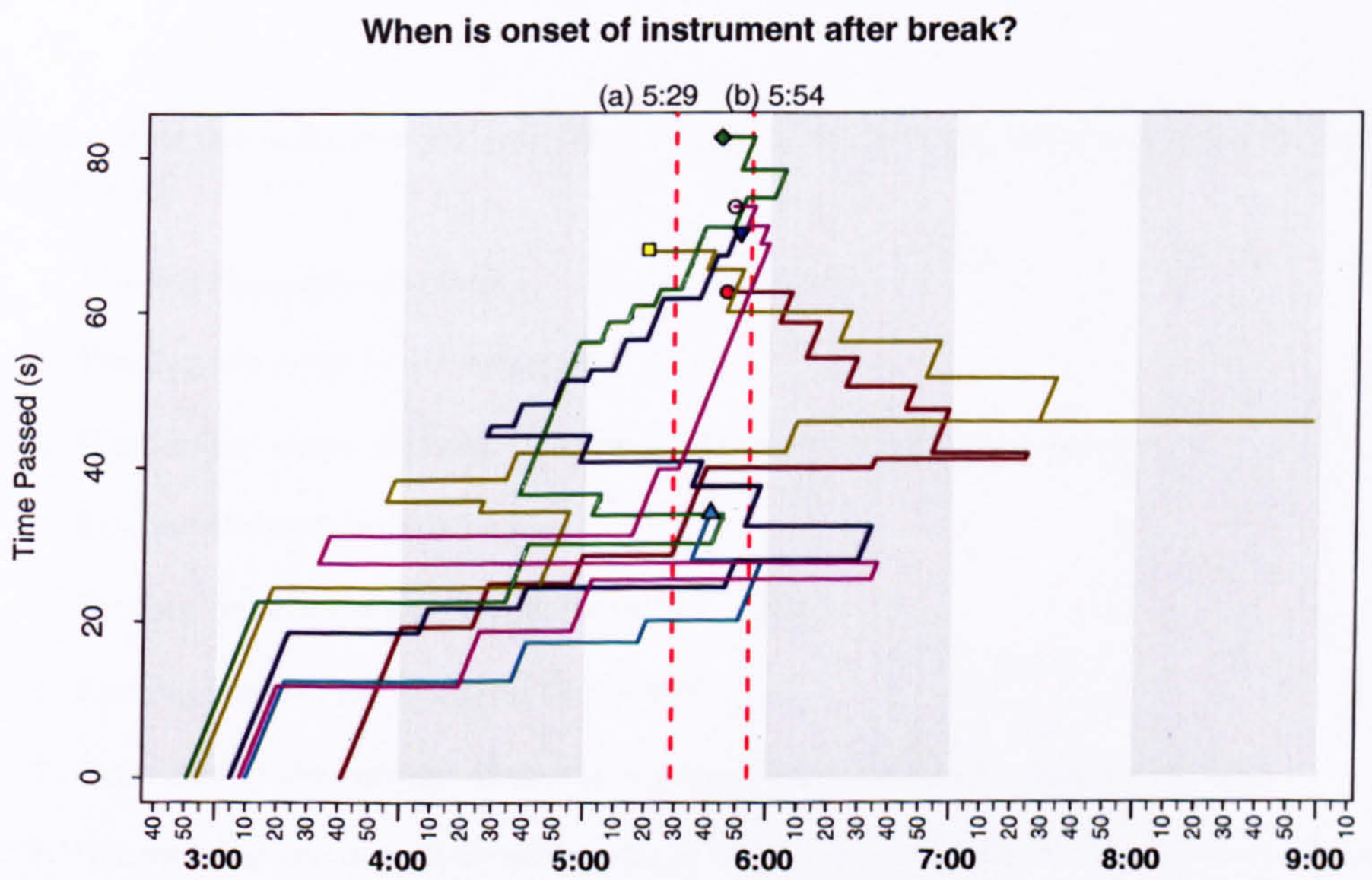


Figure B.16: Task 25: (a) marks the end of the break, (b) marks the onset of the instrument.

Appendix C

Reference Questionnaire

Please circle the tasks you are typically utilise a music playback application's navigation facilities for.

1. Finding the onset of vocals.
2. Finding the onset of an instrument.
3. Finding the starts and ends of verses, choruses and bridges in songs.
4. Finding the ending of the song.
5. Finding repetitions and variations of the music.
6. Finding breaks and pauses in the music.
7. Determining the content of vocals (e.g. wanting to listen for a certain lyric).
8. Determining the overall structure of the music (e.g. whether it has a verse/chorus structure).

Please indicate in the space below any other uses you would typically have for a navigation facility:

Appendix D

Main Task Trial Questionnaire

1. How often do you listen to music?

   (a) I rarely listen to music.

   (b) I usually listen to music for upto 3 hours a week.

   (c) I usually listen to music for upto an hour a day.

   (d) I usually listen to music for upto 3 hours a day.

   (e) I usually listen to music for at least 3 hours a day.

2. What is the extent of your experience as a musician?

   (a) I have never voluntarily written music or played an instrument.

   (b) I have not played an instrument or written music for over 10 years.

   (c) I have not played an instrument or written music for over 3 years.

   (d) I have played an instrument or written music in the past 3 years.

   (e) I regularly play an instrument.

3. How much time do you spend on a computer?

   (a) I rarely use a computer.

   (b) I usually use a computer for upto 3 hours a week.

   (c) I usually use a computer for upto an hour a day.

   (d) I usually use a computer for upto 3 hours a day.

   (e) I usually use a computer for at least 3 hours a day.

4. How familiar are you with audio playback software?

   (a) I am familiar with WinAmp, iTunes, Windows Media Player or some other application for playing music on a computer.

(b) I have rarely used applications for playback of music on a computer, but am familiar with the idea.

(c) I am unfamiliar with using a computer to playback music.

5. Of each of the media in the tasks, please indicate which you were already familiar (summed):

6. How useful did you find the software for helping to answer the tasks given?

   (a) Unhelpful; I think I could have done better with my usual means of music/audio playback.

   (b) Not useful; I don't think this helped make the tasks any easier or quicker than my usual means of music/audio playback would have.

   (c) Potentially useful; I don't think it helped me this time but given more time to learn I think it might be helpful.

   (d) Fairly helpful; I think some tasks were made substantially easier to complete.

   (e) Very helpful; I think most tasks were made substantially easier to complete.

7. How much training time do you think you need to use the this player application and navigation features?

   (a) I think the six minutes given was overkill. It's quite clear anyway.

   (b) The six minutes was about right. I don't think I could have learnt much more given more training time.

   (c) The six minutes was too little. I think another 10-20 minutes would have resulted in a substantially better performance.

   (d) The six minutes was far too short a time. I think more than 30 minutes solid training is needed before the system would be effective.

8. How representative do you think the tasks given are of your need for navigation?

   (a) I never need to navigate in music or audio.

   (b) When I do navigate in music/audio I would not need to do these tasks or tasks like them.

   (c) When I do navigate in music/audio I would need to do these tasks or tasks like them.

Bibliography

S. A. Abdallah, K. Noland, M. Sandler, M. Casey, and C. Rhodes. Theory and evaluation of a Bayesian music structure extractor. *Proceedings of the Sixth International Conference on Music Information Retrieval, JD Reiss and GA Wiggins, Eds*, pages 420–425, 2005.

S. A. Abdallah, M. Sandler, C. Rhodes, and M. Casey. Using duration models to reduce fragmentation in audio segmentation. *Machine Learning*, 65(2):485–515, 2006.

P. Aigrain. Audio Retrieval and Navigation Interfaces. *Readings in Multi-media Computing and Networking*, 2001.

P. Aigrain, P. Joly, P. Lepain, and V. Longueville. Representation-based user interfaces for the audiovisual library of year 2000. *Proceedings of SPIE, 2417, Multimedia Computing and Networking*, pages 35–45, 1995.

J. F. Alm and J. S. Walker. Time-Frequency Analysis of Musical Instruments. *Society for Industrial and Applied Mathematics Review*, 44(3):457–476, 2002.

X. Amatriain, J. Massaguer, D. Garcia, and I. Mosquera. The CLAM Annotator: A Cross-platform Audio Descriptors Editing Tool. *Proceedings of the 6 thInternational Conference on Music Information Retrieval, London, UK*, 2005.

M. Attik, L. Bougrain, and F. Alexandre. Self-organizing Map Initialization. *LECTURE NOTES IN COMPUTER SCIENCE*, 3696:357, 2005.

J. J. Aucouturier and M. Sandler. Using Long-Term Structure to Retrieve Music: Representation and Matching. *Proc. International Symposium on Music Information Retrieval*, 2001.

M. A. Bartsch and G. H. Wakefield. To catch a chorus: using chroma-based representations for audiothumbnailing. *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*, pages 15–18, 2001.

M. A. Bartsch and G. H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *Multimedia, IEEE Transactions on*, 7(1):96–104, 2005.

M. Biggs. Visualisation and Wittgenstein's Tractatus. 2002.

S. Blackburn and D. DeRoure. A Tool for Content Based Navigation of Music. In *Proceedings of ACM Multimedia.* ACM, Bristol, UK, 1998. ISBN 1-58113-036-8.

R. A. W. Bladon and B. Lindblom. Modeling the judgment of vowel quality differences. *The Journal of the Acoustical Society of America*, 69:1414, 1981.

A. Brinkman and M. Mesiti. Graphic modeling of musical structure. *Computers in Music Research*, 3:1–42, 1991.

C. J. C. Burges, D. Plastina, J. C. Platt, E. Renshaw, and H. S. Malvar. Using audio fingerprinting for duplicate detection and thumbnail generation. *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, 3, 2005.

R. Callan. *The Essence of Neural Networks.* Prentice-Hall, 1999. ISBN 0-13-908732-X.

C. Cannam, C. Landone, M. Sandler, and J. P. Bello. The Sonic Visualiser: A Visualisation Platform For Semantic Descriptiors From Musical Signals. *Proceedings of ISMIR, Victoria, Canada*, 2006.

A. Cemgil and F. Gürgen. Classification of Musical Instrument Sounds using Neural Networks. In *Proceedings of the SIU97*, 1997. URL `citeseer.nj.nec.com/cemgil97classification.html`.

W. Chai and B. Vercoe. Music Thumbnailing via Structural Analysis. *Proceedings of the eleventh ACM international conference on Multimedia, November*, pages 02–08, 2003a.

W. Chai and B. Vercoe. Structural analysis of musical signals for indexing and thumbnailing. *Digital Libraries, 2003. Proceedings. 2003 Joint Conference on*, pages 27–34, 2003b.

E. Chew and A. François. Real-time Music Information Processing. *Proceedings of the 31st International Conference for Computers and Industrial Engineering, ICCIE2003, San Francisco, CA, February*, pages 2–4, 2003.

P. Clarke. Folktronica. *BBC Collective, Editor's Review*, 2003. URL `http://www.bbc.co.uk/dna/collective/A1120555`.

R. Cogan. *New Images of Musical Sound.* Harvard University Press, 1984.

M. Cooper and J. Foote. Summarizing Popular Music via Structural Similarity Analysis. In *Proceedings of ACM Multimedia*, pages 364–373, Berkeley, California, US, 2003. ACM.

P. Couprie. Graphical Representation: An Analytical and Publication Tool for Electroacoustic Music. *Organised Sound*, 9:109–113, 2004.

R. B. Dannenberg. A Brief Survey of Music Representation Issues, Techniques, and Systems. *Computer Music Journal*, 17(3):20–30, 1992.

M. Dittenbach, R. Neumayer, and A. Rauber. PlaySOM: An Alternative Approach to Track Selection and Playlist Generation in Large Music Collections. 2003.

A. Dix, J. Finlay, G. D. Abowd, and R. Beale. Human Computer Interaction. *Upper Saddle River, NJ*, 2004.

S. Dixon, W. Goebl, and G. Widmer. The Performance Worm: Real Time Visualisation of Expression based on Langners Tempo-Loudness Animation. *Proceedings of the International Computer Music Conference (ICMC 2002)*, 2002a.

S. Dixon, W. Goebl, and G. Widmer. Real Time Tracking and Visualisation of Musical Expression. *Music and Artificial Intelligence: Second International Conference, ICMAI2002*, pages 58–68, 2002b.

S. Dixon, E. Pampalk, and G. Widmer. Classification of Dance Music by Periodicity Patterns. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR '03)*, Baltimore, Maryland, US, 2003. Johns Hopkins University. URL `http://ismir2003.ismir.net/papers/Dixon.PDF`.

G. W. Don and J. S. Walker. Music: A Time-Frequency Approach. *Journal of Mathematics and Music*, 2006.

M. Dorfler. *Gabor Analysis for a Class of Signals called Music.* Ph. D. thesis, University of Vienna, Institut of Mathematics, August 2002, 2002.

J. P. Eckmann, S. O. Kamphorst, and D. Ruelle. Recurrence plots of dynamical systems. *Europhysics Letters*, 5:973–977, 1987. doi: 10.1209/0295-5075/4/9/004. URL `http://dx.doi.org/10.1209%2F0295-5075%2F4%2F9%2F004`.

L. Fausett. *Fundamentals of Neural Networks.* Prentice-Hall, 1994. ISBN 0-13-334186-0.

J. Foote. Automatic Audio Segmentation Using A Measure of Audio Novelty. In *Proceedings of IEEE International Conference on Multimedia and Expo*, volume 1, pages 452–455. IEEE, 2000a. URL `http://www.fxpal.com/people/foote/papers/footeICME00.pdf`.

J. Foote. ARTHUR: Retrieving Orchestral Music by Long-Term Structure. *International Symposium on Music Information Retrieval*, 1, 2000b.

J. Foote. Methods for the Automatic Analysis of Music and Audio. Technical report, 1999a. URL `citeseer.nj.nec.com/foote99methods.html`.

J. Foote. Visualizing Music and Audio Using Self Similarity. In *Proceedings of ACM Multimedia*, pages 77–80, Orlando, Florida, US, nov 1999b. ACM.

J. Foote and M. Cooper. Media Segmentation using Self-Similarity Decomposition. In *Proceedings of the SPIE Storage and Retrieval for Multimedia Databases*, volume 5021, pages 167–175, San Jose, California, US, 2003. SPIE. URL `http://www.fxpal.com/people/foote/papers/SPIE02.PDF`.

E. Gómez and J. Bonada. Tonality Visualization of Polyphonic Audio. *Proceedings of the International Computer Music Conference*, pages 57–60, 2005.

Google. Google web search, 2007. URL `http://www.google.com`.

J. Guessford, H. Kaper, and S. Tipei. Loudness Scaling in a Digital Synthesis Library. *Proceedings of the International Computer Music Conference, Miami, Florida*, 2004.

S. Hainsworth. *Techniques for the Automated Analysis of Musical Audio.* PhD thesis, Ph. D. thesis, Cambridge University Engineering Department, Cambridge, UK, 2004, 2004.

R. Hamming. *Digital Filters.* Dover Publications, 1998.

R. Hiraga. Musical Performance Visualization, 2006. URL `http://www.bunkyo.ac.jp/~rhiraga/researchhi.html`.

R. Hiraga and N. Matsuda. Graphical expression of the mood of music. *Multimedia and Expo, 2004. ICME'04. 2004 IEEE International Conference on*, 3, 2004a.

R. Hiraga and N. Matsuda. Visualization of music performance as an aid to listener's comprehension. *Proceedings of the working conference on Advanced visual interfaces*, pages 103–106, 2004b.

R. Hiraga, R. Mizaki, and I. Fujishiro. Performance visualization: a new challenge to music through visualization. *Proceedings of the tenth ACM international conference on Multimedia*, pages 239–242, 2002a.

R. Hiraga, F. Watanabe, and I. Fujishiro. Music learning through visualization. *Web Delivering of Music, 2002. WEDELMUSIC 2002. Proceedings. Second International Conference on*, pages 101–108, 2002b.

J. Hirschberg and J. Choi. "I just played that a minute ago!:" Designing User Interfaces for Audio Navigation. *Proceedings of Content Visualization and Intermedia Representations CVIR'98*, 1998.

J. Hirschberg, S. Whittaker, D. Hindle, F. Pereira, and A. Singhal. Finding information in audio: A new paradigm for audio browsing/retrieval. *Proceedings of the ESCA workshop: Accessing information in spoken audio*, pages 117–122, 1999.

B. Holland. A man who sees what other people hear. *The New York Times*, page C28, 19 November 1981.

K. A. S. Immink. The compact disc story. *Journal of the Audio Engineering Society*, 46 (5):458–465, 1998.

E. Isaacson. Content visualization in a digital music library. *Third International Workshop on Information Visualization Interfaces for Retrieval and Analysis (IVIRA) at JCDL*, 2003.

E. Isaacson. What You See is What You Get: On Visualizing Music. *Proceedings of the Sixth International Conference on Music Information Retrieval*, pages 389–395, 2005.

I. Jolliffe. *Principal Component Analysis*. Springer, 2002.

H. Kaper. Manifold compositions, music visualization, and scientific sonification in an immersive virtual-reality environment. *1998 International Computer Music Conference (ICM98), Ann Arbor, MI (US), 10/01/1998-10/06/1998*, 1998.

J. Kasson and W. Plouffe. An analysis of selected computer interchange color spaces. *ACM Transactions on Graphics (TOG)*, 11(4):373–405, 1992.

D. Kimber and L. Wilcox. Acoustic segmentation for audio browsers, 1996. URL `citeseer.ist.psu.edu/article/kimber96acoustic.html`.

D. Kimber, L. Wilcox, F. Chen, and T. Moran. Speaker segmentation for browsing recorded audio. *Conference on Human Factors in Computing Systems*, pages 212–213, 1995.

P. Knees, T. Pohle, M. Schedl, and G. Widmer. Automatically Describing Music on a Map. *Proc. LSAS '06*, 2006.

M. Kobayashi and C. Schmandt. Dynamic Soundscape: mapping time to space for audio browsing. *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 194–201, 1997.

T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

T. Kohonen. Fast Evolutionary Learning with Batch-Type Self-Organizing Maps. *Neural Processing Letters*, 9(2):153–162, 1999.

T. Kohonen. *Kohonen Network*, page 7421. Scholarpedia, 2007a. URL `http://www.scholarpedia.org/article/Kohonen_Network`.

T. Kohonen. The self-organizing map (som). WWW, 2007b. URL `http://www.cis.hut.fi/projects/somtoolbox/theory/somalgorithm.shtml`.

T. Kohonen. Self-Organised Formation of Topologically Correct Feature Maps. *Biological Cybernetics*, 43:59–69, 1982.

P. Kolhoff, J. Preuß, and J. Loviscach. Music Icons: Procedural Glyphs for Audio Files. *Computer Graphics and Image Processing, 2006. SIBGRAPI'06. 19th Brazilian Symposium on*, pages 289–296, 2006.

T. Kosonen and A. Eronen. Rhythm metadata enabled intra-track navigation and content modification in a music player. *Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*, 2006.

P. Lepain. SATIE: an interactive software for listening to musical recordings. *Proceedings of the fourth ACM international conference on Multimedia*, pages 413–414, 1997.

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

M. Levy, M. Sandler, and M. Casey. Extraction of high-level musical structure from audio data and its application to thumbnail generation. *Proc. ICASSP*, 2006.

A. Lillie. Visualizing Music. 2007. URL `http://flyingpudding.com/projects/viz_music/`.

B. Logan and S. Chu. Music summarization using key phrases. *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, 2, 2000.

B. Logan and A. Salomon. A Content-Based Music Similarity Function. Technical report, Cambridge Research Laboratory, 2001. URL `http://crl.research.compaq.com/publications/techreports/reports/crl-tr-2001-2.pdf`.

D. Lubbers. Sonixplorer: Combining Visualization and Auralization for Content-based Exploration of Music Collections. *Proc. of the 6th International Conference on Music Information Retrieval (ISMIR05), London, UK*, 2005.

D. MacAdam. Visual sensitivities to color differences in daylight. *J. Opt. Soc. Am*, 32(5): 247–274, 1942.

S. Malinowski. The Music Animation Machine. 2001. URL `http://www.well.com/user/smalin/mam.html`.

D. Mazzoni and R. Dannenberg. Audacity Sound Editor, 2005. URL `audacity.sf.net`.

R. Middleton. *Studying Popular Music.* Open University Press, 1990.

F. Morchen, A. Ultsch, M. Nocker, and C. Stamm. Databionic visualization of music collections according to perceptual distance. *Proc. of the 6th International Conference on Music Information Retrieval (ISMIR05), London, UK*, 2005.

A. Munsell. A Pigment Color System and Notation. *The American Journal of Psychology*, 23(2):236–244, 1912.

C. Nakatani, S. Whittaker, and J. Hirshberg. Now you hear it, now you dont: Empirical Studies of Audio Browsing Behavior. *Proceedings of the Fifth International Conference on Spoken Language Processing,(SLP98)*, 1998.

M. Oja, S. Kaski, and T. Kohonen. Bibliography of self-organizing map (SOM) papers: 1998-2001 addendum. *Neural Computing Surveys*, 3(1):1–156, 2003.

A. Okabe, B. Boots, and K. Sugihara. *Spatial tessellations: concepts and applications of Voronoi diagrams.* John Wiley, 2000. ISBN 978-0471986355.

E. Pampalk. *Islands of Music.* PhD thesis, Institut fur Softwaretechnik und Interaktive Systeme der Technischen Universitat Wien, December 2001.

E. Pampalk, A. Rauber, and D. Merkl. Content-based Organization and Visualization of Music Archives. In *Proceedings of the ACM Multimedia*, Juan les Pins, France, 2002. ACM. URL `citeseer.nj.nec.com/pampalk02contentbased.html`.

E. Pampalk, S. Dixon, and G. Widmer. On the evaluation of perceptual similarity measures for music. *Proc Intl Conf on Digital Audio Effects*, 2003.

E. Pampalk, S. Dixon, and G. Widmer. Exploring Music Collections by Browsing Different Views. *Computer Music Journal*, 28(2):49–62, 2004.

J. Paulus and A. Klapuri. Music structure analysis by finding repeated parts. *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*, pages 59–68, 2006.

K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

P. J. Ponce de León and J. M. Iñesta. Feature-driven recognition of music styles. *1st Iberian Conference on Pattern Recognition and Image Analysis. Lecture Notes in Computer Science*, 2652:773–781, 2003.

P. J. Ponce de León and J. M. Iñesta. Musical Style Identification Using Self-Organising Maps. *Proceedings of the Second International Conference on WEB Delivering of Music WEDELMUSIC*, 2:82–92, 2002.

J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey. *Human-Computer Interaction.* Addison Wesley, Wokingham, UK, 1994. ISBN 0-201-62769-8.

C. Raphael. Automatic Segmentation of Acoustic Musical Signals Using Hidden Markov Models. In *Transactions on Pattern Analysis and Machine Intelligence*, volume 31. IEEE, 1999.

A. Rauber and M. Frühwirth. Automatically Analyzing and Organizing Music Archives. In *ECDL '01: Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries*, volume 2163, pages 402–414, London, UK, 2001. Springer-Verlag. ISBN 3-540-42537-3. URL `citeseer.nj.nec.com/rauber01automatically.html`.

A. Rauber, E. Pampalk, and D. Merkl. Using Psycho-Acoustic Models and Self-Organizing Maps to Create a Hierarchical Structuring of Music by Sound Similarities. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR '02)*, Centre Pompideu, Paris, France, 2002. IRCAM. URL `citeseer.nj.nec.com/article/rauber02using.html`.

C. Rhodes, M. Casey, S. A. Abdallah, and M. Sandler. A Markov-chain Monte-Carlo approach to musical audio segmentation. *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2006.

H. Ritter. Self-organizing maps on non-euclidean spaces. *Kohonen Maps*, pages 97–110, 1999.

D. Roy and C. Schmandt. NewsComm: a hand-held interface for interactive access to structured audio. *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, pages 173–180, 1996.

C. Sapp. Harmonic Visualizations of Tonal Music. *Proceedings of the International Computer Music Conference*, pages 423–430, 2001.

N. Sawhney and A. Murphy. ESPACE 2: an experimental hyperaudio environment. *Conference on Human Factors in Computing Systems*, pages 105–106, 1996.

B. Scharf. Critical bands. *Foundations of Modern Auditory Theory*, 1:157–202, 1970.

C. Schmandt and D. Roy. Using acoustic structure in a hand-held audio playback device. *IBM Systems Journal*, 35(3):453–472, 1996.

J. Simpson, E. Weiner, et al. The Oxford English Dictionary. 1989.

K. Sjölander and J. Beskow. WaveSurfer–an open source speech tool. *Proc. ICSLP*, 4: 464–467, 2000.

D. M. Skapura. *Building Neural Networks.* ACM Press Books, 1996. ISBN 0-201-53921-7.

S. M. Smith and G. N. Williams. A Visualization of Music. IEEE, 1997. ISBN 0-8186-8262-0.

J. Snydal and M. Hearst. ImproViz: Visual Explorations of Jazz Improvisations. In *Proceedings of the Conference on Human Factors in Computing Systems (SIGCHI '05)*, Portland, Oregon, US, 2005. ACM Press.

R. Spence. *Information visualization.* Addison-Wesley Harlow, 2001.

S. Sterrett. Pictures of Sounds: Wittgenstein on Gramophone Records and the Logic of Depiction. 2004. URL `h:ttp://philsci-archive.pitt.edu/archive/00002019/01/SterrettPicturesOfSoundsR1.pdf`.

S. Stevens and J. Volkmann. The Relation of Pitch to Frequency: A Revised Scale. *The American Journal of Psychology*, 53(3):329–353, 1940.

E. Terhardt. Impact of computers on music: an outline. *Music, Mind, and Brain: The Neuropsychology of Music*, pages 353–369, 1982.

The Amarok Team. amaroK Music Player, 2005. URL `amarok.kde.org`.

P. Toiviainen. Visualization of tonal content with self-organizing maps and self-similarity matrices. *Computers in Entertainment (CIE)*, 3(4):1–10, 2005.

P. Toiviainen and T. Eerola. A method for comparative analysis of folk music based on musical feature. *Conference Program, Proceedings & List of Participants, VII International Symposium on Systematic and Comparative Musicology III International Conference on Cognitive Musicology 2001 Jyväskylä, Finland*, 2001.

P. Toiviainen and T. Eerola. A computational model of melodic similarity based on multiple representations and self-organizing maps. *Proceedings of the 7th International Conference on Music Perception and Cognition, Sydney*, pages 236–239, 2002.

G. Tzanetakis and P. R. Cook. Audio Information Retrieval Tools. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR).* ISMIR, 2000a.

G. Tzanetakis and P. R. Cook. Experiments in Computer-Assisted Annotation of Audio . In *Proceedings of the International Conference on Auditory Display (ICAD '00).* ICAD, 2000b. URL `http://www.icad.org/websiteV2.0/Conferences/ICAD2000/PDFs/Tzanetakis.pdf/experiments-in-computer-assisted.pdf`.

G. Tzanetakis and P. R. Cook. MARSYAS: A Framework for Audio Analysis. *Organized Sound*, 2000c.

G. Tzanetakis and P. R. Cook. Multifeature Audio Segmentation for Browsing and Annotation. In *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, US, oct 1999. IEEE. URL `http://soundlab.cs.princeton.edu/publications/1999_waspaa_mfas.pdf`.

G. Tzanetakis, G. Essl, and P. R. Cook. Automatic Musical Genre Classification of Audio Signals. In *Proceedings of the Internarional Conference on Music Information Retrieval (ISMIR '01)*, pages 205–210, Bloomington, Indiana, USA, oct 2001. ISMIR. URL `citeseer.nj.nec.com/tzanetakis01automatic.html`.

S. Vembu and S. Baumann. A Self-Organizing Map Based Knowledge Discovery for Music Recommendation Systems. *Proc. of the 2nd Internation Symposium on Computer Music Modeling and Retrieval (CMMR04), Esbjerg, Denmark*, 2004.

T. Weyde. Dynamic and Interactive Visualisations of MPEG Symbolic Music Representation. Presented to the Fifth Open Workshop of the Interactive Music Network, 2005. URL `http://www.google.co.uk/url?sa=t\&ct=res\&cd=4\&url=http%3A%2F%2Fwww.interactivemusicnetwork.org%2Fevents%2FFifth_OpenWorkshop_2005%2Fpapers%2F12.doc\&ei=0_xNRtvJL42E0gTi-4CKDg\&usg=AFrqEzctpTKDy21nw-qY0aNJHk4DUBGHHQ\&sig2=YluV3NUZpfy2IN7i0AfW8w`.

T. Weyde and J. Wissmann. Visualization of musical structure with maps. *Proceedings of the First Conference on Interdisciplinary Musicology. Graz, Austria.*, 2004.

S. Whittaker, L. Terveen, and B. Nardi. Let's stop pushing the envelope and start addressing it: a reference task agenda for HCI. *Human-Computer Interaction*, 15(2/3): 75–106, 2000.

G. Widmer and W. Goebl. Computational Models of Expressive Music Performance: The State of the Art. *Journal of New Music Research*, 33(3):203–216, 2004.

L. Wittgenstein. *Tractatus Logico-Philosophicus.* Kessinger Publishing, 2004.

G. Wood and S. E. O'Keefe. Quantitative Comparisons into Content-based Music Recognition with the Self-Organising Map. In *Proceedings of the Internarional Conference on Music Information Retrieval (ISMIR '03)*, Baltimore, Maryland, US, 2003. Johns Hopkins University.

G. Wood and S. E. O'Keefe. A Case Study of Distributed Musical Audio Analysis Using the Geddei Processing Framework. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR '04)*, pages 44–47, Barcelona, Spain, 2004. Audiovisual Institute, Pompeu Fabra University. ISBN 84-8804244-2. URL `http://ismir2004.ismir.net/proceedings/p009-page-44-paper158.pdf`.

G. Wood and S. E. O'Keefe. On Techniques for Content-Based Visual Annotation to Aid Intra-Track Music Navigation. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR '05)*, pages 58–65, London, UK, 2005. Queen Mary, University of London. ISBN 0-9551179-0-9. URL `exscalibar.sourceforge.net/files/ismir-2005.pdf`.

R. Wright. Art and Science in Chaos: Contesting Readings of Scientific Visualization. *Proceedings of the 5th International Symposium on Electronic Art, 1994*, 1994.

W. D. Wright. A re-determination of the trichromatic coefficients of the spectral colours. *Transactions of the Optical Society*, 30(4):141–164, 1929.

B. Yandell. *Practical Data Analysis for Designed Experiments.* Chapman & Hall/CRC, 1997.