Making music through real-time voice timbre analysis: machine learning and timbral control

Stowell, Dan

Making music through real-time voice timbre analysis: machine learning and timbral control

Dan Stowell

PhD thesis

School of Electronic Engineering and Computer Science
Queen Mary University of London

2010

Abstract

People can achieve rich musical expression through vocal sound – see for example human beatboxing, which achieves a wide timbral variety through a range of extended techniques. Yet the vocal modality is under-exploited as a controller for music systems. If we can analyse a vocal performance suitably in real time, then this information could be used to create voice-based interfaces with the potential for intuitive and fulfilling levels of expressive control.

Conversely, many modern techniques for music synthesis do not imply any particular interface. Should a given parameter be controlled via a MIDI keyboard, or a slider/fader, or a rotary dial? Automatic vocal analysis could provide a fruitful basis for expressive interfaces to such electronic musical instruments.

The principal questions in applying vocal-based control are how to extract musically meaningful information from the voice signal in real time, and how to convert that information suitably into control data. In this thesis we address these questions, with a focus on timbral control, and in particular we develop approaches that can be used with a wide variety of musical instruments by applying machine learning techniques to automatically derive the mappings between expressive audio input and control output. The vocal audio signal is construed to include a broad range of expression, in particular encompassing the extended techniques used in human beatboxing.

The central contribution of this work is the application of supervised and unsupervised machine learning techniques to automatically map vocal timbre to synthesiser timbre and controls. Component contributions include a delayed decision-making strategy for low-latency sound classification, a regression-tree method to learn associations between regions of two unlabelled datasets, a fast estimator of multidimensional differential entropy and a qualitative method for evaluating musical interfaces based on discourse analysis.

Acknowledgements

I'd like to thank everyone in the **C4DM** at QMUL, who've been brilliant folks to work with, with such a wide range of backgrounds and interests that have made it an amazing place to develop my ideas. In particular my supervisor Mark Plumbley, whose wide-ranging knowledge and enthusiasm has been invaluable in every aspect of my PhD work.

Also I must acknowledge the **SuperCollider** community, since it's not just the main tool I use but also the software & community that enabled me to dabble with these ideas before even contemplating a PhD. Thanks to Nick Collins and Julian Rohrhuber, who got me going with it during their summer school, plus all the other friends I've made at symposiums/workshops as well as online. And James McCartney for creating SuperCollider in the first place, and his forward-thinking approach to language design which made it a pleasure to work with.

Philippa has to get major credit for at least some of this thesis, not only for much proofreading and discussion, but also moral & practical support throughout the whole three or four years. It wouldn't have been possible without you.

Shout outs to:

toplap

humanbeatbox.com

F-Step massive

Ladyfest

dorkbot london

goto10

openlab london

Also a shout out to Matt Sharples for some inspiration in the 8-bit department which found its way into all this, and to my family for all their support.

Most of the tools I use are **open-source**, meaning that they're maintained by miscellaneous hordes of (often unpaid) enthusiasts around the world. So I'd also like to take the time to acknowledge the thousands who contributed to the excellence of the open-source tools that allowed me to do my PhD research so smoothly. Besides SuperCollider, these were crucial: vim, Python+numpy, gnuplot, BibDesk, LaTeX, TeXshop, jackd, Inkscape, Sonic Visualiser.

Licence

Contents

List of Figures

List of Tables

List of abbreviations

AAMRT	Auto-Associative Multivariate Regression Tree
ANN	All Nearest Neighbours
ANSI	American National Standards Institute
ASR	Automatic Speech Recognition
CART	Classification And Regression Trees
DA	Discourse Analysis
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform
FM	Frequency Modulation
GMM	Gaussian Mixture Model
HCI	Human-Computer Interaction
HMM	Hidden Markov Model
IPA	International Phonetic Alphabet
KDE	Kernel Density Estimation
LTI	Linear Time-Invariant
MDS	Multi-Dimensional Scaling
MFCC	Mel-Frequency Cepstral Coefficient
MI	Mutual Information
MIR	Music Information Retrieval
MRT	Multivariate Regression Tree
MUSHRA	Multi-Stimulus test with Hidden Reference and Anchor
NIME	New Interfaces for Musical Expression
NN	Nearest-Neighbour
PCA	Principal Components Analysis
PIT	Probability Integral Transform
SOM	Self-Organising Map
SVS	Singing Voice Synthesis
Synth	Synthesiser (musical instrument)
XAMRT	Cross-Associative Multivariate Regression Tree
ZCR	Zero Crossing Rate

Chapter 1

Introduction

1.1 Motivation

The human voice is a wonderfully, perhaps uniquely, expressive instrument. It can exhibit a bewildering number of expressive variations beyond those of pitch and loudness, including trill, effort level, breathiness, creakiness, growl, twang [Soto-Morettini, 2006]. One may scarcely believe that the same basic apparatus is used to create such disparate sounds as heard in Mongolian/Tuvan throat singing [Lindestad et al., 2001], Inuit vocal games [Nattiez, 2008], twentieth-century art music [Mabry, 2002] and human beatboxing (Section 2.2). Even in Western popular music, singers regularly exploit a variety of modulation techniques for musical expression [Soto-Morettini, 2006]. Further, most people are able to use their voice expressively – in speech even if not necessarily in a trained musical manner.

Such vocal expression is a rich source of information, which we perceive aurally and which may be amenable to automatic analysis. There has been much research into automatic speech analysis, and relatively little on automatic singing analysis (see Chapter 2); and very little indeed that aims to encompass the breadth of vocal timbral expression which we might call *extended technique*. Yet if we can analyse/parametrise vocal expression in a suitable manner in real time, then a voice-based musical interface has the potential to offer a level of expression that could be intuitive and fulfilling for the performer.

Conversely, although traditional musical instruments such as the guitar or piano come with their own physical interface, many modern techniques for music synthesis do not imply any particular interface. For example, algorithmic processes such as granular synthesis [Roads, 1988] or concatenative synthesis [Schwarz, 2005] can be controlled by manipulating certain numerical parameters. Should a given parameter be controlled via a MIDI keyboard, or a slider/fader,

or a rotary dial? The history of electronic instruments throughout the twentieth century has shown a tendency for the piano-like MIDI keyboard to prevail. We concur with Levitin et al. [2002] who argue:

> Our approach is the consequence of one bias that we should reveal at the outset: we believe that electronically controlled (and this includes computer-controlled) musical instruments need to be emancipated from the keyboard metaphor; although piano-like keyboards are convenient and familiar, they limit the musician's expressiveness (Mathews 1991, Vertegaal and Eaglestone 1996, Paradiso 1997, Levitin and Adams 1998). This is especially true in the domain of computer music, in which timbres can be created that go far beyond the physical constraints of traditional acoustic instruments. [Levitin et al., 2002]

Such motivation spurs a wide range of research on new interfaces for musical expression [Poupyrev et al., 2001]. We believe that automatic vocal analysis could provide a fruitful basis for expressive interfaces to electronic musical instruments. Indeed, there is evident appetite for technology which extends the range of possibilities for vocal expression, shown in musicians' take-up of vocoder and Auto-Tune effects [Tompkins, 2010, Dickinson, 2001] (note that these technologies alter a vocal signal rather than using it to control another sound source).

The principal questions in applying vocal-based control are how to extract musically meaningful information from the voice signal in real time, and how to convert that information suitably into control data. In the present work we address these questions, and in particular we develop approaches that can be used with a wide variety of musical instruments by applying machine learning techniques to automatically derive the mappings between expressive audio input and control output.

1.2 Aim

The aim of this work is to develop methods for real-time control of synthesisers purely using a vocal audio signal. The vocal audio signal is construed to include a broad range of expression, in particular encompassing the extended techniques used in human beatboxing. The real-time control should be suitable for live expressive performance, which brings requirements such as low-latency and noise robustness. The choice of synthesiser should be left open, which means that we must apply machine learning techniques to automatically analyse the relationship between the synthesiser's controls and output.

1.3 Thesis structure

Chapter 2 introduces the main bodies of existing research which we will build upon. It begins by considering the physiology of the human vocal tract and the sounds used in beatboxing, and then surveys relevant research topics including speech analysis, singing voice analysis, musical timbre, and machine learning. The chapter concludes by reflecting on this existing work to consider a strategy for achieving the research aim.

Chapter 3 focuses on the representation of timbre using features measured on the audio signal. We investigate the relative merits of a diverse set of features, according to perceptual and other criteria which are each relevant to our choice of features for use in our timbral applications. The chapter finds some commonalities and tensions between these criteria, and makes some recommendations about choice of features.

Chapter 4 investigates the event-based paradigm applied to musical control by voice timbre. We describe a human beatboxing dataset which we compiled, and classification experiments performed on these data. In particular, we investigate latency issues, finding that a small latency is beneficial to the classifier, and perform a perceptual experiment with human listeners, determining the acceptable bounds on latency in a novel "delayed decision-making" real-time classification approach.

Chapter 5 investigates the continuous (event-agnostic) paradigm applied to musical control by voice timbre. We introduce our concept of "timbre remapping" from voice timbre onto synthesiser timbre, and consider various strategies for automatic machine learning of mappings from unlabelled data. In particular, we introduce a novel regression-tree method, and demonstrate that it outperforms a nearest-neighbour-type mapping.

Chapter 6 evaluates timbre remapping in use with actual beatboxers. We first discuss evaluation issues for expressive musical systems, finding that some of the traditional HCI techniques are not ideally suited to such evaluation. We then introduce a rigorous qualitative evaluation method, and apply it to evaluate a timbre remapping system, illuminating various aspects of the technique in use.

Chapter 7 concludes the thesis, drawing comparisons and contrasts between the event-based and continuous approaches to vocal timbral control, and considering the prospects for further research.

1.4 Contributions

The principal contributions of this thesis are:

- Chapter 4: a "delayed decision-making" strategy to circumvent the issue of latency in real-time audio event classification, and perceptual results indicating bounds on its applicability.
- Chapter 5: a nonparametric method based on regression trees which can learn associations between regions of two unlabelled datasets.
- Chapter 5: The use of the above-mentioned tree-based method to improve "timbre remapping" from one type of sound to another, by accounting for the differences in timbre distributions of sound sources.
- Chapter 6: a novel approach to evaluating creative/expressive interfaces in a rigorous qualitative fashion, using discourse analysis.
- Appendix A: a fast estimator of the differential entropy of multidimensional distributions.

1.5 Associated publications

Portions of the work detailed in this thesis have been presented in national and international scholarly publications, as follows (journal publications highlighted in bold):

- Chapter 2: Section 2.2 on beatboxing was published as a technical report [Stowell and Plumbley, 2008a].
- Chapter 3: An early version of some of the feature-selection work was presented at the International Conference on Digital Audio Effects [Stowell and Plumbley, 2008b].
- Chapter 4: Accepted for publication in the **Journal of New Music Research** [Stowell and Plumbley, in press].
- Chapter 5: The early timbre remapping work presented in sections of this chapter was presented at a meeting of the Digital Music Research Network [Stowell and Plumbley, 2007].

  A version of the regression tree work (Section 5.2) is submitted to a journal.
  A briefer presentation (focusing on the application to concatenative synthesis) was presented at the Sound and Music Conference [Stowell and Plumbley, 2010].
  A discussion of the three timbre remapping methods is accepted for presentation at the 2010 Workshop on Applications of Pattern Analysis [Stowell and Plumbley, accepted].
- Chapter 6: The discourse analytic approach to evaluation was presented in an early form at the International Conference on New Interfaces for Musical Expression [Stowell et al., 2008], and in a more complete form in a collaborative article of which I was the lead author, in the **International Journal of Human-Computer Studies** [Stowell et al., 2009].
- Appendix A: Published in **IEEE Signal Processing Letters** [Stowell and Plumbley, 2009].

Chapter 2

Background

To establish the basis upon which this thesis will be developed, in this chapter we introduce the main research areas which relate to our aim. We start by discussing the components and operation of the human vocal system, which will be useful in our discussion of speech and singing research and in later chapters. We also discuss specific characteristics of the beatboxing vocal style. We then introduce the main research fields which bear on our thesis. We conclude the chapter by reflecting upon how the state of the art in these fields bears upon our choice of strategy.

2.1 The human vocal system

Figure 2.1 gives a functional model of the vocal tract [Clark and Yallop, 1995]. The energy used to produce vocal sound comes primarily from the respiratory forces moving air into or out of the lungs.[1] To produce *vocalic* sounds (vowels and similar sounds such as voiced consonants or humming) the vocal folds are brought close together such that the passage of air is constricted, creating a pressure drop across the vocal folds which can cause them to oscillate. Variations in the muscular tension in the vocal folds are used to modulate the fundamental frequency of the oscillation as well as some of its harmonic characteristics: for example the relative amount of time during an oscillation that the folds remain apart (characterised by the *glottal open quotient* or conversely the *glottal closed quotient*) determines the relative strengths of harmonics in the glottal oscillation [Hanson, 1995].

[1] The vast majority of vocalisations are performed while exhaling rather than inhaling. Inhaled sounds are phonetic units in some languages [Ladefoged and Maddieson, 1996] and are used for performed sounds in traditions such as Inuit vocal games [Nattiez, 2008] and human beatboxing (Section 2.2).

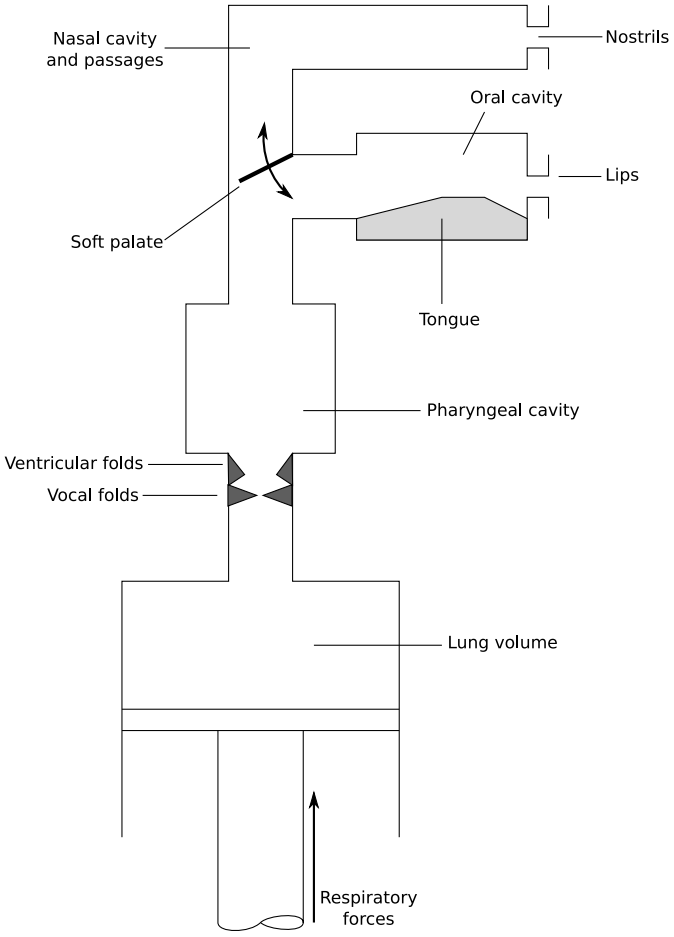


Figure 2.1: Functional model of the vocal tract, after Clark and Yallop [1995, Figure 2.2.1].

The vocal folds are therefore the main source of acoustic oscillations that propagate through the rest of the vocal system. The vocal tract contains regions which we model as resonant chambers, in particular the pharyngeal cavity and the oral cavity. The size and shape of these cavities can be modulated by various muscle movements (including the position and shape of the tongue and lips) to determine the main resonant frequencies excited by the glottal signal, which will be audible in the sound emitted to the outside. These resonances are called *formants* in literature on voice, and their frequencies and character are important in differentiating vowels from one another.

The nasal cavity also has a role in modulating the sound. The soft palate (velum), normally open in breathing to allow air through the nasal cavity, is closed when producing vowels to force most or all of the air to pass through the

oral cavity. Some vocal sounds are *nasalised*, with the soft palate opening to allow some of the energy to pass into the nasal branch of the vocal tract. The audible result is a set of formants due to nasal radiation as well as *antiformants* as energy is removed from the oral radiated sound [Fujimura and Erickson, 1999, Section 2.4.3]. (In the terminology of filter design, a formant corresponds to a pole, and an antiformant to a zero, in a filter response.)

The musculature around the vocal folds is highly configurable. We have already mentioned that it can change the fundamental frequency of the oscillation and the glottal closed quotient; it can also induce a variety of different phonation types, called *modes* of phonation [Laver, 1980][Clark and Yallop, 1995, Section 2.6]:

- The most common mode is (a little confusingly) referred to as *modal phonation*, in which the vocal folds oscillate regularly and with a glottal closure on each cycle.
- *Whispering* is a mode in which the vocal folds are held moderately wide apart such that no oscillation occurs; rather, the slight constriction creates a turbulence in the airstream which creates a broadband noise (resulting in an inharmonic sound).
- *Breathy* voice is a related mode in which the vocal folds meet along only some of their length, resulting in a glottal source signal which is a mixture of regular oscillation and turbulent noise.
- *Creaky* voice (often called *vocal fry* in musical contexts) is produced when the vibration of the vocal folds turns on and off repeatedly (because the system is on the cusp of oscillation), causing the glottal signal to contain significant sub- and interharmonics [Gerratt and Kreiman, 2001].
- *Ventricular mode phonation* occurs when the ventricular vocal folds (also called "false" or "vestibular" vocal folds; see Figure 2.1) are brought into sympathetic resonance with the vocal folds, causing a rich low-pitched oscillation used notably in Tibetan chant and Tuvan/Mongolian throat-singing [Fuks, 1998, Lindestad et al., 2001].

The taxonomy of vocal modes varies among authors but the above are quite common. Differences between modes are sometimes used in language to mark phonemic differences (e.g. two vowels may differ only in whether they are modal or creaky), but in most languages they do not [Ladefoged and Maddieson, 1996]. The variation of vocal mode and its perception are partly categorical and partly continuous in nature [Gerratt and Kreiman, 2001], and can reflect the emotional or physiological state of the speaker.

The above description of the vocal tract has focused on vocalic phonation, with the vocal folds as the primary sound source. However, human vocalisations include a wide range of sounds with excitation sources at various points in the vocal tract [Fry, 1996/1979], used to varying extent in language. Some consonant-type sounds (fricatives such as /f/ /θ/ /s/ /h/)[2] are created by constricting the airflow at specific points such as the lips/tongue/teeth to create turbulence which results in audible noise. Trills are relatively slow oscillations (often 20–30 Hz [Fujimura and Erickson, 1999, Section 2.4]) produced by forcing air past a loose obstruction in the vocal tract (formed e.g. by the tongue or lips) which then oscillates between open and closed. Plosives are caused by blocking the airflow to build up pressure which is then released in a burst of sound. Clicks are percussive sounds caused for example by the tongue hitting the floor of the mouth.

In language and in vocal expression generally, these non-vocalic sounds can often be used in conjunction with vocalic phonation or independently. Vocalic phonation is usually the primary source of sound energy, and so other sources are often neglected in discussions of human voice and – as we will see in Section 2.3.1 – in automatic analyses. However if we wish to consider a wide range of human vocal expression, we must bear in mind that the human vocal apparatus includes various different potential sound sources. For example, the percussive sounds obtained by plosives, trills and clicks are important to vocal percussion performers such as human beatboxers, as we will next discuss specifically.

2.2 The beatboxing vocal style

Beatboxing is a tradition of vocal percussion which originates in 1980s hip-hop, and is closely connected with hip-hop culture. It involves the vocal imitation of drum machines as well as drums and other percussion, and typically also the simultaneous imitation of basslines, melodies and vocals, to create an illusion of polyphonic music. It may be performed *a capella* or with amplification. Here we describe some characteristics of the beatboxing vocal performance style, as relevant for the music signal processing which we will develop in our thesis. In particular we focus on aspects of beatboxing which are different from other vocal styles or from spoken language.

Beatboxing developed well outside academia, and separate from the vocal styles commonly studied by universities and conservatories, and so there is (to our knowledge) very little scholarly work on the topic, either on its history or

[2]Characters given between slashes are International Phonetic Alphabet (IPA) representations of vocal sounds [International Phonetic Association, 1999] (see also Fukui [2004]). For example, /θ/ represents an unvoiced "th" as in "theory".

on its current practice. Beatboxing is mentioned in popular histories of the hip-hop movement, although rarely in detail. An undergraduate thesis looks at phonetic aspects of some beatboxing sounds [Lederer, 2005]. Some technical work is inspired by beatboxing to create (e.g.) a voice-controlled drum-machine [Hazan, 2005a,b, Kapur et al., 2004, Sinyor et al., 2005], although these authors don't make explicit whether their work has been developed in contact with practising beatboxers.

In the following we describe characteristics of beatboxing as contrasted against better-documented traditions such as popular singing [Soto-Morettini, 2006] or classical singing [Mabry, 2002]. Because of the relative scarcity of literature, many of the observations come from the author's own experiences and observations: both as a participant in beatboxing communities in the UK and online, and during user studies involving beatboxers as part of the work described in this thesis.

In this section we will describe certain sounds narratively as well as in IPA notation; note that the IPA representation may be approximate, since the notation is not designed to accommodate easily the non-linguistic and "extended technique" sounds we discuss.

2.2.1 Extended vocal technique

Perhaps the most fundamental distinction between the sounds produced while beatboxing and those produced during most other vocal traditions arises from beatboxing's primary aim to create *convincing impersonations* of drum tracks. (Contrast this against vocal percussion traditions such as jazz *scat* singing or indian *bol*, in which percussive rhythms are imitated, but there is no aim to disguise the vocal origin of the sounds.) This aim leads beatboxers to do two things: (1) employ a wide palette of vocal techniques to produce the desired timbres; and (2) suppress some of the linguistic cues that would make clear to an audience that the source is a single human voice.

The extended vocal techniques used are many and varied, and vary according to the performer. Many techniques are refinements of standard linguistic vowel and consonant sounds, while some involve sounds that are rarely if at all employed in natural languages. We do not aim to describe all common techniques here, but we will discuss some relatively general aspects of vocal technique which have a noticeable effect on the sound produced.

Non-syllabic patterns

The musical sounds which beatboxers imitate may not sound much like conventional vocal utterances. Therefore the vowel-consonant alternation which is

typical of most use of voice may not be entirely suitable for producing a close auditory match. Instead, beatboxers learn to produce sounds to match the sound patterns they aim to replicate, attempting to overcome linguistic patternings. Since human listeners are known to use linguistic sound patterns as one cue to understanding a spoken voice [Shannon et al., 1995], it seems likely that avoiding such patterns may help maintain the illusion of non-voice sound.

As mentioned above, vocal traditions such as *scat* or *bol* do not aim to disguise the vocal origin of the sounds. Hence in those traditions, patterns are often built up using syllable sounds which do not stray far from the performers' languages.

Use of ingressive sounds

In most singing and spoken language, the vast majority of sounds are produced during exhalation; a notable characteristic of beatboxing is the widespread use of ingressive sounds. We propose that this has two main motivations. Firstly it enables a continuous flow of sounds, which both allows for continuous drum patterns and helps maintain the auditory illusion of the sounds being imitated (since the sound and the pause associated with an ordinary intake of breath are avoided). Secondly it allows for the production of certain sounds which cannot be produced equally well during exhaling. A commonly-used example is the "inward clap snare" /k͡p͡ˡ/.[3]

Ingressive sounds are most commonly percussive. Although it is possible to phonate while breathing in, the production of pitched notes while inhaling does not seem to be used much at all by beatboxers.

Although some sounds may be specifically produced using inward breath, there are many sounds which beatboxers seem often to be able to produce in either direction, such as the "closed hi-hat" sound /tˀ/ (outward) or /t͡sˀ/ (inward). This allows some degree of independence between the breathing patterns and the rhythm patterns.

Vocal modes/qualities

Beatboxers make use of different vocal modes (Section 2.1) to produce specific sounds. For example, growl/ventricular voice may be used to produce a bass tone, and falsetto is used as a component of some sounds, e.g. vocal scratch, "synth kick". In these cases the vocal modes are employed for their timbral effects, not (as may occur in language) to convey meaning or emotional state.

Some beatboxing techniques involve the alternation between voice qualities. If multiple streams are being woven into a single beat pattern, this can involve

[3] `http://www.humanbeatbox.com/inward_snares`

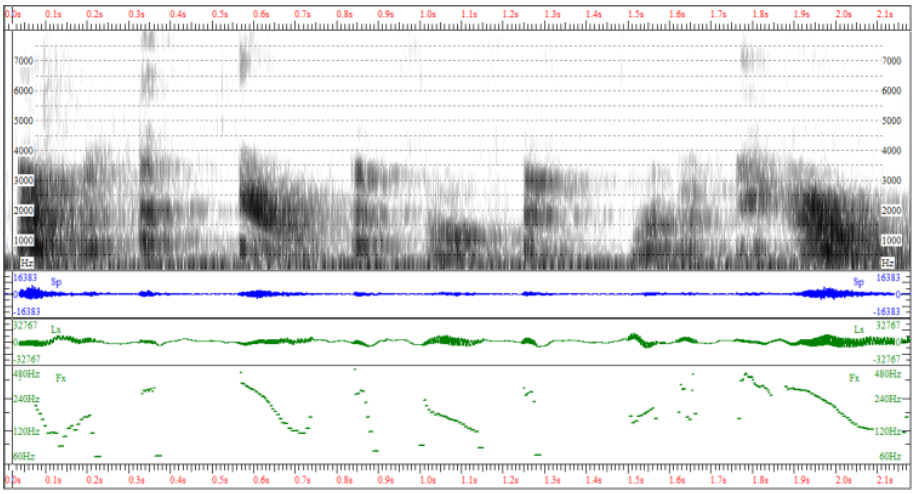


Figure 2.2: Laryngograph analysis of two seconds of "vocal scratching" performed by the author. The image shows (from top to bottom): spectrogram; waveform; laryngograph signal (which measures the impedance change when larynx opens/closes – the signal goes up when vocal folds close, goes down when they open); and fundamental frequency estimated from the laryngograph signal. The recording was made by Xinghui Hu at the UCL EAR Institute on 11th March 2008.

rapid alternation between (e.g.) beats performed using modal voice, "vocals" or sound effects performed in falsetto, and basslines performed in growl/ventricular voice. The alternation between voice qualities can emphasise the separation of these streams and perhaps contribute to the illusion of polyphony.

Fast pitch variation

Fast pitch variation is a notable feature of beatboxing, sometimes for similar reasons to the alternation of vocal modes described above, but especially in "vocal scratching". This is the vocal imitation of scratching (i.e. manually moving the record back and forth) used by DJs. Since real scratching involves very rapid variations in the speed of the record and therefore of the sound produced, its imitation requires very rapid variation in fundamental frequency, as well as concomitant changes in other voice characteristics. In laryngograph measurements made with Xingui Hu at the UCL EAR Institute (Figure 2.2), we observed pitch changes in vocal scratching as fast as one-and-a-half octaves in 150 ms.

Trills / rolls / buzzes

Beatboxers tend to use a variety of trills to produce oscillatory sounds. (Here we use the term "trill" in its phonetic sense, as an oscillation produced by a repeated blocking and unblocking of the airstream; *not* in the musical sense of a rapid alternation between pitches.) The IPA explicitly recognises three trill types:

- /r/ (alveolar trill or "rolled R")
- /ʙ/ (voiced bilabial trill)
- /ʀ/ (uvular trill)

These have a role in beatboxing, as do others: trills involving the palate, inward-breathed trills and click-trills.

The frequency of vocal trills can vary from subsonic rates (e.g. 20–30 Hz) to low but audible pitches (e.g. 100 Hz). This leads to trills being employed in two distinct ways: (1) for rapidly-repeated sounds such as drum-rolls or "dalek" sound (the gargling effect of uvular trill); and (2) for pitched sounds, particularly bass sounds. In the latter category, bilabial trill ("lip buzz") is most commonly used, but palatal trills and inward uvular trills ("snore bass") are also used.

Notably, beatboxers improve the resonant tone of pitched trills (particularly /ʙ/) by matching the trill frequency with the frequency of voicing. This requires practice (to be able to modify lip tension suitably), but the matched resonance can produce a very strong bass tone, qualitatively different from an ordinary voiced bilabial trill.

A relatively common technique is the "click roll", which produces the sound of a few lateral clicks in quick succession: /$\overset{\smile}{|||||}$/ . This is produced by the tongue and palate and does not require the intake or exhaling of air, meaning (as with other click-type sounds) that beatboxers can produce the sound simultaneously with breathing in or with humming. (There exist click-roll variants produced using inhaled or exhaled breath.)

Although trilling is one way to produce drum-roll sounds, beatboxers do also use fast alternation of sounds as an alternative strategy to produce rapidly-repeated sounds, e.g. /bˀdˀbˀdˀbˀdˀ/ for kicks or /tˀfˀtˀfˀtˀfˀ/ for hi-hats.[4]

2.2.2 Close-mic technique

Beatboxing may be performed *a capella* or with a microphone and amplification. In the latter case, many beatboxers adopt a "close-mic" technique: while

[4] `http://www.humanbeatbox.com/rolls`

standard dynamic microphones are designed to be used at a distance of around 15–20 centimetres from the mouth for a "natural" sound quality [Shure Inc., 2006], beatboxers typically use a standard dynamic vocal mic but positioned around one or two centimetres from the mouth.[5] This is to exploit the response characteristics of the microphone at close range, typically creating a bassier sound [Shure Inc., 2006]. The performer may also cup the microphone with one or both hands to modulate the acoustic response.

For some sound qualities or effects the microphone may be positioned against the throat or the nose. Against the throat, a muffled "low-pass filter" effect can be produced.

A beatbox routine may be performed with the microphone held in a relatively constant position against the mouth, but some beatboxers rapidly reposition the microphone (e.g. pointing it more towards the nose for some sounds) to modulate the characteristics of individual sounds, which may help to differentiate sounds from each other in the resulting signal. It requires some skill to synchronise these movements with the vocal sounds, but it is not clear that fast mic repositioning is used by all skilled beatboxers.

Close-mic techniques alter the role of the microphone, from being a "transparent" tool for capturing sound to being a part of the "instrument". There is an analogy between the development of these techniques, and the developments following the invention of the electric guitar, when overdrive and distortion sounds (produced by nonlinearities in guitar amplifiers) came to be interpreted, not as deviations from high fidelity, but as specific sound effects.

2.2.3 Summary

Beatboxing is a relatively recently-developed performance style involving some distinct performance techniques which affect the nature of the audio stream, compared against the audio produced in most other vocal performance styles. The use of non-syllabic patterns and the role of inhaled sounds typically leads to an audio stream in which language-like patterns are suppressed, which we argue may facilitate the illusion of a non-vocal sound source or sources. These and other extended vocal techniques are employed to provide a diverse sound palette. Close-mic techniques are used explicitly to modify the characteristics of the sound.

In this discussion we have documented aspects of these performance techniques, and have provided details to illuminate how the performance style may affect the nature of the recorded sound, as contrasted against other vocal musical performance styles. We next introduce the research fields which will be

[5] `http://www.humanbeatbox.com/techniques/p2_articleid/128`

important in our work on vocal sounds including beatboxing.

2.3 Research context

2.3.1 Speech analysis

Spoken language is perhaps the main use of the human voice and so the vast majority of voice research has been dedicated to understanding and automatically analysing speech. Research into automatic speech analysis systems flourished in the 1960s and 1970s with the widespread application of computers, and continues to the present day. We discuss topics in this field as relevant to our purpose, rather than aiming to give a complete overview.

A prominent topic in this field is Automatic Speech Recognition (ASR), with the aim of enabling machines to extract the words and sentences from natural spoken language audio [Rabiner and Schafer, 1978, Rabiner and Juang, 1993, O'Shaughnessy, 2003]. The basic unit of analysis is typically the *phoneme*, a term for the smallest segmental units of spoken language corresponding roughly to what many people would think of as "vowels and consonants" [International Phonetic Association, 1999, Chapter 2]. We emphasise at this point that ASR is not intended to extract all the information that may be available in a speech signal (e.g. emotional or physiological information), nor typically to extract information from non-speech voice signals. A typical audio signal presents thousands of samples per second, while speech contains only roughly 12 phonemes per second (chosen from a relatively small dictionary of phonemes) [O'Shaughnessy, 2003]. Hence ASR is a kind of pattern-recognition process that implicitly performs a *data reduction*.

The first main step in a typical ASR process is to divide the audio signal into frames (segments of around 10–20 ms, short enough to be assumed to reside within one phoneme and treated as "quasi-stationary" signals) and then to represent each frame using a model designed to capture the important aspects either of the state of the vocal apparatus (*physical modelling*) or of the sound characteristics deemed useful to our auditory system (*perceptual modelling*). The evolution of the model state over a sequence of frames is then used to infer the most likely combination of phonemes to assign to a particular time series. The dominant approach for such inference is the Hidden Markov Model (HMM) which models transitions between "hidden states" (e.g. phonemes) [Bilmes, 2006]. In this thesis we will not be focusing on the temporal evolution of sequences such as phonemes; however we will need to consider mid-level signal representations, so we next discuss the main models used for this in ASR.

The formants and antiformants of the vocal tract, discussed in Section 2.1,

can be observed by inspecting spectrograms of voice signals [Fry, 1996/1979]. Further, they are commonly modelled directly using the *source-filter* physical model of the vocal tract: if the glottal oscillations are treated as an independent source signal, and the modulations due to the vocal tract as a combination of linear time-invariant (LTI) filters, then a variety of mathematical tools can be applied to analyse the combined system. Notable is linear prediction analysis [Markel, 1972, McCandless, 1974], which can be used to estimate parameters for the LTI filters used to model the vocal tract resonances and therefore to derive formant information such as frequency and bandwidth. An estimate of the glottal source signal can also be produced as the "residual" from the linear prediction model. Linear prediction has been an important tool in speech analysis despite the many simplifying assumptions made by the model (e.g. independence of glottal source from the rest of the system, LTI nature of the resonances) and is used for example in speech audio compression algorithms [Schroeder and Atal, 1985].

An alternative to the physical modelling used in linear prediction is perceptual modelling. Auditory models exist which replicate many of the behaviours of the components in the human auditory system, and could be used as input to ASR [Duangudom and Anderson, 2007]. However, the most common such model is the Mel-frequency cepstrum, which parametrises the shape of an audio spectrum after warping the frequency axis to roughly represent the salience of different frequency bands in the auditory system [Rabiner and Schafer, 1978] (see also Fang et al. [2001]). The Mel-frequency cepstral coefficients (MFCCs) are therefore designed to represent perceptually salient aspects of spectral shape in a few coefficients. Compared against fuller auditory models they neglect many known phenomena of the auditory system (such as temporal masking, which can render a sound imperceptible depending on the sounds occurring immediately before or after) yet are computationally relatively lightweight to calculate. To capture some measure of temporal variability, the MFCCs are often augmented with their deltas (ΔMFCCs, the difference between coefficient values in the current and previous frames) and sometimes also the double-deltas (deltas-of-deltas, $\Delta\Delta$MFCCs).

Both linear prediction and MFCCs derive information largely about resonances such as vowel formants, and little detail about consonant sounds such as fricatives; but this information content is sufficient that good speech recognition performance can be obtained from an ASR system which uses them [O'Shaughnessy, 2003]. In fact ASR systems often neglect quite a lot of information that is readily perceivable by a human in speech, including phase, pitch and mode of phonation, since the small improvement over baseline accuracy that could be achieved is considered not to be worth the complexity costs

[O'Shaughnessy, 2003].

Linear prediction and MFCCs are the two most common mid-level representations used in speech systems, with MFCCs dominant in speech recognition [O'Shaughnessy, 2003]. For example, European Standard ES 201 108 for distributed speech recognition specifies an MFCC implementation for the signal representation [Pearce, 2003].

The ASR task is not the only automated analysis of interest to speech researchers, of course. Issues such as detecting emotional states and recognising or verifying speaker identity have been the subject of a growing body of work. Often a similar toolset is applied as in ASR: MFCCs are commonly used in e.g. emotion recognition [Zeng et al., 2009] and speaker recognition [Ganchev et al., 2005, Mak et al., 2005, Hosseinzadeh and Krishnan, 2008] for example, although in research systems these may often be supplemented with other features. Such tasks involve some analysis of what we might call voice "quality" or "timbre"; here we will briefly focus on emotion recognition, since musical expression can be said to be connected to the conveying of emotional meaning [Soto-Morettini, 2006, Introduction].

The state of the art in emotion recognition in speech is largely based on two types of mid-level feature from the audio signal: instantaneous spectral/temporal features and *prosodic* features (concerning the rhythm, stress and intonation of speech) [Schuller et al., 2009, Zeng et al., 2009]. In the former category, MFCCs are popular as well as harmonics-to-noise ratio, zero-crossing rate (ZCR) of the time signal, energy and pitch [Schuller et al., 2009]. The latter category may include measures such as the distribution of phoneme/syllable durations, or whether a sentence/phrase/word has a pitch trajectory that is downwards/upwards/flat or matches one of a set of linguistically-informed templates.

The mechanism for deciding on emotional state from these features varies among researchers. HMMs may be employed and/or other machine learning techniques (see Section 2.3.4). As an example, the winning system in a recent emotion-recognition challenge employs a decision tree algorithm which combines elements of both expert knowledge and automatic classification into the design [Lee et al., 2009].

Note that prosodic features are very strongly bound to linguistic expression, and depend on some kind of segmentation of the audio stream into linguistic units. This means that they are problematic to translate to a context which encompasses non-linguistic vocalisations. Even in singing it is not clear that prosodic analyses developed for speech would be usefully applicable: although singing often has linguistic content, the pitch trajectories and durations of syllables are strongly influenced by musical factors not present in speech.

It is the singing voice to which we next turn, to consider singing-oriented

research that may be relevant for our topic.

2.3.2 Singing voice analysis

Research on the singing voice, although related, is distinct from that on the speaking voice. This is in part because intended applications are different (e.g. applications in music technologies) but perhaps more fundamentally because of important differences between speech and singing. In the following we discuss these differences before indicating some singing-voice research relevant to our topic, as well as considering how such research relates to musical voice construed more broadly than singing.

The singing voice

Important differences between singing and speech include the use of pitch and duration. In speech, pitch modulation is an aspect of prosody, whereas in singing musical principles usually dominate pitch modulation (although both musical and prosodic influences may be present). Pitch is also often higher and covers a wider range in singing than speech [Howard, 1999, Loscos et al., 1999]. Musical considerations of rhythm and metre also strongly affect the duration of syllables, with one consequence that vowels are generally longer than in speech [Loscos et al., 1999]. In some traditions a deliberate vibrato (rapid pitch modulation) is added which is not found in language, for example in Western classical/opera [van Besouw et al., 2008] or Indian classical [Rao and Rao, 2008] styles. The resonances of the voice may also be deliberately modulated by trained singers for aesthetic effect or volume, as in the strong resonance called the "singer's formant" observed in Western classical/opera singers [Sundberg, 2001].

In Section 2.1 we discussed vocal phonation modes, which to some extent convey linguistic or emotional information in speech. These are relevant in singing too (see for example Soto-Morettini [2006] on the use of creaky and breathy voice in Western popular singing). A further set of vocal configurations used to modulate singing voice quality are referred to under the term of *vocal register* [Henrich, 2006]. Different vocal registers were originally described according to perceived differences in voice quality and pitch range, and/or introspection by singers on the way it felt to produce different sounds, although a strong tradition developed following Garcia [1854] of considering vocal registers fundamentally to be different mechanical oscillatory modes in the vocal cords [Henrich, 2006]. Traditional labels used by singers might describe "chest voice", "head voice" and "falsetto" as the main vocal registers (with the latter two sometimes synonymous), where chest voice was used for the lower part of the singer's ordinary pitch range and the other two for the upper part. Less per-

vasively used were: the "whistle" or "flute" register, a very high register with a ringing tone more commonly described in women than men; the very low-pitched "strohbass" register described in men; and the loud mid-range "belt" register used more often in Western popular than classical music.

As discussed by Henrich [2006] there is still a tension between vocal register as a practical term for singers' techniques or perceived vocal qualities, and the physiological-mechanical conception of vocal register as different types of oscillation in the larynx, although the latter has developed significant insights since Garcia's day. Four main modes of vocal fold oscillation (different from the vocal modes discussed for speech in Section 2.1, but with some overlap) have been observed and standardised under the labels M0 through M3: M0 is the very low "pulse" register including strohbass and the ventricular mode phonation mentioned in Section 2.1; M1 is the more common low-to-mid-range chest register; M2 is the mid-to-upper head register (including falsetto); M3 is the very high whistle/flute register. These different registers create glottal source waves with audibly different characteristics, and so can produce different vocal qualities. In the Western classical/opera tradition, singers train to minimise the differences between the sound of the registers, so as to minimise audible transitions of register; a fact which emphasises that the physiologically-defined registers M0–M3 are not exactly identical with the registers defined according to perception and singing practice, since the changes in oscillation mode will still occur even if their timbral effect can be suppressed. Singers may modulate other aspects of their voice such as vocal tract resonances [Story et al., 2001], either to bring two registers closer together in sound, or to create differences in sound. As an example of the latter, the physiological description of the belt register has been found to consist of M1 combined with a high glottal pressure and a high glottal closed quotient, which together create the loud, harmonically-rich sound and its perceived differences from chest voice [Henrich, 2006].

At this point we highlight that much singing voice research has focused primarily on professional singers in the Western classical/opera traditions (mentioned by Henrich [2006]; see also the spread of topics covered in conferences such as the International Conference on the Physiology and Acoustics of Singing[6]). Such singers are highly trained in a particular style, and so we must take care to distinguish results which apply generally to human voice (such as the oscillation modes of the vocal folds) and those which might be developed within specific vocal traditions (such as the singer's formant). At this point therefore we briefly note some facets of other vocal traditions identified in the literature, to broaden our perspective slightly on the varieties of expression used in singing.

[6] `http://projects.dlc.utsa.edu/pas/`

- Indian classical singing uses vibrato as does Western classical singing, but with a much more dramatic depth of modulation [Rao and Rao, 2008].
- Rock singers often use the belt register mentioned above [Soto-Morettini, 2006], and sometimes produce a so-called "dist" register apparently by modulating the glottal oscillations with a low-frequency oscillation of material around the vocal folds [Zangger Borch et al., 2004].
- Traditional pygmy African singing employs numerous vocal effects including "hoots, screams, ... yodeling and hocketing" as well as "falsetto singing and ... holding the nose and singing" [Frisbie, 1971]. (Some of these effects were heard in the West through the pygmy-influenced music of Zap Mama [Feld, 1996].)
- Western avant-garde art music of the twentieth century explored a variety of extended vocal techniques including laughter, weeping, heavy breathing, and muffling the mouth with the hands [Mabry, 2002, Part II][Mitchell, 2004].
- Overtone singing styles originating in Asian traditions involve the singer learning to manipulate vocal tract resonances to produce a strong, clear, high-pitched ringing resonance in addition to the ordinary vocal formants [Bloothooft et al., 1992, Kob, 2004].
- The Croatian *dozivački* folk singing style involves singing "very loud in a high register (male singing), and to a Western ear this singing may not be perceived as singing at all, but as shouting" [Kovačić et al., 2003].

This disparate list serves to indicate some of the techniques singers can employ, and to remind us that vocalists have available a highly varied palette of modulations – beyond the basics of pitch and timing, and beyond the phonation modes and vocal registers we have discussed.

We will next survey research into singing voice analysis technologies, and we will see that the pitch evolution of the vocal signal has been quite a common object of study, either in itself or as the basis for other techniques. Through this overview it is worth remembering that pitch is but one component of the expression available to a vocal performer.

Singing voice technology

Singing voice technologies and speech technologies exhibit some overlap and common history. Pitch tracking has been extensively studied both in speech and singing [Gerhard, 2003], and very good general pitch tracking algorithms

are now available at least for solo monophonic sources such as a single voice [de Cheveigné and Kawahara, 2002, McLeod and Wyvill, 2005]. The physical model of linear prediction and the perceptual model of MFCCs (Section 2.3.1), both developed primarily in the speech context, can be applied to singing, although with some caveats: as the fundamental frequency of singing is often higher than speech, MFCC values may exhibit unwanted dependence upon frequency, since the harmonics of the voice will sample the vocal tract spectrum increasingly sparsely [Gu and Rose, 2001]. ASR techniques (Section 2.3.1) can be applied to singing, e.g. for transcribing the words in singing or for time-aligning a singing signal with known lyrics, although perhaps with some modifications to standard ASR to account for characteristic aspects of singing [Loscos et al., 1999].

However, singing voice technology research also comprises topics not found under the umbrella of speech research. Often these are connected with the field of Music Information Retrieval (MIR) which has developed particularly since the growth of digital music formats, and which studies music signals and data, and informatic tasks relevant for music creation/consumption and musicology [Orio, 2006]. One example related to monophonic pitch tracking is the issue of detecting a lead vocal line in polyphonic music audio. In many musical styles a human vocalist provides a sung melody which is the focus of the music – so the detection and tracking of this lead melody, despite the presence of harmonically-related accompaniment, is a common topic [Li and Wang, 2005, Sutton, 2006, Rao and Rao, 2008]. Related is the application of source separation techniques to recover the singing signal from the audio mixture [Ozerov et al., 2005].

Some research studies specific aspects of singers' vocal styles in order to inform musicological analyses or information retrieval. For example Garner and Howard [1999] characterise aspects of the singing voice to investigate the differences between trained and untrained singers (e.g. differences in glottal closed quotient) with potential applications in voice training. Nwe and Li [2007] detect vibrato characteristics of singing (in polyphonic audio) to perform vibrato-based singer identification. Maestre et al. [2006] model singers' articulation gestures (the way they move from one note to the next). Nordstrom and Driessen [2006] study effort in singing voice; they criticise standard linear prediction as unable to model flexibly the different glottal spectra produced by different singing effort levels, and develop a variant of linear prediction which allows for the variations in glottal source spectrum.

Having extracted information from a singing signal, it should be possible either to resynthesise the audio or to use the data as musical control information. Kim [2003] develops a linear-prediction model for singing voice analysis-synthesis, with applications including singer voice transformation (from tenor to

bass or vice versa). Fabig and Janer [2004] use a phase vocoder (Discrete Fourier Transform) analysis-synthesis technique with some inspiration from source-filter models, to modify timbral characteristics of a singing voice recording such as the smoothness of pitch transitions or the apparent vocal effort. With advances in computing power and in algorithms it even becomes feasible to use singing voice analysis for such purposes in real time (e.g. for a live performance): singing-controlled-synthesis has been studied by Loscos and Celma [2005] and Janer [2008], developing relatively simple analyses which run in real time to control pitched synthesisers.

Thus far we have not touched upon the complement to voice analysis research, namely that into voice synthesis. Although active, and with conceptual connections to the speech and singing research discussed above, it has less direct relevance to the topic of this thesis. However, we note that one application of the information derived from a singing voice signal could be to control singing voice synthesis (SVS) [Janer, 2008].

Beyond singing

This section so far has focused on singing; however singing is but a subset of musical vocalisation, let alone of vocalisation. In this thesis we wish to encompass a wide range of vocal expression, so we note some vocal traditions which stretch beyond singing.

We have already discussed beatboxing in some detail; however vocal percussion arises in many world music traditions. For example Indian tabla-players use a vocal imitation of tabla patterns called *bol* [Gillet and Richard, 2003], perhaps the oldest continuing vocal percussion tradition. More recently a rhythmic wheezing style of vocal percussion called *eefing* or *eephing* is recorded as developing in Appalachia in the late nineteenth and early twentieth centuries [Sharpe, 2006], heard widely in Western media when used by Rolf Harris in the 1960s.

Other vocal techniques which lie beyond the realm of singing include screams and growls. Punk and other hard rock styles often employ unpitched shouting or screaming rather than sung vocals. In a related but different technique, death metal vocalists typically "growl" or "grunt" in a roaring sound [Cross, 2007] which probably involves ventricular mode phonation.

As described in the previous section, singing voice research primarily attends to the pitched vocalic sounds, with the source-filter model strongly influential. Our brief discussion of non-singing vocal styles suggests that that approach may encounter limitations when faced with the wide variety of sounds of which the human voice is capable. If we wish to preserve broad applicability to musical vocal expression, we may need to use representations of vocal character which do

not depend entirely on the pitched vocalic model. Bearing this in mind, we next consider the psychological and acoustical study of musical timbre encompassing musical sounds of various types.

2.3.3 Musical timbre

The musical term *timbre* is used broadly to refer to the variability in sonic characteristics that different instruments produce. (Some writers use terms such as *sound quality* or *tone colour* to a similar end, e.g. Kreiman et al. [2004].) It is generally considered conceptually separate from the aspects of pitch, loudness and duration, encompassing attributes which musicians might describe as "bright" vs "dull", "rough" vs "smooth", etc.; although its definition has been a matter of some debate [Papanikolaou and Pastiadis, 2009].

Hermann von Helmholtz's studies in the nineteenth century are perhaps the first formal investigations of timbre and its relation to acoustic properties [von Helmholtz, 1954/1877]. He found that the distribution of energy among the harmonics of a note was an important determinant of timbre: for example a clarinet's distinctive "hollow" sound is largely due to the absence of energy in the even-numbered harmonics (those whose frequency is $2NF_0$, where N is an integer and F_0 the fundamental frequency). This harmonic-strengths approach to timbre is still employed today in various works. Note however that its utility is largely confined to pitched harmonic sounds.

Von Helmholtz's approach separates the concepts of timbre and pitch, which is common. However it is worth noting the contrary opinion expressed by Arnold Schoenberg, influential as both a composer and music theorist in the twentieth century:

> I cannot readily admit that there is such a difference, as is usually expressed, between timbre and pitch. It is my opinion that the sound becomes noticeable through its timbre and one of its dimensions is pitch. In other words: the larger realm is the timbre, whereas the pitch is one of the smaller provinces. The pitch is nothing but timbre measured in one direction. If it is possible to make compositional structures from timbres which differ according to height, [pitch] structures which we call melodies, sequences producing an effect similar to thought, then it must also be possible to create such sequences from the timbres of the other dimension from what we normally and simply call timbre. Such sequences would work with an inherent logic, equivalent to the kind of logic which is effective in the melodies based on pitch. All this seems a fantasy of the future, which it probably is. Yet I am firmly convinced that it can be

realized. [Schoenberg, 1922, p471]

Schoenberg's position on pitch as a dimension of timbre is not commonplace in musical discussion, but suggests an interconnected way of thinking about the two which will become salient shortly when we consider perceptual studies on musical timbre and its relation to pitch. It also appears to have been shared by other prominent musical thinkers such as Edgar Varèse [Mitchell, 2004].

Disagreements over the nature of timbre have never completely been resolved, although many aspects have been elucidated by perceptual and some neural studies (discussed shortly). In absence of a true consensus, one of the most widely-used definitions is that given by the American National Standards Institute (ANSI). A concise positive definition is rather difficult to state, and so the ANSI definition is curiously negative, based primarily on what timbre is not:

> [Timbre is] that attribute of sensation in terms of which a listener can judge that two sounds having the same loudness and pitch are dissimilar. [ANSI, 1960]

This definition of timbre implies very little about its form: for example, is it one-dimensional or multidimensional, continuous or categorical? (The ANSI definitions of both pitch and loudness give them as one-dimensional continuous attributes.) The definition has been criticised on such grounds, but no stronger definition has reached widespread acceptance or usefulness [Kreiman et al., 2004]. As we will describe shortly, no widely-agreed complete analysis of timbre into measurable components yet exists, although the literature shows consensus on some specific aspects of musical timbre – so it is perhaps reasonable that no more precise definition exists, as long as the psychoacoustics of the phenomenon are not fully mapped out.

Note that the ANSI definition allows for timbre to be a differentiator between instruments or between settings of a single instrument: the "two sounds" could be that of a saxophone and a guitar, or two different notes from the same saxophone. Some research concentrates exclusively on inter-instrument differences while some focuses on intra-instrument variation; in this thesis we will have cause to consider both these types of difference. The ANSI definition would also seem to include the duration of notes as an aspect of timbre, although in many works duration is considered separately, as an aspect of musical expression or of the musical score.

In the following we will consider some themes that have emerged from perceptual studies into musical timbre, including the important question of whether perceived timbre attributes can generally be predicted by measured attributes of the acoustic signal. We will then consider some applications of timbre analysis

technology in the MIR domain, in order to inform our development of timbre-based technology.

Perceptual studies of timbre and its acoustic correlates

The ANSI definition of timbre quoted above seems almost directly to suggest an experimental framework for investigating timbre: present a listener with two sounds having the same loudness and pitch, and determine the extent to which they can judge their dissimilarity. After multiple such presentations, data about the dissimilarity judgements could be used somehow to create a general map of timbre.

However there is a problem in that the definition tells us nothing about the kind of map we could expect to produce, which would bear upon the tools we should use to produce the map. Should we expect timbre to be categorical, with the important differences being between major groups such as voice vs. percussion, or more a smoothly continuous attribute? If timbre can be portrayed in an underlying "timbre space", is that space Euclidean, or would it be better represented for example as a tree-like space? (See especially Lakatos [2000] on this question.) If a Euclidean space can represent timbre, how many dimensions should it have? To our knowledge there has never been a persuasive argument which specifies what geometry a timbre space should have. However it is quite common to base investigations on tools which assume a Euclidean geometry.

The most influential work on musical timbre perception has been that of Grey and co-workers in the 1970s [Grey, 1977, Grey and Gordon, 1978], using a mathematical technique called *multi-dimensional scaling* (MDS). MDS is designed to recover a Euclidean space of a user-specified number of dimensions from a set of dissimilarity data [Duda et al., 2000, Chapter 10], although generalisations of MDS to non-Euclidean geometries also exist. Grey and co-workers presented listeners with pairs of recordings of individual notes recorded from orchestral instruments, and asked them to rate their dissimilarity on a numerical scale. They then applied MDS to these results, thus producing Euclidean spaces in which each orchestral instrument (or more strictly, each recorded sound) was positioned relative to the others. These positions could then be analysed to probe correspondences, for example whether instruments tended to cluster together based on instrument family. Although the MDS algorithm cannot directly decide the dimensionality of the space, if one creates solutions in a selection of dimensionalities one can then choose the solution which has a low "stress" – a measure of disagreement between the allocated positions and the input dissimilarities. Grey and others found a three-dimensional space useful for representing the perceived differences between orchestral instruments.

Subsequent studies have expanded on the MDS theme. Iverson and Krumhansl [1993] studied "dynamic aspects" of timbre, specifically the influence of the attack and decay portions of the signal, finding significant redundancy in timbral information across the signal, since the MDS space recovered from judgements about attacks contained much the same structure as those recovered from judgements about decays or about entire sounds. McAdams et al. [1995] applied extensions of the MDS algorithm to account for possibilities including subgroup effects in experimental participants (e.g. perhaps musicians and non-musicans use different strategies to judge timbral differences) and instrument *specificities* (additional distances not accounted for by the Euclidean space but which separate some instruments further from others); they found evidence of both subgroup effects and specificities in their three-dimensional model. Lakatos [2000] investigated whether pitched (harmonic) and percussive sounds were best treated within a single space or separately, by using a clustering analysis in conjunction with MDS. He found that spectral centroid and attack time worked well as continuous-valued predictors of the leading dimensions in MDS, but also found strong evidence of categorical judgements when listeners judged a wide variety of sounds, in that categories of instrument tended to form well-separated groups in MDS space, and the categories were well represented in a tree structure recovered by clustering. Burgoyne and McAdams [2007, 2009] applied a nonlinear extension of MDS (called Isomap [Tenenbaum et al., 2000]) to reanalyse data from the work of Grey and McAdams et al., finding that the Isomap technique (with specificities) successfully improved the representation of the data, suggesting that nonlinearity effects in timbre judgements may be important.

The history just recalled does not give strong grounds for confidence in timbre as a simply-defined (e.g. Euclidean) space, given the various modifications for categories of sound and of listener plus specificities and other nonlinearities. Despite such reservations, MDS spaces have been one of the main bases for inferences about which acoustic features, measurable on the audio signal, may best represent perceptual timbre. The canonical approach is to find acoustic features whose values (for the sounds used in MDS experiments) correlate well with sounds' positions on one or more of the axes of the MDS space produced. The aim is largely to predict rather than explain timbral judgements – in other words, there is no general claim that highly-correlating features are likely to represent calculations that are actually made in the human auditory system. Features based on detailed auditory models may yet become more prevalent (see e.g. Howard and Tyrrell [1997], Pressnitzer and Gnansia [2005]); however simpler statistics of the time- or frequency-domain signal are widely used [Casey, 2001].

From the start, MDS analysis of timbre was accompanied by exploration of acoustic correlates to the timbre spaces produced. Grey and co-workers [Grey, 1977, Grey and Gordon, 1978, Grey, 1978] interpreted axes as relating to "spectral shape", "attack time" and "harmonic onset irregularity", and tested the strength of correlations between the axes and some features chosen to capture such phenomena. They found the *spectral centroid* feature (analogous to the "centre of mass" measured on a spectrum, thus characterising the general location of the sound energy on the frequency axis) to be a good correlate of the axis characterised as denoting an instrument's "brightness". In subsequent research this correlate has been the most persistent: in perceptual experiments and in applications, both the musical concept of "brightness" and its characterisation using the spectral centroid is highly common. Similarly common is the importance of "attack time", often measured in the log-time domain, in interpreting such spaces. Wessel [1979] performed an MDS analysis and found by inspection that brightness and attack time corresponded to the axes of his two-dimensional solution. Krimphoff et al. [1994] performed MDS on the synthetic instrumental sounds used in Krumhansl [1989], finding that the three axes correlated well with spectral centroid, log attack time and a measure of irregularity of the spectrum. Caclin et al. [2005] created synthetic stimuli with variations in four timbral parameters, and found spectral centroid and attack time to be the main two axes recovered by MDS. McAdams et al. [2006] performed a meta-analysis of 10 MDS timbre spaces, investigating 72 potential acoustic correlates. They found that the log attack time, spectral centroid and spectral spread were among the best correlates.

Perceptual studies have probed these putative dimensions of timbre to determine whether they are perceptually separable from one another and from pitch, by testing for correlations between measurements on these axes. This is an important issue not only because it bears upon whether timbral and pitch processing is a holistic process in human audition, but also because it will have implications for technical procedures we may wish to apply such as targeted timbre modifications. Marozeau and de Cheveigné [2007] performed an MDS analysis using synthetic tones in which fundamental frequency (F_0) and spectral centroid were manipulated, and found significant interaction between the two dimensions, proposing a corrective factor for the spectral centroid dependent on F_0. Applying an experimental paradigm called *Garner interference* (based on participant reaction times in an identification task) to musical sounds, interactions have been recorded between pitch and timbre, and between acoustic timbre dimensions including attack time and spectral centroid [Krumhansl and Iverson, 1992, Caclin et al., 2007]. However a study of nerve potentials in auditory sensory memory suggests timbre dimensions are coded separately in the early

stage of the auditory chain [Caclin et al., 2006], suggesting that any interaction occurs in later processing stages. Timbre can affect listener judgements about pitch deviations [Vurma and Ross, 2007], and pitch differences between notes from the same instrument can cause listeners to misidentify them as coming from different instruments [Handel and Erickson, 2004].

Studies have also demonstrated contextual effects of timbre perception. Grey [1978] found timbral similarity judgements could differ depending on whether the sounds were presented as single notes or in monophonic or polyphonic sequences. Krumhansl and Iverson [1992] found that the timbre of sounds preceding or following a target sound could influence timbre recognition in sequences, but that this effect vanished when pitch also varied, suggesting pitch may be the dominant percept. Margulis and Levine [2006] found participants' timbre recognition to improve when a stimulus was presented in a sequence which fit with melodic expectations, as compared against the stimulus without any melodic context; and conversely, recognition worsened when presented in sequences which contravened melodic expectations. Such contextual effects demonstrate that timbre is a complex phenomenon and reinforce the potential difficulty in generalising results such as the timbre spaces derived from MDS experiments.

The results discussed so far have largely concerned differences between recorded/synthesised single notes representing standard orchestral instruments, or additive-synthetic tones designed to cover the acoustic correlates of spaces derived using such instruments. Lakatos [2000] offered some generalisation to include percussive sounds, but also found evidence that instruments clustered together strongly into percussive and non-percussive types, whose timbre may be represented best by different acoustic descriptors (see also McAdams et al. [2006]). However, some authors have applied MDS to variations within a single instrument, thus investigating what might be considered a narrower range of timbral variation. Barthet [accepted] applied MDS to clarinet notes played on the same instrument but with different techniques, finding that the resulting space correlated well with spectral centroid and attack time and with the ratio of odd-to-even harmonic strengths. Martens and Marui [2005] performed a similar analysis on electric guitar sounds, varying the nature of the distortion effect applied, finding that a measure related to brightness strongly predicted the leading MDS axis. Stepánek [2004] derived MDS spaces based on violin notes played on various violins and with various playing styles, and compared the MDS axes with adjectival descriptions. Kreiman et al. [1993] applied MDS to supraperiodic voices (ones showing oscillations on longer timescales than the vocal pitch period, such as the creaky or ventricular modes described in Section 2.1), and confirmed that these vocal sounds were strongly distinguishable by listeners.

To summarise, there is still much scope for work in elucidating musical timbre perception; but the MDS approach initiated by Grey and others in the 1970s has been applied in a variety of contexts and has led to a consensus on two of the most important attributes in musical timbre, namely the brightness (with spectral crest as a good acoustic correlate) and the attack time (often measured in the log-time domain). There is a wealth of evidence that the relationship between timbral attributes is not simple: they can interact with each other and with pitch perception; their perception can depend strongly on context; and timbral judgement appears to exhibit significant nonlinearities. The total number of dimensions needed to account satisfactorily for timbral variation is uncertain, and it is quite likely to be context-dependent: many studies find three- or four-dimensional solutions acceptable, although it must be borne in mind that MDS studies are always based on a small selection of example sounds (limited by the number of pairwise comparisons that it is feasible for participants to draw).

In this thesis we will be making use of acoustic timbre features and in Chapter 3 we will consider timbre features further. We will be applying machine learning techniques to create automated real-time timbre manipulations, and so in the next section we introduce selected topics in the field of machine learning, as well as describing some previous applications of machine learning to timbre-related issues in Music Information Retrieval (MIR).

2.3.4 Machine learning and musical applications

Machine learning research applies statistical and algorithmic techniques to allow computers to adapt their behaviour based on data input [Mitchell, 1997, Marsland, 2009]. It is conceptually related to artificial intelligence, having perhaps a difference of emphasis in applying algorithms that can learn to solve specific problems rather than to create a more general machine intelligence. The reader is referred to Mitchell [1997], MacKay [2003], Marsland [2009] for comprehensive introductions; here we introduce some general concepts in machine learning which we will be using in this thesis, as well as the application of such techniques to musical data.

Classification: The most common type of task in machine learning is *classification*, applying one of a discrete set of labels to data [Duda et al., 2000]. A classifier must first be trained using a labelled dataset, assumed to be representative of the data which is to be classified. The classifier adapts internal parameters based on the training data, after which it can apply labels to new data presented. A wide variety of real-world tasks can be expressed in this framework, including medical diagnosis and automatic

speech recognition (Section 2.3.1). It can be demonstrated that no classification algorithm is universally optimal for all tasks [Duda et al., 2000, Section 9.2], and so a range of such algorithms continues to be studied. Common issues in classification include assumptions made by the classifier (e.g. smoothness or Gaussianity of data distributions), and the danger when learning from necessarily limited training data of *overfitting* – learning the particularities of the training data points rather than the more general distribution they represent.

Clustering: Related to classification is *clustering*, which takes a set of unlabelled data and attempts to collect the data points into clusters such that points within clusters are more similar to each other (by some measure) than they are to points in other clusters [Xu and Wunsch II, 2005]. Unlike classification, clustering typically does not label the produced clusters, and the number of clusters may be decided algorithmically rather than user-specified. Clustering is a type of *unsupervised* learning, and classification a type of *supervised* learning, where the supervision in question refers to the supply of the "ground truth" labelled training set.

Regression: A third machine learning task is *regression*, which is similar to classification except that the aim is to predict some variable rather than a class label. This variable is called the response variable or dependent variable; it is typically continuous-valued and may be multivariate. Some regression frameworks are adaptations of classification frameworks (e.g. regression trees [Breiman et al., 1984, Chapter 8]), while some are unrelated to classification (e.g. Gaussian processes [Rasmussen and Williams, 2006]).

These key tasks in machine learning will appear in various forms in this thesis, and we will introduce specific algorithms where they are used. We next discuss some themes which apply across all these tasks.

An important issue in many applications of learning from data is the *curse of dimensionality* [Chávez et al., 2001][Hastie et al., 2001, Chapters 2 and 6]. Adding extra dimensions to a mathematical space causes an exponential increase in volume, with the consequence that the number of data points needed to sample the space to a given resolution has an exponential dependence on the number of dimensions. The amount of training data available for training a classification/regression algorithm is generally limited in practice (as is the amount of computation effort for training), which means that the algorithm's ability to generalise correctly will tend to deteriorate as the dimensionality of the input data becomes large. In clustering too, high dimensional spaces incur a curse

of dimensionality, as similarity (proximity) search is similarly affected [Chávez et al., 2001]. This has practical implications: although we might wish machine learning algorithms to uncover regularities in data irrespective of whether or not the regularities are represented in few or many dimensions and whether other irrelevant input dimensions are supplied, in practice we need to provide learning algorithms with a relatively small number of informative input features, in order to generalise well from training data.

Strategies for choosing a parsimonious data representation are therefore important. One strategy is *feature extraction* based on expert domain knowledge; in Section 2.3.1 we discussed compact feature representations such as MFCCs and linear prediction, which contain no information beyond what is in the audio signal, but compress information from on the order of 1000 dimensions (in a 10–20 ms audio frame) down to perhaps 10 dimensions intended to capture the important aspects of the signal. Other more general *dimension reduction* strategies attempt automatically to compress high-dimensional data into a smaller number of dimensions [Marsland, 2009, Chapter 10]. One of the most common techniques is Principal Components Analysis (PCA) which identifies linear combinations of dimensions along which the data have the largest variance, producing a new orthogonal basis in which most of the variance is captured in the first few dimensions [Morrison, 1983, Section 7.4]. Data reduction is therefore achieved by keeping only some of the principal components. An alternative to dimension reduction is *feature selection* which does not transform the input features but decides to keep only a subset of them [Guyon and Elisseeff, 2003]. Most commonly feature selection is applied in the context of classification tasks, where training data can be used to identify which features best predict the class labels. A relatively recent development is feature selection in unsupervised situations, where features must be selected according to measures made on the feature set itself – such as the amount of redundancy between features or the extent to which features support clustering [Mitra et al., 2002, Li et al., 2008].

Another consideration in machine learning is *online* vs. *batch* learning. Many algorithms operate in two distinct stages, with an initial training stage which precedes any application to new data (or in clustering, training precedes output of cluster allocations), with the algorithmic parameters determined in the first stage and thereafter held fixed. (This is often called batch learning.) It is sometimes desirable to have an online algorithm which learns at the same time as it outputs decisions, perhaps because the data is arriving in a temporal stream and decisions about early data points are required before the later input data points are available. Batch algorithms may be adapted for online application (e.g. Duda et al. [2000, Section 10.11], Davy et al. [2006], Artac et al. [2002]), or algorithms may intrinsically be amenable to online use. Notable in the latter

category are artificial neural networks such as multilayer neural networks [Duda et al., 2000, Chapter 6] or the Self-Organising Map (SOM) [Kohonen, 1990]: since they were developed by analogy with natural neural networks, which generally experience no distinct training stage but adapt through the process of interacting with the world, such algorithms are often capable of online learning.

In real-time systems online learning can be useful to adapt to changes of context, or to begin operation quickly without an explicit training stage, so as we apply machine learning techniques to real-time vocal control we will consider the desirability of online learning.

Timbre and machine learning in Music Information Retrieval

Music Information Retrieval (MIR) applies machine learning and other techniques to topics related to musical information [Orio, 2006]. It covers a wide variety of analyses and tasks which we will not cover here (concerning e.g. pitch, tempo, rhythm), but also timbre-oriented topics. Such topics are generally informed by the research on timbre perception discussed in Section 2.3.3. Here we consider some of the existing MIR approaches to timbre.

Quite often "timbre" in MIR is taken to refer to the sound character of entire polyphonic music recordings, with timbral similarity measures then defined between the recordings [Tzanetakis et al., 2001, Aucouturier and Pachet, 2004]. Various acoustic features have been used in this context, including MFCCs, spectral centroid, ZCR, and MPEG-7 features. Note that log attack time (see Section 2.3.3) is less useful in this context because it cannot be measured without some segmentation of audio into events and possibly into different sources. Modelling of the distribution of features has employed strategies such as HMMs and Gaussian Mixture Models (GMMs) which model each data point as generated by one of a set of Gaussian distributions whose mean and covariance are to be inferred. Aucouturier and Pachet [2004] investigate the limits of MFCC-plus-GMM-based music similarity measures, finding an upper limit at around 65% accuracy compared against a ground truth, but also arguing for additional features such as spectral contrast measures.

Timbre models have been applied for analysis at the instrumental level: De Poli and Prandoni [1997] develop an MFCC-based model for characterising the timbral differences between instruments, while Jensen [1999, 2002] develops a detailed timbre model based around sinusoidal analysis and harmonic strengths. As previously remarked, such a harmonics-based approach may be productive for analysis of harmonic instruments but its relevance diminishes for percussive and inharmonic sounds. Herrera et al. [2002] investigate the automatic classification of a database of drum sounds, applying a feature selection

technique to determine useful acoustic features (with MFCCs not found to be highly useful in this context). Tindale et al. [2004] perform a study in a similar vein, but focusing on a narrower range of sounds, automatically classifying different playing styles in snare drum hits.

Machine listening and real-time signal processing

Many of the MIR systems described in the previous section are designed for offline use, processing prerecorded musical datasets in non-real-time. However it is often desirable, and with improvements in computer processing power increasingly feasible, to analyse an audio stream in real time. When MIR-type techniques are applied to a real-time audio signal to extract semantic musical information, or to enable interactive musical tasks, this begins to take on a role analogous to that of the auditory system in human music listening, and we call it *machine listening* [Collins, 2006]. The term is also used in the context of non-musical real-time tasks [Foote, 1999] but in this thesis we primarily consider musical machine listening.

Real-time applications impose specific demands on a system which are not necessarily present in offline processing:

Causality: Decisions can only be based on input from the past and present – only that part of the signal's evolution is available to the system.

Low latency: It is often desirable for the system to react within a short time frame to events or changes in the audio stream. The acceptable bounds will depend on the task. For example, in real-time onset detection (detecting the beginning of musical notes) [Collins, 2004], we may desire a system to react to events such that the latency is imperceptible by humans – in music, the threshold of perception for event latency can be held to be around 30 ms, depending on the type of musical signal [Mäki-Patola and Hämäläinen, 2004]. The latency of the machine listening process will typically be in addition to other latencies in the overall system such as the analogue-to-digital audio conversion [Wright et al., 2004].

Efficiency: The system must be able to run on the available hardware and make decisions within the bounds of acceptable latency, meaning that computation-intensive algorithms are often impractical. Even if an algorithm can run in real time on a standard desktop computer, there are often other tasks running (e.g. music playback or synthesis, or control of MIDI instruments) meaning that only a portion of the computational resources may be available for the machine listening system.

See for example the work of Brossier [2007] who develops techniques for musical onset detection and pitch tracking with attention to these three constraints.

Real-time systems have been developed which apply signal processing techniques to derive parameters either for modulating effects [Verfaille et al., 2006] or controlling musical synthesisers [Janer, 2008]; however in these works the connection between input and output is manually specified rather than derived by machine learning. Automatic classification is applied in Hazan [2005b] to trigger events based on detecting and classifying audio onsets in real time. In a related but non-musical context, the Vocal Joystick system classifies non-speech vocalisations in real time for joystick-like control [Bilmes et al., 2006]. Collins [2006] develops real-time beat tracking and event segmentation algorithms, and applies them to develop agent-based systems which can interact musically in real time.

We highlight in the above examples the issue of event segmentation (onset detection). In some applications it is highly desirable to segment the audio stream into events, such as in the percussion classification of Hazan [2005b], whereas in some applications the continuous audio stream is used without segmentation [Verfaille et al., 2006]. The Vocal Joystick combines aspects of continuous and discrete control, able to identify discrete command sounds or sounds with continuous modulation (of the vowel formants) [Bilmes et al., 2006]. We will consider both event-based and continuous approaches in later chapters of this thesis, and attempt to investigate the differing affordances of each.

To conclude this section: we have seen that machine learning has a broad applicability in extracting information from data, and in particular that it can be applied to enhance the extraction of musical information from audio. When this can be applied in real-time systems it has the potential to enrich human-machine musical interactions, and indeed some interesting applications have already been studied, despite the important real-time constraints of causality, low latency and efficiency.

2.4 Strategy

In this chapter we have set the context for our research topic, introducing the topics of human voice analysis (both speech and musical voice), musical timbre, machine learning and real-time music processing. We are now in a position to reflect upon our aim (Section 1.2) in light of this context, and reflect on our strategy for achieving the aim.

We have seen various analytical models, such as the source-filter model of vocal production and linear prediction analysis which derives from it. This is a simplified model which concentrates on vocalic sound and therefore cap-

tures a lot of linguistically important information, but largely neglects sounds in which vocal tract resonances are less relevant. We aim to make use of a wide range of vocal sounds including non-vocalic sounds – and further, we will have cause to analyse the timbre not just of the voice but of synthesisers we wish to control. Therefore it may be preferable to avoid a model of the production system such as the source-filter model, in favour of models based on timbre perception. The requirement to treat a wide range of sounds also argues against purely harmonics-based models since many sounds are not well characterised by decomposition into harmonics. We have seen that simple auditory models such as MFCCs are commonly used in speech recognition and MIR. We have also seen that human timbre perception is a complex phenomenon with significant outstanding questions, but that there is consensus on at least some of the perceptually important factors – and there are acoustic features which largely correlate with these factors albeit not explain them. We therefore have acoustic features available which can model timbre quite generally, although there may still be questions over which combination of features works best for our purpose.

We have also seen different approaches to the temporal nature of sounds. Typical ASR systems model speech sound (using HMMs) as a temporal evolution from one discrete phoneme to another, while prosodic modelling for emotion recognition in speech is often based on segmentation of the stream into words or phrases. Such segmentation schemes are clearly only appropriate to linguistic audio. In MIR, some applications segment a signal into events (e.g. musical notes), while some do not perform segmentation and use the continuous audio stream. There are advantages to both approaches: segmentation may allow for analysis such as automatically determining the attack time of notes, yet it may lead to unnecessary focus on the chunked events as opposed to the continuous evolution of auditory attributes, and may make subsequent analysis dependent on the quality of the real-time segmentation. Therefore we will investigate both approaches, event-based and continuous, and reflect on the different affordances of the processes thus created – the range of musical expression they allow, and how easy or difficult, obvious or obscure, they make the task of expressive performance. This will be explored in evaluation experiments with users.

Our discussion of machine learning in the music/audio context shows that there are prospects for applying machine learning to automatically determine mappings from vocal audio input to synth controls. However real-time constraints, in particular those of low latency and efficiency, will limit our choice of technique. These constraints suggest that offline learning may be preferable, so that the computational effort used for training can be performed in advance of a real-time music performance; however this is not an absolute requirement, and we will consider possibilities for online learning where appropriate.

Chapter 3

Representing timbre in acoustic features

An important question for the design of any system based on automatic timbre analysis is how timbre will be measured. In Section 2.3.3 we saw that timbre is not straightforward to define and measure, but some features are common in the literature; in Section 2.4 we stated a strategy based on features which can be measured on signals quite generally. In this chapter we will devote attention to the choice of such features, to ensure that we are using good features for our timbre-based systems - improving the likelihood that our machine learning processes, fed with good data, can learn useful generalisations about timbre.

There exists an unbounded set of possible features one could extract from an audio signal. In this work we will focus our attention on those that can be measured on an arbitrary frame of audio, so as to be able to characterise the instantaneous timbre of voice and synthesiser (synth) sounds. This excludes some features such as those that can only be measured on a harmonic signal (harmonic strengths, odd-even harmonic strength ratio) and those that require segmentation (attack time – although this will be included in Section 3.2 for comparison).

However there is still a large selection of features available. In this chapter we consider a variety of features found in the MIR literature, and analyse them to determine which are the most suitable for our task of timbral synthesis control.

Many feature selection studies relate to a classification task, and so feature selection algorithms can be applied which directly evaluate which features enable the best classification performance [Guyon and Elisseeff, 2003]. In this work our core interest will be with features that enable expressive vocal control of a synth. This is not purely a classification-type application since we construe

vocal expression very broadly and not always as a selection from a small set of discrete categories, and we will also (in Chapter 5) be controlling continuous-valued synth parameters. Our desire for feature selection could perhaps be addressed by user studies where different features are used in an interactive system, but it would be prohibitively expensive in time and resources to probe more than a handful of feature combinations in such a way.

Therefore we will evaluate candidate features without direct reference to the target task but to requirements which we can evaluate across a wide range of features. Our three requirements will be **perceptual relevance**, **robustness** and **independence**. We next describe these requirements and outline our reasons for choosing them.

Perceptual relevance: Perhaps the core requirement for acoustic timbre features is that they reflect to some extent the variation that we hear as timbral variation. This requirement might perhaps outweigh others if not for the fact that definitional issues in timbre remain open (Section 2.3.3) and so there is still some ambiguity in how we might measure the ability of a feature to fulfil this requirement. However, as discussed in Section 2.3.3 the Multi-Dimensional Scaling (MDS) approach leads to one such analysis: we can measure how well different features correlate against coordinates of the spaces recovered from MDS studies of musical timbre.

In Section 3.2 we will perform a re-analysis of some MDS data from the literature. It is worth noting in advance that MDS data is necessarily derived from a limited set of musical instrument stimuli and that different spaces are recovered when using different stimuli. The focus in the literature has largely been on inter-instrument timbral differences, whereas our concern might be described more as with intra-instrument timbral differences, i.e. the expressive modulations in timbre that can be achieved with a single instrument.

Robustness: It is also important that our features are robust or repeatable: repeated timbre measures taken on a stationary sound should not exhibit too much variation. For example one might expect that a synth playing a sustained note at a fixed setting which sounds timbrally steady should yield acoustic timbre measures that change little. This expectation will underlie our operational definition of feature robustness applied to synth sounds in Section 3.3.1.

Robustness becomes particularly important when timbre analysis is placed in a machine learning setting. Many machine learning techniques involve a training phase on limited data before application to new data, and there-

fore assume that measurements on the training data are representative of those that will be made on new data.

Further, we will consider a second type of robustness. Robustness to degradations (such as additive noise) is important in many systems since real-world data often contains noise. In our context there are two principal types of sound source: vocal sounds captured by a microphone and synth sounds captured directly from the synth audio output. Both such sounds may contain line noise (Johnson–Nyquist noise or thermal noise, having essentially a flat spectrum [Johnson, 1928]); additionally the vocal signal may be contaminated by background noise from the environment such as crowd noise or music noise in a live performance. We will primarily focus on the vocal signal in Section 3.3.2 in order to characterise the noise robustness of the timbre features under consideration.

Independence: Given an arbitrary set of acoustic features, we have no guarantee that there is not significant overlap (redundancy) in the information provided by the features. If a pair of features is strongly correlated, for example, then it may be possible to exclude one feature from consideration with very little detriment to further analysis, since the excluded feature provides very little information that is not otherwise present. Reducing such overlap should allow us to capture the necessary timbral information in a small set of features, which can reduce both the computational load of timbral analysis and the effect of curse of dimensionality issues. Correlations have a strong history of use in the sciences for analysing associations between variables; in Section 3.4 we will apply information-theoretic measures that attempt to capture more general types of dependence.

Each of these requirements can be operationalised as measurements which we will investigate separately during this chapter, before concluding by drawing together the results relating to the three requirements. As we will see, none of the experiments in themselves will suggest a specific compact feature set; rather, they tend to allow us to rank features relative to one another, identifying some which are particularly good (or bad) according to each criterion. In some cases there will be a tension between the satisfaction of different requirements. Our final choice of features will therefore involve a degree of judgement in generalising over the findings of this chapter. First we describe the acoustic features we selected for investigation.

Label	Feature
centroid	Spectral centroid (power-weighted mean frequency)
spread	Spectral spread (power-weighted standard deviation)
mfcc1–mfcc8	Eight MFCCs, derived from 42 Mel-spaced filters (zero'th MFCC not included)
dmfcc1–dmfcc8	Delta MFCCs (temporal differences of *mfcc1–mfcc8*)
power	Spectral power
pow1–pow5	Spectral power in five log-spaced subbands (50–400, 400–800, 800–1600, 1600–3200 and 3200–6400 Hz)
pitch	Autocorrelation pitch estimate (in log-frequency domain)
clarity	Clarity measure of autocorrelation pitch estimate (normalised height of first peak in autocorrelation)
pcile25–pcile95	Spectral distribution percentiles: 25%, 50%, 75%, 95%
iqr	Spectral distribution interquartile range
tcrest	Temporal crest factor (TCF)
crest	Spectral crest factor (SCF)
crst1–crst5	Spectral crest factor in five log-spaced subbands (50–400, 400–800, 800–1600, 1600–3200 and 3200–6400 Hz)
zcr	Zero-crossing rate (ZCR)
flatness	Spectral flatness
flux	Spectral flux
slope	Spectral slope

Table 3.1: Acoustic features investigated.

3.1 Features investigated

We chose to investigate the set of features summarised in Table 3.1. Many of the features are as given by Peeters [2004]; we now discuss each family of features in turn.

Spectral centroid & spread: As discussed in Section 2.3.3 spectral centroid is often held to carry timbral information, in particular relating to the "brightness" of a sound. The exact calculation varies across authors (for example, whether it is measured on a linear or bark frequency scale); in this work the spectral centroid is the amplitude-weighted mean frequency measured on a linear frequency scale:

$$\text{Spectral centroid} = \frac{\sum_{i=1}^{N} |S_i| f_i}{\sum_{i=1}^{N} |S_i|} \tag{3.1}$$

where N is the number of Discrete Fourier Transform (DFT) bins (ranging from zero to the Nyquist frequency), f_i is the centre frequency of bin i (in Hz) and S_i the value of the DFT in that bin. A related feature is the spectral spread, being the amplitude-weighted variance of the spectrum.

MFCCs & ΔMFCCs: The popularity of MFCCs for speech analysis and in MIR was discussed in Chapter 2. To capture some aspect of the local dynamics, MFCCs are often augmented with their deltas, meaning their temporal first difference [O'Shaughnessy, 2003]. We measured 8 MFCCs (not including the zero'th coefficient) and their deltas.

Spectral power: The instantaneous power in a signal may convey expressive information, and the relative balance of energy within frequency bands, used for example by Hazan [2005b], Wegener et al. [2008]. We measured the overall spectral power in a frame, as well as the proportion of that power that was contained in each of a log-spaced set of bands (50–400, 400–800, 800–1600, 1600–3200, 3200–6400 Hz).

Pitch & clarity: Although pitch is commonly construed as separate from timbre, as discussed in Section 2.3.3 this is not always accepted, and timbre perception can show significant interactions with pitch perception. For these reasons as well as to compare timbre features against a pitch feature, we used an autocorrelation-based estimate of instantaneous pitch [McLeod and Wyvill, 2005] (recorded on a log frequency scale).

Using an autocorrelation-based pitch tracker yields not only a pitch estimate, but also a measure of pitch clarity: the normalised strength of the second peak of the autocorrelation trace [McLeod and Wyvill, 2005]. This clarity measure gives some indication of how "pure" the detected pitch will sound; it has been used as a timbral feature in itself, and so we also included it in our analysis.

Spectral percentiles: We also measured various percentiles on the amplitude spectrum. The name "spectral rolloff" is used for a high percentile, meaning it represents the frequency below which the majority of the spectral energy is found; however its definition varies between 85-, 90- and 95-percentile [Paulus and Klapuri, 2003, Sturm, 2006]. Further, as in many analyses where the median is a useful alternative to the mean, we consider that the spectral 50-percentile (median) is worthy of consideration as an alternative to the spectral centroid. In this work we measured the spectral 25-, 50-, 75- and 95-percentiles. We also recorded the spectral interquartile range (i.e. the difference between the 25- and 75-percentiles) which has an analogy with the spectral spread.

Spectral and temporal crest factors: A further set of features we investigated were spectral and temporal crest factors. A crest factor is defined as the ratio of the largest value to the mean value, indicating the degree

to which the peak value rises above the others. The spectral crest factor is then

$$\text{Spectral crest} = \frac{N \max |S_i|}{\sum_{i=1}^{N} |S_i|} \tag{3.2}$$

where notation is as for Equation (3.1). Spectral crest factors can be measured across the whole spectrum or in specific bands, and have been investigated by Hosseinzadeh and Krishnan [2008], Ramalingam and Krishnan [2006], Herre et al. [2001]. We measured the overall spectral crest factor (SCF) as well as that for the same log-spaced frequency bands as for power ratios (above). We also measured the temporal crest factor (TCF) derived from the time-domain signal, which has occasionally been found useful [Hill et al., 1999].

Zero-crossing rate: The zero-crossing rate (ZCR) is the number of times the time-domain signal crosses zero during the current frame.

Spectral flatness: The geometric mean of the amplitude spectrum divided by its arithmetic mean is often used as a measure of the flatness of the spectrum, designed to distinguish noisy (and therefore relatively flat) spectra from more tonal spectra.

Spectral flux: Spectral flux is the sum over all the DFT bins of the change in amplitude of each bin between the previous and current frame. It reflects short-term spectral instability and thus may be relevant for timbral roughness.

Spectral slope: The slope of the best-fit regression line for the amplitude spectrum is another way to summarise the balance of energy across frequencies.

For examples of these features in use in the literature see e.g. Herre et al. [2001], Hazan [2005b], McAdams et al. [2006].

Features were measured on 44.1 kHz monophonic audio using a frame size of 1024, and Hann windowing for FFT analysis. The hop size between frames was 0.125 – a relatively high degree of overlap, to increase the amount of data available for analysis. The audio signals analysed vary according to the experiment and will be described in later sections of this chapter.

Having described the features we chose to investigate, we can now proceed with experimental explorations of our requirements of those features, starting with their perceptual relevance.

3.2 Perceptual relevance

The features we have chosen have all been used in the past as timbre-related statistics, but it is not necessarily clear how far they actually capture perceivable timbre differences between sounds. It would be unwise to proceed without investigating whether there is a measurable connection between our acoustic timbre features and perceptual timbre. In this section we will perform an analysis which contributes towards this goal. Ideally we would select some subset of the features which captures most of the important timbral variation, but as we will see, the analysis will not yield a simple subset of features, though it will yield some useful observations e.g. on the relation between spectral centroid and brightness.

The tradition of Multi-Dimensional Scaling (MDS) timbre studies was introduced in Section 2.3.3, including the correlation-based approach to compare acoustic features with the results. Briefly, if the values of a given acoustic feature measured on the sound stimuli correlate well with the positions of the stimuli in the MDS space, then we infer that it captures some perceptually relevant information and may be useful when measured on other sounds. Note that this is an inference rather than a deduction: high correlation between a feature and one axis of an MDS space can occur by chance, especially since the number of points in an MDS timbre space is typically small (on the order of 10–20 audio stimuli).[1] Our confidence in such inferences will increase if correlations emerge from multiple separate MDS experiments, since that would be much less likely to arise by chance.

It is also unclear how far such correlations might generalise, again since only a few audio stimuli are used. On this point see especially Lakatos [2000] who investigated whether pitched and percussive sounds could be described in a common MDS space. He found that both pitched and percussive sounds led to MDS spaces with spectral centroid and attack time as acoustic correlates, but that listeners showed a tendency to group sounds according to source properties, suggesting that differences between diverse sounds may be better explained categorically rather than through continuous spatial correlates.

A further issue comes from recent reanalyses of MDS experiments such as Burgoyne and McAdams [2009] who argued that MDS with certain nonlinear extensions yielded spaces which better accounted for the dissimilarity data.

Although the original MDS studies do explore acoustic correlates [Grey, 1978, Grey and Gordon, 1978, Grey, 1977, McAdams et al., 1995], they do not

[1] The small number of stimuli comes from a practical limitation. The raw data for MDS studies consists of pairwise comparisons. Listeners must therefore judge the similarity of on the order of $\frac{1}{2}N^2$ pairs if there are N stimuli, which for around $N > 20$ becomes too many for a listener to judge without fatigue affecting the data.

explore all the features we are considering, so we wished to perform our own correlation analysis. Having the choice of using the published MDS coordinates from the original studies or from the reanalysis of Burgoyne and McAdams [2009], we chose the latter, on the basis of Burgoyne and McAdams' evidence that their model accounts for more of the structured variation in the data.

3.2.1 Method

We used MDS coordinates from Burgoyne and McAdams [2009] in conjunction with our own acoustic timbre features measured on the audio stimuli from three experiments in the literature (stimuli kindly provided by McAdams, pers. comm.). In addition to our instantaneous timbre features, averaged over the duration of the non-silent portion of each audio stimulus, we also included the log attack time measured on each stimulus as a potential correlate, since the stimuli were amenable to that measure and it has been found useful in previous studies (as discussed in Section 2.3.3). This will be listed in the results tables as *attacktimeLOG*.

We measured the Pearson correlation between each acoustic feature and each dimension from the selected MDS spaces. One might argue that the MDS spaces should be treated as a whole, e.g. by analysing the multivariate correlation between every feature and each 2D or 3D space. In a simple MDS analysis this is particularly justified since the orientation of the solution space would be arbitrary; however the MDS spaces from Burgoyne and McAdams [2009] accommodate latent-class weights as well as perceptual distances, giving a consistent orientation (Burgoyne, pers. comm.), meaning each dimension should represent some perceivable factor whose acoustic correlates are of individual interest.

We wished to analyse correlations and derive significance measures for our results. However, the phenomenon of strong correlations arising through chance (e.g. through random fluctuations in the data) becomes very likely if a large number of correlation measures are taken, and so it is important to control for multiple comparisons [Shaffer, 1995]. For the set of correlations we measured, we used Holm's sequentially-rejective procedure [Shaffer, 1995] to control for multiple comparisons at a *familywise error rate* of $p < 0.05$ (in other words, to test for the significance of all measured correlations such that our chance of falsely rejecting one or more null hypotheses was maintained at less than 0.05).

We further mitigated the issue of multiple comparisons by choosing only one MDS space for analysis from each dataset. There are various options available in deriving an MDS space, such as the number of dimensions in the output space and the inclusion of nonlinearities into the the model. These are explored by Burgoyne and McAdams [2009] who produce a selection of output spaces

and explore the goodness-of-fit of various spaces derived using varieties of MDS processing. From the data and discussion in that paper we selected the space which was said to best represent the data, and applied our correlation analysis to that.

Our three datasets were therefore (where each "model" is from Burgoyne and McAdams [2009]):

***G77*:** Stimuli from Grey [1977],
with coordinates from the 3D-with-specificities model

***G78*:** Stimuli from Grey and Gordon [1978],
with coordinates from the 2D-without-specificities model

***M95*:** Stimuli from McAdams et al. [1995],
with coordinates from the 3D-without-specificities model

3.2.2 Results

Results are shown in Table 3.2, listing the strongest-correlating features for each dimension and with significant correlations in bold.

The leading dimensions of *G77* and *G78* show multiple significant correlations with our timbre features, while the other dimensions show only one or zero significant correlations. This occurs too for *M95* except with the second dimension showing the correlations. This indicates that not all of the MDS timbre space dimensions are predictable from the acoustic features we have measured: either the remaining dimensions are predictable using features we have not investigated, or they are not directly predictable from acoustic features – for example they may represent cultural or learned responses to particular sounds.

As in previous studies (discussed in Section 2.3.3) we find the spectral centroid (*centroid*) correlates well with the leading dimension in all three of the spaces analysed (although this did not reach our specified significance level for *M95*, it still ranked highly) – yet the exact same can be said of the spectral 95-percentile (*pcile95*), which in fact outranks the spectral centroid in two of the three spaces. (*mfcc2* also shows a strong correlation with the leading dimension of *G77* and *G78*.) Since spectral percentiles are not tested as often as the spectral centroid in the literature, we cannot be sure whether this similarity in the predictive power of these two features holds very generally, but we note from our results that it seems likely that either of these two features would appear to serve equally well as a representative of this leading dimension of timbre. This is the dimension referred to by previous authors as the "brightness" dimension since informal listening indicates that it serves to differentiate

Feature	r_1	Feature	r_2	Feature	r_3
centroid	**0.967***	**clarity**	**-0.764***	mfcc4	0.69
mfcc2	**-0.964***	mfcc3	-0.664	attacktimeLOG	0.666
pcile95	**0.954***	spread	-0.603	slope	0.61
pcile75	**0.947***	pow2	0.58	dmfcc3	0.584
pcile50	**0.87***	mfcc4	-0.508	mfcc8	-0.557
iqr	**0.865***	pcile95	-0.506	dmfcc2	0.547
pow4	**0.82***	mfcc1	0.479	pow1	0.544
power	**-0.809***	mfcc2	0.463	pow3	-0.513
spread	**0.796***	flatness	-0.449	pcile25	-0.501
flatness	**0.792***	pitch	0.437	clarity	0.413

(a) *G77*

Feature	r_1	Feature	r_2
pcile95	**0.973***	**pow1**	**-0.805***
mfcc2	**-0.829***	mfcc4	-0.729
iqr	**0.812***	pcile25	0.695
centroid	**0.808***	clarity	-0.648
pcile75	**0.804***	mfcc8	0.615
spread	0.732	crest	-0.585
power	-0.715	dmfcc3	-0.578
flux	-0.63	dmfcc4	-0.572
pow4	0.583	pow3	0.569
pcile50	0.543	slope	-0.552

(b) *G78*

Feature	r_1	Feature	r_2	Feature	r_3
mfcc1	**-0.815***	**pcile50**	**-0.861***	flux	-0.701
pcile95	0.715	**crest**	**0.84***	dmfcc5	0.494
centroid	0.691	**pow1**	**0.817***	mfcc3	0.452
attacktimeLOG	-0.676	**mfcc2**	**0.779***	pow5	0.434
flatness	0.675	pcile25	-0.729	pow1	0.391
slope	0.667	mfcc5	0.713	pow3	-0.377
spread	0.624	crst3	0.709	mfcc4	0.363
crst3	0.592	pow4	-0.667	slope	0.345
clarity	-0.589	pow3	-0.646	iqr	0.329
power	-0.562	mfcc4	0.619	mfcc7	-0.316

(c) *M95*

Table 3.2: Ranked Pearson correlations of timbre features against axes of MDS spaces. Strongest 10 correlations are shown for each axis, and those judged significant (at a familywise error rate $p < 0.05$, Holm's procedure) are shown in bold and starred. The dimensions are labelled r_n.

"bright" and "dull" sounds. It may be that we come to prefer one of these two features over the other based on the other criteria explored during this chapter.

There are no other correlation patterns which show much consistency across these three spaces. Some features show significant correlation in one of the three spaces (*clarity*, *pow1*, *mfcc1*, *pcile50*) but this is generally not supported by any similarly strong correlation in the other spaces.

The log attack time is generally considered in the literature to correlate well against timbral perceptual data, but in our analysis it shows only a weak association with the timbre dimensions. For *G77* and *M95* it shows a correlation strength of around 0.67 with one axis, but in both these cases there are spectral features which correlate more strongly with that axis. Compare this with, for example, correlation strengths of around 0.8 for log attack time in some experiments reported by Burgoyne and McAdams [2009] and by Iverson and Krumhansl [1993]. This difference is likely due to differences in the datasets used, although there may be some small influence from differences in implementation of the log attack time measure.

It is worth recalling some of the limitations of MDS studies into musical timbre perception. Since participants must make a large number of comparisons, only a small number (10–20) of stimulus sounds can feasibly be used and so generalisation to a larger variety of sounds is problematic – only the associations which repeatedly emerge from such studies are likely to be broadly applicable. Also, these studies use comparisons between single notes, and so they do not directly concern the timbral variation that is possible within a single instrument's range or even within the evolution of a single note or sound. (MDS studies for within-instrument variation do exist, such as Barthet [accepted] for clarinet and Martens and Marui [2005] for electric guitar distortion. They too report a "brightness" axis as the main consistent finding.)

However, our aim here has been to investigate the predictive strength of our selection of acoustic timbre features for these MDS timbre spaces that are well-known in the musical timbre literature, as one approach to selecting features for use in a real-time timbre analysis. For the types of timbre judgements captured in these MDS studies our correlation analysis finds only one generality: the leading dimension in such timbre judgements is well predicted by both the spectral centroid and the spectral 95-percentile. For the spectral centroid this is in agreement with the literature. The log attack time, also discussed in the literature, is not confirmed by our analysis as a strong correlate.

3.3 Robustness

The perceptual data give us a reasonable confidence that a "brightness" dimension, represented by the spectral centroid or 95-percentile, is one aspect of perceptually relevant information, which we can measure on an arbitrary frame of an audio signal. It is clear that this is not the only perceptible axis of timbral variation, but we must defer further perceptual studies to future research. Instead, we turn to the other criteria against which we wish to judge timbre features.

Our application of timbre features will be as input to machine learning algorithms. Therefore we need to ensure that we are supplying "good data" to the algorithms, in particular data that is relatively tolerant to degradations such as additive noise or the inherent signal variability in sounds which are perceptually stationary.

Study of the noise-robustness of audio analysis algorithms has a long pedigree, e.g. for speech recognition systems [O'Shaughnessy, 2003, Section G] or musical instrument classifiers [Wegener et al., 2008]. In such cases the context of application provides a way to quantify the robustness, via such measures as word error rate or classification error rate, measured against ground truth annotations. If a given algorithm's error rate increases substantially when degradations are introduced, then the algorithm is said to be less robust than one whose error rate increases less under the same degradations.

Our application will involve the analysis of two types of signal: the output of synths, and voice signals captured by microphone. The latter may take place in live performance situations, where degradations such as background noise or signal distortion are more likely to occur than in a studio situation, where signal quality can be more tightly controlled. In this section we will therefore investigate the robustness of our timbre features in two ways:

- Robustness to the inherent signal variability in perceptually stationary sounds, characterised by repeatedly measuring features on constant synth settings and characterising the amount of variation;
- Robustness to degradations such as additive noise and signal distortions, measured on representative datasets of performing voice signal.

We will find some consistency in the results of these two related investigations. We proceed first with the robustness measures on synth signals.

3.3.1 Robustness to variability of synth signals

The robustness of acoustic features can be characterised by making repeated measurements of timbre features on a synth with its settings held constant, and

determining the variability of those repeated measurements, using e.g. standard deviation. If this is done for a variety of synth settings and for multiple features then we have a measure of variability which we can use to compare features. This relies on the assumption that the constant setting yields sounds which give a stable timbral percept, which is not always true: many synths include random, generative or dynamic features, meaning that the aural result of a fixed configuration may not be constant. Therefore in order to probe the robustness of features we must choose a set of synths which can be said to satisfy the assumption, as well as broadly representing the types of synth sound which we may wish to control in our application.

Method

We implemented five types of monophonic synth as patches in SuperCollider, and for each one we enumerated a set of controls which we could programmatically manipulate. The synths are described in full in Appendix B; in brief they are:

supersimple, a simple additive synth;

moogy1, a subtractive synth;

grainamen1, a granular synth using a percussive sound;

gendy1, an algorithm originally conceived by Iannis Xenakis with a parametrically varying waveform; and

ay1, an emulation of a real-world sound chip.

Three of the synths can be called "pitched" in that they have a fundamental frequency control which has a strong relationship to the perceived pitch, while two (grainamen1 and gendy1) are "unpitched". These latter synths do not have a fundamental frequency control; in some configurations they produce sounds with a cyclical nature and therefore can sound pitched, but in many configurations they produce noisy or percussive sounds with no clear impression of pitch.

Our method for measuring the variability of each feature was to make repeated feature measurements on the synths held at constant settings, choosing to sample features from non-overlapping frames from the steady-state portion of the synth sound (i.e. attack and decay portions were not included). Variability was quantified as the average standard deviation of feature values across a variety of synths and a variety of settings.

Two caveats must be introduced at this point. The first is that some normalisation must be applied to accommodate the fact that acoustic features have

different ranges. We normalised each feature across the whole range of measured timbre values to give zero mean and unit variance, so our measure of variability within a single synth setting was the standard deviation calculated on the normalised Euclidean distance. Since this normalisation involves dividing values by the overall standard deviation, this has the effect that our measure is a ratio of standard deviations: the ratio of the amount of variation *within* each synth setting to the *overall* variation.

The second caveat is a practical one: the large number of possible synth settings means that it is unfeasible to record a large number of examples from all possible settings combinations. In our experiment we could not iterate over all possible settings, so we instead used a random sample of settings for the synths. In this case we used 100 different settings of each of the five synths, and from each recorded a short segment producing 120 audio frames.

Results

Figure 3.1 summarises the standard deviation measurements on the timbre features. The long whiskers indicate that all features exhibit some degree of variability. However, there is a clear separation among the medians, indicating that some features are much more stable than others.

The ΔMFCCs (*dmfcc_*) stand out immediately as being far more variable than all other features. One might suggest this is due to their nature as differences between successive frames – their variability compounds the variability from two frames. However, the spectral flux (*flux*) is also a difference between frames yet does not exhibit such strong variability here. Also note that features measured on adjacent frames may be expected to have some concomitant variation (after all, adjacent frames share many audio samples since frames overlap), and so the delta operation might be expected to cancel out some portion of the variability.

After the ΔMFCCs, another family of features which performs poorly on this robustness measure is the spectral crest features (*crst_* and *crest*, and the temporal crest *tcrest*), with many of these features among the lowest-ranked by median variability. This may be due to the reliance of crest features on finding the maximum of a set of values, an operation which may be strongly affected by noise or variation on a single value. If crest features are desirable, it may be possible to improve their robustness for example by using the 95-percentile rather than the maximum; however we will not pursue this in the present work.

The strongest-performing families of features in this experiment are the bandwise power ratios (*pow_*) and the spectral percentiles (*pcile__*), both of which provide information about the broad spectral shape and whose value may be

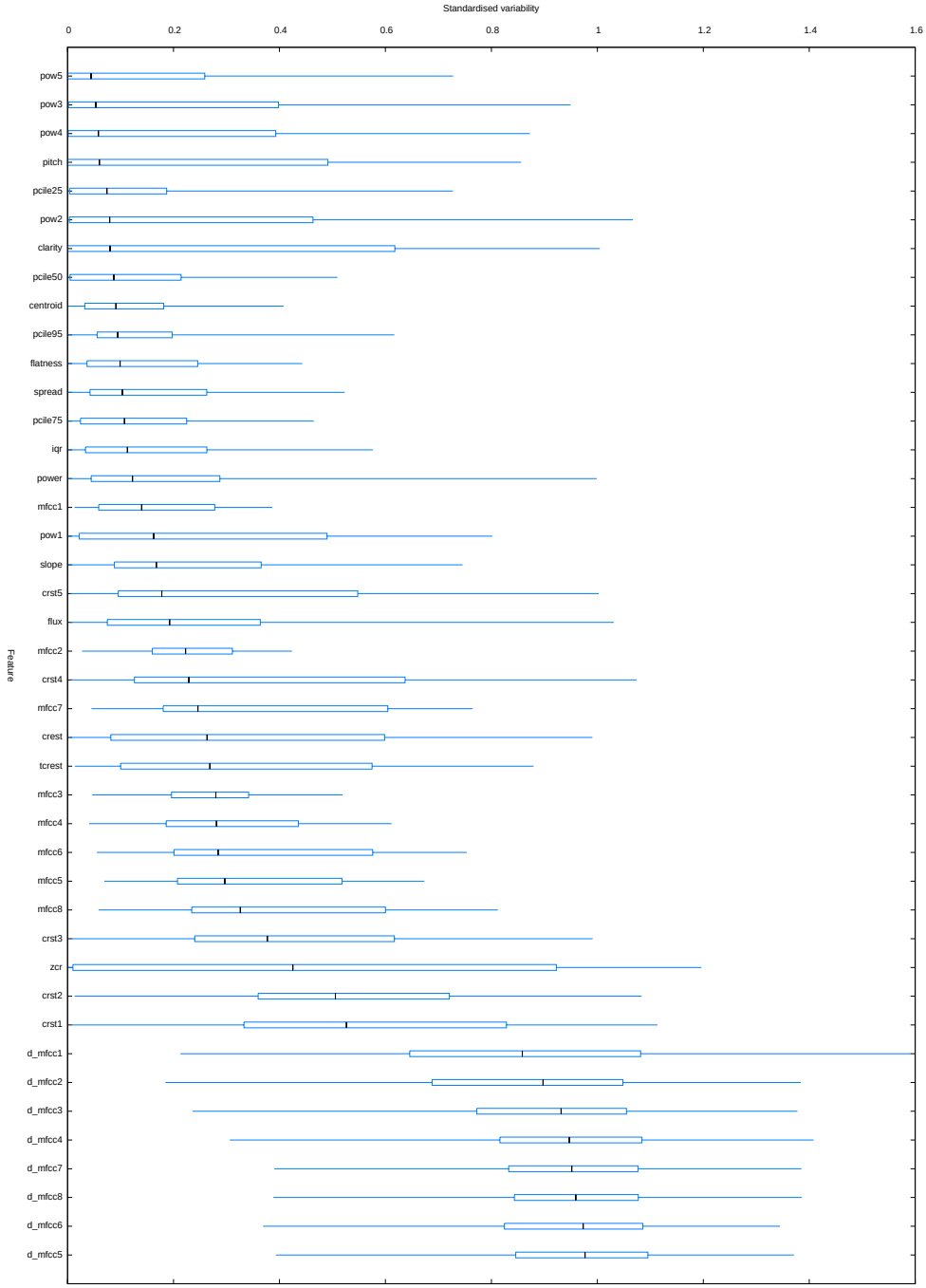


Figure 3.1: Variability (normalised standard deviation) of timbre features, measured on a random sample of synth settings and 120 samples of timbre features from each setting. The box-plots indicate the median and quartiles of the distributions, with whiskers extending to the 5- and 95-percentiles.

dominated by strong peaks in the spectrum. This can also be said of the spectral centroid (*centroid*). The spectral flatness measure (*flatness*) performs similarly well, and is one feature which (like the spectral crests) is designed to extract some information comparing the strong peaks against the background. The autocorrelation clarity (*clarity*) also performs strongly and may be said to characterise a similar aspect of the sound, although calculated in the time domain rather than the frequency domain. With such considerations in mind we may be optimistic that our most stable features are not all redundantly measuring the same aspect of the signal, a topic we will return to in Section 3.4.

Although the ΔMFCCs perform worst in this test, it is notable that the MFCCs themselves (*mfcc_*) are also relatively unstable by our measure. In general they are grouped in the lower half of the median-ranked features, although the lower-valued MFCCs (particularly *mfcc1*) yield more acceptable robustness performance.

In the next section we will move from synthesiser timbre to vocal timbre as we turn to investigate the robustness of features to noise and signal distortions.

3.3.2 Robustness to degradations of voice signals

We aim to develop methods which can be driven by timbre measured from a voice signal in a live vocal performance. Voice signals captured from a microphone may be subject to different types of degradation than synthesiser signals captured directly from the instrument, such as background noise from music or a crowd. Hence in this section we will use performing voice signals and analyse the robustness of the continuous-valued timbre features to degradations applied to those signals. We will characterise robustness to degradations as the extent to which information remains in the timbre features even after the degradations, evaluated with an information-theoretic measure.

In Section 3.3.1 we used synthesiser settings as a ground truth against which to measure robustness. However, we typically do not have access to analogous datasets of voice with detailed timbral annotations. Therefore we will employ a slightly different method, in which we analyse the variability of timbre features as we apply synthetic degradations to recorded voice signals; the features measured on the original voice recordings take the role of ground truth.

There are many ways to degrade an audio signal. Speech recognition algorithms may commonly be evaluated for their robustness to the addition of background noise (babble, street noise) or to the compression used in mobile telephony [Kotnik et al., 2003]. Musical analysis systems may be evaluated for robustness to MP3 compression or reverberation [Wegener et al., 2008]. Here we are interested in real-time analysis of a microphone voice signal, used in a

Label	Description	Duration (secs)	Total non-silent frames (3 s.f.)
SNG	Singing	529	30,300
SPC	Speech	795	34,000
BBX	Beatboxing	414	15,700

Table 3.3: The three datasets investigated.

live music performance. In this situation we will want to consider robustness to additive white noise (as a generic model for the line or thermal noise which affects many signal conductors [Johnson, 1928, Nyquist, 1928]), crowd noise, clipping distortion due to saturated components in the signal chain, or feedback echoes due to microphone placement.

We first describe the voice datasets used for this investigation, before describing the degradations applied and our measure of robustness given those degradations.

Voice datasets and degradations

For our experiments we prepared three datasets representing three types of performing voice: singing, speech and beatboxing. These datasets we refer to as *SNG*, *SPC* and *BBX* respectively. These three types were selected because they exhibit differences which may be relevant to timbral analysis: singing voice signals contain relatively more vowel phonation than speech [Soto-Morettini, 2006], while beatboxing signals contain less vowel phonation and also employ an extended palette of vocal techniques (Section 2.2). Participants were aged 18–40 and with varying levels of musical training. For *SNG* and *SPC* we recorded 5 male and 3 female participants; for *BBX* we recorded 4 male participants (the beatboxing community is predominantly male). All recordings were made in an acoustically-treated studio, using a Shure SM58 microphone and Focusrite Red 1 preamp, recorded at 44.1 kHz with 32-bit resolution. Each recording was amplitude-normalised and long pauses were manually edited out. After feature analysis, low-power frames (silences) were discarded. The datasets are summarised in Table 3.3.

We designed a set of signal degradations representative of the degradations that may occur in a live vocal performance, listed in Table 3.4. For each of the seven degradation types, we applied the degradation separately to the voice signals at four different effect levels, measuring the timbre features on the resulting audio.

Description	Effect settings
Additive white noise	-60 dB, -40 dB, -20 dB, 0 dB
Additive crowd noise *(BBC Sound Effects, crowd, vol. 48)*	-60 dB, -40 dB, -20 dB, 0 dB
Additive music noise *(The Cardiacs, Guns, track 7, ALPHCD027)*	-60 dB, -40 dB, -20 dB, 0 dB
Clipping distortion $y_t = \max(\min(x_t, k), -k)$	0.3, 0.5, 0.7, 0.9
Delay with no feedback $y_t = x_t + \frac{1}{2}x_{(t-k)}$	5 ms, 25 ms, 40 ms, 70 ms
Delay with feedback $y_t = x_t + \frac{1}{2}y_{(t-k)}$	5 ms, 25 ms, 40 ms, 70 ms
Reverberation y_t = FreeVerb.ar(x_t, 0.5, room:k, 0.9)	0.1, 0.4, 0.7, 1.0

Table 3.4: Audio signal degradations applied. Note that *FreeVerb.ar* is the SuperCollider implementation of the public-domain Freeverb reverb algorithm, see e.g. `http://csounds.com/manual/html/freeverb.html`.

Method

Having described the audio datasets and the degradations applied to them, it remains to specify how the deviations of the features due to the degradations can usefully be summarised and compared. Summarising the absolute or relative deviation of the feature values directly (as in Section 3.3.1) is one possibility, but here we wish to apply a general method based on the idea that our degradations will tend to destroy some of the information present in the signal. Such concepts find a mathematical basis in information theory [Arndt, 2001], where the differential (Shannon) entropy $H(X)$ of a continuous variable X quantifies the information available in the signal:

$$H(X) = \int_X p(x) \log p(x) dx \tag{3.3}$$

The mutual information $I(X;Y)$ is a related information-theoretic quantity which quantifies the information which two variables X and Y have in common, and therefore the degree to which one variable is predictable or recoverable from the other:

$$I(X;Y) = \int_Y \int_X p(x,y) \log \left(\frac{p(x,y)}{p(x)\,p(y)} \right) dx\, dy \tag{3.4}$$

$$= H(X) + H(Y) - H(X,Y) \tag{3.5}$$

where $p(x,y)$ is the joint probability density of the random variables X and Y, $p(x)$ and $p(y)$ are their marginal probability densities. The mutual information

measure then directly indicates the degree of informational overlap between X and Y, a more general measure of redundancy than correlation.

If we measure timbre features on a clean signal and a degraded signal, and then find the mutual information between those two measurements, this therefore provides a general quantification of how much information from the clean timbre features is recoverable from the degraded timbre features. The mutual information between two continuous variables is not bounded from above and could theoretically be infinite, meaning one continuous variable gives perfect information about the other. In practice we are working with sampled signals and finite numerical precision, meaning our measurements will not diverge to infinity.

We applied this information-theoretic approach to robustness of our timbre features by measuring the timbre features on each of our voice datasets, both clean and with the degradations applied. We then normalised the scaling of each timbre feature such that the continuous entropy of the feature measured on the clean audio had a fixed entropy of 10 nats (to remove the possibility of biases introduced due to numerical precision error), before calculating the mutual information (3.5) between each degraded feature set and its corresponding clean set.

This process produced a large number of comparisons (having 7 degradations each at four effect levels). We applied Kendall's W test [Kendall and Smith, 1939] across the different degradations, as well as visually inspecting graphs of the results, to determine whether results showed consistency across the different degradation types. In all cases we found consistent effects (various values of W, all yielding $p < 0.001$), so in the following we report the results aggregated across all effect types.

Results

Table 3.5 summarises the robustness measures for each of the three voice datasets, showing the mean of the mutual information between the degraded and the clean feature values. The three tables show some commonalities with each other, but also with the ranked lists derived from robustness measures in Section 3.3.1. The overall agreement among the four rankings (i.e. Table 3.5 together with Figure 3.1) is significant (Kendall's W=0.381, p=0.0168, 41 d.f.).

The ΔMFCCs perform particularly poorly by our measure, with the crest features and MFCCs also performing rather poorly. As noted in the previous experiment, this does not appear to be explicable merely by the nature of the ΔMFCCs as inter-frame difference measures, since *flux* is also an inter-frame difference measure yet performs relatively strongly.

Feature	MI
pitch	3.49
zcr	2.06
clarity	1.99
pow1	1.81
power	1.79
pcile25	1.73
slope	1.73
pow2	1.72
pow3	1.58
crst1	1.54
pcile50	1.54
crest	1.46
pow4	1.45
pow5	1.37
pcile75	1.32
pcile95	1.28
flux	1.28
crst2	1.25
iqr	1.21
mfcc2	1.12
centroid	1.12
mfcc4	1.1
mfcc1	1.08
spread	0.989
mfcc3	0.967
mfcc5	0.965
tcrest	0.957
crst3	0.954
mfcc6	0.904
mfcc7	0.881
crst4	0.877
flatness	0.876
mfcc8	0.87
crst5	0.81
dmfcc2	0.301
dmfcc1	0.3
dmfcc3	0.259
dmfcc6	0.258
dmfcc7	0.226
dmfcc5	0.219
dmfcc4	0.214
dmfcc8	0.212

(a) SNG dataset

Feature	MI
pitch	3.05
clarity	2.37
zcr	1.91
power	1.76
pow1	1.7
slope	1.7
pow2	1.64
pcile50	1.5
pow3	1.48
flux	1.43
crest	1.41
pcile25	1.39
pow4	1.38
pcile75	1.36
pow5	1.3
iqr	1.26
crst1	1.24
pcile95	1.14
centroid	1.1
crst2	1.08
mfcc2	1.02
mfcc1	0.974
tcrest	0.938
mfcc3	0.932
spread	0.914
mfcc6	0.905
mfcc4	0.885
mfcc5	0.843
flatness	0.842
crst3	0.768
mfcc7	0.765
mfcc8	0.764
crst4	0.733
crst5	0.674
dmfcc1	0.604
dmfcc2	0.536
dmfcc3	0.443
dmfcc6	0.399
dmfcc4	0.378
dmfcc5	0.349
dmfcc8	0.336
dmfcc7	0.332

(b) SPC dataset

Feature	MI
pitch	2.02
zcr	1.87
power	1.86
pcile25	1.84
pow1	1.82
clarity	1.8
pcile50	1.74
flux	1.74
slope	1.68
pcile75	1.62
centroid	1.52
pow3	1.5
pcile95	1.5
pow4	1.45
crest	1.45
pow5	1.42
iqr	1.4
mfcc1	1.37
spread	1.36
flatness	1.34
mfcc2	1.23
pow2	1.19
crst1	1.19
tcrest	1.14
mfcc3	1.09
mfcc4	0.938
mfcc7	0.88
mfcc5	0.862
crst3	0.833
mfcc6	0.822
mfcc8	0.798
crst4	0.682
crst2	0.681
crst5	0.647
dmfcc1	0.607
dmfcc2	0.516
dmfcc3	0.405
dmfcc4	0.372
dmfcc6	0.352
dmfcc8	0.347
dmfcc5	0.343
dmfcc7	0.337

(c) BBX dataset

Table 3.5: Noise robustness of timbre features, summarised across all degradations. "MI" is the mean mutual information in nats.

Strongest-performing are various features including autocorrelation pitch

and clarity, ZCR, power-based features and spectral slope. The spectral percentiles and centroid rank moderately highly in these figures, though not as highly as in the previous robustness tests.

These investigations into robustness have shed some light on the relative merits of individual features, the strongest conclusion being the recommendation against the ΔMFCCs. In order to work towards a more integrated perspective we must consider interactions between features, which we turn to in the final section of this chapter.

3.4 Independence

Our investigations so far have been concerned with attributes of individual timbre features. However we are likely to be using multiple timbre features together as input to machine learning procedures which will operate on the resulting multidimensional timbre space. We therefore need to consider which features together will maximise the amount of useful information they present while minimising the number of features, to minimise the risk of curse of dimensionality issues. We do this by studying the mutual information (MI) between variables.

Mutual information was introduced in Section 3.3.2 in the context of considering how MI was shared between a feature and its degraded version, where the aim was to maximise the value; here we wish to avoid choosing featuresets in which pairs of features have high MI, since high MI indicates needless redundancy in the information represented.

In the following we report an experiment using MI calculated pairwise between features. This gives a useful indication of where informational overlaps exist. It would also be useful to consider the interactions between larger feature subsets. In Appendix C we report preliminary results from an information-theoretic feature selection approach which aims to consider such interactions; however, we consider such methods currently need further development, so we concentrate here on the mutual information results.

3.4.1 Method

We used the same three voice datasets *SNG SPC* and *BBX* as described in Section 3.3.2. We applied the probability integral transform to normalise each of the features' values and ensure that our measures were not influenced by differences in the distributions of the features. (This standardisation of the marginal variables is closely related to the use of empirical copulas to study dependency between variables, see e.g. Nelsen [2006, Chapter 5], Diks and

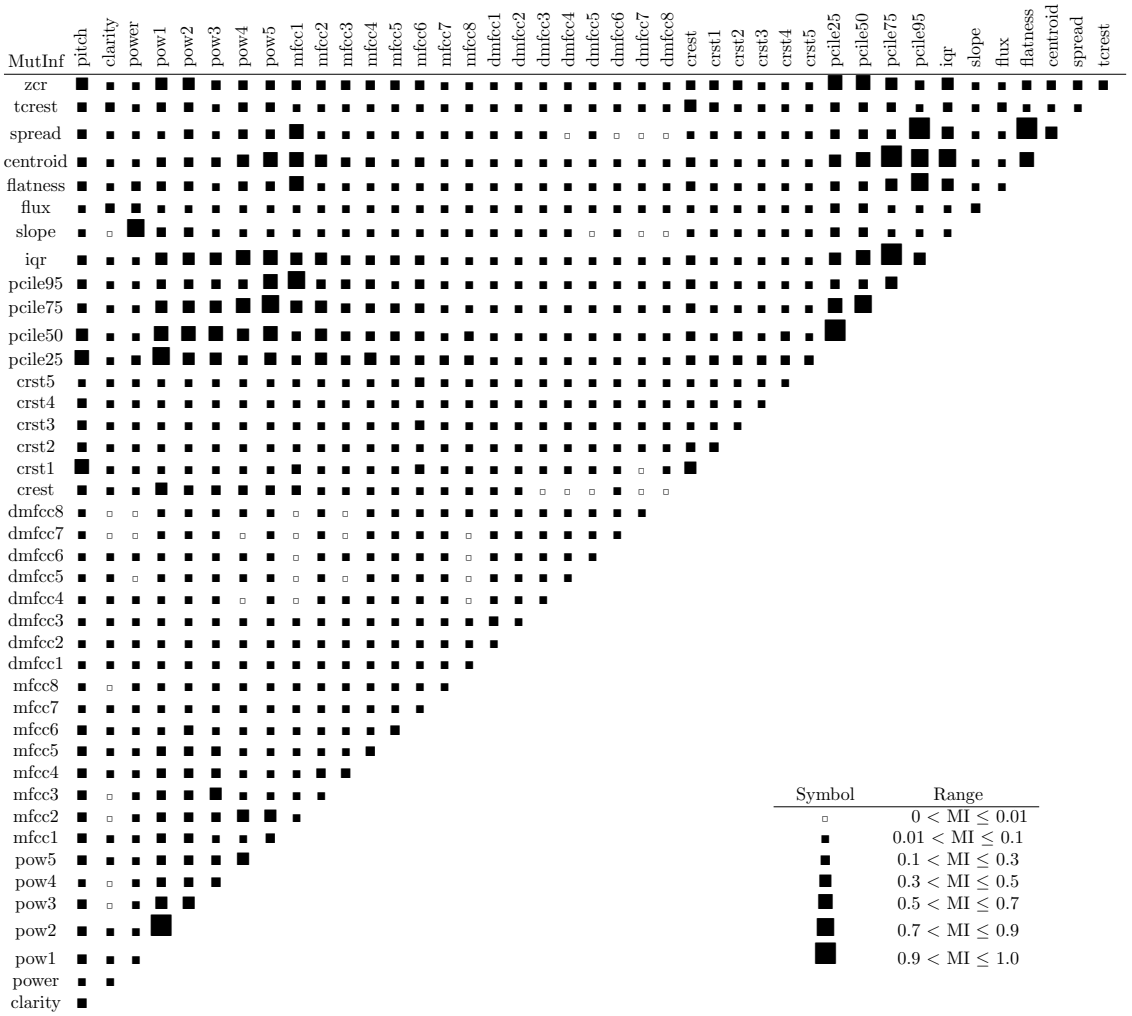


Table 3.6: Mutual Information (bits) between features, for the aggregate of the three voice datasets.

Panchenko [2008].) We then used our partition-based entropy estimator (Appendix A) to estimate the mutual information (MI) by Equation 3.5.

3.4.2 Results

We first measured the MI between features using each of the three performing voice datasets separately. However, on comparing the results we found very strong agreement between the three sets (Pearson's $r \geq 0.944$, one-tailed, $N = 276$, $p < 10^{-10}$), so we report MI measured over the aggregate of all three voice datasets (Table 3.6). A general pattern which is visually apparent is for the larger MI values to be confined to a subset of features: the central features in

the table – MFCCs, ΔMFCCs, and spectral crests – each show only small MI with any other feature, while the larger MIs are confined to other features, in particular the spectral percentile and subband power measures. (Other features exhibiting only small MIs are clarity, spectral slope and spectral flux.)

The MFCC calculation does include an approximate decorrelation using a Discrete Cosine Transform [Rabiner and Schafer, 1978] (done in order to compact spectral energy into the lower coefficients as well as for approximate decorrelation), which provides a theoretical reason to expect the within-set independence of MFCCs. The spectral crest calculation does not deliberately decorrelate the subbands, so the within-set independence is perhaps more notable.

The features *pitch* and *power* are not usually considered timbral features (cf. Section 2.3.3), and are included to probe dependencies between them and timbre-related features. In this dataset we see only small interactions: *slope* is the only feature which shares more than 0.5 bits of information with *power*, and no feature shares that much information with *pitch*.

The larger MI values are mainly found among feature pairs drawn from the spectral percentile and subband power measures. This is perhaps unsurprising given the strong formal connection between the calculations: the 95-percentile is inherently constrained never to take a value lower than the 75-percentile, for example, while a *pow1* value greater than 0.5 would tell us that at least 50% of the spectral power lies below 400 Hz (the top of the subband) and therefore that the lower percentiles must be below that level.

The spectral centroid and spread also show some interaction with the spectral percentile measures. The centroid has its strongest interaction with *pcile75*, and the spread with *pcile95*, suggesting that these parametric and nonparametric representations (i.e. moments and percentiles, respectively) are alternatives which to some extent capture the same information about the spectral shape. Compare this with the results of Section 3.2, which found both *centroid* and *pcile95* to be strong correlates with the perceptually-derived dimensions said to relate to brightness. This would lead us to expect a rather high informational overlap between the two features, which is what we find. However this overlap is not the highest MI detected, suggesting that there may be scope to tease apart the relation between the two measures and perceptual brightness, in future studies.

3.5 Discussion and conclusions

In choosing acoustic features to represent timbre, we wish to select features which capture perceptible variation yet which are robust to minor signal variations, and ideally which form a compact subset without too much informational

overlap. The criteria we have considered in this chapter go some way toward helping us to make such a selection, each leading to recommendations for or against some subsets of the features we have investigated.

The perceptual tests (Section 3.2) confirmed spectral centroid and spectral 95-percentile as strong predictors of a timbral dimension recovered from MDS experiments. However there was little agreement across the perceptual tests about correlates for other axes. In particular the log attack time was not confirmed as a consistent strong correlate. Despite this, timbral variation is indeed richer than just this one dimension, as indicated by the MDS experiments of others, so in order to try and capture some of this richness we should make use of other features, to present further information to our machine learning algorithms which may help make useful decisions.

Robustness of measurements is important to avoid passing too much irrelevant information (e.g. originating from background noise) on to the later processing. Our robustness tests (Sections 3.3.1 and 3.3.2) yielded some agreement over the relative merits of features. In particular the ΔMFCCs were shown to be highly sensitive to noise and variation, and to a lesser extent so were the spectral crest measures. This is useful information given that the ΔMFCCs are quite commonly used in e.g. speech analysis [Mak et al., 2005]. Strongly-performing features from the robustness tests include spectral centroid, spectral percentiles, spectral spread, subband powers, and spectral flatness. Notably, the spectral centroid and spectral 95-percentile recommended from our perceptual experiment generally exhibited good robustness, indicating that the brightness dimension can be characterised quite dependably.

The MFCCs performed relatively poorly in the robustness tests, although not as poorly as the ΔMFCCs and crest features. These results reflect a theme found in the literature, that MFCCs although useful are quite sensitive to noise – this issue and some potential remedies have been discussed for analysis of speech [Gu and Rose, 2001, Chen et al., 2004, Tyagi and Wellekens, 2005] and music [Seo et al., 2005].

The independence experiment (Section 3.4) shows that MFCCs, ΔMFCCs and spectral crest factors all show a particularly low degree of information overlap among themselves or with other features. Conversely there is a general indication that the subband powers and the spectral percentiles, taken together, form a subset with quite a lot of redundancy. Therefore a multidimensional timbre space need only use some of those features in order to capture much of the information they provide.

Despite the tensions in the experimental data, it is possible to draw some conclusions about the suitability of the timbre features studied, for our application in real-time timbre analysis of voice and of synthesisers. Spectral centroid

and spectral 95-percentile are recommended for their perceptual relevance and robustness. Some subset of subband powers and spectral percentiles are recommended as a robust class of features albeit with some redundancy. Spectral crests and ΔMFCCs are not recommended since they show particularly poor robustness in our tests. We will take account of these conclusions in future chapters, when designing machine learning techniques based on continuous timbre features.

Chapter 4

Event-based paradigm

In real-time signal processing it is often useful to identify and classify events represented within a signal. With music signals this need arises in applications such as live music transcription [Brossier, 2007] and human-machine musical interaction [Collins, 2006, Aucouturier and Pachet, 2006]. This could be a fruitful approach for voice-driven musical systems, detecting vocal events and triggering sounds such as synthesiser notes or samples. Indeed some prior work has explored this potential in non-real time [Sinyor et al., 2005] and in real time [Hazan, 2005b, Collins, 2004].

Yet to respond to events in real time presents a dilemma: often we wish a system to react with low latency, perhaps as soon as the beginning of an event is detected, but we also wish it to react with high precision, which may imply waiting until all information about the event has been received so as to make an optimal classification. The acceptable balance between these two demands will depend on the application context. In music, the perceptible event latency can be held to be around 30 ms, depending on the type of musical signal [Mäki-Patola and Hämäläinen, 2004].

We propose to deal with this dilemma by allowing event triggering and classification to occur at different times, thus allowing a fast reaction to be combined with an accurate classification. Triggering prior to classification implies that for a short period of time the system would need to respond using only a provisional classification, or some generic response. It could thus be used in reactive music systems if it were acceptable for some initial sound to be emitted even if the system's decision might change soon afterwards and the output updated accordingly. To evaluate such a technique applied to real-time music processing, we need to understand not only the scope for improved classification at increased latency, but also the extent to which such delayed decision-making affects the listening experience, when reflected in the audio output.

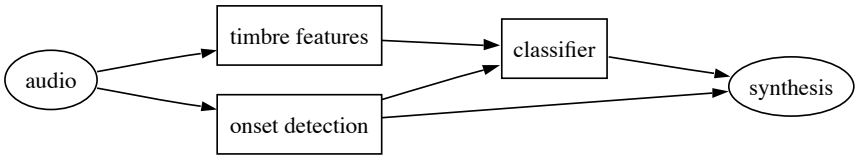


Figure 4.1: An approach to real-time beatbox-driven audio, using onset detection and classification.

In this chapter we investigate delayed decision-making in the context of musical control by vocal percussion in the human beatbox style (discussed in Section 2.2). We consider the imitation of drum sounds commonly used in Western popular music such as kick (bass) drum, snare and hihat (for definitions of drum names see Randel [2003]). The classification of vocal sounds into such categories offers the potential for musical control by beatboxing.

This chapter investigates two aspects of the delayed decision-making concept. In Section 4.1 we study the relationship between latency and classification accuracy: we present an annotated dataset of human beatbox recordings, and describe classification experiments on these data. Then in Section 4.2 we describe a perceptual experiment using sampled drum sounds as could be controlled by live beatbox classification. The experiment investigates bounds on the tolerable latency of decision-making in such a context, and therefore the extent to which delayed decision-making can help resolve the tension between a system's speed of reaction and its accuracy of classification.

4.1 Classification experiment

We wish to be able to classify percussion events in an audio stream such as beatboxing, for example a three-way classification into kick/hihat/snare event types. We might for example use an onset detector to detect events, then use acoustic features measured from the audio stream at the time of onset as input to a classifier which has been trained using appropriate example sounds (Figure 4.1) [Hazan, 2005b]. In such an application there are many options which will bear upon performance, including the choice of onset detector, acoustic features, classifier and training material. In the present experiment we factor out the influence of the onset detector by using manually-annotated onsets, and we introduce a real-world dataset for beatbox classification which we describe below.

We wish to investigate the hypothesis that the performance of some real-time classifier would improve if it were allowed to delay its decision so as to receive more information. In order that our results may be generalised we will use a

classifier-independent measure of class separability, as well as results derived using a specific (although general-purpose) classifier.

To estimate class separability independent of a classifier we use the Kullback-Leibler divergence (KL divergence, also called the relative entropy) between the continuous feature distributions for classes [Cover and Thomas, 2006, Section 9.5]:

$$D_{KL}(f||g) = \int f(x) \log \frac{f(x)}{g(x)} dx \tag{4.1}$$

where f and g are the densities of the features for two classes. The KL divergence is an information-theoretic measure of the amount by which one probability distribution differs from another. It can be estimated from data with few assumptions about the underlying distributions, so has broad applicability. It is nonnegative and non-symmetric, although can be symmetrised by taking the value $D_{KL}(f||g) + D_{KL}(g||f)$ [Arndt, 2001, Section 9.2]; in the present experiment we will further symmetrise over multiple classes by averaging D_{KL} over all class pairs to give a summary measure of the separability of the distributions. Because of the difficulties in estimating high-dimensional densities from data [Hastie et al., 2001, Chapter 2] we will use divergence measures calculated for each feature separately, rather than in the high-dimensional joint feature space.

To provide a more concrete study of classifier performance we will also apply a Naïve Bayes classifier [Langley et al., 1992], which estimates distributions separately for each input feature and then derives class probabilities for a datum simply by multiplying together the probabilities due to each feature. This classifier is selected for multiple reasons:

- It is a relatively simple and generic classifier, and well-studied, and so may be held to be a representative choice;
- Despite its simplicity and unrealistic assumptions (such as independence of features), it often achieves good classification results even in cases where its assumptions are not met [Domingos and Pazzani, 1997];
- The independence assumption makes possible an efficient updateable classifier in the real-time context: the class probabilities calculated using an initial set of features can be later updated with extra features, simply by multiplying by the probabilities derived from the new set of features.

Both our KL divergence estimates and our Naïve Bayes classification results operate on features independently. In this chapter we do not consider issues of redundancy between features.

4.1.1 Human beatbox dataset: *beatboxset1*

To facilitate the study of human beatbox audio we have collected and published a dataset which we call *beatboxset1*.[1] It consists of short recordings of beatboxing recorded by amateur and semi-professional beatboxers recorded under heterogenous conditions, as well as onset times and event classification annotations marked by independent annotators. The audio and metadata are freely available and published under the Creative Commons Attribution-Share Alike 3.0 license.

Audio: The audio files are 14 recordings each by a different beatboxer, between 12 and 95 seconds in length (mean duration 47 seconds). Audio files were recorded by the contributors, in a range of conditions: differing microphone type, recording equipment and background noise levels. The clips were provided by users of the website `humanbeatbox.com`.

Annotations: Annotations of the beatbox data were made by two independent annotators. Individual event onset locations were annotated, along with a category label. The labels used are given in Table 4.1. Files were annotated using Sonic Visualiser 1.5,[2] via a combination of listening and inspection of waveforms/spectrograms. A total of 7460 event annotations were recorded (3849 from one annotator, 3611 from the other).

The labelling scheme we propose in Table 4.1 was developed to group sounds into the main categories of sound heard in a beatboxing stream, and to provide for efficient data entry by annotators. For comparison, the table also lists the labels used for a five-way classification by Sinyor et al. [2005], as well as symbols from Standard Beatbox Notation (SBN – a simplified type of score notation for beatbox performers[3]). Our labelling is oriented around the sounds produced rather than the mechanics of production (as in SBN), but aggregates over the fine phonetic details of each realisation (as would be shown in an International Phonetic Alphabet transcription).

Table 4.2 gives the frequency of occurrence of each of the class labels, confirming that the majority (74%) of the events fall broadly into the kick, hihat, and snare categories.

[1] `http://archive.org/details/beatboxset1`
[2] `http://sonicvisualiser.org`
[3] `http://www.humanbeatbox.com/tips/`

Label	Description	SBN	Sinyor
k	Kick	b / .	kick
hc	Hihat, closed	t	closed
ho	Hihat, open	tss	open
sb	Snare, *bish* or *pss*-like	psh	p-snare
sk	Snare, *k*-like (clap or rimshot snare sound)	k	k-snare
s	Snare but not fitting the above types	–	–
t	Tom	–	–
br	Breath sound (not intended to sound like percussion)	h	–
m	Humming or similar (a note with no drum-like or speech-like nature)	m	–
v	Speech or singing	[words]	–
x	Miscellaneous other sound	–	–
?	Unsure of classification	–	–

Table 4.1: Event labelling scheme used in *beatboxset1*.

4.1.2 Method

To perform a three-way classification experiment on *beatboxset1* we aggregated the labelled classes into the three main types of percussion sound:

- kick (label `k`; 1623 instances),
- snare (labels `s`, `sb`, `sk`; 1675 instances),
- hihat (labels `hc`, `ho`; 2216 instances).

The events labelled with other classes were not included in this experiment.

We analysed the soundfiles to produce the set of 24 features listed in Table 4.3. Features were derived using a 44.1 kHz audio sampling rate, and a frame size of 1024 samples (23 ms) with 50% overlap (giving a feature sampling rate of 86.1 Hz). This set of features is slightly different from that used in Chapter 3 (Table 3.1) since the experiment was conducted before that work concluded, although majority of features are the same.

Each manually-annotated onset was aligned with the first audio frame containing it (the earliest frame in which an onset could be expected to be detected in a real-time system). In the following, the amount of delay will be specified in numbers of frames relative to that aligned frame, as illustrated in Figure 4.2. We investigated delays of zero through to seven frames, corresponding to a latency of 0–81 ms.

Label	Count
k	1623
hc	1840
ho	376
sb	469
sk	1025
s	181
Sum	5514 (74%)

(a) Main

Label	Count
t	201
br	132
m	404
v	76
x	1072
?	61
Sum	1946 (26%)

(b) Others

Table 4.2: Frequencies of occurrence of classes in *beatboxset1* annotations, grouped into the main kick/hihat/snare sounds versus others.

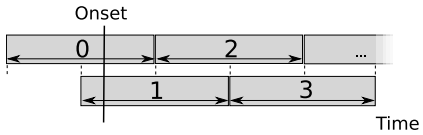


Figure 4.2: Numbering the "delay" of audio frames relative to the temporal location of an annotated onset.

To estimate the KL divergence from data, we used a Gaussian kernel estimate for the distribution of each feature separately for each class. For each feature we then estimated the KL divergence pairwise between classes, by numerical integration over the estimated distributions (since the KL divergence is a directed measure, there are six pairwise measures for the three classes). To summarise the separability of the three classes we report the mean of the six estimated divergences, which gives a symmetrised measure of divergence between the three classes.

In applying the Naïve Bayes classifier, we investigated various strategies for choosing features as input to the classifier, exploring "stacking" as well as feature selection:

Feature stacking: We first used only the features derived from the frame at a single delay value (as with the divergence measures above). However, as we delay the decision, the information from earlier frames is in principle available to the classifier, so we should be able to improve classification performance by making use of this extra information – in the simplest case by "stacking" feature values, creating a larger featureset of the union of the features from multiple frames [Meng, 2006, Section 4.2]. Therefore we also performed classification at each delay using the fully stacked featuresets, aggregating all frames from onset up to the specified delay. Our 24-feature set at zero delay would become a 48-feature set at one frame delay, then a 72-feature set at two frames' delay,

Label	Feature
mfcc1–mfcc8	Eight MFCCs, derived from 42 Mel-spaced filters (zero'th MFCC not included)
centroid	Spectral centroid
spread	Spectral spread
scf	Spectral crest factor
scf1–scf4	Spectral crest factor in subbands (50–400, 400–800, 800–1600, and 1600–3200 Hz)
25%ile–95%ile	Spectral distribution percentiles: 25%, 50%, 90%, 95% ("rolloff")
HFC	High-frequency content
ZCR	Zero-crossing rate
flatness	Spectral flatness
flux	Spectral flux
slope	Spectral slope

Table 4.3: Acoustic features measured for classification experiment (cf. the features used in Chapter 3 [Table 3.1]).

and so forth.

Feature selection: Stacking features creates very large featuresets and so risks incurring curse of dimensionality issues, well known in machine learning: large dimensionalities can reduce the effectiveness of classifiers, or at least require exponentially more training data to prevent overfitting (see Section 2.3.4). To circumvent the curse of dimensionality yet combine information from different frames, we applied two forms of feature-selection. The first used each of our 24 features once only, but taken at the amount of delay corresponding to the best class separability for that feature. The second applied a standard feature-selection algorithm to choose the 24 best features at different delays, allowing it to choose a feature multiple times at different delays. We used the Information Gain selection algorithm [Mitchell, 1997, Section 3.4.1] for this purpose.

In total we investigated four featuresets derived from our input features: the plain non-stacked features, the fully stacked featureset, the stacked featureset reduced by class-separability feature-selection, and the stacked featureset reduced by Information Gain feature-selection.

We used SuperCollider 3.3 [McCartney, 2002] for feature analysis, with Hann windowing applied to frames before spectral analysis. KL divergence was estimated using *gaussian_kde* from the SciPy 0.7.1 package, running in Python 2.5.4, with bandwidth selection by Scott's Rule. Classification experiments were performed using Weka 3.5.6 [Witten and Frank, 2005], using ten-fold cross-validation.

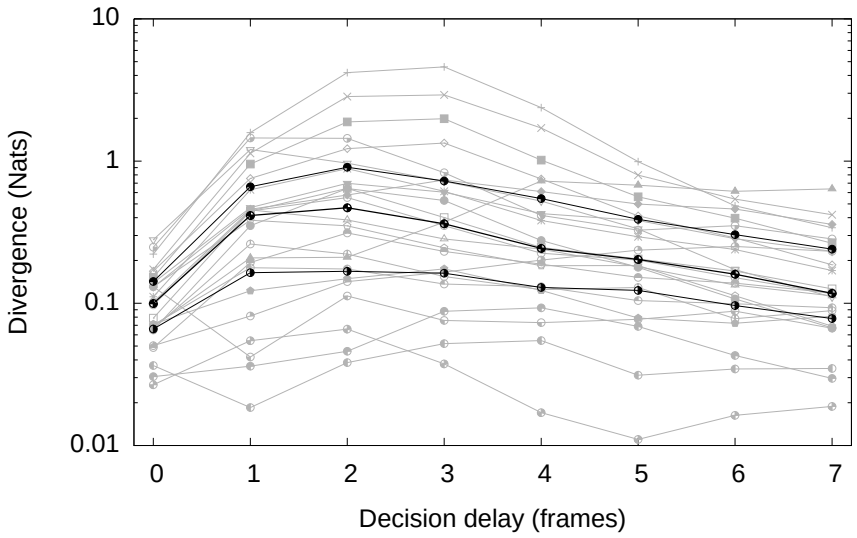


Figure 4.3: Separability measured by average KL divergence, as a function of the delay after onset. At each frame the class separability is summarised using the feature values measured only in that frame. The grey lines indicate the individual divergence statistics for each of the 24 features, while the dark lines indicate the median and the 25- and 75-percentiles of these values.

4.1.3 Results

The class separability measured by average KL divergence between classes is given in Figure 4.3, and the peak values for each feature in Table 4.4. The values of the divergences cover a broad range depending on both the feature type and the amount of delay, and in general a delay of around 2 frames (23 ms) appears under this measure to give the best class separation. Note that this analysis considers each amount of delay separately, ignoring the information available in earlier frames. The separability at zero delay is generally the poorest of all the delays studied here, which is perhaps unsurprising, as the audio frame containing the onset will often contain a small amount of unrelated audio prior to the onset plus some of the quietest sound in the beginning of the attack. The peak separability for the features appears to show some variation, occurring at delays ranging from 1 to 4 frames. The highest peaks occur in the spectral 25- and 50-percentile (at 3 frames' delay), suggesting that the distribution of energy in the lower part of the spectrum may be the clearest differentiator between the classes.

The class separability measurements are reflected in the performance of the Naïve Bayes classifier on our three-way classification test (Figure 4.4). When using only the information from the latest frame at each delay the data show a

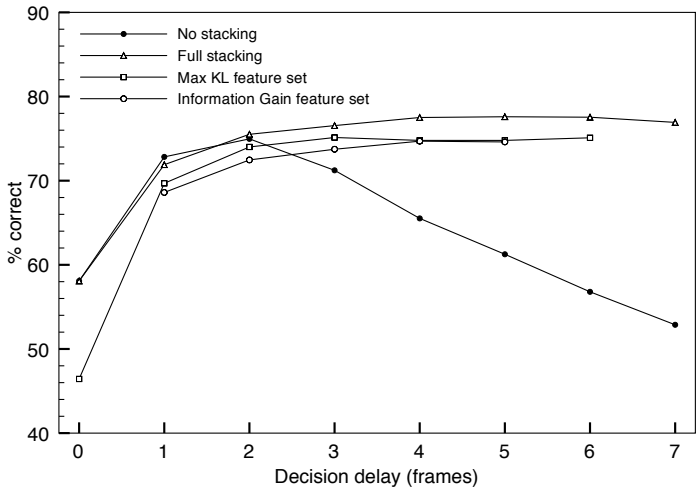


Figure 4.4: Classification accuracy using Naïve Bayes classifier.

similar curve: poor performance at zero delay, rising to a strong performance at 1 to 3 frames' delay (peaking at 75.0% for 2 frames), then tailing off gradually at larger delays.

When using feature stacking the classifier is able to perform strongly at the later delays, having access to information from the informative early frames, although a slight curse of dimensionality effect is visible in the very longest delays we investigated: the classification accuracy peaks at 5 frames (77.6%) and tails off afterwards, even though the classifier is given the exact same information plus some extra features. Overall, the improvement due to feature stacking is small compared against the single-frame peak performance. Such a small advantage would need to be balanced against the increased memory requirements and complexity of a classifier implemented in a real-time system – although as previously mentioned, the independence assumption of the classifier allows frame information to be combined at relatively low complexity.

We also performed feature selection as described earlier, first using the peak-performing delays given in Table 4.4 and then using features/delays selected using Information Gain (Table 4.5). In both cases some of the selected features are unavailable in the earlier stages so the feature set is of low dimensionality, only reaching 24 dimensions at the 5- or 6-frame delay point. The performance of these sets shows a similar trajectory to the full stacked feature set although consistently slightly inferior to it. The Information Gain approach is in a sense less constrained than the former approach – it may select a feature more than once at different delays – yet does not show superior performance, suggesting that the variety of features is more important than the varieties of delay in

Feature	Delay	Divergence
mfcc1	3	1.338
mfcc2	3	0.7369
mfcc3	1	0.3837
mfcc4	3	0.1747
mfcc5	1	0.2613
mfcc6	6	0.2512
mfcc7	1	0.1778
mfcc8	2	0.312
centroid	3	1.9857
spread	2	0.5546
scf	2	0.6975
scf1	0	0.1312
scf2	2	0.0658
scf3	4	0.0547
scf4	4	0.0929
25%ile	3	4.6005
50%ile	3	2.9217
90%ile	2	0.8857
95%ile	2	0.6427
HFC	4	0.7245
ZCR	1	0.454
flatness	2	0.6412
flux	1	1.2058
slope	1	1.453

Table 4.4: The delay giving the peak symmetrised KL divergence for each feature.

classification performance.

The Information Gain feature selections (Table 4.5) also suggest which of our features may be generally best for the beatbox classification task. The 25- and 50-percentile are highly ranked (confirming our observation made on the divergence measures), as are the spectral centroid and spectral flux.

In summary, we find that with this dataset of beatboxing recorded under heterogeneous conditions, a delay of around 2 frames (23 ms) relative to onset leads to stronger classification performance.[4] Feature stacking further improves classification results for decisions delayed by 2 frames or more, although at the cost of increased dimensionality of the feature space. Reducing the dimensionality by feature selection over the different amounts of delay can provide good classification results at large delays with low complexity, but fails to show im-

[4]Compare e.g. Brossier [2007, Section 5.3.3], who finds that for real-time pitch-tracking of musical instruments, reliable note estimation is not possible until around 45 ms after onset. This suggests for example that for a system performing real-time pitch-tracking as well as event classification, a delay of 23 ms could well be acceptable since it would not be the limiting factor on overall analysis latency.

Rank	Feature	Delay	Rank	Feature	Delay
1	*50%ile*	2	13	*mfcc1*	2
2	*centroid*	2	14	*90%ile*	2
3	*50%ile*	3	15	*slope*	2
4	*centroid*	3	16	*25%ile*	1
5	*25%ile*	2	17	*50%ile*	5
6	*flux*	1	18	*flux*	3
7	*flux*	2	19	*ZCR*	1
8	*50%ile*	4	20	*25%ile*	4
9	*50%ile*	1	21	*centroid*	4
10	*slope*	1	22	*mfcc1*	1
11	*centroid*	1	23	*mfcc1*	3
12	*25%ile*	3	24	*90%ile*	1

Table 4.5: The 24 features and delays selected using Information Gain, out of a possible 192.

provement over the classifier performance simply using the features at the best delay of 2 frames.

In Figure 4.5 we show the waveform and spectrogram of a kick and a snare from the dataset. The example shows that the snare and kick sounds do not differ strongly in their spectral content at first – the main difference between the two sounds is that the snare "fills out" with more energy in the mid and upper frequencies (above $\sim$ 1 kHz) after the initial attack. From such evidence and from our experience of beatboxing techniques, we suggest that this reflects the importance of the beatboxer's manipulation of the resonance in the vocal cavity to create the characteristics of the different sounds. This can induce perceptibly different sounds, but its effect on the signal does not develop immediately. It therefore suggests that the experimentally observed benefit of delayed decision-making may be particularly important for beatboxing sounds as opposed to some other percussion sounds.

In designing a system for real-time beatbox classification, then, a classification at the earliest possible opportunity is likely to be suboptimal, especially when using known onsets or an onset detector designed for low-latency response. Classification delayed until roughly 10–20 ms after onset detection would provide better performance. Features characterising the distribution of the lower-frequency energy (the spectral 25- and 50-percentiles and centroid) can be recommended for this task.

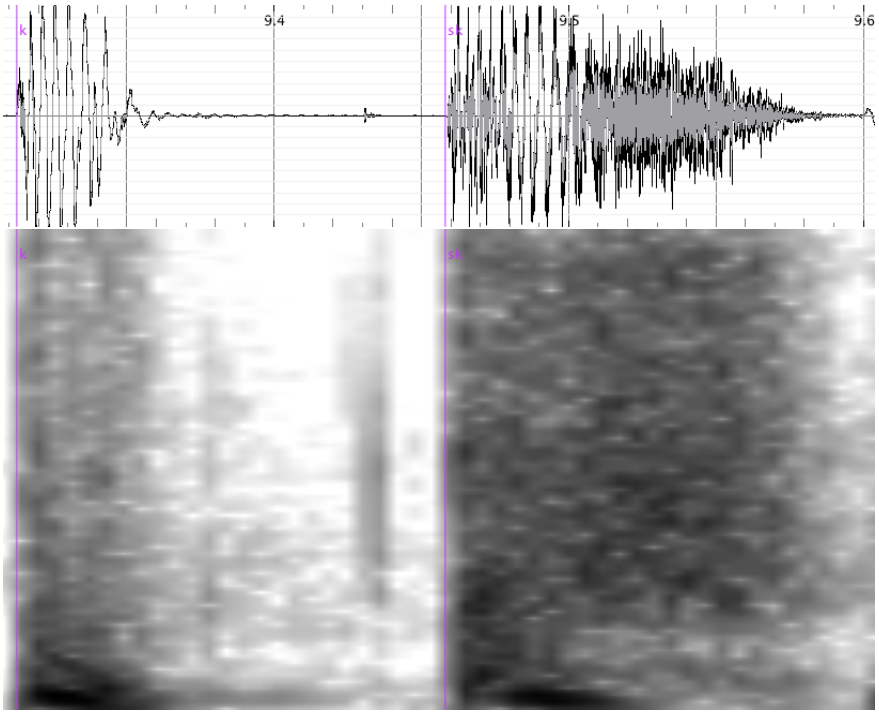


Figure 4.5: Waveform and spectrogram of a kick followed by a snare, from the *beatboxset1* data. The duration of the excerpt is around 0.3 seconds, and the spectrogram frequencies shown are 0–6500 Hz.

4.2 Perceptual experiment

In Section 4.1 we confirmed that beatbox classification can be improved by delaying decision-making relative to the event onset. Adding this extra latency to the audio output may be undesirable in a real-time percussive performance, hence our proposal that a low-latency low-accuracy output could be updated some milliseconds later with an improved classification. This two-step approach would affect the nature of the output audio, so we next investigate the likely effect on audio quality via a listening test.

Our test will be based on the model of an interactive musical system which can trigger sound samples, yet which allows that the decision about which sound sample to trigger may be updated some milliseconds later. Between the initial trigger and the final classification the system might begin to output the most likely sample according to initial information, or a mixture of all the possible samples, or some generic "placeholder" sound such as pink noise. The resulting audio output may therefore contain some degree of inappropriate or distracting content in the attack segments of events. It is known that the attack portion of musical sounds carries salient timbre information, although that information

is to some extent redundantly distributed across the attack and later portions of the sound [Iverson and Krumhansl, 1993]. Our research question here is the extent to which the inappropriate attack content introduced by delayed decision-making impedes the perceived quality of the audio stream produced.

4.2.1 Method

We first created a set of audio stimuli for use in the listening test. The delayed-classification concept was implemented in the generation of a set of drum loop recordings as follows: for a given drum hit, the desired sound (e.g. kick) was not output at first, rather an equal mixture of kick, hihat and snare sounds was output. Then after the chosen delay time the mixture was crossfaded (with a 1 ms sinusoidal crossfade) to become purely the desired sound. The resulting signal could be considered to be a drum loop in which the onset timings were preserved, but the onsets of the samples had been degraded by contamination with other sound samples. We investigated amounts of delay corresponding to 1, 2, 3 and 4 frames as in the earlier classifier experiment (Section 4.1) - approximately 12, 23, 35 and 46 ms.

Sound excerpts generated by this method therefore represent a kind of idealised and simplified delayed decision-making in which no information is available at the moment of onset (hence the equal balance of all drum types) and 100% classification accuracy occurs after the specified delay. Our classifier experiment (Section 4.1) indicates that in a real-time classification system, some information is available soon after onset, and also that classification is unlikely to achieve perfect classification accuracy. The current experiment factors out such issues of classifier performance to focus on the perceptual effect of delayed decision-making in itself.

The reference signals were each 8 seconds of drum loops at 120bpm with one drum sample (kick/snare/hihat) being played on every eighth-note. Three drum patterns were created using standard dance/pop rhythms, such that the three classes of sound were equally represented across the patterns. The patterns were (using notation k=kick, h=hihat, s=snare):

```
k k s h h k s h
k h s s k k s h
k h s k h s h s
```

We created the sound excerpts separately with three different sets of drum sound samples, which were chosen to be representative of standard dance/pop drum sounds as well as providing different levels of susceptibility to degradation induced by delayed classification:

Immediate-onset samples, designed using SuperCollider to give kick/hihat/snare

sounds, but with short duration and zero attack time, so as to provide a strong test for the delayed classification. This drum set was expected to provide poor acceptability at even moderate amounts of delay.

Roland TR909 samples, taken from one of the most popular drum synthesisers in dance music [Butler, 2006, p. 326], with a moderately realistic sound. This drum set was expected to provide moderate acceptability results.

Amen break, originally sampled from "Amen brother" by The Winstons and later the basis of jungle, breakcore and other genres, now the most popular breakbeat in dance music [Butler, 2006, p. 78]. The sound samples are much less "clean" than the other sound samples (all three samples clearly contain the sound of a ride cymbal, for example). Therefore this set was expected to provide more robust acceptance results than the other sets, yet still represent a commonly-used class of drum sound.

The amplitude of the three sets of audio excerpts was adjusted manually by the first author for equal loudness.

Tests were performed within the "MUlti Stimulus test with Hidden Reference and Anchor" (MUSHRA) standard framework [International Telecommunication Union, 2003]. In the MUSHRA test participants are presented with sets of processed audio excerpts and asked to rate their *basic audio quality* in relation to a reference unprocessed audio excerpt. Each set of excerpts includes the unprocessed audio as a hidden reference, plus a 3.5 kHz low-pass filtered version of the excerpt as a low-quality anchor, as well as excerpts produced by the systems investigated.

Our MUSHRA tests were fully balanced over all combinations of the three drum sets and the three patterns, giving nine trials in total. In each trial, participants were presented with the unprocessed reference excerpt, plus six excerpts to be graded: the hidden reference, the filtered anchor, and the delayed-decision versions at 1, 2, 3 and 4 frames' delay (see Figure 4.6 for a screenshot of one trial). The order of the trials and of the excerpts within each trial was randomised.

Participants: We recruited 23 experienced music listeners (17 men and 6 women) aged between 23 and 43 (mean age 31.3). Tests took around 20–30 minutes in total to complete, including initial training, and were performed using headphones.

Post-screening was performed by numerical tests combined with manual inspection. For each participant we calculated correlations (Pearson's r and Spearman's ρ) of their gradings with the median of the gradings provided by

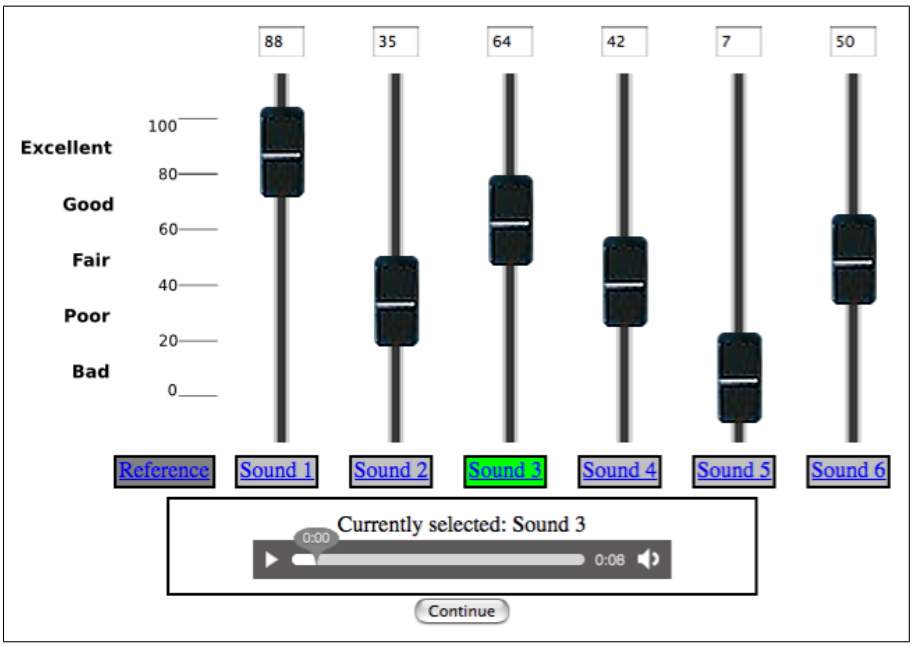


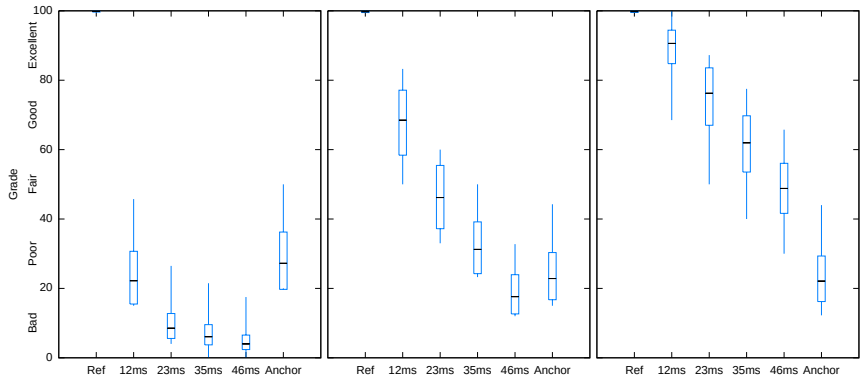
Figure 4.6: The user interface for one trial within the MUSHRA listening test.

the other participants. Any set of gradings with a low correlation was inspected as a possible outlier. Any set of gradings in which the hidden reference was not always rated at 100 was also inspected manually. (Ideally the hidden reference should always be rated at 100 since it is identical to the reference; however, participants tend to treat MUSHRA-type tasks to some extent as ranking tasks [Sporer et al., 2009], and so if they misidentify some other signal as the highest quality they may penalise the hidden reference slightly. Hence we did not automatically reject these.)

We also plotted the pairwise correlations between gradings for every pair of participants, to check for subgroup effects. No subgroups were found, and one outlier was identified and rejected. The remaining 22 participants' gradings were analysed as a single group.

The MUSHRA standard [International Telecommunication Union, 2003] recommends calculating the mean and confidence interval for listening test data. However, the grading scale is bounded (between 0 and 100) which can lead to difficulties using the standard normality assumption to calculate confidence intervals, especially at the extremes of the scale. To mitigate these issues we applied the logistic transformation [Siegel, 1988, Chapter 9]:

$$z = \log \frac{x + \delta}{100 + \delta - x}\,, \tag{4.2}$$



(a) Immediate-onset (b) TR909 (c) Amen break

Figure 4.7: Results from the listening test, showing the mean and 95% confidence intervals (calculated in the logistic transformation domain) with whiskers extending to the 25- and 75-percentiles. The plots show results for the three drum sets separately. The durations given on the horizontal axis indicate the delay, corresponding to 1/2/3/4 audio frames in the classification experiment.

where x is the original MUSHRA score and the δ is added to prevent boundary values from mapping to $\pm\infty$ (we used $\delta = 0.5$). Such transformation allows standard parametric tests to be applied more meaningfully (see also Lesaffre et al. [2007]). We calculated our statistics (mean, confidence intervals, t-tests) on the transformed values z before projecting back to the original domain.

The audio excerpts, participant responses, and analysis script for this experiment are published online.[5]

4.2.2 Results

For each kit, we investigated the differences pairwise between each of the six conditions (the four delay levels plus the reference and anchor). To determine whether the differences between conditions were significant we applied the paired samples t-test (in the logistic z domain; d.f. = 65) with a significance threshold of 0.01, applying Holm's procedure to control for multiple comparisons [Shaffer, 1995]. All differences were significant with the exception of the following pairs:

- Immediate-onset samples:
  - anchor and 12 ms
  - 23 ms and 35 ms
  - 35 ms and 46 ms

[5] `http://archive.org/details/dsmushradata09`

- Roland TR909 samples:
  - anchor and 35 ms
  - anchor and 46 ms

The logistic transformation mitigates against boundary effects when applying parametric tests. However the MUSHRA standard does not propose such transformation, so as an additional validation check we also applied the above test on the data in its original domain. In this instance the significance testing produced the same results.

Figure 4.7 summarises the results of the listening test. It confirms that for each of the drum sets, the degradation is perceptible by listeners since the reference is readily identifiable, and also that the listening quality becomes worse as the delay lengthens. It also demonstrates that the three drum sets vary in their robustness to this degradation, as expected.

The immediate-onset drum set was designed to provide a kind of lower bound on the acceptability, and it does indeed show very poor gradings under all of the delay lengths we investigated. Participants mostly found the audio quality to be worse than the low-pass filtered anchor, except in the 12 ms condition where no significant difference from the anchor was found, so we say that participants found the audio quality to be similarly poor as the anchor. For such a drum set, this indicates that delayed decision-making would likely be untenable.

The other two sets of drum sounds are more typical of drum sounds used in popular music, and both are relatively more robust to the degradation. Sound quality was rated as 60 or better (corresponding in the MUSHRA quality scale to *good* or *excellent*) at 12 ms for the TR909 set, and up as far as 35 ms for the Amen set. Even at 46 ms delay, the acceptability for the Amen set is much greater than that for the immediate-onset set at 12 ms delay.

When applied in a real-world implementation, the extent to which these perceptual quality measures reflect the amount of delay acceptable will depend on the application. For a live performance in which real-time controlled percussion is one component of a complete musical performance, the delays corresponding to good or excellent audio quality could well be acceptable, in return for an improved classification accuracy without added latency.

4.3 Conclusions

We have investigated delayed decision-making in real-time classification, as a strategy to allow for improved characterisation of events in real time without increasing the triggering latency of a system. This possibility depends on the

notion that small signal degradations introduced by using an indeterminate onset sound might be acceptable in terms of perceptual audio quality.

We introduced a new real-world beatboxing dataset *beatboxset1* and used it to investigate the improvement in classification that might result from delayed decision-making on such signals. A delay of 23 ms generally performed strongly out of those we tested. This compares favourably with e.g. the 45 ms minimum delay for pitch-tracking reported by Brossier [2007, Section 5.3.3]. Neither feature stacking nor feature selection across varying amounts of delay led to strong improvements over this performance.

In a MUSHRA-type listening test we then investigated the effect on perceptual audio quality of a degradation representative of delayed decision-making. We found that the resulting audio quality depended strongly on the type of percussion sound in use. The effect of delayed decision-making was readily perceptible in our listening test, and for some types of sound delayed decision-making led to unacceptable degradation (poor/bad quality) at any delay; but for common dance/pop drum sounds, the maximum delay which preserved an excellent or good audio quality varied from 12 ms to 35 ms.

Chapter 5

Continuous paradigm: timbre remapping

Chapter 4 represents an event-based paradigm for synthesiser control, an approach which has to some extent been dominant in digital music research (see for example the classification papers referred to in that chapter). However we wish in the longer term to move towards systems which can reflect the rich complexity of vocal expression, which means moving beyond a simplistic model such as classification over a small number of event types. It may mean augmenting such events with information that serves a kind of adjectival role (a "soft" snare, a "crisp" snare, etc.), or some aspect of fuzzy categorisation for data whose boundaries are themselves fuzzy. It may mean augmenting the events with information about modulations over time (a humming sound may begin gently but increase in harshness) or with longer-term information such as recognition of patterns (e.g. a drum-and-bass breakbeat pattern, which implies genre-derived roles for the constituent sounds which may not be discernible from the events considered in isolation).

But it may mean moving away from such categorisations, since the event model may well break down in various cases such as: sounds which combine aspects of two categories; sounds which overlap in time; indeterminate sounds which mean different things to different listeners. Further, the categorical approach could be said to apply a false emphasis to the basic categories chosen, even if modulations and variants are incorporated as extensions to the model. Human music perception is sufficiently rich, context-sensitive and culturally informed that it may be better to attempt to reproduce timbral variation in a continuous way, and allow the listener to interpret the continuous audio stream as an interplay of events and modulations as appropriate.

This is the motivation for this chapter, to develop methods for voice timbral control of synthesisers that are continuous in nature. We wish to take expressive vocal timbre modulations and reproduce them as timbre modulations in a synthesiser's output, which presents a kind of mapping problem in two senses. We must derive relationships between the input controls for the synthesiser and its output timbre. But we must also map vocal timbre usefully into target synthesiser timbre, in a way which accounts for broad differences between the two – the underlying distributions of the two are not the same (since they are not capable of the same range of timbres) and so the mapping should be able to infer timbral analogies. For example, if a singer produces their brightest sound, then it is reasonable that the best expressive mapping would be to the brightest sound that the target synthesiser can achieve, whether that is brighter or duller than the input.

In the present work we are considering instantaneous timbre as represented in the features discussed in Chapter 3. The temporal evolution of sounds can be modelled and may provide useful information for timbre-based control, but we leave that consideration for future work and focus on control through instantaneous timbre.

The organisation of this chapter is as follows: we first introduce our approach to this task, which we call *timbre remapping*, comparing it to related research in the field, and describing our early explorations based on existing machine learning methods. Those methods showed some limitations for the task in hand, so we then describe a novel method based on regression tree learning (Section 5.2). We demonstrate the application of this approach to timbre remapping in an experiment using concatenative synthesis, before concluding by discussing prospects for the future of such methods.

Note that the following chapter (Chapter 6) develops a user evaluation method and applies it to a timbre remapping system. Our empirical perspective on timbre remapping will therefore consist of that user evaluation taken together with the numerical experiments described in this chapter (Section 5.3).

5.1 Timbre remapping

The basis of what we call timbre remapping is outlined in Figure 5.1. We consider two probability distributions within a common space defined by a set of timbre features: one for the voice source, and one for the target synthesiser (synth). These two distributions have common axes, yet they may have different ranges (e.g. if the synth has a generally brighter sound than the voice) or other differences in their distributions. Given a timbre space as defined using acoustic features as discussed in Chapter 3, then, timbre remapping consists of

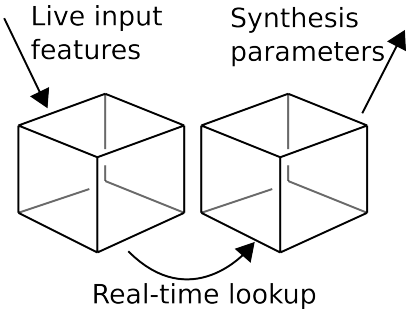


Figure 5.1: Overview of timbre remapping. Timbral input is mapped to synthesiser parameters by a real-time mapping between two timbre spaces, in a fashion which accounts for differences in the distribution of source and target timbre.

taking each vocal timbre coordinate and inferring a good choice of synth timbre coordinate to produce in response, and inferring the synth controls to use to create this timbre.

The mapping from vocal timbre coordinate to synth timbre coordinate could take a number of forms. It could be an identity mapping, making no correction for the different distributions of vocal vs. synth timbre. In some cases this might be useful, but generally we consider that this is less likely to be useful for expressive performance since it may generally put some synth settings beyond reach. This could be accommodated by a relatively simple normalisation (e.g. of mean and variance), which would eliminate broad differences of location but would not in general bring the distributions into strong alignment – it would not account for differences in the shapes of the distributions. Figure 5.3 (later in this chapter) is an illustrative example, in which the timbral distributions of two sound excerpts visibly exhibit general structural similarities but differences in shape.

Whatever the mapping, we wish it to be induced automatically from unlabelled timbre data, so that it can be applied to large datasets and/or a wide variety of synthesisers without requiring a large investment of human effort in annotation. In this chapter we will consider different types of mapping. Note that we broadly wish mappings to preserve *orientation* in timbre space: increased brightness in the source signal should generally produce increased brightness in the target signal, and so on.[1]

There is also a choice to be made in how synth timbre coordinates could be mapped (or "reverse engineered") into synth controls. If one can assume some parametric relationship between a control and a timbral attribute then one could infer the continuous mapping. For example, many synths have low-pass filters,

[1] "Brightness" was discussed in Sections 2.3.3 and 3.2.2.

whose cutoff frequency often has a direct connection with brightness; in such a case one could restrict mappings to linear or polynomial functions, to be fitted to data. However more generally such a neat relationship cannot be assumed. For example, frequency modulation (FM) synthesis is a relatively simple and widespread parametric synthesis technique, yet the relationship between input parameters and the output timbre is famously non-trivial [Chowning, 1973]. Many other modern techniques such as granular synthesis [Roads, 1988] or concatenative synthesis [Schwarz, 2005] have similarly intricate and nonlinear relationships between controls and timbre, as do commercial synthesiser circuits [General Instrument, 1979]. Therefore it may be preferable to use nonparametric techniques such as nearest-neighbour (NN) search to connect timbres with the synth controls which could produce them. This has some disadvantages – we lose the smooth interpolation of parameters that a parametric model could provide – but will preserve the general applicability of the technique to a wide variety of real-world synthesisers.

5.1.1 Related work

Previous work has investigated audio-driven systems which use continuous timbre features as input, whether for controlling audio effects [Verfaille et al., 2006] or synthesising sound [Beauchamp, 1982] – note in particular the work of Janer [2008] who like us focuses on real-time mapping from voice to instrumental timbre. However these depend on a fixed or user-specified mapping between input timbre and the algorithm controls, rather than automatic inference of the relationship.

Work also exists which performs automatic inference by finding a closest match between input and output timbre spaces [Puckette, 2004, Hoffman and Cook, 2007, Janer and de Boer, 2008]. These all operate via a relatively straightforward NN search, typically using a Euclidean distance metric, and so they may not address issues discussed above about accommodating the differences between distributions and learning to make the desired "analogies" between timbral trajectories.

5.1.2 Pitch–timbre dependence issues

It is standard practice and often convenient to treat pitch and timbre as separable aspects of musical sound (Section 2.3.3), whether considered as perceptual phenomena or as the acoustic features we measure to represent them. For example many synthesisers have a fundamental frequency control: in such cases, although there may be other controls which affect the fundamental frequency (such as a vibrato control), the frequency control is typically the overwhelm-

ing determinant of the pitch of the output, while other controls may separately affect the timbre. Yet as discussed in Section 2.3.3 there is psychoacoustic evidence of some interactions between the perception of pitch and timbre, and some common acoustic features used for timbre analysis can be affected when the fundamental frequency of the signal varies. So although our focus is on acoustic timbre features, it is worth considering the role of pitch estimation in our approach.

In Section 3.1 we included an autocorrelation-based pitch estimate as a candidate feature. One approach to handling pitch could be to follow Schoenberg (see quote in Section 2.3.3) and treat this simply as if it were any other timbre feature. This has a conceptual simplicity, and may have particular advantages – for example if we apply a decorrelation process to such data then the inclusion of the pitch dimension could help to separate the influence of pitch out from the other dimensions. However, it could also have important drawbacks: as we have argued, timbre remapping will need to take account of relative/contextual aspects of timbre – yet human pitch perception is closely related to fundamental frequency [van Besouw et al., 2008] and very sensitive to the octave relationships between notes [Houtsma and Smurzynski, 1990]. This tends to imply that our mapping process should not be deriving a nonlinear mapping of pitch, but rather should be able pass the estimated pitch directly to the target synthesiser, if it has a fundamental frequency input.

We therefore allocate a dual role for pitch estimation in the remapping process, illustrated in Figure 5.2. It is included in our set of potential timbre features, creating a timbre space in which pitch-dependencies can be implicitly accounted for, since this leaves open the possibility for mappings from two sounds to differ even if they differ only in estimated pitch and not in our timbre-features. Yet the pitch estimate can also be passed directly through to the target synth if there is a fundamental frequency control. If so, then any settings for the fundamental frequency control which are retrieved by the remapping are overridden by the information from the pitch tracker.

The remainder of this chapter discusses the development of two approaches to timbre remapping. In both of them, pitch tracking takes the dual role just described, although in Section 5.3.1 we present an experimental evaluation in which the role of pitch is deliberately minimised in order to focus on timbral aspects.

5.1.3 Nearest-neighbour search with PCA and warping

Using nearest-neighbour (NN) search is an obvious candidate for a mapping scheme such as timbre remapping, being simple in concept and in implementa-

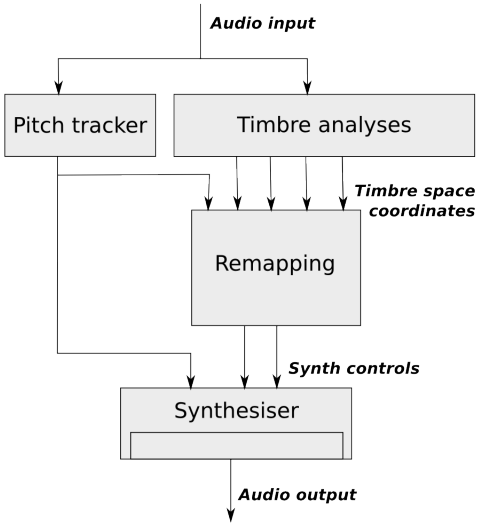


Figure 5.2: Pitch tracking serves a dual role in the timbre remapping process. It is used as an input "timbre feature", and if the target synth has a frequency control then it also directly drives that control. If the target synth does not have a frequency control then the estimated pitch is treated like any other timbre feature.

tion. The NN concept can be applied to a wide variety of metric spaces and a variety of distance metrics can be used [Chávez et al., 2001], although in the current context is typically implemented using Euclidean distance (see cited works in Section 5.1.1), on raw or normalised timbre features.

Two problems for the basic form of the NN search have already been raised. One is the curse of dimensionality, affecting search in high-dimensional spaces; and one is the difference in data distributions which may inhibit the ability of the NN search to produce useful timbral analogies. However, it may be that some modification or preprocessing steps could mitigate these issues and allow NN search to be applied usefully.

Our first approach to timbre remapping is indeed based on a NN search using Euclidean distance, with preprocessing applied to the timbre data to alleviate these two potential issues. We next consider each of the two issues and introduce the preprocessing steps which our implementation uses to alleviate them.

Curse of dimensionality; dimension reduction

Timbre spaces may often be of high dimensionality, being derived from a large number of acoustic features (e.g. the large featuresets used in Chapters 3 and

4). With high-dimensional spaces, the *curse of dimensionality* (Section 2.3.4) becomes a concern, and may reduce the effectiveness of NN search.

In Section 2.3.4 we introduced the concepts of feature selection and dimension reduction, either of which can be applied to mitigate the curse of dimensionality by projecting the data into a lower-dimensional space. One well-understood dimension reduction technique is Principal Component Analysis (PCA), which finds an orthogonal set of axes along which most of the variance in the data set lies [Morrison, 1983, Section 7.4]. By projecting the data onto these axes, a lower-dimensional dataset is created, which will typically discard some of the variation from the full dataset; however the PCA axes produced will conserve the largest amount of variance possible given the number of dimensions in the output. (The dimensionality of the output is a free parameter, not determined by the PCA algorithm, and so must be user-specified based on requirements or heuristics.) Further, the PCA axes are decorrelated, which can be beneficial for some tasks.

PCA is relatively simple to implement, and once the projection has been determined it is easy to apply: the projection is simply a matrix rotation, which can typically be carried out in a real-time system without imposing a large processing burden. Therefore in our NN lookup we use a PCA projection onto four dimensions as a preprocessing step. Choosing a 4D projection (i.e. the first 4 principal components) is relatively arbitrary but is motivated by the timbre literature discussed in Section 2.3.3 as well as studies such as reported by Alder et al. [1991] who argue that the intrinsic dimensionality of speech audio "may be about four, in so far as the set can be said to have a dimension".

Differing data distributions; warping

A key feature of the timbre remapping process should be the ability to map from one type of sound input onto a very different sound type. One issue is that the timbral measurements made on the 'source' and 'target' audio will often occupy different regions of the timbral space, as discussed in Section 5.1. Range normalisation could be used to align the source and target timbre spaces, but would be unable to account for differences in the shapes of the distributions, and so is only a partial solution.

One way to mitigate the effect of differences between data distributions is to transform the data to satisfy specific requirements on the distribution shape. Standardising the mean and variance, or the range, are simple transformations in this category; others include those which transform distributions to a more Gaussian shape (*Gaussianisation*) [Xiang et al., 2002], and the *probability integral transform* (PIT) which transforms univariate data (or the marginal distri-

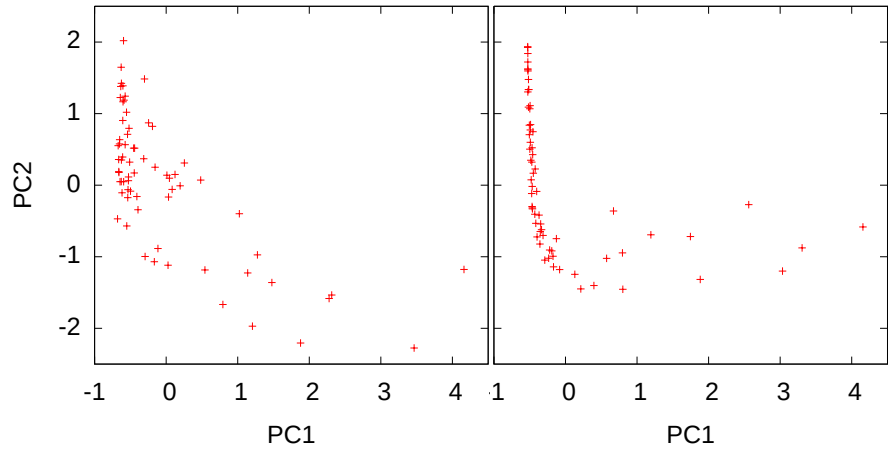


Figure 5.3: Two-dimensional PCA projections of timbre coordinates derived from analysis of the *Amen breakbeat* (left) and *thunder* (right) sound excerpts (described in Section 5.3.1). The timbre distributions have broad similarities in structure as well as differences: both show a non-linear interaction between the two axes yielding a curved profile; yet the second plot exhibits a sharper bend and a narrower distribution in the upper-left region. The common PCA rotation used for both projections was calculated using the balanced concatenation of the separately-standardised datasets (Equation (5.4)).

butions of multivariate data) to the uniform distribution $U(0, 1)$ [Angus, 1994, Nelsen, 2006]. Such methods are typically quite generally applicable, and the choice of which to use will depend on what is to be done with the data in later processing.

In this context – timbre-remapping using NN search on PCA-transformed timbre data – we wished to transform the data so that the data space was "well-covered" in the sense that any input data point would have a roughly equal chance of finding a nearest neighbour within a small radius. This translates quite naturally into a requirement to produce approximately uniform output distributions. We also wished to design a transformation which was efficient enough to run in real time and amenable to online learning (Section 2.3.4). The PIT is slightly problematic in this regard: it could be estimated from partial data (and therefore usable in online learning) but this would require the maintenance and updating of a large number of data quantiles in memory, which requires the maintenance of a list of data points received so far (or another layer of approximation [Chen et al., 2000]).

Instead we designed a linear piecewise warping using the statistics of minimum, maximum, mean and standard deviation, all of which statistics can easily be calculated online for an unbounded number of inputs. Given those statistics,

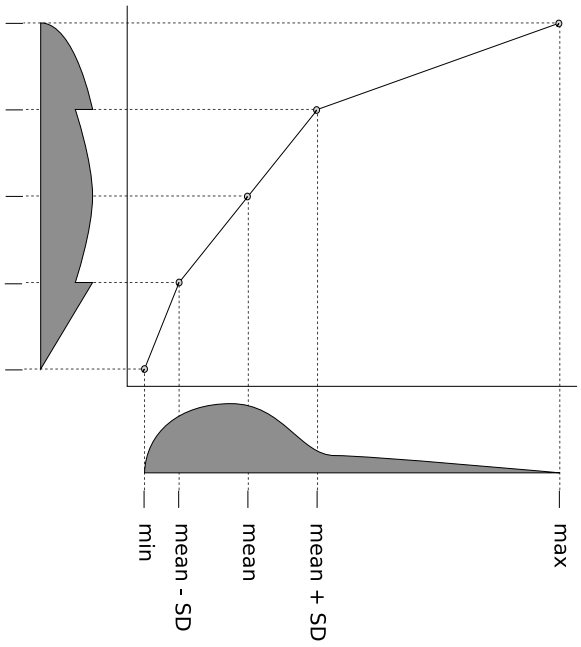


Figure 5.4: Illustration of the linear piecewise warping procedure, mapping regions of the data distribution to fixed intervals in the output (y axis).

our warping transformation is

$$f(x) = \begin{cases} (0.5-k)\frac{x-\alpha_x}{(\bar{x}-\sigma_x)-\alpha_x} & \text{if } x < (\bar{x}-\sigma_x), \\ 2k\frac{x-(\bar{x}-\sigma_x)}{(\bar{x}+\sigma_x)-(\bar{x}-\sigma_x)} + (0.5-k) & \text{if } (\bar{x}-\sigma_x) \leq x \leq (\bar{x}+\sigma_x), \\ (0.5-k)\frac{x-(\bar{x}+\sigma_x)}{\omega_x-(\bar{x}+\sigma_x)} + (0.5+k) & \text{if } (\bar{x}+\sigma_x) < x \end{cases} \quad (5.1)$$

where α_x, ω_x, $\bar{x}$ and σ_x are respectively the minimum, maximum, mean and standard deviation of input data x (estimated from sample statistics), and k a constant which controls the shape of the output distribution ($0 < k < 0.5$). Figure 5.4 depicts the application of $f(x)$ graphically. A typical warping with $k = 0.25$ might remap the minimum to 0, the mean-minus-one-standard-deviation to 0.25, the mean to 0.5, the mean-plus-one-standard-deviation to 0.75, and the maximum to 1. This is applied separately to each axis of our data. Figure 5.5 shows examples of the piecewise linear warping applied to different types of distribution.

The flow of information processing from audio through the PCA and warping steps to the well-covered timbre space is illustrated in Figure 5.6a. Timbre remapping in such a space is implemented by mapping an input point into the space (with a warping dependent on the source type, e.g. voice) and then

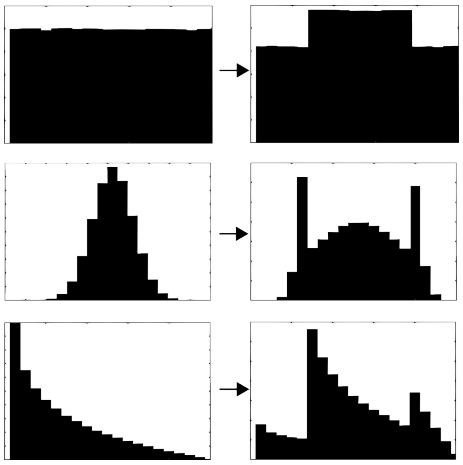


Figure 5.5: Illustration of the linear piecewise warping used in the PCA-based system, applied to sampled data from three types of distribution (uniform, Gaussian, and exponential). The distributions become more similar in the way they span the space. In this example all distributions are changed (for illustrative purposes) but with a suitable choice of the linear piecewise warping parameters, a transform can be produced which tends to leave e.g. uniformly-distributed data unchanged.

performing a NN search for a datum from the training set for the target synth. (The coordinates in the training set for the target synth are projected and warped in analogous fashion.) The control settings associated with that nearest neighbour are then sent to the synthesiser.

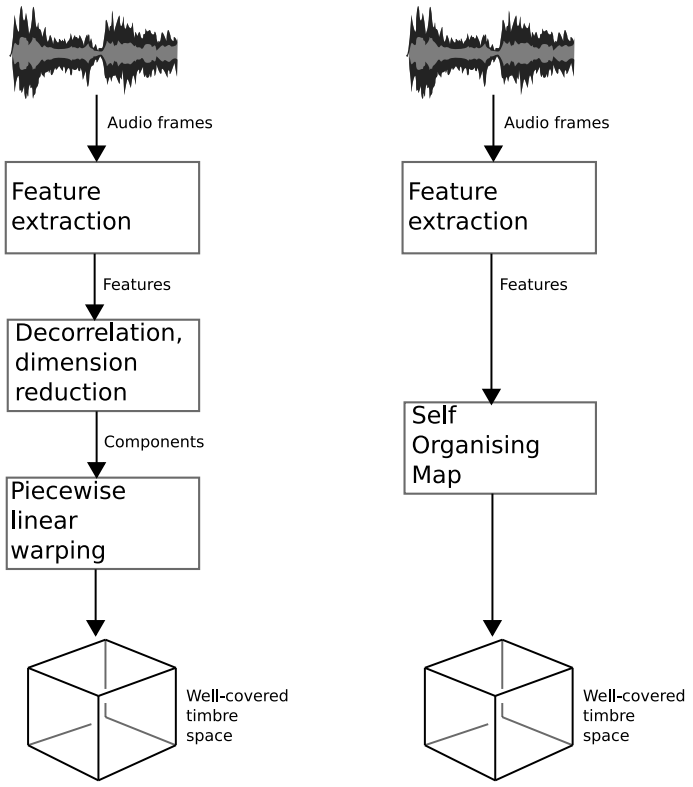
Implementation

We implemented the system in SuperCollider 3.3, providing components for the PCA rotation, the linear piecewise warping, and the NN lookup. All components were implemented to be amenable to online learning, with the exception of the learning of the PCA rotation matrix (although that is possible [Artac et al., 2002]) since offline PCA analysis was simplest to implement for prototyping.

Results

The PCA-based method was applied to a small selection of synths. We derived warping statistics for each synth as well as for a voice dataset, and built NN lookup tables for each synth based on random sampling of synth control settings.

In informal testing (with live microphone input) during development, we found that the method produced good mappings for some synths. The **ay1** synth and this PCA-based method formed the basis of a system used for live performances and demonstrations – notably, it was selected as a finalist in the



(a) PCA-based (see Section 5.1.3) (b) SOM-based (see Section D)

Figure 5.6: The PCA- and SOM-based approaches used to create a "well-covered" timbre space from audio data.

Guthman New Musical Instrument Competition 2009, held at Georgia Tech university (Altanta, Georgia, USA).[2] We then conducted a formal evaluation with a group of users, using the PCA-based method with one particular synth; the study found good results, encouraging the development of timbre remapping as an interface to vocal music-making. We defer discussion of the formal user evaluation study to Chapter 6, when we will broadly consider issues of evaluating such systems before concentrating on the evaluation of our timbre remapping system.

However we encountered some difficulties in applying the PCA-based method to some synths, in particular those with a large number of control settings. This may be because of the difficulties in sampling a large space of possible

[2] `http://www.wired.com/gadgets/mods/multimedia/2009/03/gallery_instruments`

control setting combinations. During the development process this led us to consider that the approach may be limited by its rather fragmented treatment of timbre space, as we will next discuss. In fact, in a quantitative experiment to be described later in this chapter (Section 5.3.1) we derived numerical results which suggest the improvement over a standard NN search is only modest.

Issues

One issue with the PCA-based method is that the piecewise warping is a rather arbitrary approach to standardising the shapes of distributions, and has some practical problems. One problem is visible in Figure 5.5, in that the piecewise nature of the transformation leads to rather odd changes in distribution densities at the transition points between differently-warped regions. There are also questions about NN search for data points near the transition points: if for example a point has two nearest neighbours in the warped space, one of which lies in the same region and one of which lies in a neighbouring region, is it reasonable to treat them as equally near?

Strongly skewed data would also cause a technical issue for our chosen warping scheme (Equation 5.1) since for example the mean-plus-one-standard deviation could extend beyond the data maximum, which would cause problems for our mapping function. One could swap or limit the mapping points in such cases, but such considerations primarily serve to highlight the arbitrary nature of the mapping.

A more fundamental issue with the scheme is that it is unable to account for dependences between the data axes. Since the warping is applied independently for each of the axes, it can only affect aspects of the marginal distribution, and cannot remove interactions in the joint distribution. For example, the interaction between dimensions shown in Figure 5.3 means that the warping process would leave a large unoccupied region within the joint density (in the top-right of the plots), where the nearest neighbour to an input point could actually be rather far away. Results from the quantitative experiment described later (Section 5.3.1) provide some evidence that such issues may indeed limit the usefulness of our modifications to NN search.

There are many ways one could address such issues, e.g. by designing some multidimensional warping scheme. However, there exist algorithms in the existing machine learning literature which can learn the structure of a data distribution in a continuous multidimensional space, and even provide a data structure which could be useful in performing the remapping. These hold the potential to support the timbre remapping process in a more theoretically elegant way than the PCA-based method, combining aspects of dimension reduction, nonlinear

mapping, and lookup into one scheme. In Appendix D we report investigations in using the Self-Organising Map (SOM) algorithm [Kohonen, 2001] for this purpose – investigations which were not ultimately fruitful, for reasons we consider in the appendix. In the remainder of this chapter we investigate a novel approach based on regression trees which is well-suited to our task.

5.2 The cross-associative multivariate regression tree (XAMRT)

To recap, we seek a technique which can learn the structure (including nonlinearities) of separate timbre data distributions in a timbre space (where the data distributions may be of relatively low intrinsic dimensionality compared against the extrinsic dimensionality, i.e. that of the space), and can learn to project from one such distribution into another so as to retrieve synth control settings. In this section we introduce a family of algorithms which perform an efficient nonparametric analysis of data distributions, and then introduce a novel variant which is well-suited to timbre remapping. We demonstrate this with a quantitative experiment on timbre remapping, and also show the potential application of our algorithm to other domains, through an experiment on speech vowel data.

The family of techniques known as classification and regression trees (CART) [Breiman et al., 1984] was developed as a computationally efficient nonparametric approach to analysing structure in a multivariate dataset, with a class label or a continuous-valued response variable to be predicted by the independent variables. The core concept is to recursively partition the dataset, at each step splitting it into two subsets using a threshold on one of the independent variables (i.e. a splitting hyperplane orthogonal to one axis). The choice of split at each step is made to minimise an "impurity" criterion (defined later) for the value of the response variable in the subsets. When the full tree has been grown it is likely to overfit the distribution, so it is typically then pruned by merging branches according to a cross-validation criterion to produce an optimally-sized tree.

CART methods have found application in a variety of disciplines and have spawned many variants [Murthy, 1998]. Classification and regression using such an algorithm are different but thematically similar; Breiman et al. [1984] develop both types, giving methods for choosing which split to make at each step, as well as pruning criteria. Classification trees are perhaps more commonly used than regression trees; here we focus on the latter. Note that tree-based methods are not restricted to datasets with an underlying hierarchical structure, rather they provide an efficient approach to general nonparametric modelling of the variation

and structure within a dataset. Tree methods are attractive in our context of timbre remapping because the recursive partitioning provides a generic approach to partitioning multidimensional distributions into regions of interest at multiple scales, with a common structure (e.g. a binary tree) that we might be able to use to association regions of different distributions one with another.

The standard CART is univariate in two senses: at each step only one variable is used to define the splitting threshold; and the response variable is univariate. The term "multivariate" has been used in the literature to refer to variants which are multivariate in one or other of these senses: for example Questier et al. [2005] regress a multivariate response variable, while Brodley and Utgoff [1995] use multivariate splits in constructing a classification tree; Gama [2004] considers both types of multivariate extension. In the following we will refer to "multivariate-response" or "multivariate-splits" variants as appropriate. Multivariate-splits variants can produce trees with reduced error [Gama, 2004], although the trees will usually be harder to interpret since the splitting planes are more conceptually complex.

We next consider a particular type of regression tree which was proposed for the unsupervised case, i.e. it does not learn to predict a class label or response variable, rather the structure in the data itself. We will extend this tree to include multivariate splits, before considering the cross-associative case.

5.2.1 Auto-associative MRT

Regression trees are studied in a feature-selection context by Questier et al. [2005], including their application in the unsupervised case, where there is no response variable for the independent variables to predict. The authors propose in that case to use the independent variables also as the response variables, yielding a regression tree task with a multivariate response which will learn the structure in the dataset. In their feature-selection application, this allows them to produce an estimate of the variables that are "most responsible" for the structure in the dataset. However the strategy is quite general and could allow for regression trees to be used on unlabelled data for a variety of purposes. It is related to other data-dependent recursive partitioning schemes, used for example in estimation of densities [Lugosi and Nobel, 1996] or information-theoretic quantities (Appendix A).

Splitting criterion

In constructing a regression tree, a choice of split must be made at each step. The split is chosen which minimises the sum of the "impurity" of the two resulting subsets, typically represented by the mean squared error [Breiman et al.,

1984, Section 8.3]:

$$\text{impurity}(\alpha) = \sum_{i=1}^{n_\alpha} (y_i - \bar{y})^2 \tag{5.2}$$

where n_α is the number of data points in the subset α under consideration, and $\bar{y}$ the mean of the sampled values of the response variable y_i for the points in α.

Questier et al. [2005] use the multivariate-response generalisation

$$\text{impurity}(\alpha) = \sum_{i=1}^{n_\alpha} \sum_{j=1}^{p} (y_{ij} - \bar{y}_j)^2 \tag{5.3}$$

with definitions as in (5.2) except that the y_i (and therefore also $\bar{y}$) are now p-dimensional vector values, with j indexing over the dimensions. In the auto-associative case the y_{ij} are the same as the x_{ij}, the variables by which the splitting planes will be defined.

The impurity measures (5.2) and (5.3) are equivalent to the sum of variances in the subsets, up to a multiplication factor which we can disregard for the purposes of minimisation. By the law of total variance (see e.g. Searle et al. [2006, Appendix S]), minimising the total variance within the subsets is the same as maximising the variance of the centroids; therefore the impurity criterion selects the split which gives the largest difference of the centroids of the response variable in the resulting subsets.

In the feature-selection task of Questier et al. [2005] it is the univariate splits which are counted for feature evaluation, so a multivariate-splits extension would not be appropriate. We are not performing feature-selection but characterising the data distributions; as explored by Gama [2004] it may be advantageous to allow multivariate splits to reduce error. Further, if we are not performing feature-selection then we wish to allow all dimensions to contribute towards our analysis of the data structure, which may not occur in cases of limited data: if there are N data points then there can be no more than around $\log_2 N$ splits used to reach a leaf in a balanced binary tree, which could be fewer than the number of dimensions. We therefore extend the AAMRT approach by allowing multivariate splits.

The hyperplane which splits a dataset into two subsets with the furthest-separated centroids is simply the hyperplane perpendicular to the first principal component in the centred data. This multivariate-splits variant of AAMRT allows for efficient implementation since the leading principal component in a dataset can be calculated quickly e.g. by expectation-maximisation [Roweis, 1998].

5.2.2 Cross-associative MRT

Auto-associative MRT may be useful for discovering structure in an unlabelled dataset [Questier et al., 2005]. Here we wish to adapt it such that it can be used to analyse structural commonalities between two unlabelled datasets, and learn associations between the two. Therefore we now develop a variant that is cross-associative rather than auto-associative; we will refer to it as cross-associative MRT or XAMRT.

Our assumptions will be that the two datasets are i.i.d. samples from two distributions which have broad commonalities in structure and orientation in the measurement space, but that there may be differences in location of regions between the distributions. These may be broad differences such as the location (centroid) or dispersion (variance) along one or many dimensions, or smaller-scale differences such as the movement of a small region of the distribution relative to the rest of the distribution. Some examples of situations where these assumptions are reasonable will be illustrated in the experiments of Section 5.3.

The AAMRT approach is adaptable to the case of two data distributions simply by considering the distributions simultaneously while partitioning – in other words, we determine the splitting plane based on the union of the datasets (or of subsets therefrom). However, we allow the two distributions to have differences in location by perform centring separately on each distribution, before combining them for the purpose of finding a common principal component. Therefore the orientation of the splitting plane is common between the two, but the exact location of the splitting plane can be tailored to the distribution of each separate dataset. We perform this centring at each level of the recursion, which creates an algorithm which allows for differences in location both overall and in smaller subregions of the distributions. This is illustrated schematically in Figure 5.7.

If the datasets contain unequal numbers of data points then the larger set will tend to dominate over the smaller in calculating the principal component. To eliminate this issue we weight the calculation so as to give equal emphasis to each of the datasets, equivalent to finding the principal component of the union J of weighted datasets:

$$J = (N_Y(X - C_X)) \mathbin{++} (N_X(Y - C_Y)) \tag{5.4}$$

where X and Y represent the data (sub)sets, C_X and C_Y their centroids, and N_X and N_Y the number of points they contain.

By recursively partitioning in this way, the two datasets are simultaneously partitioned in a way which reflects both the general commonalities in structure (using splitting hyperplanes with a common orientation) and their differences

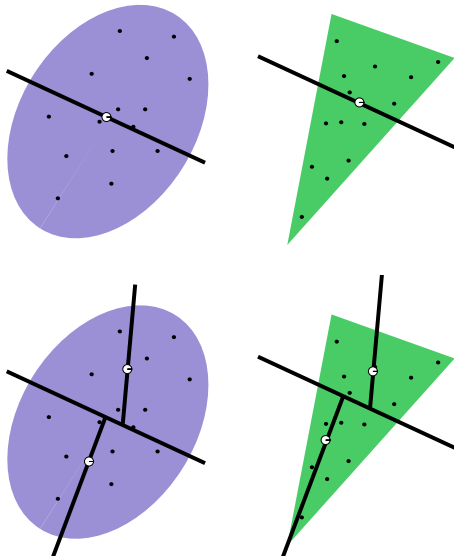


Figure 5.7: Schematic representation of the first two steps in the XAMRT recursion. In the first step (top), the centroids of each dataset are calculated separately, and then a splitting plane with a common orientation is chosen. The second step (bottom) is the same but performed separately on each of the partitions produced in the first step.

in location (the position of the hyperplanes, passing through the centroids of subsets of each dataset) (Figure 5.7). The tree structure defines two different partitions of the space, approximating the densities of the two distributions, and pairing regions of the two distributions.

The tree thus produced is similar to a standard (i.e. neither auto-associative nor cross-associative) multivariate-response regression tree, in that it can predict a multivariate response from multivariate input. However it treats the two distributions symmetrically, allowing projection from either dataset onto the other. Unlike the AAMRT it does not require the input data to be the same as the response data.

Pruning criterion

Allowing a regression tree to proceed to the maximum level of partitioning will tend to overfit the dataset. Criteria may be used to terminate branching, but a generally better strategy (although more computationally intensive) is to grow the full tree and then prune it back by merging together branches [Breiman et al., 1984, Chapter 3]. In the CART framework, the standard measure for pruning both classification and regression trees is *crossvalidation error* within a branch: a normalised average over all data points of the error that results

from estimating the label of each datum from the other data labels [Breiman et al., 1984, Chapters 3 and 8]. Branches which exhibit crossvalidation error above a user-specified threshold are merged into leaf nodes, so as to improve the stability and generality of the tree.

In our case this approach cannot be applied directly because we consider the unsupervised case, i.e. without labels. In Questier et al. [2005] the unlabelled data are used to predict themselves, meaning that the tree algorithm does in fact see (multivariate) labels attached to the data and the crossvalidation measure can be used. We wish to associate two separate distributions whose data points are not paired, and so such a strategy is not available to us.

Instead, we propose to apply the crossvalidation principle to the splitting hyperplanes themselves, producing a measure of the stability of a multivariate split. This would penalise splitting hyperplanes which were only weakly justified by the data, and so produce a pruned tree whose splits were relatively robust to outliers and noise. Our crossvalidation measure is calculated using a leave-one-out ("jackknife") procedure as follows: given a set of N data points whose first principal component p has been calculated to give the proposed splitting plane, we calculate

$$R = \frac{1}{N} \sum_{i=1}^{N} \mathrm{abs}(p \cdot \hat{p_i}) \tag{5.5}$$

where $\hat{p_i}$ is the first principal component calculated after excluding datum i. A measured principal component may be flipped by 180° yet define the same splitting hyperplane (cf. Gaile and Burt [1980]), hence our measure is designed to consider the orientation but not the direction of the principal component vectors – this is achieved by taking the absolute value of each item in the sum. Both p and $\hat{p_i}$ are unit vectors, so R is the average cosine distance between the principal component and its jackknife estimates.

As with the standard CART, we then simply apply a threshold, merging a given branch if its value of R is below some fixed value. Our measure ranges between 0 and 1, where 1 is perfect stability (meaning the principal component is unchanged when any one data point is excluded from the calculation). In this work we use manually-specified thresholds when applying our algorithm, as in CART. Alternatively one could derive thresholds from explicit hypothesis tests by modelling the distribution of the jackknife principal components on the hypersphere [Figueiredo, 2007].

Summary of algorithm

The algorithm is summarised as pseudocode in Figure 5.8. Given two datasets X and Y, both taking values in $\mathcal{X} = \mathbb{R}^D$, the recursive function GROW creates the

regression tree from X and Y, and the recursive function PRUNE prunes the tree given a user-specified stability threshold. We have published our implementation of the algorithm in Python,[3] as well as a real-time tree lookup component for SuperCollider.[4]

To test the efficient operation of the real-time lookup component, we derived a tree from voice recordings and the *gendy1* synth (see Appendix B), having 9 timbre dimensions and around 4000 nodes, and then ran a tree lookup in real time on a laptop (Mac 10.4.11, 1.67 GHz PowerPC G4), driving the synthesiser based on a recorded voice sample. CPU usage (analysed with Apple's profiler, Shark 4.5.0) showed the lookup component to use less than 0.06% of the available CPU power. As expected for a regression tree, the lookup is highly efficient.

5.3 Experiments

We next describe two experiments we conducted to explore the use of the tree regression algorithm (XAMRT) developed in the previous section, in different application domains.

The first directly relates to our goal of timbre remapping, using concatenative synthesis as an established technique in which timbre remapping can be used, and which can be evaluated numerically. This experiment will compare standard nearest neighbour (NN) search with the PCA-based method (Section 5.1.3) as well as with XAMRT, all applied to the same concatenative synthesis task.

The second experiment demonstrates application of XAMRT to a different domain – vowel formant frequencies, using a published dataset from the study of phonetics. This is done to explore the potential of the algorithm for use in other applications, as well as to provide an example of remapping from one distribution to another in the case where ground-truth-labelled data are available to compare against the output of the algorithm.

5.3.1 Concatenative synthesis

Our first experiment applies the regression tree method for our intended purpose of timbre remapping. In order to be able to evaluate the procedure numerically, we choose to apply timbre remapping in the context of concatenative synthesis (or "audio mosaicing"), which can use the timbral trajectory of one sound recording to create new audio from segments of existing recordings [Schwarz, 2005, Jehan, 2004, Sturm, 2006]. These brief segments (on the order of 100 ms duration, henceforth called "grains") are stored in large numbers in a database.

[3] `http://www.elec.qmul.ac.uk/digitalmusic/downloads/xamrt/`
[4] `http://sc3-plugins.sourceforge.net/`

GROW(X, Y)

$C_X \leftarrow$ centroid of X
$C_Y \leftarrow$ centroid of Y
$J \leftarrow$ result of equation (5.4)
$p \leftarrow$ principal component of J
$X_l \leftarrow X \cap ((X - C_X) \cdot p > 0)$
$X_r \leftarrow X \cap ((X - C_X) \cdot p \leq 0)$
$Y_l \leftarrow Y \cap ((Y - C_Y) \cdot p > 0)$
$Y_r \leftarrow Y \cap ((Y - C_Y) \cdot p \leq 0)$
if X_l is singular or Y_l is singular
  then $L = [X_l, Y_l]$
  else $L =$ GROW(X_l, Y_l)
if X_r is singular or Y_r is singular
  then $R = [X_r, Y_r]$
  else $R =$ GROW(X_r, Y_r)
return [L, R]

PRUNE(*tree*, *threshold*)

PRUNE(left child, *threshold*)
PRUNE(right child, *threshold*)
if children of left child are both leaf nodes
  then PRUNEONE(left child, *threshold*)
if children of right child are both leaf nodes
  then PRUNEONE(right child, *threshold*)

PRUNEONE(*tree*, *threshold*)

$R \leftarrow$ result of equation (5.5)
if $R <$ *threshold*
  then merge child nodes into a single node

Figure 5.8: The cross-associative MRT algorithm. X and Y are the two sets of vectors between which associations will be inferred.

Description	Duration (sec)	No. of grains
Amen breakbeat	7	69
Beatboxing	93	882
Fireworks	16	163
Kitchen sounds	49	355
Thunder	8	65

Table 5.1: Audio excerpts used in timbre experiment. "No. of grains" is the number of 100 ms grains segmented and analysed from the audio (excluding silent frames) – see text for details.

It is typically impractical to manually annotate the grains, so our unsupervised technique may be practically useful; at the same time, we can use the indices of the selected grains to design an evaluation statistic based on the pattern of grain use.

Concatenative synthesisers typically operate not only on timbre, but use pitch and duration as well as temporal continuity constraints in their search strategy, and then modify the selected grains further to improve the match [Maestre et al., 2009]. While recognising the importance of these aspects in a full concatenative synthesis system, we designed an experiment in which the role of pitch, duration and temporal continuity were minimised, by excluding such factors from grain construction/analysis/resynthesis, and also by selecting audio excerpts whose variation is primarily timbral.

For this synthesis application, a rich and varied output sound is preferable to a repetitious one, even if the fine variation is partly attributable to measurement noise, and so in the present experiment we do not prune trees derived from timbre data. In a full concatenative synthesiser it may be desirable to use pruned trees which would return a large number of candidate grains associated with a typical leaf, and then to apply other criteria to select among the candidates; we leave this for future work.

We first describe the audio excerpts we used and how timbre was analysed, before describing the concatenative synthesiser and our performance metric.

Audio data

In order to focus on the timbral aspect, we selected a set of audio excerpts in which the interesting variation is primarily timbral and pitch is less relevant. The five excerpts – two musical (percussive) and three non-musical – are listed in Table 5.1 (with spectrograms illustrated in Figure 5.9) and are also available online.[5] The excerpts are 44.1 kHz mono recordings.

The excerpts are quite heterogeneous, not only in sound source but also in

[5] `http://archive.org/details/xamrtconcat2010`

duration (some differ by an order of magnitude). They each contain various amounts/types of audio event, which are not annotated.

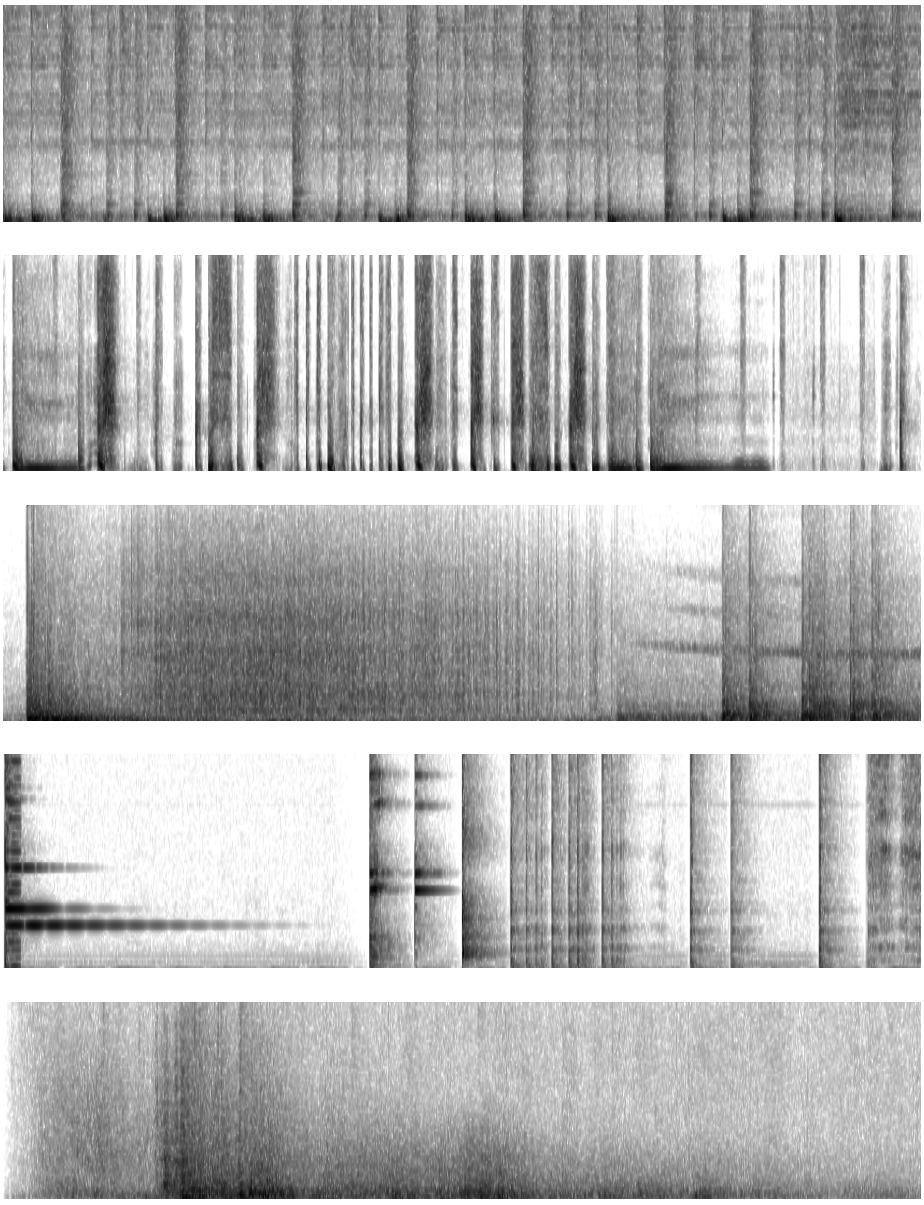


Figure 5.9: Spectrograms of the audio excerpts listed in Table 5.1 (from top to bottom: Amen breakbeat, beatboxing, fireworks, kitchen sounds, thunder). Each shows a duration of 7 seconds and a frequency range of 0–6500 Hz.

Timbre features

We chose a set of 10 strongly-performing features from Chapter 3 to represent signal timbre: *centroid*, *power*, *pow1*–*pow5*, *pcile25*, *pcile95*, and *zcr* (labels as given in Table 3.1 [page 53]). Analysis was performed on audio grains of fixed 100 ms duration taken from the audio excerpt every 100 ms (i.e. with no overlap). Each grain was analysed by segmenting into frames of 1024 samples (at 44.1 kHz sampling rate) with 50% overlap, then measuring the feature values for each frame and recording the mean value of each feature for the grain. Grains with a very low spectral power (< 0.002) were treated as silences and discarded. (Power values were measured on a relative scale where 1 is full time-domain amplitude, meaning the power threshold is around -54 dB.) The timbre features of the remaining grains were normalised to zero mean and unit variance within each excerpt. Analysis was performed in SuperCollider 3.3.1 [McCartney, 2002].

Figure 5.3 plots a PCA projection of the grain timbre data for two of the sound excerpts, illustrating the broad similarities yet differences in detail of the timbre distributions.

Timbral concatenative synthesiser

We designed a simple concatenative synthesiser using purely timbral matching, by one of three methods:

NN, a standard nearest-neighbour search

NN+, the NN search augmented with PCA and warping as developed in Section 5.1.3

XAMRT, the cross-associative regression tree developed in Section 5.2 (without pruning).

Given two excerpts – one which is the source of grains to be played back, and one which is the control excerpt determining the order of playback – and the timbral metadata for the grains in the two excerpts, the synthesis procedure works as follows: For each grain in the control excerpt, if the grain is silent (power < 0.002) then we replace it with silence. Otherwise we replace it with a grain selected from the other excerpt by performing a lookup of the timbre features using the selected method. For numerical evaluation, the choice of grain is recorded. For audio resynthesis, the new set of grains is output with a 50 ms linear crossfade between grains.

The NN search uses the standard Euclidean distance, facilitated using a k-d tree data structure [Bentley, 1975]. Note that the timbre features are normalised

for each excerpt, meaning the NN search is in a normalised space rather than the space of the raw feature values.

In both the NN/NN+ and XAMRT lookup there is an issue of tie-breaking. More than one source grain could be retrieved – at the minimum distance from the query (for NN/NN+) or in the leaf node retrieved from the query (for XAMRT) – yet we must choose only one. This is not highly likely for NN/NN+ search (depending on the numerical precision of the implementation) because the distance measure is continuous-valued, but will occur in XAMRT when mapping from a small to a large dataset, since the tree can grow only to the size allowed by the smaller dataset. Additional criteria (e.g. continuity) could be used to break the tie, but for this experiment we keep the design simple and avoid confounding factors by always choosing the grain from the earliest part of the recording in such a case.

To validate that the system was performing as expected, we performed two types of unit test: firstly we applied the XAMRT algorithm to some manually-defined "toy" datasets of specific shapes and inspected the results; and secondly we confirmed that for all three search strategies, the self-to-self mapping (i.e. using the same audio file as both the grain source and the control excerpt) recovered the sequence of grains in their original temporal order. The outcome of these tests was successful.

Evaluation method

For development and comparison purposes it is particularly helpful to have objective measures of success. It is natural to expect that a good concatenative synthesiser will make wide use of the "alphabet" of available sound grains, so as to generate a rich as possible output from the limited alphabet. Here we develop this notion into an information-theoretic evaluation measure.

Communication through finite discrete alphabets has been well studied in information theory [Arndt, 2001]. A key information-theoretic quantity is the (Shannon) *entropy*. This was applied in earlier chapters but primarily while considering continuous variables; the entropy of a discrete random variable X taking values from an alphabet $\mathcal{A}$ is defined as

$$H(X) = -\sum_{i=1}^{|\mathcal{A}|} p_i \log p_i \tag{5.6}$$

where p_i is the probability that $X = \mathcal{A}_i$ and $|\mathcal{A}|$ is the number of elements in $\mathcal{A}$. The entropy $H(X)$ is a measure of the information content of X, and has the range

$$0 \leq H(X) \leq \log |\mathcal{A}| \tag{5.7}$$

Query type	Efficiency (%)
NN	70.8 ± 4.4
NN+	72.3 ± 4.2
XAMRT	84.5 ± 4.8

Table 5.2: Experimental values for the information-theoretic efficiency of the lookup methods. Means and 95% confidence intervals are given. The improvement of XAMRT over the others is significant at the $p < 0.0001$ level (paired t-test, two-tailed, 19 degrees of freedom, $t > 10.01$). The improvement of NN+ over NN is significant at the $p = 0.0215$ level ($t = 2.506$).

with the maximum achieved iff X is uniformly distributed.

If the alphabet size is known then we can define a normalised version of the entropy called the *efficiency*

$$\text{Efficiency}(X) = \frac{H(X)}{\log |\mathcal{A}|} \tag{5.8}$$

which indicates the information content relative to some optimised alphabet giving a uniform distribution. This can be used for example when X is a quantisation of a continuous variable, indicating the appropriateness of the quantisation scheme to the data distribution.

We can apply such an analysis to our concatenative synthesis, since it fits straightforwardly into this framework: timbral expression is measured using a set of continuous acoustic features, and then "quantised" by selecting one grain from an alphabet to be output. It does not deductively follow that a scheme which produces a higher entropy produces the most pleasing audio results: for example, a purely uniform random selection would have high entropy. However, a scheme which produces a low entropy will tend to be one which has an uneven probability distribution over the grains, and therefore is likely to sound relatively impoverished – for example, some grains will tend to be repeated more often than in a high-entropy scheme. Therefore the efficiency measure is useful in combination with the resynthesised audio results for evaluating the efficacy of a grain selection scheme.

Results

We applied the concatenative synthesis of Section 5.3.1 to each of the 20 pairwise combinations of the 5 audio excerpts (excluding self-to-self combinations, which are always 100% efficient) using each of the three lookup methods (NN, NN+, and XAMRT). We then measured the information-theoretic efficiency (5.8) of each run. Table 5.2 summarises the efficiencies for each lookup method. NN+ yields a small improvement over the basic NN method. The XAMRT method

is seen to produce a dramatic improvement over both of the other search types, improving average efficiency by over 12 percentage points.

This difference in performance suggests that the inability of NN+ to accommodate dependencies between dimensions may indeed be limiting its ability to create a well-covered timbre space (as discussed in Section 5.1.3) and thus to encourage a uniform use of grains. More detailed investigation would be needed to confirm that as the cause.

Audio examples of the output from the system are available online.[6] Note that the reconstructed audio examples sound rather unnatural because the experiment is not conducted in a full concatenative synthesis framework. In particular we use a uniform grain duration of 100 ms and impose no temporal constraints, whereas a full concatenative synthesis system typically segments sounds using detected onsets and includes temporal constraints for continuity, and therefore is able to synthesise much more natural attack/sustain dynamics [Maestre et al., 2009].

The XAMRT technique therefore shows promise as the timbral component of a multi-attribute search which could potentially be used in concatenative synthesis, as well as more generally in timbral remapping and in other applications requiring timbral search from audio examples (e.g. query-by-example [Foote, 1999, Section 4.2]).

Note that this experiment shows that the XAMRT algorithm improves the mapping in the sense of better matching the distributions, but does not directly tell us that it produces better audio results. Although audio examples are available for judging this informally, in future it would be worthwhile to design a perceptual experiment in which listeners rated the audio produced – compare for example the perceptual experiment of the previous chapter (Section 4.2). However, it is harder to evaluate perceived quality in this case, because we would not be measuring the perception of degradation but of the musicality/pleasantness/appropriateness of the output. Although there is inter-rater variation in assessing the quality of degraded audio, the variation is relatively small and the nature of what is being assessed is typically well understood and shared among raters. Any quantitative perceptual experiment testing success of the musical "analogies" created by timbre remapping would need to be designed with careful attention to what is being measured, and the potential effect of listeners' musical and cultural background on their ratings.

Having demonstrated that the XAMRT technique works well as intended for the application to timbre remapping, our second experiment turns to an application domain outside of the main focus of this thesis, showing the potential for using XAMRT for other tasks.

[6] `http://www.archive.org/details/xamrtconcat2010`

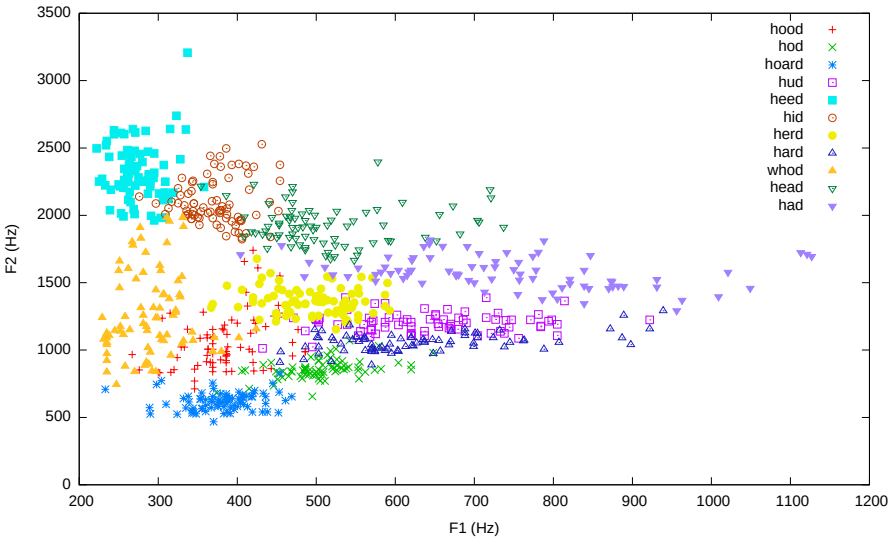


Figure 5.10: Frequencies of the first two vocal formants, measured by Hawkins and Midgley [2005] for specific hVd words as given in the legend.

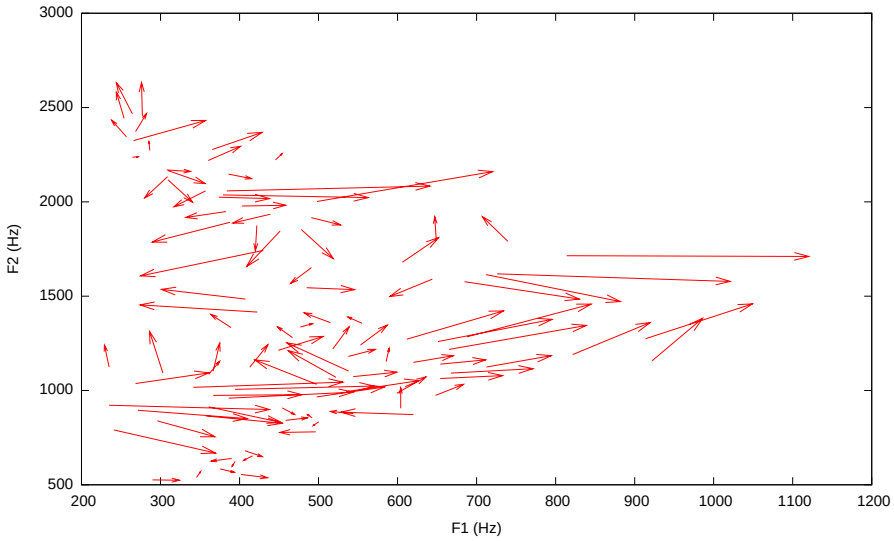
5.3.2 Vowel formant analysis

In our second experiment on the performance of the XAMRT algorithm we analyse data representing the change in vowel pronounciation between different generations of speakers of British English. In Hawkins and Midgley [2005] the first two formant frequencies F1 and F2 (the two main resonances of the vocal tract) are measured for different age groups of speakers of Received Pronounciation (RP) British English, and comparisons are then drawn between generations. These data are labelled: each measurement is made on a single-syllable word of the form hVd, where the V stands for a monophthong vowel. The labelled data is displayed (aggregated over all age groups) in Figure 5.10.

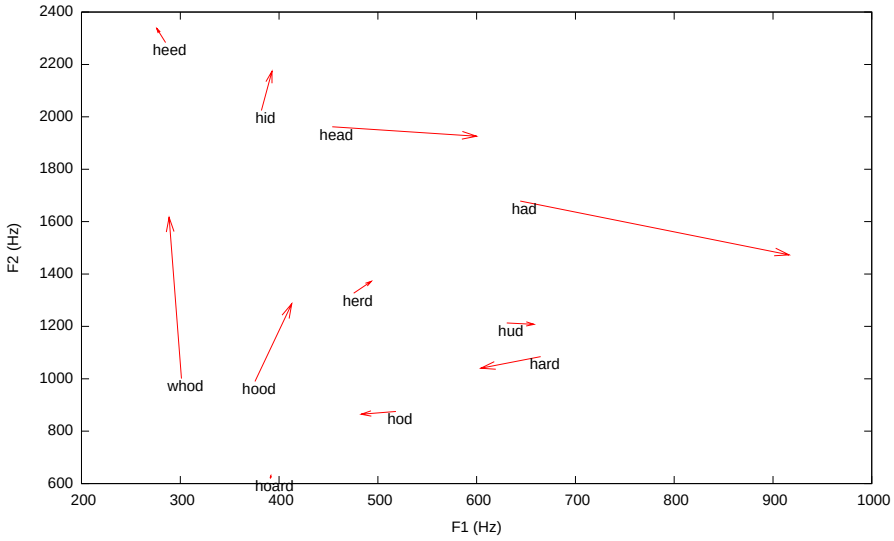
Such data allows us apply our unsupervised analysis to the formant frequencies (ignoring the labels), pairing the data distribution for one generation of speakers with that of another, and compare this analysis with the expert observations about intergenerational change made by the authors of the original study.

We took the formant data for the oldest and youngest group of speakers (group 1 and 4 respectively), and applied our tree-based partitioning algorithm. We then calculated the two-dimensional centroid locations for each cluster, and visualised the movement from a centroid in the older generation, to the corresponding centroid in the younger generation (Figure 5.11a).

The results indicate quite a lot of movement between the two data distributions. Notable are three regions with long right-pointing arrows, which suggest



(a) Movement of the centroids of clusters determined automatically by our algorithm.



(b) Movement of the centroids of word-labelled groups.

Figure 5.11: Movement of formant positions determined either automatically or using the word labels. Each arrow connects paired regions of density, going from Hawkins and Midgley [2005]'s group 1 (age 65+ years) to their group 4 (age 20–25 years). Axes represent the frequencies of the first two vocal formants.

that the F1 frequency in these regions may have raised in the younger generation while the F2 stayed roughly constant. The upper two of those three regions represent the vowels /ɛ/ /æ/ (*heed*, *had*) and directly match the authors' first observations about F1 (although less so for F2): "The mean frequencies of /ɛ/ and /æ/ are successively slightly lower in F2 and markedly higher in F1 in each

age group from oldest to youngest, consistent with the percept that they are becoming progressively more open with younger speakers" (p. 188).

The authors continue: "In contrast, the mean frequency of /uː/ has a higher F2 in successive age groups, with F1 unchanged or little changed". This too can be seen in our analysis: the /uː/ (*who'd*) vowels are to be found at the left and near the bottom of the plot, in a region whose arrows point upwards indicating the raised F2. However, just above are some arrows pointing leftwards (suggesting a lowered F1) which can also be considered to belong to the domain of the /uː/ (*who'd*) vowels, a shift which does not exist in the authors' description.

The authors next observe for the vowel /ʊ/ (*hood*) that "the youngest group has a rather higher F1 ... and a markedly higher F2". This vowel is at the lower left in our figure, and although our analysis shows the raised F1 it does not capture the raised F2.

The authors go on to note that the vowels /iː/ /ɪ/ /ɑː/ /ɒ/ /ɔː/ /ʌ/ /ɜː/ remain largely unchanged across the generations. These vowels occupy the upper-left through to the centre of our figure, regions showing changes which are generally small in magnitude and inconsistent in direction. These small changes may represent noise in the data, artifacts due to our algorithm or real changes in pronounciation which were too small to be remarked by the authors.

We can visually compare our results with a plot of formant movements grouped according to the vowel labels, showing the change in the mean formant location for each vowel (Figure 5.11b). At the upper right of Figure 5.11b are two large increases in F1 which are strongly similar to shifts identified by our unsupervised analysis. Some other arrows show similar orientations and directions; however the plot makes clear that our algorithm has not identified the notable rise in F2 displayed by the two vowels /uː/ /ʊ/ (*who'd* and *hood*, at the lower left of the plot), perhaps because those vowels appear to have moved into a region at the same time as other vowels move out of it.

To summarise, our technique has highlighted some of the phonetically important changes observed by Hawkins and Midgley [2005], despite being unsupervised and hence ignorant of the phoneme labels. This demonstrates the potential of this technique to highlight changes between two data distributions which may be of interest for further study. The data we have used happens to be labelled with corresponding words from a controlled vocabulary; however, large corpuses of data may be unlabelled, and so the procedure could be applied for preliminary analysis in such cases. One difference between the Hawkins and Midgley [2005] data and a large-corpus analysis is that the latter would not use a controlled distribution of words, and so the analysis would reflect changes in formants balanced over the distribution of phoneme use in the corpus rather than over the controlled vocabulary.

5.4 Conclusions

In this chapter we have charted the development of our approach to timbre remapping as a technique to create useful real-time mapping from voice timbre to synth timbre. Our first method was based on a simple nearest-neighbour (NN) search, with modifications to create a more even coverage of the search space. It produced adequate results and has been used in live performances and demonstrations, but is somewhat ad-hoc, and was seen in an experiment to yield only a small improvement over the basic NN search. Our second method (discussed more fully in Appendix D) aimed to bring a more coherent framework to bear on the process using Self-Organised Maps (SOM); however this approach was severely hampered by difficulties in controlling the alignment of maps, difficulties which are inherent in the standard SOM algorithms. Our third approach was to develop a novel regression-tree based method (XAMRT). This was designed specifically to pair two different yet related timbre distributions; in numerical experiments with a simple concatenative synthesiser, we demonstrated that it makes significantly better use of the source material than both the basic and augmented NN search.

Throughout the chapter we have been concerned to develop techniques that can be applied in real time, so as to be usable in a live expressive vocal performance. The XAMRT method fulfils this since regression trees are very computationally efficient. However, we also wished to leave open the possibility of online learning rather than having to train the system in advance. This is one attraction of the SOM, which was indeed first developed as an online learning algorithm [Kohonen, 2001]. At present we only have a batch method for training the XAMRT; as future work it would be useful to develop techniques for online adaptation of the regression tree, e.g. to allow it to adapt to the vocal range of a particular performer.

Our concatenative synthesis experiment (Section 5.3.1) demonstrated the technique used in a simplified synth using only timbral criteria. In order to use timbre remapping in a full concatenative synthesiser, or in some similar system, future work would need to consider how to combine timbre remapping with other criteria, such as the pitch, duration and continuity criteria used in more sophisticated concatenative synths [Maestre et al., 2009].

Chapter 6

User evaluation

In Chapters 4 and 5 we developed two different approaches to real-time, low-latency control of synthesisers through vocal timbre – one based on event-based control, and one based on continuous mapping. Throughout this development we have measured statistics to demonstrate the efficacy of our approach or to determine the effect of parameters (classification accuracy [Section 4.1], listening test data [Section 4.2], information-theoretic mapping efficiency [Section 5.3.1]). Yet the aim of this thesis (Section 1.2) is to develop such methods of vocal timbre control "suitable for live expressive performance". We therefore consider it important to evaluate such methods in a way which sheds light on the usefulness of such methods for the live performance, from the performer's perspective and with a bearing upon the interaction between the system and the performer's creativity/expressiveness.

In our discussion thus far, issues of creativity or expressiveness have only indirectly been considered. In part this is because statistics derived from algorithm output (such as classification accuracy) do not tell us much about these issues – but also because live technologically-mediated expression is a dynamic situation involving continuous feedback between the system and the performer, which creates difficulties in designing experiments to probe the situation. Yet this interaction between performer and system is a critical aspect of the technology, which we take to be an important factor in determining whether (and how) a particular technology is taken up.

In this chapter we first consider issues in evaluating expressive/creative musical systems and describe previous research in the area, before developing a performer-centred qualitative approach to evaluation. We then describe an evaluation study performed with human beatboxers, on an early version of the timbre remapping system of Chapter 5, illuminating some aspects of the vocal interaction with this technology. As we will discuss, our development fits into a

Evaluation type	NIME conference year		
	2006	2007	2008
Not applicable	*8*	*9*	*7*
None	18	14	15
Informal	12	8	6
Formal qualit.	1	2	3
Formal quant.	2	3	3
Total formal	3 *(9%)*	5 *(19%)*	6 *(22%)*

Table 6.1: Survey of oral papers presented at the conference on New Interfaces for Musical Expression (NIME), indicating the type of evaluation described. The last line indicates the total number of formal evaluations presented, also given as a percentage of the papers (excluding those for which evaluation was not applicable).

current research context in Human-Computer Interaction (HCI) which aims to move beyond task-focused evaluation to include affective and context-sensitive evaluation techniques, sometimes referred to as the "third paradigm" in HCI research [Harrison et al., 2007].

6.1 Evaluating expressive musical systems

Live human-computer music-making, with reactive or interactive systems, is a topic of recent artistic and engineering research [Collins and d'Escriván, 2007, esp. Chapters 3, 5, 8]. However, the formal evaluation of such systems is relatively little-studied [Fels, 2004]. A formal evaluation is one presented in rigorous fashion, which presents a structured route from data collection to results (e.g. by specifying analysis techniques). It therefore establishes the degree of generality and repeatability of its results. Formal evaluations, whether quantitative or qualitative, are important because they provide a basis for generalising the outcomes of user tests, and therefore allow researchers to build on one another's work. As one indicator of the state of the field, we carried out a survey of recent research papers presented at the conference on New Interfaces for Musical Expression (NIME – a conference about user interfaces for music-making). It shows a consistently low proportion of papers containing formal evaluations (Table 6.1).

Live human-computer music making poses challenges for many common HCI evaluation techniques. Musical interactions have creative and affective aspects, which means they cannot be described as tasks for which e.g. completion rates can reliably be measured. They also have dependencies on timing (rhythm, tempo, etc.), and feedback interactions (e.g. between performers, between performer and audience), which further complicate the issue of developing valid

and reliable experimental procedures.

Evaluation could be centred on a user (performer) perspective, or alternatively could be composer-centred or audience-centred (e.g. using expert judges). In live musical interaction the performer has privileged access to both the intention and the act, and their experience of the interaction is a key part of what determines its expressivity. Hence in the following we focus primarily on performer-centred evaluation, as have others (e.g. Wanderley and Orio [2002]).

"Talk-aloud" protocols [Ericsson and Simon, 1984, Section 2.3] are used in many HCI evaluations. However, in some musical performances (such as singing or playing a wind instrument) the use of the speech apparatus for music-making precludes concurrent talking. More generally, speaking may interfere with the process of rhythmic/melodic performance: speech and music cognition can demonstrably interfere with each other [Salamé and Baddeley, 1989], and the brain resources used in speech and music processing partially overlap [Peretz and Zatorre, 2005], suggesting issues of cognitive "competition" if subjects are asked to produce music and speech simultaneously.

Other observational approaches may be applicable, although in many cases observing a participant's reactions may be difficult: because of the lack of objectively observable indications of "success" in musical expression, but also because of the participant's physical involvement in the music-making process (e.g. the whole-body interaction of a drummer with a drum-kit).

Some HCI evaluation methods use models of human cognition rather than actual users in tests – e.g. GOMS [Card et al., 1983] – while others such as cognitive walkthrough [Wharton et al., 1994] use structured evaluation techniques and guidelines. These are good for task-based situations, where cognitive processes are relatively well-characterised. However we do not have adequate models of the cognition involved in live music-making in order to apply such methods. Further, such methods commonly segment the interaction into discrete ordered steps, a process which cannot easily be carried out on the musical interactive experience.

Another challenging aspect of musical interface evaluation is that the participant populations are often small [Wanderley and Orio, 2002]. For example, it may be difficult to recruit many virtuoso violinists, human beatboxers, or jazz trumpeters, for a given experiment. Therefore evaluation methods should be applicable to relatively small study sizes.

6.1.1 Previous work in musical system evaluation

There is a relative paucity of literature in evaluating live sonic interactions, perhaps in part due to the difficulties mentioned above. Some prior work has looked

at HCI issues in "offline" musical systems, i.e. tools for composers (e.g. Buxton and Sniderman [1980], Polfreman [1999]). Borchers [2001] applies a pattern-language approach to the design of interactive musical exhibits. Others have used theoretical considerations to produce recommendations and heuristics for designing musical performance interfaces [Hunt and Wanderley, 2002, Levitin et al., 2002, Fels, 2004, De Poli, 2004], although without explicit empirical validation. Note that in some such considerations, a "Composer→Performer→Audience" model is adopted, in which musical expression is defined to consist of timing and other variations applied to the composed musical score [Widmer and Goebl, 2004, De Poli, 2004]. In this work we wish to consider musical interaction more generally, encompassing improvised and interactive performance situations.

Wanderley and Orio [2002] provide a particularly useful contribution to our topic. They discuss pertinent HCI methods, before proposing a task-based approach to musical interface evaluation using "maximally simple" musical tasks such as the production of glissandi or triggered sequences. The authors propose a user-focused evaluation, using Likert-scale feedback (i.e. allowing users to report their experience in a simple rank-scale format [Grant et al., 1999]) as opposed to an objective measure of gesture accuracy (e.g. relative pitch error on a task involving production of pitches), since such objective measures may not be a good representation of the musical qualities of the gestures produced. The authors draw an analogy with Fitts' law, the well-known law in HCI which predicts the time required to move to a target (e.g. by moving a mouse cursor) based on distance and target size [Card et al., 1978]; they suggest that numbers derived from their task-based approach may allow for quantitative comparisons of musical interfaces.

Wanderley and Orio's framework is interesting but may have some drawbacks. The reduction of musical interaction to maximally simple tasks risks compromising the authenticity of the interaction, creating situations in which the affective and creative aspects of music-making are abstracted away. In other words, the reduction conflates *controllability* of a musical interface with *expressiveness* of that interface [Dobrian and Koppelman, 2006]. The use of Likert-scale metrics also may have some difficulties. They are susceptible to cultural differences [Lee et al., 2002] and psychological biases [Nicholls et al., 2006], and may require large sample sizes to achieve sufficient statistical power [Göb et al., 2007].

Acknowledging the relative scarcity of research on the topic of live human-computer music-making, we may look to other areas which may provide useful analogies. The field of computer games is notable here, since it carries some of the features of live music-making: it can involve complex multimodal interactions, with elements of goal-oriented and affective involvement, and a degree of

learning. For example, Barendregt et al. [2006] investigates the usability and affective aspects of a computer game for children, during first use and after some practice. Mandryk and Atkins [2007] use a combination of physiological measures to produce a continuous estimate of the emotional state (arousal and valence) of subjects playing a computer game.

In summary, although there have been some useful forays into the field of expressive musical interface evaluation, and some work in related disciplines such as that of computer games evaluation, the field could certainly benefit from further development. Whilst task-based methods are suited to examining usability, the *experience* of interaction is essentially subjective and requires alternative approaches for evaluation. Therefore in the next section we develop a method based on a rigorous qualitative method which analyses language in context, before applying this method to a vocal timbre remapping interface.

6.2 Applying discourse analysis

When a sonic interactive system is created, it is not "born" until it comes into use. Its users construct it socially using analogies and contrasts with other interactions in their experience, a process which creates the affordances and contexts of the system. This primacy of social construction has been recognised for decades in strands of the social sciences and psychology (e.g. Pinch and Bijker [1984], Norman [2002]), but is often overlooked by technologists. It is reflected to some extent in the use of the term "affordances" in HCI research: it originally referred to the possibilities for action offered by a system, but found wide application in HCI after an emphasis on *perceived* possibilities developed, meaning affordances are dependent not only on the system itself and the user's capabilities, but also on their goals, beliefs and past experiences [Norman, 2002].

Discourse Analysis (DA) is an analytic tradition that provides a structured way to analyse the construction and reification of social structures in discourse [Banister et al., 1994, Chapter 6][Silverman, 2006, Chapter 6]. The source data for DA is written text, which may be appropriately-transcribed interviews or conversations.

Interviews and free-text comments are sometimes reported in studies on musical interfaces. However, often they are conducted in a relatively informal context, and only quotes or summaries are reported rather than any structured analysis, therefore providing little analytic reliability. DA's strength comes from using a *structured method* which can take apart the language used in discourses (e.g. interviews, written works) and elucidate the connections and implications contained within, while remaining faithful to the content of the original text [Antaki et al., 2003]. DA is designed to go beyond the specific sequence of

phrases used in a conversation, and produce a structured analysis of the conversational resources used, the relations between entities, and the "work" that the discourse is doing.

DA is not a single method but an analytic tradition developed with a social constructionist basis. Discourse-analytic approaches have been developed which aim to elucidate social power relations, or the details of language use. Our interest lies in understanding the conceptual resources brought to bear in constructing socially a new interactive artefact. Therefore we derive our approach from a Foucauldian tradition of DA found in psychology [Banister et al., 1994, Chapter 6], which probes the reification of existing social structures through discourse, and the congruences and tensions within.

We wish to use the power of DA as part of a qualitative and formal method which can explore issues such as expressivity and affordances for users of interactive musical systems. Longitudinal studies (e.g. those in which participants are monitored over a period of weeks or months) may also be useful, but imply a high cost in time and resources. Therefore we aim to provide users with a brief but useful period of exploration of a new musical interface, including interviews and discussion which we can then analyse.

We are interested in issues such as the user's conceptualisation of musical interfaces. It is interesting to look at how these are situated in the described world, and particularly important to avoid preconceptions about how users may describe an interface: for example, a given interface could be: an instrument; an extension of a computer; two or more separate items (e.g. a box and a screen); an extension of the individual self; or it could be absent from the discourse.

In any evaluation of a musical interface one must decide the context of the evaluation. Is the interface being evaluated as a successor or alternative to some other interface (e.g. an electric cello vs an acoustic cello)? Who is expected to use the interface (e.g. virtuosi, amateurs, children)? Such factors will affect not only the recruitment of participants but also some aspects of the experimental setup.

6.2.1 Method

As discussed, we based our method on that of Banister et al. [1994, Chapter 6], but wished to stimulate participants to talk in a relatively unconstrained manner during and after using a musical interface, so as to elicit talk in reaction to the interface (the raw data for DA). We therefore designed study sessions in which participants would be encouraged to use and explore the system in question, while recording their speech and actions and aiming to stimulate discussion.

Our method is designed either to trial a single interface with no explicit

comparison system, or to compare two similar systems (as is done in our case study of timbre remapping). The method consists of two types of user session, solo sessions followed by group session(s), plus the Discourse Analysis of data collected.

We emphasise that DA is a broad tradition, and there are many designs which could bring DA to bear on evaluating sonic interactions. The method described in the following is just one approach.

Solo sessions

In order to explore individuals' personal responses to the interface(s), we first conduct solo sessions in which a participant is invited to try out the interface(s) for the first time. If there is more than one interface to be used, the order of presentation is randomised in each session.

The solo session consists of three phases for each interface:

Free exploration. The participant is encouraged to try out the interface for a while and explore it in their own way.

Guided exploration. The participant is presented with audio examples of recordings created using the interface, in order to indicate the range of possibilities, and encouraged to create recordings inspired by those examples. This is not a precision-of-reproduction task; precision-of-reproduction is explicitly not evaluated, and participants are told that they need not replicate the examples.

Semi-structured interview [Preece et al., 2004, Chapter 13]. The interview's main aim is to encourage the participant to discuss their experiences of using the interface in the free and guided exploration phases, both in relation to prior experience and to the other interfaces presented if applicable. Both the free and guided phases are video recorded, and the interviewer may play back segments of the recording and ask the participant about them, in order to stimulate discussion.

The raw data to be analysed is the interview transcript. Our aim is to allow the participant to construct their own descriptions and categories, which means the interviewer must be critically aware of their own use of language and interview style, and must (as far as possible) respond to the terms and concepts introduced by the participant rather than dominating the discourse.

Group session

To complement the solo sessions we also conduct a group session. Peer group discussion can produce more and different discussion around a topic, and can

demonstrate the group negotiation of categories, labels, comparisons, and so on. The focus-group tradition provides a well-studied approach to such group discussion [Stewart et al., 2007]. Our group session has a lot in common with a typical focus group in terms of the facilitation and semi-structured group discussion format. In addition we make available the interface(s) under consideration and encourage the participants to experiment with them during the session.

As in the solo sessions, the transcribed conversation is the data to be analysed. An awareness of facilitation technique is also important here, to encourage all participants to speak, to allow opposing points of view to emerge in a non-threatening environment, and to allow the group to negotiate the use of language with minimal interference.

Data analysis

Our DA approach to analysing the data is based on that of Banister et al. [1994, Chapter 6], adapted to the experimental context. The DA of text is a relatively intensive and time-consuming method. It can be automated to some extent, but not completely, because of the close linguistic attention required. Our approach consists of the following five steps:

(a) Transcription. The speech data is transcribed, using a standard style of notation which includes all speech events (including repetitions, speech fragments, pauses). This is to ensure that the analysis can remain close to what is actually said, and avoid adding a gloss which can add some distortion to the data. For purposes of analytical transparency, the transcripts (suitably anonymised) should be published alongside the analysis results.

(b) Free association. Having transcribed the speech data, the analyst reads it through and notes down surface impressions and free associations. These can later be compared against the output from the later stages.

(c) Itemisation of transcribed data. The transcript is then broken down by itemising every single object in the discourse (i.e. all the entities referred to). Pronouns such as "it" or "he" are resolved, using the participant's own terminology as far as possible. For every object an accompanying description of the object is extracted from that speech instance – again using the participant's own language, essentially by rewriting the sentence/phrase in which the instance is found.

The list of objects is scanned to determine if different ways of speaking can be identified at this point. For example, there may appear to be a technical music-production way of speaking, as well as a more intuitive music-performer way of speaking, both occurring in different parts of the

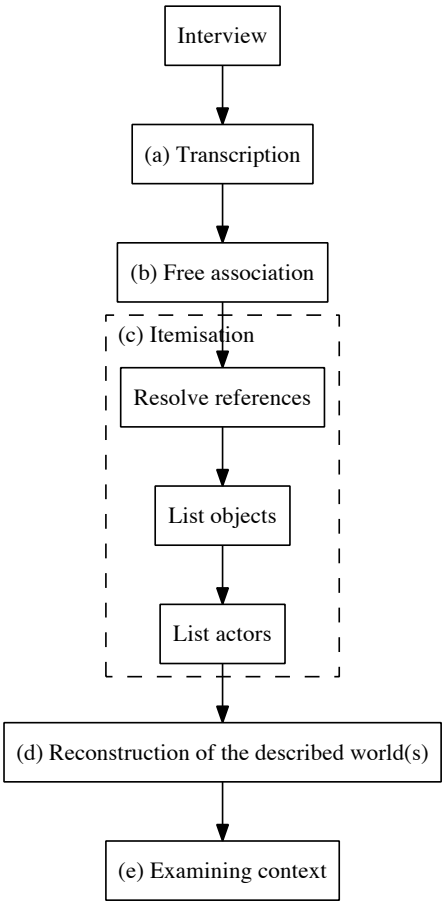


Figure 6.1: Outline of our Discourse Analysis procedure.

discourse; they may have overlaps or tensions with each other. Also, those objects which are also "actors" are identified – i.e. those which act with agency/sentience in the speech instance; they need not be human.

It is helpful at this point to identify the most commonly-occurring objects and actors in the discourse, as they will form the basis of the later reconstruction.

Figure 6.2 shows an excerpt from a spreadsheet used during our DA process, showing the itemisation of objects and subjects, and the descriptions extracted.

(d) Reconstruction of the described world. Starting with the list of most commonly-occurring objects and actors in the discourse, the analyst reconstructs the depictions of the world that they produce, in terms of the interrelations between the actors and objects. This could for example be

Transcription	Object (referent)	Description	Is actor?
I was trying to work out what the other person was,	Participant	was trying to work out what the other person was ((doing))	Y
yeah I'm curious to see how the other person did it,	((person recording the examples)) the other person	Ptcpt was trying to work out what this person was ((doing))	Y
Because it was more fun	((Y))	Ptcpt preferred this ((to X)) because it was more fun	
I just think the noises were a bit more,	the noises ((made by Y))	were a bit more, bit different	
you could come up with some slightly more funky noises.	((general person))	could come up with some slightly more funky noises ((in Y, than X))	Y
	noises	((general person)) could come up with some slightly more funky ones ((in Y, than X))	

Figure 6.2: Excerpt from a spreadsheet used during the itemisation of interview data, for step (c) of the Discourse Analysis.

represented using concept maps. If different ways of speaking have been identified, there will typically be one reconstructed "world" per way of speaking. Overlaps and contrasts between these worlds can be identified. Figure 6.3 shows an excerpt of a concept map representing a "world" distilled in this way.

The "worlds" we produce are very strongly tied to the participant's own discourse. The actors, objects, descriptions, relationships, and relative importances, are all derived from a close reading of the text. These worlds are essentially just a methodically reorganised version of the participant's own language.

(e) Examining context. One of the functions of discourse is to create the context(s) in which it operates, and as part of the DA process we try to identify such contexts, in part by moving beyond the specific discourse act. For example, the analyst may feel that one aspect of a participant's discourse ties in with a common cultural paradigm of an dabbling amateur, or with the notion of natural virtuosity.

In our design we have parallel discourses originating with each of the participants, which gives us an opportunity to draw comparisons. After running the previous steps of DA on each individual transcript, we compare and contrast the described worlds produced from each transcript, examining commonalities and differences. We also compare the DA of the focus group session(s) against that of the solo sessions.

Our approach is summarised in Figure 6.1. In the next section we apply this method to evaluate an instance of our timbre remapping system.

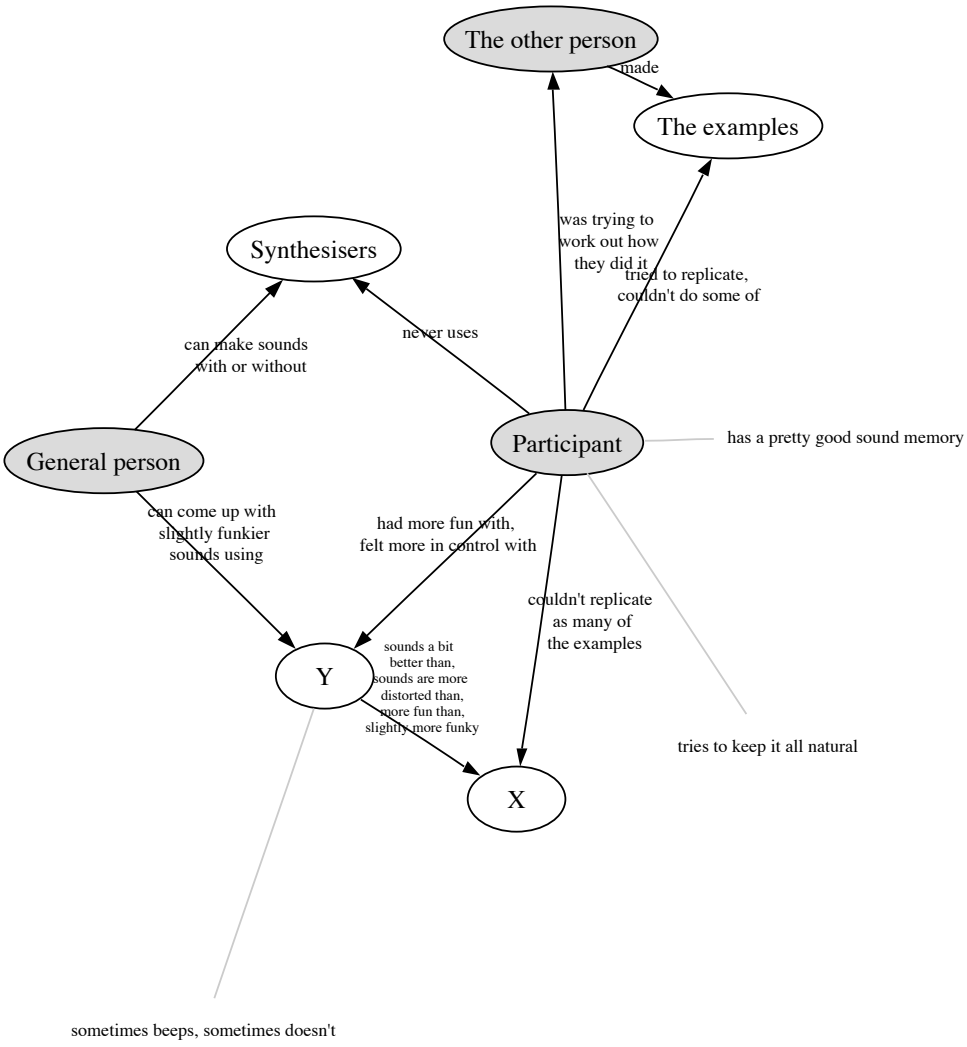


Figure 6.3: An example of a reconstructed set of relations between objects in the described world. This is a simplified excerpt of the reconstruction for User 2 in our study. Objects are displayed in ovals, with the shaded ovals representing actors.

6.3 Evaluation of timbre remapping

We performed an evaluation of the timbre remapping approach described in Chapter 5. The system used was a relatively early version, using the PCA-based remapping technique (Section 5.1.3) rather than the regression tree method advocated in the later part of that chapter. Our primary aim was to evaluate timbre remapping as a general approach to vocal musical control, rather than a particular variant of the technique.

In our study we wished to evaluate the timbre remapping system with beatboxers (vocal percussion musicians), for two reasons: they are one target audience for the technology in development; and they have a familiarity and level of comfort with manipulation of vocal timbre that should facilitate the study sessions. They are thus not representative of the general population but of a kind of expert user.

After piloting the evaluation method successfully with a colleague, we recruited by advertising online (a beatboxing website) and with posters around London for amateur or professional beatboxers. Participants were paid £10 per session plus travel expenses to attend sessions in our (acoustically-isolated) university studio ("Listening Room"). We recruited five participants from the small community, all male and aged 18–21. One took part in a solo session; one in the group session; and three took part in both. Their beatboxing experience ranged from a few months to four years. Their use of technology for music ranged from minimal to a keen use of recording and effects technology (e.g. Cubase). The facilitator was known to the participants by his membership of the beatboxing website.

In our study we wished to investigate any effect of providing the timbre remapping feature. To this end we presented two similar interfaces: both tracked the pitch and volume of the microphone input, and used these to control a synthesiser, but one also used the timbre remapping procedure to control the synthesiser's timbral settings. The synthesiser used was an emulated General Instrument AY-3-8910 [General Instrument, 1979], which was selected because of its wide timbral range (from pure tone to pure noise) with a well-defined control space of a few integer-valued variables. The emulation was implemented in a very similar way to the *ay1* synth given in Appendix B. Participants spent a total of around 30–60 minutes using the interfaces, and 15–20 minutes in interview. Analysis of the interview transcripts using the procedure of section 6.2.1 took approximately 9 hours per participant (around 2000 words each).

We do not report a detailed analysis of the group session transcript here: the group session generated information which is useful in the development of our system, but little which bears directly upon the presence or absence of timbral control. We discuss this outcome further in Section 6.4.

In the following, we describe the main findings from analysis of the solo sessions, taking each user one by one before drawing comparisons and contrasts. We emphasise that although the discussion here is a narrative supported by quotes, it reflects the structures elucidated by the DA process – the full transcripts and Discourse Analysis tables are available online[1] and excerpts from the analysis are given in Appendix E. In the study, condition "X" was used to

[1] `http://www.elec.qmul.ac.uk/digitalmusic/papers/2008/Stowell08ijhcs-data/`

refer to the system with timbre remapping inactive, "Y" for the system with timbre remapping active.

6.3.1 Reconstruction of the described world

User 1 expressed positive sentiments about both X (without timbre remapping) and Y (with timbre remapping), but preferred Y in terms of sound quality, ease of use and being "more controllable". In both cases the system was construed as a reactive system, making noises in response to noises made into the microphone; there was no conceptual difference between X and Y – for example in terms of affordances or relation to other objects.

The "guided exploration" tasks were treated as reproduction tasks, despite our intention to avoid this. User 1 described the task as difficult for X, and easier for Y, and situated this as being due to a difference in "randomness" (of X) vs. "controllable" (of Y).

User 2 found the the system (in both modes) "didn't sound very pleasing to the ear". His discussion conveyed a pervasive structured approach to the guided exploration tasks, in trying to infer what "the original person" had done to create the examples and to reproduce that. In both Y and X the approach and experience was the same.

Again, User 2 expressed preference for Y over X, both in terms of sound quality and in terms of control. Y was described as more fun and "slightly more funky". Interestingly, the issues that might bear upon such preferences are arranged differently: issues of unpredictability were raised for Y (but not X), and the guided exploration task for Y was felt to be more difficult, in part because it was harder to infer what "the original person" had done to create the examples.

User 3's discourse placed the system in a different context compared to others. It was construed as an "effect plugin" rather than a reactive system, which implies different affordances: for example, as with audio effects it could be applied to a recorded sound, not just used in real time; and the description of what produced the audio examples is cast in terms of an original sound recording rather than some other person. This user had the most computer music experience of the group, using recording software and effects plugins more than the others, which may explain this difference in contextualisation.

User 3 found no difference in sound or sound quality between X and Y, but found the guided exploration of X more difficult, which he attributed to the input sounds being more varied.

User 4 situated the interface as a reactive system, similar to Users 1 and 2. However, the sounds produced seemed to be segregated into two streams rather

than a single sound – a "synth machine" which follows the user's humming, plus "voice-activated sound effects". No other users used such separation in their discourse.

"Randomness" was an issue for User 4 as it was for some others. Both X and Y exhibited randomness, although X was much more random. This randomness meant that User 4 found Y easier to control. The pitch-following sound was felt to be accurate in both cases; the other (sound effects / percussive) stream was the source of the randomness.

In terms of the output sound, User 4 suggested some small differences but found it difficult to pin down any particular difference, but felt that Y sounded better.

6.3.2 Examining context

Users 1 and 2 were presented with the conditions in the order XY; Users 3 and 4 in the order YX. Order-of-presentation may have some small influence on the outcomes: Users 3 and 4 identified little or no difference in the output sound between the conditions (User 4 preferred Y but found the difference relatively subtle), while Users 1 and 2 felt more strongly that they were different and preferred the sound of Y. It would require a larger study to be confident that this difference really was being affected by order-of-presentation.

In our study we are not directly concerned with which condition sounds better (both use the same synthesiser in the same basic configuration), but this is an interesting aspect to come from the study. We might speculate that differences in perceived sound quality are caused by the different way the timbral changes of the synthesiser are used. However, participants made no conscious connection between sound quality and issues such as controllability or randomness.

Taking the four participant interviews together, no strong systematic differences between X and Y are seen. All participants situate Y and X similarly, albeit with some nuanced differences between the two. Activating/deactivating the timbre remapping facet of the system does not make a strong enough difference to force a reinterpretation of the system.

A notable aspect of the four participants' analyses is the differing ways the system is situated (both X and Y). As designers of the system we may have one view of what the system "is", perhaps strongly connected with technical aspects of its implementation, but the analyses presented here illustrate the interesting way that users situate a new technology alongside existing technologies and processes. The four participants situated the interface in differing ways: either as an audio effects plugin, or a reactive system; as a single output stream or as two. We emphasise that none of these is the "correct" way to conceptualise the

interface. These different approaches highlight different facets of the interface and its affordances.

The discourses of the "effects plugin" and the "reactive system" exhibit some tension. The "reactive system" discourse allows the system some agency in creating sounds, whereas an effects plugin only alters sound. Our own preconceptions (based on our development of the system) lie more in the "reactive system" approach; but the "effects plugin" discourse seemed to allow User 3 to place the system in a context along with effects plugins that can be bought, downloaded, and used in music production software.

During the analyses we noted that all participants maintained a conceptual distance between themselves and the system, and analogously between their voice and the output sound. There was very little use of the "cyborg" discourse in which the user and system are treated as a single unit, a discourse which hints at mastery or "unconscious competence". This fact is certainly understandable given that the participants each had less than an hour's experience with the interface. It demonstrates that even for beatboxers with strong experience in manipulation of vocal timbre, controlling the vocal interface requires learning – an observation confirmed by the participant interviews.

The issue of "randomness" arose quite commonly among the participants. However, randomness emerges as a nuanced phenomenon: although two of the participants described X as being more random than Y, and placed randomness in opposition to controllability (and to preference), User 2 was happy to describe Y as being more random and also more controllable (and preferable).

A uniform outcome from all participants was the conscious interpretation of the guided exploration tasks as precision-of-reproduction tasks. This was evident during the study sessions as well as from the discourse around the tasks. As one participant put it, "If you're not going to replicate the examples, what are you gonna do?" This issue did not appear in our piloting.

A notable absence from the discourses, given our research context, was discussion which might bear on expressivity, for example the expressive range of the interfaces. Towards the end of each interview we asked explicitly whether either of the interfaces was more expressive, and responses were generally non-commital. We propose that this was because our tasks had failed to engage the participants in creative or expressive activities: the (understandable) reduction of the guided exploration task to a precision-of-reproduction task must have contributed to this. We also noticed that our study design failed to encourage much iterative use of record-and-playback to develop ideas. In the next section we suggest some possible implications of these findings on future study design.

6.4 Discussion

Our DA-based method was designed to extract a detailed reconstruction of users' conceptualisation of a system, and it has achieved that. Our investigation of a voice-controlled interface provides us with interesting detail on the interaction between such concepts as controllability and randomness in the use of the interface, and the different ways of construing the interface itself. These findings would be difficult to obtain by other methods such as observation or questionnaire.

However, we see evidence that the discourses obtained are influenced by the experimental context: the solo sessions, structured with tasks in using both variants of our interface, produced discourse directly related to the interface; while the group session, less structured, produced wider-ranging discourse with less content bearing directly on the interface. The order of presentation also may have made a difference to the participants. It is clear that the design of such studies requires a careful balance: experimental contexts should be designed to encourage exploration of the interface itself, while taking care not to "lead" participants in unduly influencing the categories and concepts they might use to conceptualise a system. It is therefore appropriate to consider our method in contrast with other approaches.

A useful point of comparison is the approach due to Wanderley and Orio [2002], involving user trials on "maximally simple" tasks followed by Likert-scale feedback. As previously discussed, this approach raises issues of task authenticity, and of the suitability of the Likert-style questionnaire. Indeed, Kiefer et al. [2008] investigate the Wanderley and Orio approach, and find qualitative analysis of interview data to be more useful than quantitative data about task accuracy. The Wanderley and Orio method may therefore only be appropriate to cases in which the test population is large enough to draw conclusions from Likert-scale data, and in which the musical interaction can reasonably be reduced or separated into atomic tasks. We suggest the crossfading of records by a DJ as one specific example: it is a relatively simple musical task that may be operationalised in this way, and has a large user-base. (We do not wish to diminish the DJ's art: there are creative and sophisticated aspects to the use of turntables, which may not be reducible to atomic tasks.)

One advantage of the Wanderley and Orio method is that Likert-scale questionnaires are very quick to administer and analyse. In our study the ratio of interview time to analysis time was approximately 1:30 or 1:33, a ratio slightly higher than the ratio of 1:25–1:29 reported for observation analysis of video data [Barendregt et al., 2006]. This long analysis time implies practical limitations for large groups.

Our approaches (as well as that of Wanderley and Orio) are "retrospective" methods, based on users' self-reporting after the musical act. We have argued that concurrent verbal protocols and observation protocols are problematic for experiments involving live musicianship. A third alternative, which is worthy of further exploration, is to gather data via physiological measurements. Mandryk and Atkins [2007] present an approach which aims to evaluate computer-game-playing contexts, by continuously monitoring four physiological measures on computer-game players, and using fuzzy logic to infer the players' emotional state. Analogies between the computer-gaming context and the music-making context suggest that this method could be adopted for evaluating interactive music systems. However, there are some issues which would need to be addressed:

- Most importantly, the inference from continuous physiological variables to continuous emotional state requires more validation work before it can be relied on for evaluation.
- The evaluative role of the inferred emotional state also needs clarification: the mean of the *valence* (the emotional dimension running from happiness to sadness) suggests one simple figure for evaluation, but this is unlikely to be the whole story.
- Musical contexts may preclude certain measurements: the facial movements involved in singing or beatboxing would affect facial electromyography [Mandryk and Atkins, 2007], and the exertion involved in drumming will have a large effect on heart-rate. In such situations, the inference from measurement to emotional state will be completely obscured by the other factors affecting the measured values.

We note that the literature, the present work included, is predominantly concerned with evaluating musical interactive systems from a performer-centred perspective. Other perspectives are possible: a composer-centred perspective (for composed works), or an audience-centred perspective. We have argued in introducing this chapter that the performer should typically be the primary focus of evaluation, in particular for the techniques evaluated here; but in some situations it may be appropriate to perform e.g. audience-centred evaluation. Our methods can be adapted for use with audiences – indeed, the independent observer in our musical Turing Test case study takes the role of audience. However, for audience-centred evaluations it may be the case that other methods are appropriate, such as voting or questionnaire approaches for larger audiences. Labour-intensive methods such as DA will tend to become impractical with large audience groups.

A further aspect of evaluation focus is the difference between solo and group music-making. Wanderley and Orio's set of simple musical tasks is only applicable for solo experiments. Our evaluation method can apply in both solo and group situations, with the appropriate experimental tasks for participants. The physiological approach may also apply equally well in group situations.

6.5 Conclusions

This chapter contributes to our topic in two ways:

Firstly, we contribute to the fledgling topic of evaluation methodology for expressive musical systems, by developing a rigorous qualitative method based on Discourse Analysis (DA). The method was trialled with a small user group and found to yield useful information, although we hope to refine the method in future iterations – perhaps by conducting experiments using pairs of users rather than solo users, to encourage the generation of more relevant talk to be analysed.

Secondly, we have illuminated aspects of the timbre remapping concept developed in Chapter 5 through a contextual user evaluation. With our cohort of beatboxers, we found that the timbre remapping feature was an unproblematic addition to a voice-controlled synthesiser system, not creating unwelcome associations with e.g. uncontrollability. The DA also revealed various different approaches to conceptualising the system, which may be useful information for future design.

Chapter 7

Conclusions and further work

In fulfilment of our aim "to develop methods for real-time control of synthesisers purely using a vocal audio signal" (Section 1.2), the central part of this thesis has been the development of two different ways to apply machine learning techniques to this task – an event-based approach and a continuous timbre remapping approach. To support these techniques we have investigated the choice of timbre features to use, and how to evaluate such systems as expressive interfaces for real-time music making.

To conclude this thesis, we first summarise the contributions made; then we reflect upon the event-based and continuous approaches in comparison with one another. Finally, we consider some potential avenues for future work, including specific consequences of our studies as well as a broader consideration of vocal interfaces to music-making.

7.1 Summary of contributions

- As a preliminary we explored a variety of acoustic features used to represent timbre (Chapter 3). We found that spectral centroid and spectral 95-percentile each could serve well as a representative of perceptual "brightness", but that correlation analysis of timbre perception data did not support any compact set of features to represent the remaining variation in timbral judgements. We also analysed timbre features with respect to criteria of robustness and independence, finding that spectral crest features and ΔMFCCs performed particularly poorly on the robustness measures and therefore are not recommended for our purpose.

- We developed a novel estimator of the differential entropy of multidimensional data (Appendix A), which is computationally efficient and broadly applicable. This was applied as part of the work on feature independence.

- In Chapter 4 we studied the event-based approach to real-time control by vocal timbre (particularly beatboxing), where our particular contribution was to circumvent the dilemma of low-latency vs. good-classification by introducing a delayed decision-making strategy. We evaluated this new strategy by measuring the deterioration in listening quality of some standard drum loops as a function of the amount of delay, and found that a delay of around 12–35 ms for some common drum loops was acceptable, in line with the delay which allows peak classifier performance.

- In Chapter 5 we developed a new regression-tree technique (XAMRT) which can learn associations between the distributions of two unlabelled datasets. We demonstrated that this technique could be used to perform real-time timbre remapping in a way which accommodates differences in the timbre distributions of the source and target, outperforming nearest-neighbour based searches.

  In fact the XAMRT procedure is quite a general technique and may find uses in other domains – we presented one potential application in the analysis of vowel sounds, comparing two populations of English speakers.

- In Chapter 6 we developed a novel approach to the evaluation of expressive musical interfaces, by applying the rigorous qualitative method of Discourse Analysis to participants' talk. In a small study we applied the method and found positive indications for the timbre remapping approach generally. We also gained some insights into the evaluation procedure and suggested future improvements. This provides a contribution to this fledgling topic within musical HCI.

Many of these contributions are represented in international peer-reviewed conference and journal articles, as listed in Section 1.5.

Other outputs include: contributions towards the academic understanding of the beatbox vocal style, both descriptively (Section 2.2) and through producing a publicly-available annotated beatbox dataset (Section 4.1.1); and open-source implementations of algorithms (entropy estimator, XAMRT, SOM) for use in various programming languages.

7.2 Comparing the paradigms: event classification vs. timbre remapping

While carrying out the investigations described in this thesis, we have had opportunities to witness our voice-controlled systems in practice, through informal testing, evaluation with beatboxers, demonstrations at conferences, and musical performances in a variety of settings. Drawing on these experiences as well as the evidence presented, we now consider the relative merits of the two paradigms for vocal musical expression.

The event-based approach with real-time classification appears to be relatively limited in its expressive potential. Our experiments used a simple classifier with only three event types, which is a very small number – a classifier with many more event types might provide for more expression. However, the accuracy of the classification is difficult to maintain as the number of classes grows (as observed in informal testing), and the effect of misclassification during a live vocal performance can be quite distracting for the performer, when a clearly unintended sound is produced by the system. The delayed decision-making strategy helps to mitigate this but misclassifications will still occur.

The continuous timbre-remapping approach has shown a lot of potential. It appears to be quite approachable, at least for performers such as beatboxers, and doesn't tend to make obvious "errors" as would be heard from a misclassified event in a classifier-based system (instead it tends to make less glaring errors, such as some amount of jitter on control parameters during what is intended as a held note). Importantly, the relatively "unbounded" nature of the interaction allows users to discover a wide variety of sounds they can make within a timbre remapping system (e.g. by making popping sounds with the lips, or through vocal trills), sounds which were *not specifically designed in by the developer*. Our relatively basic approach of analysing instantaneous timbre (with no attention to trajectory over time, for example) produces quite a simple mapping whose character is easily learnable by a performer. However the instantaneous approach neglects the opportunity to reduce measurement noise for example by smoothing the timbre trajectory over time, which may be useful to add.

7.3 Further work

Further work that could follow on from the research of this thesis includes:

Temporal modelling: The timbre remapping technique has been developed without any temporal modelling; but temporal evolution is relevant in var-

ious aspects of timbre and vocalisation (such as trill, discussed in Sections 2.1 and 2.2), and temporal considerations could help to reduce noise artefacts, so the integration of temporal modelling could be a fruitful avenue to pursue. Discrete event models such as HMMs may be applicable, for example applied to transitions from one leaf to another in a XAMRT tree – but methods which model continuous timbral evolution should also be investigated.

Voice as expressive music interface: The real-time non-speech voice interface is still underdeveloped in terms of research understanding, in music as well as in other fields (e.g. non-speech command interfaces [Harada et al., 2009]). Future work should investigate psychological aspects such as the split of attention between input and output sounds, and the benefits/disadvantages of de-personalising the sound by transforming it. Explicit formal comparisons between vocal interfaces and other modalities should also be conducted.

Combining event-based and continuous: There may be benefit in combining aspects of both event-based and continuous paradigms into future voice-based systems. For example, event segmentation would give access to analysis of attack times, not possible in a purely instantaneous approach.

Study of group interaction: As one particular aspect of the voice interface, group interactive aspects deserve more research attention. Beatboxers and other vocal performers often perform and even improvise together, as do other musicians, and there is scope for exploring the nature of group interaction (such as self-identification and the exchange of musical ideas) in technologically-mediated vocal performance. There is some work on the interactive aspects of group improvisation [Healey et al., 2005, Bryan-Kinns et al., 2007]; future work should investigate this for vocal group interaction, teasing out any aspects specific to the vocal modality. More traditional vocal interactions should be studied first, so that technologically-mediated vocal interactions can usefully be compared with them.

Use in other contexts: Our user study was conducted with beatboxers, but the potential application of vocal technology such as timbre remapping could exist in other areas. We envisage studies which explore its use in populations such as novice users, children and users of music therapy services. Some recent work has explored "vocal sketching" for sound design,[1] another potentially fruitful application domain for these technologies.

[1] `http://www.michalri.net/sid/category/about/`

7.3.1 Thoughts on vocal musical control as an interface

Vocal control of music technology is at present not mainstream. The MIDI keyboard remains the *de facto* standard interface for digital musicians (referred to in Chapter 1). The robotic sound of the "vocoder effect" has established a niche for itself in popular music [Tompkins, 2010], and so too has exaggerated Auto-Tune vocal processing (the "Cher effect") [Dickinson, 2001]. Some voice-interactive mobile music applications have gained media attention,[2] though it remains to be seen whether the latter will have long-term traction. The question arises whether vocal musical control could or should ever become the mainstream interface for musical expression. For example, could composers do away with their MIDI keyboard and use the microphone built into their computer?

One obvious limitation on vocal interfaces is on polyphony. A solo vocalist cannot directly produce the range of chords available to a solo keyboardist – even the polyphony available through techniques such as overtone singing and ventricular phonation (Section 2.3.2) is relatively limited. There are workarounds, such as layering multiple sounds, but this inability of the interface directly to enable musical effects such as harmony suggests that we would be unwise to propose a vocal interface as the main tool for all solo composition tasks. However, the keyboard and the microphone can co-exist; and since microphones are increasingly commonly included as standard in consumer computers, we suggest that vocal interaction may increasingly become a component of music technology, perhaps as part of a multimodal interaction.

An issue peculiar to the vocal interface is that people can often be inhibited about vocal musical expression – confronting someone with a microphone can induce them to opt out saying, "I can't sing", more often due to inhibition than inability [Sloboda et al., 2005, Abril, 2007]. However, the popularity of karaoke [Kelly, 1998] and of its transformation into computer games such as SingStar [Hämäläinen et al., 2004] shows that many people can overcome this barrier given the right social context. Further, techniques such as timbre remapping might help to de-personalise the output sound (cf. Section 6.3.2) and therefore help to overcome inhibitions.

Unlike the MIDI keyboard controller and various other interfaces, vocal sound occupies the same modality (audio) as the musical result. It therefore raises issues such as unwanted feedback, and the extent to which a performer can/must pay attention both to the input and output sounds simultaneously. Live performance styles such as beatboxing indicate that such issues are not overly inhibiting. From practical experience performing with a timbre remapping system, we note that feedback suppression would be useful for club-type

[2] `http://www.smule.com/products`, `http://www.rjdj.me/`

environments where PA/monitor sound can bleed back into the microphone signal – feedback suppression schemes exist, but would probably need to be customised to the timbre remapping case where the input and output are different types of sound.

Current voice-based interfaces also preclude silent practice, since the performer must make noise even if the output sound is muted or played on headphones. One development which may bear upon this in future is that of silent voice interfaces, which use non-audio analysis of the vocal system to react to mimed vocalisations [Denby et al., 2010].

The phone "form factor"

Many developments in 20th century music seem to have been stimulated by the equipment that was becoming widely available. Cheap drum machines and samplers were important in rave and house music [Butler, 2006], vinyl turntables in hip-hop and DJ culture. The massive growth of the general-purpose home computer stimulated many musical scenes in the late 20th century, allowing "bedroom producers" to make multitrack home recording studios with little additional costs (such as the Atari ST in the mid 1980s, which even had built-in MIDI ports) [Collins and d'Escriván, 2007, Chapter 5].

It is therefore worth noting the general rise of the mobile phone in the first decade of the 21st century, and in particular the growing popularity of "smartphones" capable of general-purpose computing. The smartphone platform differs from the home computer – it has no keyboard (or a limited one) and few buttons, but comes with a microphone built in. There is already some academic and commercial work which aims to capitalise on the affordances of this form factor for music,[2] although we consider the topic to be still in its infancy, and look forward to further development stimulated by the wide availability of advanced mobile phones. In our view this form factor will lead to an interaction paradigm that is multimodal by default, using audio as well as camera-based and touch interaction.

7.3.2 Closing remarks

Broadly, we note from experience that the subtlety of the sounds which people can and do produce vocally is staggering, and is beyond the wit of current algorithms to reproduce entirely. The techniques we have developed provide expressive tools which performers enjoy and find useful; but from personal experience we assert that there is still more under-utilised information in the input – both in the signal itself and in the cultural and musical significations which a human listener can pick up. There is still a long way to go before systems can

produce a really musically intelligent response to any given vocal input. That would probably require systems trained with cultural and musical information as well as rich processing of the immediate input stream(s).

Reflecting on the results of the qualitative study (Chapter 6) together with our own experience of performing with the systems developed in this thesis, we conclude that timbre remapping in particular is a viable approach to expanding the palette of beatboxing, and hopefully also other types of vocal performance. Since we do not claim that our current timbre remapping system is able to translate all the subtleties of an expressive vocal performance into the output sound, one might argue that the system is constraining rather than enabling, since it is likely that there is less variety at the output than the input. However, in performance situations a vocalist would have the option of switching between the remapped or raw sound (in the focus-group session, participants did exactly this), and/or switching among different synthesisers controlled – meaning the overall effect is to extend a performer's timbral repertoire and to allow them move between different sonic palettes during the course of a performance.

Some beatboxers take a purist approach and prefer not to add further technological mediation to their performance – while a larger portion of performing beatboxers use technology to build on top of the basic beatbox sound and to extend the musical interest of a live solo performance (e.g. by using audio-looping effects). Based on our research and our experience of beatbox performance, we look forward to timbre remapping techniques being available as part of performers' live setup – not as an exclusive vehicle for expression, but as one such tool in the vocal performer's toolkit.

Appendix A

Entropy estimation by k-d partitioning

For a multivariate random variable X taking values $x \in \mathcal{X}$, $\mathcal{X} = \mathbb{R}^D$, the differential Shannon entropy is given as

$$H = -\int_{\mathcal{X}} f(x) \log f(x)\, dx \tag{A.1}$$

where $f(x)$ is the probability density function (pdf) of X [Arndt, 2001]. Estimating this quantity from data is useful in various contexts, for example image processing [Chang et al., 2006] or genetic analysis [Martins Jr et al., 2008]. While estimators can be constructed based on an assumed parametric form for $f(x)$, non-parametric estimators [Beirlant et al., 1997] can avoid errors due to model mis-specification [Victor, 2002].

In this appendix we describe a new non-parametric entropy estimator, based on a rectilinear adaptive partitioning of the data space. The partitioning procedure is similar to that used in constructing a k-d tree data structure [Bentley, 1975], although the estimator itself does not involve the explicit construction of a k-d tree. The method produces entropy estimates with similar bias and variance to those of alternative estimators, but with improved computational efficiency of order $\Theta\big(N \log N\big)$.

In the following, we first state the standard approach to entropy estimation by adaptive partitioning (Section A.1), before describing our new recursive partitioning method and stopping criterion in Section A.2, and considering computational complexity issues in Section A.3. We present empirical results on the bias, variance and efficiency of the estimator in Section A.4.

A.1 Entropy estimation

Consider a partition A of $\mathcal{X}$, $A = \{A_j \,|\, j = 1, ..., m\}$ with $A_j \cap A_k = \emptyset$ if $j \neq k$ and $\bigcup_j A_j = \mathcal{X}$. The probability mass of $f(x)$ in each cell A_j is $p_j = \int_{A_j} f(x)$. We may construct an approximation $f_A(x)$ having the same probability mass in each cell as $f(x)$, but with a uniform density in each cell:

$$f_A(x) = \frac{p_j}{\mu(A_j)} \quad , \quad j \text{ s.t. } x \in A_j \tag{A.2}$$

where $\mu(A_j)$ is the D-dimensional volume of A_j.

Often we do not know the form of $f(x)$ but are given some empirical data points sampled from it. Given a set of N D-dimensional data points $\{x_i \,|\, i = 1, ..., N\}, x_i \in \mathbb{R}^D$, we estimate p_j by n_j/N where n_j is the number of data points in cell A_j. An empirical density estimate can then be made:

$$\hat{f}_A(x) = \frac{n_j}{N\mu(A_j)} \quad , \quad j \text{ s.t. } x \in A_j \,. \tag{A.3}$$

This general form is the basis of a wide range of density estimators, depending on the choice of partitioning scheme used to specify A. A surprisingly broad class of data-adaptive partitioning schemes can be used to create a consistent estimator, meaning $\hat{f}_A(x) \to f(x)$ as $N \to \infty$ [Breiman et al., 1984, Chapter 12][Zhao et al., 1990].

The within-cell uniformity of $f_A(x)$ allows us to rewrite (A.1) to give the following expression for its entropy:

$$H_A = \sum_{j=1}^{m} p_j \log \frac{\mu(A_j)}{p_j} \tag{A.4}$$

and so our partition-based estimator from data points x_j is

$$\hat{H} = \sum_{j=1}^{m} \frac{n_j}{N} \log \left(\frac{N}{n_j} \mu(A_j) \right) \tag{A.5}$$

To estimate the entropy from data, it thus remains for us to choose a suitable partition A for the data.

A.1.1 Partitioning methods

A computationally simple approach to choose a partition A is to divide a dataset into quantiles along each dimension, since quantiles provide a natural way to divide a single dimension into regions of equal empirical probability. Indeed, in one dimension this approach leads to estimators such as the "m_N-spacing"

estimator of Vasicek [1976] (see also Learned-Miller and Fisher [2003]). In the multidimensional case, by dividing each dimension of $\mathbb{R}^D$ into q-quantiles, we would create a product partition having q^D cells. However, such a product partition can in fact lead to poor estimation at limited number of data points N because $f(x)$ is not in general equal to the product of its marginal densities, and so the product partition may be a poor approximation to the structure of the ideal data partition [Darbellay and Vajda, 1999].

Data-driven non-product partitioning methods exist. Voronoi partitioning divides the space such that each data point is the centroid of a cell, and the boundary between two adjacent cells is placed equidistant from their centroids. Delaunay triangulation partitions the space using a set of simplices defined with the data points at their corners [Edelsbrunner, 1987, Chapter 13]. Such partitions are amenable to entropy estimation by (A.5), as considered by Learned-Miller [2003]. However, the complexity of such diagrams has a strong interaction with dimensionality: although two-dimensional diagrams can be $\mathrm{O}\big(N \log N\big)$ in time and storage, at $D \geq 3$ they require $\mathrm{O}\big(N^{\lceil \frac{D+1}{2} \rceil}\big)$ time and $\mathrm{O}\big(N^{\lceil \frac{D}{2} \rceil}\big)$ storage [Edelsbrunner, 1987, Chapter 13].

Partitioning by tree-like recursive splitting of a dataset is attractive for a number of reasons. It is used in nonparametric regression [Breiman et al., 1984] as well as in constructing data structures for efficient spatial search [Bentley, 1975]. The non-product partitions created can take various forms, but in many schemes they consist of hyperrectangular cells whose faces are axis-aligned. Such hyperrectangle-based schemes are computationally advantageous because the storage complexity of the cells does not diverge strongly, requiring only $2D$ real numbers to specify any cell. A notable example here is Darbellay and Vajda's 2D mutual information estimator [Darbellay and Vajda, 1999], which recursively splits a dataset into four subpartitions until an independence criterion is met. In Section A.2 we will describe our new method which has commonalities with this approach, but is specialised for the fast estimation of multidimensional entropy.

A.1.2 Support issues

If the support of the data is not known or unbounded then there will be open cells at the edges of A. These are problematic because they have effectively infinite volume and zero density, and cannot be used to calculate (A.5). One solution is to neglect these regions and adjust N and m to exclude the regions and their data points [Learned-Miller, 2003]. But for small datasets or high dimensionality, this may lead to the estimator neglecting a large proportion of the data points, leading to an estimator with high variance. It also leads to a biased estimate, tending to underestimate the support.

An alternative is to limit edge cells to finite volume by using the Maximum Likelihood Estimate of the hyperrectangular support. This reduces to the estimate that the extrema of the data sample define the support (since any broadening of the support beyond the extrema cannot increase the posterior probability of the data sample). This is of course likewise a biased estimate, but does not exclude data points from the calculation of (A.5), and so should provide more efficient estimation at low N. We use this approach in the following.

A.2 Adaptive partitioning method

Since the approximation $\hat{f}_A(x)$ has a uniform distribution in each cell, it is reasonable to design our adaptive partitioning scheme deliberately to produce cells with uniform empirical distribution, so that $\hat{f}_A(x)$ best approximates $f(x)$ at limited N. Partitioning by recursively splitting a dataset along quantiles produces a consistent density estimator [Breiman et al., 1984, Chapter 12][Zhao et al., 1990], so we design such a scheme whose stopping criterion includes a test for uniformity.

At each step, we split a set of data points by their sample median along one axis, producing two subpartitions of approximately equal probability. This has a close analogy in the approach used to create a k-d tree data structure [Bentley, 1975], hence we will refer to it as *k-d partitioning*. Such rectilinear partitioning is computationally efficient to implement: not only because the splitting procedure needs only consider one dimension at a time, but because unlike in the Voronoi or Delaunay schemes any given cell is a hyperrectangle, completely specified by only $2D$ real numbers.

It remains to select a test of uniformity. Various tests exist [Quesenberry and Miller, 1977], but in the present work we seek a computationally efficient estimator, so we require a test which is computationally light enough to be performed many times during estimation (once at each branch of the recursion). Since our partitioning scheme requires measurement of the sample median, we might attempt to use the distribution of the sample median in a uniform distribution to design a statistical test for uniformity.

The distribution of the sample median tends to a normal distribution [Chu, 1955] which can be standardised as

$$Z_j = \sqrt{n_j} \frac{2 \cdot \mathrm{med}_d(A_j) - \min_d(A_j) - \max_d(A_j)}{\max_d(A_j) - \min_d(A_j)} \tag{A.6}$$

where $\mathrm{med}_d(A_j)$, $\min_d(A_j)$, $\max_d(A_j)$ respectively denote the median, minimum and maximum of the hyperrectangular cell A_j along dimension d. An improbable value for Z_j (we use the 95% confidence threshold for a standard

```
KDPEE({x_i}, D, N)
    L_N ← result of equation (A.7)
    A_0 ← range({x_i})
    return KDPEE_RECURSE(A_0, 1)

KDPEE_RECURSE(A, level)
    d ← level  mod D
    n ← count(x_i ∈ A)
    med ← median along dth dimension of x_i ∈ A
    Z ← result of equation (A.6)
    if level ≥ L_N and |Z| ≥ 1.96
       then
            return (n/N) log((N/n) μ(A))
       else
            U ← A ∩ (dimension_d < med)
            V ← A ∩ (dimension_d ≥ med)
            return KDPEE_RECURSE(U, level +1)
                   +KDPEE_RECURSE(V, level +1)
```

Figure A.1: The k-d partitioning entropy estimation algorithm for a set of N D-dimensional data points $\{x_i\}$. Note that the dimensions are given an arbitrary order, $0...(D-1)$. A_0 is the initial partition with a single cell containing all the x_i.

normal distribution, $|Z_j| > 1.96$) indicates significant deviation from uniformity, and that the cell should be divided further.

This test is weak, having a high probability of Type II error if the distribution is non-uniform along a dimension other than d, and so can lead to early termination of branching. We therefore combine it with an additional heuristic criterion that requires partitioning to proceed to at least a minimum branching level L_N, so that the cell boundaries must reflect at least some of the structure of the distribution. We use the partitioning level at which there are $\sqrt{N}$ data points in each partition,

$$L_N = \left\lceil \frac{1}{2} \log_2 N \right\rceil . \tag{A.7}$$

This is analogous to the common choice of $m = \sqrt{N}$ in the m-spacings entropy estimator, which in that case is chosen as a good compromise between bias and variance [Learned-Miller, 2003]. Our combined stopping criterion is therefore $L \geq L_N$ and $|Z_j| > 1.96$.

The recursive estimation procedure is summarised as pseudocode in Figure A.1.

To produce a reasonable estimate, we expect to require a minimum amount

of data values. We require the estimator to be able to partition at least once along each dimension–in order that no dimension is neglected in the volume estimation–so the estimator must have the potential to branch to D levels. The number of levels in the full binary tree approximates $\log_2 N$, which gives us a lower limit of $N \geq 2^D$. This limit will become important at high dimensionality.

A.3 Complexity

The complexity of all-nearest-neighbour-based estimators such as that of Kybic [2006] is dominated by their All Nearest Neighbour (ANN) search algorithm. The naïve ANN search takes $\Theta\big(N^2 D\big)$ time, but improved methods exist [Chávez et al., 2001]. For example, using a *cover tree* data structure, ANN can be performed in $\mathrm{O}\big(c^{12} N \log N\big)$ time, where c is a data-dependent "expansion constant" which may grow rapidly with increasing dimensionality [Beygelzimer et al., 2006]. Time complexity of $\mathrm{O}\big(N \log N\big)$ is possible in a parallel-computation framework [Callahan, 1993].

Learned-Miller's estimator based on Voronoi-region partitions [Learned-Miller, 2003] is, like ours, a multidimensional partitioning estimator. As discussed in section A.1.1 the complexity of Voronoi or Delaunay partitioning schemes is $\mathrm{O}\big(N^{\lceil \frac{D+1}{2} \rceil}\big)$ in time and $\mathrm{O}\big(N^{\lceil \frac{D}{2} \rceil}\big)$ in storage, meaning that for example a 3D Voronoi diagram is $\mathrm{O}\big(N^2\big)$ in time and storage.

Kernel density estimation (KDE) can also be a basis for entropy estimation [Beirlant et al., 1997]. Methods have been proposed to improve on the naïve KDE complexity of $\mathrm{O}\big(N^2 D\big)$, although their actual time complexity is not yet clear [Lang et al., 2004].

For our algorithm, the time complexity is dominated by the median partitioning, which we perform in $\Theta\big(N\big)$ time using Hoare's method [Hoare, 1961]. At each partitioning level we have m_L cells each containing approximately $\frac{N}{m_L}$ points, meaning that the total complexity of the m_L median-finding operations remains at $\Theta\big(N\big)$ for each level. For any given dataset, the stopping criterion (A.6) may result in termination as soon as we reach level L_N or may force us to continue further, even to the full extent of partitioning. Therefore the number of levels processed lies in the range $\frac{1}{2}\log_2 N$ to $\log_2 N$. This gives an overall time complexity of $\Theta\big(N \log N\big)$ at any dimensionality. For $D > 2$ and a single processor this is therefore an improvement over the other methods.

The memory requirements of our algorithm are also low. In-place partitioning of the data can be used, and no additional data structures are required, so space complexity is $\Theta\big(N\big)$. This is the same order as the cover-tree-based ANN estimator, and better than the Voronoi-based estimator.

A.4 Experiments

We tested our k-d estimation algorithm against samples from some common distribution types, with $N = 5000$ and D from 1 to 12. In each case we ran 100 simulations and calculated the mean deviation from the theoretical entropy value of the distribution, as well as the variance of the entropy estimates. These will be expressed as deviations from the true entropy, which in all cases was 2 nats ($\frac{2}{\ln 2}$ bits) or greater.

For comparison, we also tested two other common types of estimator: a KDE-based resubstitution estimator, and an ANN estimator. We used publicly-available implementations due to Ihler,[1] which use a k-d tree to speed up the KDE and ANN algorithms. All three implementations are Matlab code using C/C++ for the main calculations. We did not test the Voronoi-region-based estimator because it becomes impractical beyond around 4 dimensions (Learned-Miller, pers. comm.).

Fig. A.2 plots the bias for up to 12 dimensions, for each of the three different estimators. In general, the estimators all provide bias performance at a similar order of magnitude and with a similar deterioration at higher dimensionality, although our estimator exhibits roughly twice as much bias as the others. The narrow confidence intervals on the graphs (exaggerated for visibility in Fig. A.2) reflect the low variance of the estimators.

The upward bias of our estimator for non-uniform distributions at higher dimensions is likely to be due to underestimation of the support, neglecting regions of low probability (see Section A.1.2). This would lead to some overestimation of the evenness of the distribution and therefore of the entropy. Since the estimator is consistent, this bias should decrease with increasing N.

Fig. A.3 plots the CPU time taken by the same three estimators, at various data sizes and $D \in 2, 5, 8$. In all tested cases our estimator is faster, by between one and three orders of magnitude. More importantly, the times taken by the resubstitution and ANN estimators diverge much more strongly than those for our estimator, at increasing D and/or N. As we expect from Section A.3, CPU time for our estimator is broadly compatible with $\Theta\big(N \log N\big)$ (Fig. A.4).

A.5 Conclusion

We have described a nonparametric entropy estimator using k-d partitioning which has a very simple and efficient implementation on digital systems, running in $\Theta\big(N \log N\big)$ time for any dimensionality of data. In experiments with known

[1] `http://www.ics.uci.edu/~ihler/code/`

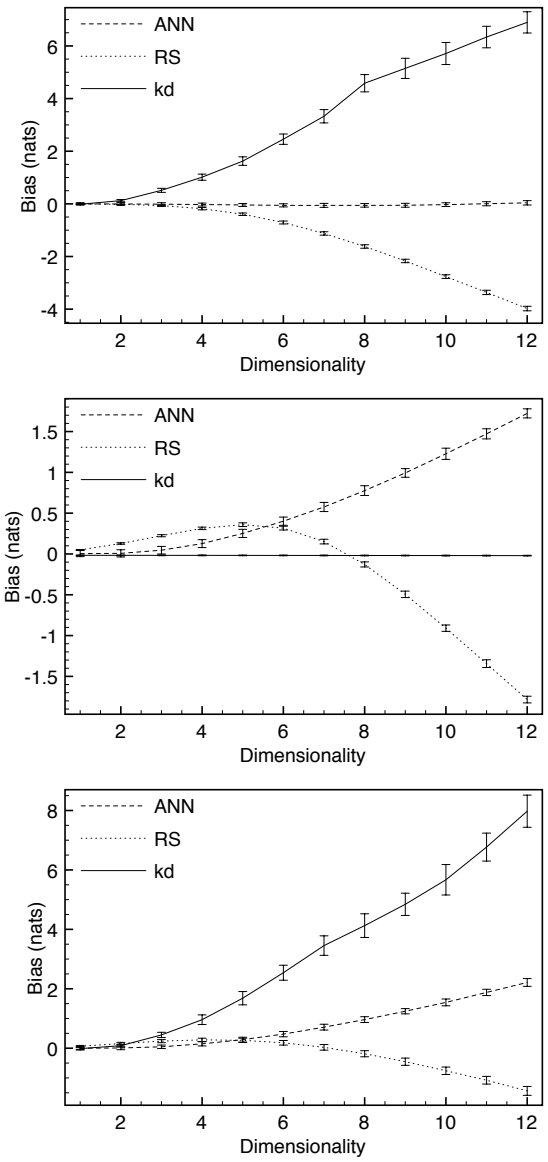


Figure A.2: Bias of some entropy estimators at increasing dimensionality. Error bars show the 95% confidence interval exaggerated by a factor of 10 for visibility. Distributions tested are gaussian (top), uniform (middle), exponential (bottom). $N = 5000$, 100 runs. **ANN** = all-nearest-neighbours estimator. **RS** = resubstitution estimator. **kd** = k-d partitioning estimator.

distributions, our estimator exhibits bias and variance comparable with other estimators.

The estimator is available for Python (numpy), Matlab or GNU Octave.[2]

[2] `http://www.elec.qmul.ac.uk/digitalmusic/downloads`

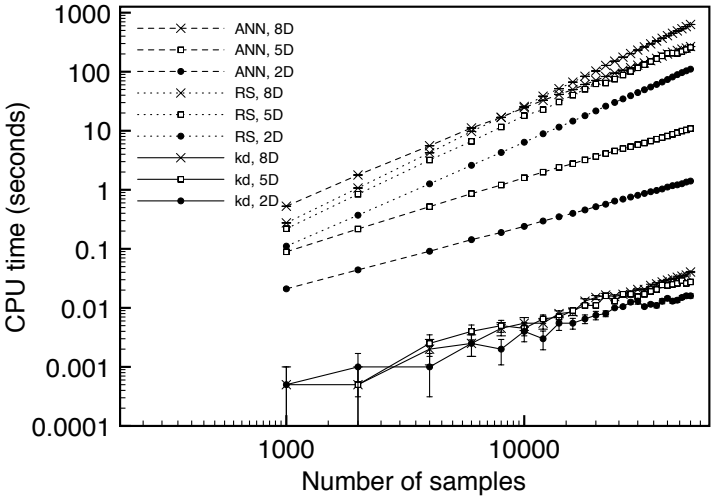


Figure A.3: CPU time for the estimators in Figure A.2, using Gaussian distributions and $D \in 2, 5, 8$. Tests performed in Matlab 7.4 (Mac OSX, 2 GHz Intel Core 2 Duo processor). Data points are averaged over 10 runs each (20 runs each for our estimator). 95% confidence intervals are shown (some are not visible).

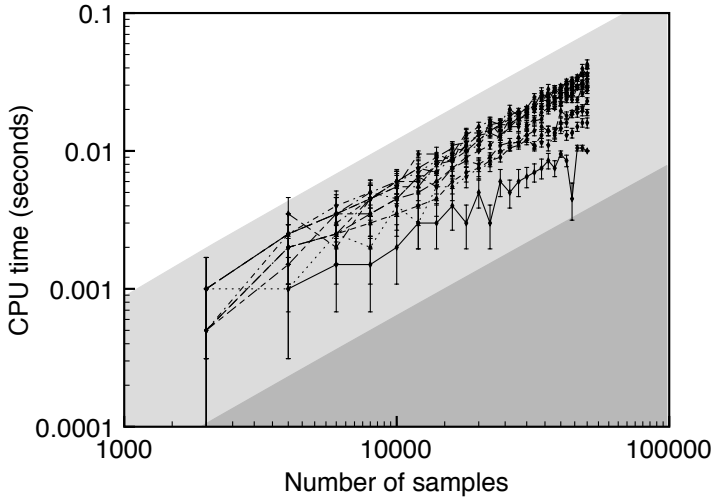


Figure A.4: CPU time for our estimator, calculated as in Figure A.3 but for all D ranging from 1 to 12. The shaded areas indicate slopes of $kN \log N$.

Appendix B

Five synthesisers

This appendix describes five simple synthesisers (synths) used for some of the work in the thesis, including stability of features (Section 3.3.1). Each of them was implemented as a "SynthDef" (synth definition) in SuperCollider 3.3.1 [McCartney, 2002], and each is given here with a brief description, plus the SuperCollider SynthDef source code and a description of the controls.

B.1 simple

A simple mixture of a sine-wave and a pink noise source, intended to represent a synthesiser with a very simple timbral range.

```
SynthDef(\_maptsyn_supersimple, { |out=0, amp=1,
freq=440, noise=0|
var son;
son = XFade2.ar(SinOsc.ar(freq), PinkNoise.ar, noise);
Out.ar(out, son * (amp));
})
```

Control inputs:

freq: fundamental frequency, 25–4200 Hz exponential

noise: noise/tone mix control, −1–1 linear

B.2 moogy1

A software implementation of a popular type of analogue-inspired sound: a saw wave with a variable amount of additive pink-noise and crackle-noise, passed

through an emulation of a Moog synthesiser's resonant low-pass filter [Fontana, 2007].

```
SynthDef(\_maptsyn_moogy1, { |out=0, amp=1,
freq=440, noise=0.1, dustiness=0.1, filtfreq=1000, filtgain=1|
var son;
son = MoogFF.ar(Saw.ar(freq) * PinkNoise.ar.range(1 - noise, 1)
+ Dust2.ar(dustiness), filtfreq, filtgain);
Out.ar(out, son * (amp * 2.8));
})
```

Control inputs:

freq: fundamental frequency, 25–4200 Hz exponential

noise: pink noise modulation depth, 0–1 linear

dustiness: additive crackle noise amplitude, 0–1 linear

filtfreq: filter cutoff, 20–20000 Hz exponential

filtgain: filter resonance gain, 0–3.5 linear

B.3 grainamen1

A granular synthesiser [Roads, 1988] applied to a recording of the Amen break-beat [Butler, 2006, p78] to produce a controllable unpitched sound varying across the timbral range of a drum-kit, yet with the granular synthesis aspect providing a controllable stationarity that is not present in many drum sounds.

```
SynthDef(\_maptsyn_grainamen1, { |out=0, amp=1,
// mapped:
phasegross=0.5, phasefine=0.05, trate=50,
// extraArgs:
bufnum=0
|
var phase, son, clk, pos, dur;
dur = 12 / trate;
clk = Impulse.kr(trate);
pos = (phasegross + phasefine) * BufDur.kr(bufnum)
+ TRand.kr(0, 0.01, clk);
son = TGrains.ar(2, clk, bufnum, 1.25, pos, dur, 0, interp: 0)[0];
Out.ar(out, son * (amp * 20));
})
```

Control inputs:

phasegross: gross position from which to take grains, 0–0.95 linear

phasefine: fine position from which to take grains, 0–0.05 linear

trate: grains per second, 16–120 exponential

B.4 ay1

A software emulation of the General Instrument AY-3-8910 sound chip [Sinclair Research Ltd., 1985], a real-world yet relatively simple sound-synthesis chip with a set of integer-valued controls for three tone generators and one noise generator, each with roughly square waveform. Only one tone generator was used in this realisation to preserve monophony.

```
SynthDef(\_maptsyn_ay1, { |out=0, amp=1,
control=1, noise=15, freq=440|
var son;
son =  AY.ar(
control: control,
noise: noise,
tonea: AY.freqtotone(freq).round,
toneb: 0,
tonec: 0,
vola: 15,
volb: 0,
volc: 0
);
Out.ar(out, son * (amp * 2.8));
})
```

Control inputs:

control: chip control (bit mask) for tone/noise/both, discrete values 1/8/9

noise: chip control for noise type, integers 0–31

freq: fundamental frequency, 25–4200 Hz exponential

B.5 gendy1

An implementation of the "dynamic stochastic synthesis generator" conceived by Iannis Xenakis [Xenakis, 1992, Chapters 9, 13, 14] and implemented in SuperCollider by Nick Collins, which is a synthesiser with some dynamic and random

elements (yet with a consistent percept at any given setting), capable of a wide range of timbres reminiscent of e.g. trumpet, car horns, bees.

```
SynthDef(\_maptsyn_gendy1, { |out=0, amp=1,
ampdist=1, durdist=1, adparam=1.0, ddparam=1.0,
minfreq=20, maxfreq=1000, ampscale= 0.5|
var son;
son = Gendy1.ar(ampdist, durdist, adparam,
ddparam, minfreq, maxfreq, ampscale, 0.5);
son = son.softclip;
Out.ar(out, son * amp);
}).writeDefFile
```

Control inputs (all directly controlling parameters of the Gendy1 algorithm, see Gendy1 helpfile or [Xenakis, 1992, Chapters 9, 13, 14] for detail on their effect):

ampdist: integers 0–5

durdist: integers 0–5

adparam: 0.0001–1 linear

ddparam: 0.0001–1 linear

minfreq: 10–2000 Hz exponential

minfreq: 200–10000 Hz exponential

ampscale: 0.1–1 linear

Appendix C

Classifier-free feature selection for independence

Our investigations into timbre features in Chapter 3 largely investigated attributes of the features individually. However we are likely to be using multiple timbre features together as input to machine learning procedures which will operate on the resulting multidimensional timbre space. We therefore wish to find a subset of features which together maximise the amount of useful information they present while minimising the number of features, to minimise the risk of curse of dimensionality issues. To do this we wish to perform a kind of feature selection based on analysing the redundancy among sets of variables.

However, our voice datasets are unlabelled and so we do not have the option of using a classifier-based feature selection [Guyon and Elisseeff, 2003]. A very few others have studied unsupervised feature selection, e.g. Mitra et al. [2002], who use a clustering technique. In this Appendix we report some preliminary experiments working towards the aim of selecting an independent subset of features in an unsupervised context. This work is unfinished; as we will discuss, it is a difficult task with issues still be resolved.

C.1 Information-theoretic feature selection

We have seen in Section 3.3.2 some use of entropy and mutual information measures to characterise the amount of information shared between variables. Generalisations of mutual information from the bivariate to the multivariate case exist [Fano, 1961][Fraser, 1989], however these are not as widely used as mutual information applied pairwise to features. Another useful measure is the *conditional entropy* [Arndt, 2001, Chapter 13]. The entropy of Y conditional

on X is given as

$$H(Y|X) = \int_X p(x)\, H(Y|X = x) \tag{C.1}$$

$$= H(Y, X) - H(X) \tag{C.2}$$

and can be seen as quantifying the amount of information provided by Y that is not also provided by X (which may be multivariate).

The conditional entropy measure (C.2) can be used as a basis for feature selection. Given a set of K features, for each feature we can calculate the conditional entropy between that feature and the remaining $K - 1$ features, to quantify the amount of information it provides that is not otherwise present in the ensemble. We emphasise the contextual nature of such a calculation: the results for each feature depend on which other features are being considered. Using such measures, a subset could be chosen in which the lowest inter-feature redundancy is found.

In feature selection, the optimal result could be determined by an exhaustive search, but the number of possible combinations to be evaluated is exponential in the number of candidate features and thus typically intractable [Dash and Liu, 1997]. Two common types of search algorithm are sequential forwards selection (or "greedy selection") – in which a small set of features is repeatedly grown by adding in an extra feature, such that some criterion is maximised – and sequential backwards selection (or "greedy rejection") – in which a large set of features is repeatedly reduced by choosing a feature to reject, such that some criterion is maximised [Jain and Zongker, 1997]. Either algorithm can be used to produce a set of features of a desired size, and/or to rank all features in order of preference. The choice/ranking is not guaranteed to be optimal but is often near optimal [Jain and Zongker, 1997], and can be improved by using a "floating" search which allows the possibility to backtrack, e.g. in forwards selection by rejecting features that had been selected in an earlier iteration [Pudil et al., 1994].

Conditional entropy can be used as an evaluation metric for sequential selection. At each step one could identify which of K features has the lowest conditional entropy with the others and can be rejected (backwards selection), or could identify which of a set of additional features has the highest conditional entropy with the K features and should be added to the set (forwards selection). However, the nature of nonlinear dependence analysis presents some difficulties which must be considered:

- Backwards selection is initialised with the full set of candidate features, i.e. with a high-dimensional feature space. Yet estimators of information-

theoretic quantities from data are known to perform worse at higher dimensionality (see Appendix A), so the earliest estimates in such an approach could introduce error such as prematurely rejecting a given feature and therefore strongly skewing its ranking.

- Forwards selection begins with a small set of candidate features, and therefore the earlier estimates of information-theoretic quantities (on e.g. one- or two-dimensional spaces) would be expected to be the more reliable. However, nonlinear dependencies may exist within larger groups of features that are not apparent when considering only small subsets. For example, three features A B and C may be relatively independent from one another when evaluated pairwise, yet there could still exist significant informational overlap in the set A B C. In such a situation, the three-way interaction would not be evident in the two-way measures such as $H(A|B)$ or $H(B|A)$, meaning that for example A and B could be selected at an early stage, producing a ranking which fails to reflect the higher-order dependencies.

We therefore use a hybrid of both forwards and backwards techniques, as follows. We choose a cardinality K at which it is tractable to evaluate all possible subsets of the candidate features, e.g. $K = 3$ or $K = 4$. Given S total features, the number of subsets to be evaluated is $\binom{S}{K} = \frac{S!}{K!(S-K)!}$. We evaluate all possible feature subsets of cardinality K to find which has the least redundancy between features – note that this can be made equivalent to finding which subset has the largest joint entropy, if the features are first normalised such that each has the same fixed univariate entropy (Equation C.2). Then having identified the best such subset, we perform both forwards and backwards selection starting from that point: backwards selection to rank the K features, and forwards floating selection to append the remaining $S - K$ features. In this way, the entire feature set is ranked, and an information-bearing subset of any size $K \in (1...S)$ can be identified, yet the problems stated above for pure forwards or backwards searches will be reduced in their effect.

C.2 Data preparation

We used the same data preparation as described in Section 3.4: the three voice datasets *SNG SPC* and *BBX* were analysed, using our entropy estimator (Appendix A) to estimate conditional entropies by Equation (C.2).

The calculation was optimised by applying the probability integral transform to each feature, which normalises the univariate entropies, meaning we could select for maximum joint entropy rather than minimum conditional entropy.

Rank	Feature	CondEnt
1	dmfcc8	-3.75242e-05
2	crst2	-0.00795948
3	dmfcc4	-0.0147636
4	mfcc3	-0.0214396
5	dmfcc2	-0.00500112
6	iqr	-0.00924548
7	dmfcc6	0.0025963
8	dmfcc7	-0.00977632
9	crst5	-0.0184311
10	mfcc8	-0.00381674
11	dmfcc5	0.000197236
12	mfcc7	0.00144796
13	mfcc6	-3.33595e-05
14	pcile75	-3.33595e-05
15	mfcc5	-3.33595e-05
16	dmfcc3	-3.33595e-05
17	crst3	-3.33595e-05
18	power	-3.33595e-05
19	flux	-3.33595e-05
20	pow5	-3.33595e-05
21	centroid	-3.33595e-05
22	mfcc4	-3.33595e-05
23	crst1	-3.33595e-05
24	mfcc2	-3.33595e-05
25	slope	-3.33595e-05
26	tcrest	-3.33595e-05
27	flatness	-3.33595e-05
28	dmfcc1	-3.33595e-05
29	zcr	-3.33595e-05
30	pow1	-3.33595e-05
31	pow4	-3.33595e-05
32	mfcc1	-3.33595e-05
33	pow3	-3.33595e-05
34	pcile95	-3.33595e-05
35	crest	-3.33595e-05
36	pow2	-3.33595e-05
37	crst4	-3.33595e-05
38	spread	-3.33595e-05
39	pitch	-5.00396e-05
40	pcile50	-0.000100082
41	pcile25	-0.00138535
42	clarity	-0.0145836

(a) SNG dataset

Rank	Feature	CondEnt
1	dmfcc7	-0.00704878
2	pow4	-0.0026457
3	dmfcc4	-0.0104024
4	dmfcc8	-0.0136911
5	dmfcc5	0.00327771
6	dmfcc3	-0.0020854
7	crst2	-0.00977744
8	crst5	-0.0181953
9	crst3	-0.00361814
10	crst4	-0.00196425
11	flux	-0.00223691
12	mfcc8	0.000262112
13	pow5	-3.31625e-05
14	pcile75	-3.31625e-05
15	dmfcc1	-3.31625e-05
16	tcrest	-3.31625e-05
17	mfcc7	-3.31625e-05
18	zcr	-3.31625e-05
19	mfcc5	-3.31625e-05
20	mfcc4	-3.31625e-05
21	crest	-3.31625e-05
22	mfcc6	-3.31625e-05
23	mfcc1	-3.31625e-05
24	slope	-3.31625e-05
25	pow1	-3.31625e-05
26	spread	-3.31625e-05
27	pcile95	-3.31625e-05
28	centroid	-3.31625e-05
29	iqr	-3.31625e-05
30	pow3	-3.31625e-05
31	pitch	-3.31625e-05
32	mfcc3	-3.31625e-05
33	mfcc2	-3.31625e-05
34	flatness	-3.31625e-05
35	pow2	-3.31625e-05
36	dmfcc6	-3.31625e-05
37	power	-3.31625e-05
38	crst1	-3.31625e-05
39	dmfcc2	-3.31625e-05
40	pcile50	-0.000182408
41	pcile25	-0.00255674
42	clarity	-0.0500113

(b) SPC dataset

Rank	Feature	CondEnt
1	dmfcc8	0.052939
2	mfcc5	-0.0284791
3	dmfcc2	-0.0388293
4	dmfcc4	-0.00424729
5	dmfcc6	-0.00141392
6	slope	-0.0984349
7	dmfcc3	-0.162284
8	spread	-0.102299
9	crst1	0.000493247
10	dmfcc7	-8.3309e-05
11	centroid	-8.3309e-05
12	pow5	-8.3309e-05
13	pow2	-8.3309e-05
14	zcr	-8.3309e-05
15	mfcc7	-8.3309e-05
16	mfcc2	-8.3309e-05
17	crest	-8.3309e-05
18	crst4	-8.3309e-05
19	crst3	-8.3309e-05
20	pcile95	-8.3309e-05
21	pow1	-8.3309e-05
22	pow3	-8.3309e-05
23	mfcc1	-8.3309e-05
24	flux	-8.3309e-05
25	crst2	-8.3309e-05
26	dmfcc1	-8.3309e-05
27	pow4	-8.3309e-05
28	mfcc8	-8.3309e-05
29	tcrest	-8.3309e-05
30	mfcc6	-8.3309e-05
31	power	-8.3309e-05
32	dmfcc5	-8.3309e-05
33	mfcc3	-8.3309e-05
34	crst5	-8.3309e-05
35	mfcc4	-8.3309e-05
36	flatness	-0.000124966
37	pitch	-0.000291612
38	iqr	-0.000333278
39	pcile75	-0.000333278
40	pcile50	-0.00333779
41	pcile25	-0.0416762
42	clarity	-0.220772

(c) BBX dataset

Table C.1: Results of feature selection: voice timbre features ranked using floating selection/rejection algorithm with conditional entropy measure.

This standardisation of the marginal variables is closely related to the use of empirical copulas to study dependency between variables, see e.g. Nelsen [2006, Chapter 5], Diks and Panchenko [2008].

C.3 Results

Table C.1 shows the results of the information-theoretic feature selection carried out on the three voice timbre datasets. Agreement between the ranking in the three datasets is moderate – some commonalities can be observed by inspection, but the overall rank agreement is not statistically significant (Kendall's W=0.369, p=0.29, 41 d.f.). Notably, the rank ordering is very different from

Rank	Feature	CondEnt
1	flux	-3.75598e-05
2	mfcc4	-0.0130079
3	spread	-0.0357503
4	pow4	-0.0694354
5	mfcc3	-0.172405
6	power	-0.121682
7	zcr	-0.0933562
8	pow5	-0.139361
9	clarity	-0.0978034
10	pitch	-0.097602
11	mfcc2	-0.0668573
12	mfcc1	-0.0432822
13	pow3	-0.00915218
14	slope	-0.000816111
15	iqr	-3.31647e-05
16	pcile95	-3.31647e-05
17	flatness	-3.31647e-05
18	centroid	-3.31647e-05
19	pcile75	-3.31647e-05
20	pow2	-3.31647e-05
21	pow1	-3.31647e-05
22	pcile50	-9.94975e-05
23	pcile25	-0.00137727

(a) SNG dataset

Rank	Feature	CondEnt
1	mfcc2	-3.7668e-05
2	spread	-0.0150767
3	mfcc4	-0.0320311
4	flux	-0.0489897
5	pitch	-0.125682
6	power	-0.106014
7	mfcc3	-0.0719297
8	zcr	-0.0881291
9	clarity	-0.134593
10	pow2	-0.100906
11	slope	-0.0943874
12	pow4	-0.0462031
13	mfcc1	-0.00635225
14	pow5	-7.88219e-06
15	pow1	-3.30328e-05
16	pcile95	-3.30328e-05
17	pow3	-3.30328e-05
18	pcile75	-3.30328e-05
19	centroid	-3.30328e-05
20	flatness	-3.30328e-05
21	iqr	-3.30328e-05
22	pcile50	-0.000181694
23	pcile25	-0.00254673

(b) SPC dataset

Rank	Feature	CondEnt
1	power	0.0140867
2	mfcc4	-0.0396685
3	mfcc2	-0.129682
4	pow2	-0.115882
5	pitch	-0.0571967
6	zcr	-0.0731364
7	pcile50	-0.130079
8	flux	-0.354576
9	mfcc3	-8.18724e-05
10	mfcc1	-8.20182e-05
11	pow3	-8.20182e-05
12	pow1	-8.20182e-05
13	centroid	-8.3309e-05
14	slope	-8.3309e-05
15	pcile95	-8.3309e-05
16	spread	-8.3309e-05
17	pow5	-8.3309e-05
18	pow4	-8.3309e-05
19	flatness	-0.000124966
20	pcile75	-0.000333278
21	iqr	-0.000333278
22	pcile25	-0.0416762
23	clarity	-0.220772

(c) BBX dataset

Table C.2: Feature selection as in Table C.1 but using a reduced set of input features.

that produced in the stability and robustness rankings (Sections 3.3.1 and 3.3.2), and in fact to some extent it is reversed: the lowest-ranking features include the autocorrelation clarity and spectral percentiles, while the highest-ranking features include the ΔMFCCs, the spectral crests and MFCCs – in agreement with the observations made on the pairwise MI values.

Knowing that some features gave poor results in the robustness tests, we also performed the feature-selection experiment on a reduced feature set excluding the ΔMFCCs, crests and MFCCs 5–8. Results are shown in Table C.2 and again show only moderate agreement among datasets (Kendall's W=0.362, p=0.35, 22 d.f.).

In all these feature selection experiments the spectral percentiles and subband powers show some tendency to be rejected early, perhaps due to the information overlap with subband power as discussed above (Section 3.4.2). However, it is difficult to generalise over these results because of the amount of variation: for example *clarity* is the first to be rejected in all three of the full-set experiments, yet curiously is ranked quite highly in two of the reduced-set experiments.

C.4 Discussion

We note that the results of the present feature-selection appear to exhibit some tension with the robustness rankings reported in Sections 3.3.1 and 3.3.2, which told us the extent to which features contain what we take to be irrelevant information (e.g. due to noise). In light of this tension, it is important to recognise that the analysis presented in this Appendix may have difficulties distinguishing between relevant and irrelevant information: the analysis is unsupervised, meaning no ground truth of relevance is considered, and very few assumptions are made about the form of the data. Therefore it is difficult to be certain whether the independence results reflect the kind of independence which may be useful in constructing a multidimensional timbre space. The relative lack of consistency in the feature selection experiments shown in Tables C.1 and C.2 does not lead to a strong confidence in their utility.

Such is the challenge of feature selection in the absence of a ground truth such as classification labels. Others have attempted feature selection in such situations. For example Mitra et al. [2002] describe a method for feature selection based on clustering features according to a similarity measure (e.g. correlation or mutual information) and choosing features which best represent those clusters. Such an approach may hold promise, but we note that it has a strong dependence on the input feature set, in that consensus among features is the main metric: for example, if the input feature set contains a particular feature duplicated many times, this would "force" the algorithm of Mitra et al. [2002] to select it since it would appear to represent a consensus, whereas our approach based on unique information would tend to reject duplicate features at an early stage. We therefore say that feature selection without classification, for non-redundant feature subsets, is a subject for further exploration and development, and should be noise-robust as well as robust to initial conditions.

Appendix D

Timbre remapping and Self-Organising Maps

In Chapter 5 we developed techniques which can learn the structure (including nonlinearities) of separate timbre data distributions in a timbre space (where the data distributions may be of relatively low intrinsic dimensionality compared against the extrinsic dimensionality, i.e. that of the space), and can learn to project from one such distribution into another so as to retrieve synth control settings. In that chapter we presented a PCA-based nearest-neighbour method, and our novel regression tree method, both of which worked and have been used in user experiments and live performances. In this appendix we report our investigations in using the Self-Organising Map (SOM) [Kohonen, 2001] as a nonlinear mapping method for this purpose. The approach did not yield successful results; we consider the reasons for this. We first briefly introduce SOMs and explain their appeal in this context, before exploring the application to timbre remapping.

The SOM is a relatively simple type of neural network – in other words, a machine learning technique inspired by the observed behaviour of biological neurons, in which a collection of similar interconnected units (called neurons or nodes) can be trained to detect patterns or learn to model an input-output relationship. The SOM is self-organising in the sense that it learns a mapping in which the topology of the input data is reflected in the topology of the output – the nodes of its network become organised around the topology of the input data. The nodes of a SOM are typically arranged in a square or hexagonal grid of interconnections, and each node also stores a location in the input space. Each incoming training data point is associated with a node whose location is nearest to it; then the location of that node as well as of nodes in a small

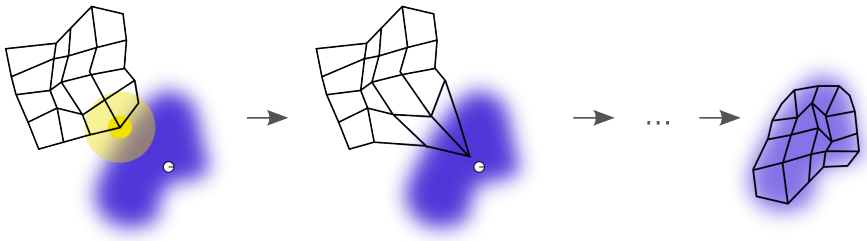


Figure D.1: An illustration of the training of a self-organising map. The blue blob represents the distribution of the training data, and the small white disc represents the current training sample drawn from that distribution. At first (left) the SOM nodes are arbitrarily positioned in the data space. The node nearest to the training node (highlighted in yellow) is selected, and is moved towards the training datum, as to a lesser extent are its neighbours on the grid. After many iterations the grid tends to approximate the data distribution (right).

neighbourhood on the grid is modified to move closer to the training data point (Figure D.1). This means that the node locations adapt to the distribution of the training data – but importantly, the effect of neighbourliness-on-the-grid means that the nodes do not tend to move to some arbitrary set of locations matching the distribution, but form a kind of manifold such that nodes which are neighbours on the grid tend to be close together in the data space. The SOM is therefore able to learn the nonlinear structure of a manifold embedded in a space, by fitting a set of discrete points (the node locations) which approximate that manifold. Any input point can be mapped to a coordinate on the SOM simply by finding the nearest node, and outputting the coordinate of that node within the SOM network.

Important in the use of SOMs is the choice of network topology for the node connections [Kohonen, 2001, Chapter 3]. The dimensionality of the network typically reflects the dimensionality of the manifold one hopes to recover and is typically quite low, e.g. a 1- 2- or 3-dimensional square grid of nodes. The SOM tends to adapt to the data even if the dimensionality is mismatched, but the resulting mappings may be less practically useful since they may contain arbitrary "twisting" of the map to fit the data. Consider for example a 1D SOM adapting to a square (2D) dataset: the SOM will typically result in a mapping which can be pictured as a piece of string arbitrarily curling around to fill a square piece of paper [Kiviluoto, 1996]. The output coordinate along this 1D SOM is likely to be not so informative about the intrinsic topology of the data as that which would come from a 2D SOM.

It is also worth noting that standard SOM algorithms are agnostic about the orientation of the map in the input space, meaning that any given mapping

will typically take one arbitrary orientation out of many possible. For example, the topology of a square grid of nodes has a symmetry which means that any one of its four corners might equally probably find itself fitting to a particular "corner" of the data distribution [Duda et al., 2000, p. 576]. This means that, although a trained SOM can map any input data point to a coordinate on the SOM network, the resulting coordinate could be dramatically different from that produced by a similar SOM trained on the same data (given some variation such as in the order of presentation of training data) [de Bodt et al., 2002]. It is quite normal for SOM grids to rotate during the training process [de Bodt et al., 2002], so even a preferred orientation given through setting the initial code coordinates may not strongly affect this. This indeterminacy of orientation will be an important consideration in our application.

We have not described all aspects of the SOM algorithm here – for example, details of the learning procedure, in which the size of the learning neighbourhood usually shrinks as learning progresses. The reader is referred to Kohonen [2001, esp. Chapter 3] for a thorough and accessible introduction. Next, we consider the application of SOMs to our remapping task.

Remapping using SOMs

To prepare a SOM-based timbre remapping system, we select a network topology and dimensionality, and then train one such SOM using timbre data for each sound source. For example, we might train one SOM using a generic voice dataset, and also train one SOM using a dataset from a synth which we wish to control vocally. In the latter case, we would also store the synth control parameters associated with each data point with its corresponding SOM node.

The SOM learning process tends to distribute nodes in a way which approximates the density of the data distribution (although often with slight "contraction" at the map edges), meaning that nodes are approximately equally likely to be selected by an input data point [Kohonen, 2001, Chapter 3]. This equalisation means that the space defined by the coordinates on the SOM grid corresponds rather well to the "well-covered space" which we seek. Figure 5.6b shows the SOM-based generation of the timbre space, illustrating that the SOM replaces both the dimension reduction and the nonlinear warping of the previous PCA-based approach (Figure 5.6a).

To actually perform the timbre remapping, we map a vocal timbre coordinate onto its coordinate in the voice-trained SOM. We then retrieve the synth controls associated with the analogous position in the synth timbre data, which is simply the node at the same coordinate but in the SOM trained on the synth timbre data (Figure D.2).

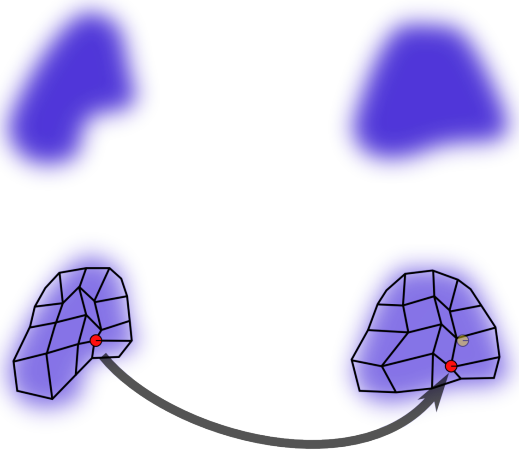


Figure D.2: Diagrammatic representation of SOM use in remapping. Upper: two distributions in the same space, with structural similarities as well as differences. Lower: SOM grids that might be fitted to these distributions. The arrow shows a remapping from one distribution to the other, achieved by matching coordinates on the SOM grid.

This process assumes a common orientation of the SOM grids, so that a coordinate in one can be unambiguously held to correspond to the same coordinate in the other (implicit in Figure D.2). As discussed, though, standard SOM algorithms do not guarantee any such orientation. To try and encourage a common alignment of SOM grids, one can initialise the node locations as a grid oriented along some common principal component axes, as well as reduce the amount by which the SOM nodes move towards training data points at each step in the learning process.

The SOM algorithm is therefore conceptually a good fit to the timbre remapping task: not only is it able to learn the shape of nonlinear timbre distributions in a feature space, but it yields a coordinate representation which enables a direct lookup of synth control parameters in one map, using coordinates retrieved from a different map.

Implementation

We implemented the system in SuperCollider 3.3, providing components for online SOM learning and for SOM lookup, with square-grid SOM topologies in 1D, 2D, 3D and 4D. Implementation of the SOM algorithms as components for SuperCollider allowed for their use as elements in an efficient real-time timbre remapping system, useful for prototyping as well as eventual use in performance.

The SOM components are publicly available.[1] In order to validate that the

[1] `http://sc3-plugins.sourceforge.net/`

SOM components were performing as intended, we designed some tests in which the SOMs were trained to fit specified shapes (e.g. a sinusoid in a 2D space, or a sinusoidally-undulating sheet in a 3D space). These were manually inspected to verify that the correct results were produced; some of these tests are available in the help files accompanying the published implementations.

Our preferred SOM dimensionality was 4D for the same reasons as in the PCA-based method (Section 5.1.3). However we also experimented with 2D and 3D mappings. In all cases we initialised the SOM before training to a grid of coordinates aligned with the leading components derived from a PCA analysis of a large human-voice dataset (the amalgamation of the speech, singing and beatbox datasets described in Section 3.3.2).

Results

The SOM-based timbre remapping never yielded satisfactory results in informal testing/development. Vocal timbral gestures tended to produce a rather arbitrary timbral output from the target synth, even when the task was simplified to a very basic synthesiser and a 2D map in a 2D feature space. See for example Figure D.3, which shows a rather typical example of a SOM trained on timbre data derived from the *gendy1* synth: the SOM manifold curls back on itself in various places and also interpenetrates itself.

Because of this, we did not bring the SOM-based timbre remapping to the point of formal evaluation. We will conclude this section by discussing the issues we encountered, which led us to leave this strand of development pending further work. The numerical evaluation in the later part of this chapter will therefore not feature a SOM-based technique.

Issues

From inspecting maps produced, we found that the main cause of this unsatisfactory performance was the tendency for maps to rotate and to develop twists/folds during training (e.g. Figure D.3). This could cause undesirable mappings such as an increase in vocal brightness causing a decrease in synth brightness. We tried to reduce these effects using the PCA initialisation of the SOM grids, by reducing the amount by which SOM nodes move towards training points, and by experimenting with different SOM dimensionalities and sizes (number of nodes). However in our tests there was no general setting which produced consistently useful mappings.

One might attempt to mitigate the effects of rotation during the map training, for example by including some global orientation constraint in the training algorithm. However, solving the rotational indeterminacy would not address

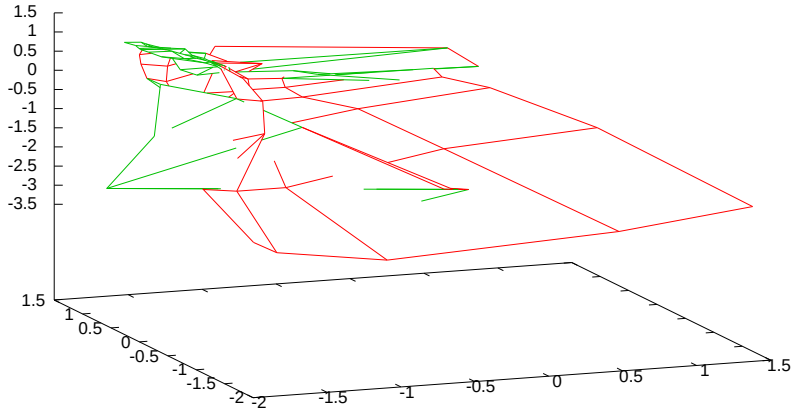


Figure D.3: An example of a SOM trained on synth timbre data, illustrating the twisting and folding that often occurred. Shown is a SOM with a 10-by-10 grid of nodes, trained using audio input from the *gendy1* synth analysed with 10 timbre features. The visualisation shows the SOM node locations in the first 3 principal components of the 10-dimensional timbre space.

the whole problem, since the tendency for twists/folds to appear in the map seems equally problematic.

The appearance of twists/folds in SOMs can be caused by a poor fit of the map topology to the data. One cause could be an inappropriate choice of the map dimensionality; more generally the distribution of timbre data in the high-dimensional space could take some unusual shape which is not well approximated by a regular grid of nodes. Kohonen [2001, Chapter 5] considers some variants on the SOM algorithm, including those with arbitrary rather than regular network topologies, and those whose network topology can change (e.g. adding/removing nodes, making/breaking connections between neighbour nodes). Taken to its extreme this adaptive approach to the network topology is represented by the neural gas [Martinetz et al., 1993] and growing neural gas [Martinetz et al., 1993] algorithms, which have no topology at initialisation and learn it purely from data. However, applying such schemes to our timbre remapping task presents a major issue: if the map topology is learnt or adapted for each dataset, how can we map from one to another (e.g. voice to synth) given that there will typically be no inherent correspondence between nodes in different maps?

Such a SOM-like algorithm with adaptive topology, or a SOM with added orientation constraints, could be the subject of future work in timbre remapping techniques; in the present work we do not pursue this. From our investigations, we believe that the issues we encountered are general issues with using SOMs

for timbre remapping, although future work could reveal a variant of the SOM algorithm which better suits the task.

Appendix E

Discourse analysis data excerpts

This Appendix lists analyst's notes for each of the four solo session participants in the evaluation study of Chapter 6. The full transcriptions and data tables are too large to include here, but are available at
`http://www.elec.qmul.ac.uk/digitalmusic/papers/2008/Stowell08ijhcs-data/`

The four participants are herein labelled by their codings P20, P21, P23, P24. In the chapter they are labelled differently: User 1 (P24), User 2 (P21), User 3 (P20), User 4 (P23).

These notes represent an intermediate stage of the analysis, after the transcription and itemisation, when the analyst is extracting the main objects, actors and ways of speaking. Concept maps were also sketched on paper as part of the process but are not included here. The final narrative representation of the results is given in the chapter text (Section 6.3).

Analysis coding

Identifying context of interviews:

Xi - interview follows mode X session

XYi - interview follows mode X session and mode Y session

Yi - interview follows mode Y session

YXi - interview follows mode Y session and mode X session

Identifying referents:

Xf - mode X session/system during free exploration

Xg - mode X session/system during guided exploration

Xo - mode X session/system during BOTH free & guided

Yf - mode Y session/system during free exploration

Yg - mode Y session/system during guided exploration

Yo - mode Y session/system during BOTH free & guided

Participant P20

i. Systematically itemise objects

Most common objects:

- P20 (33)
- ((system)) "this sort of program" (13)
- ((output sound)) "the effect" (9)

ii. Objects are organised by ways of speaking

Notes on different ways of speaking:

Sounds P20 was doing → sounds it was doing in response

vs

P20 pushing air into microphone

Effect that can be applied in realtime or afterwards

vs

Something that "responds to" what you do (may have been prompted by me? certainly described it more as an effect at the start)

iii. Systematically itemise the actors (who are a subset of the objects)

Main actors:

* P20 (26)

- listens to a lot of aphex twin and squarepusher, goes for complicated beats, doesn't do a lot of sequencing

- would definitely have to figure out exactly (what to do with system), wouldn't do what they would normally do into a mic with no effects; was definitely doing very different things from what they normally do in beatboxing

- heard what self was doing in real time rather than the new effect thing; found it a lot easier with it on bypass

- would probably cringe if heard the Yg recordings without the editing on them

- was trying to recreate sounds; started noticing from having to experiment how ((examples)) were made; got a better understanding; got a few ideas from Yg for sounds they should have done in Yf; should have used the record and playback more in Yf

- rarely uses "ahhhh" side of it, generally doesn't hum, got to use it in Xo, found it hard to get around

- got ideas after Yo, for YXo, and carried a lot through; could try crazy sounds when experimenter was out of the room; if given a longer time would have tried melodies and other things anyway

- was quite happy with some of the sounds happening (YXf)

- could use fast notes on the high end of the scale, although it would sort of be carried off in distortion

* ((system)) (5 in Yi) works in a certain way, reminds P20 of blue-glitch, picks up a lot of breath and background sounds, picks up a much cleaner sound with the more scat-singing side of bbxing

* ((output sound)) (3) depends on the force and the volume you're doing it, and is a lot more melodic in Xo

* the examples (3) influence ease of producing certain sounds, and give P20 ideas of things to do

* ((Yg) (2) helps P20 to understand how ((system)) works

* ((experimenter)) (2) went out of the room, and may or may not have heard of blue-glitch

Participant P21

i. Systematically itemise objects

Most common objects:

* P21 (60) [see actor section for descriptions]

* ((P21's vocal sound)) (14) P21 puts it in and it comes out strangely

* ((example sounds)) (10) someone originally made them in a certain way, P21 tried to work out how and learned how to do them, couldn't do all of them

* the other person (10) made the ((audio examples)), P21 is curious to see how they did it

* ((Yo)) (11) is obviously a slightly different setting than Xo, sounds a bit more distorted and better, is a bit more fun, gives a bit more control, is a bit

more interesting

* ((system)) (8) makes strange noises, doesn't sound very pleasing, sometimes beeps and sometimes doesn't

* sound on the thing (8) were made by P21's sounds, but sounded very/totally different, got distorted

* ((general person)) (7) puts in sounds and they come out different, is made to keep time with self

* ((an audio example)) (5) sounded like a human hadn't made it, was so distorted that P21 couldn't work it out, P21 is curious to see how the other person did this

* the initial noise (4) P21 was trying to work out what they were

ii. Objects are organised by ways of speaking

Notes on different ways of speaking:

No tensions evident here. Clear conceptual model of someone originally making the sounds, and P21's aim to work out how they did it in order to record their own.

The system sounds bad, nevertheless quite insistent on the difference between Y and X in that Y is more fun and interesting, despite sounding distorted and being difficult to replicate the examples. This might seem like a tension, but I'm quite sure it's only a tension to me; P21 very comfortable with the coexistence, no hedging of it or avoidance.

iii. Systematically itemise the actors (who are a subset of the objects)

Main actors:

* P21 (42):

- never uses synthesisers, tries to keep it all natural, would get lost ((making music with computers)), finds it all a bit techno

- tried to imagine and work out what the other person was doing, could on certain snare sounds pick up what the original noise was, could tell where inward K handclaps were, is curious to see how the other person did it, is not gonna be able to record the same thing

- putting sounds into the mic, trying out beats, could pick up on trying to make the timing better, suddenly realised hadn't tried doing any clicks, did the clicks, was doing a full beat pattern

- had more fun with ((Yo))

* the other person)) (9) - does things to create audio examples, P21 is trying to work out what they did, would be curious to see how they did it, in some cases it's gonna take a long time to work it out

* ((general person)) (5) - puts in a sound and it comes out quite different, can come up with some slightly more funky noises in Yo, doesn't get caught up in how good it sounds; can make sounds with or without a synthesiser; if they aren't gonna copy the examples what are they gonna do?

* ((system)) (4) - really made a strange noise, makes funny noises, sometimes beeps and sometimes doesn't

Participant P23

i. Systematically itemise objects

Most common objects:

* P23 (25)
* the sounds ((it made)) (16)
* ((Yo)) (13; 10 in Yi, 3 in YXi)
* ((system)) (12)
* ((sound P23 makes)) (4; 2 in each)
* humming (4; 1 in Yi, 3 in YXi)
* ((sound system makes)) (3; 2 in Yi, 1 in YXi)

ii. Objects are organised by ways of speaking

Notes on different ways of speaking:

randomness vs accurately following: Yo could be a bit random, Xo was more random; synth sounds followed the humming accurately.

iii. Systematically itemise the actors (who are a subset of the objects)

Main actors:

* P23 (23)

- trying stuff out, wondering how to do certain things, discovering ((the system)), getting confused occasionally

- not liking hearing self played back

* ((system)) (5) is epic, has broad ability, sounds quite random, sounds like a synth machine with sound effects

* sounds ((made by system)) (5) are a bit random, switch around, don't always work

* ((sound P23 makes)) (2) causes ("makes") the system produce a certain sound, and this is different in mode Y as in mode X

* ((general person)) (2)

Participant P24

i. Systematically itemise objects

Most common objects:

* P24 (21)
* ((Yg)) or ((Yo)) (10)
* ((system)) (10)
* ((Xo)) (7)
* ((examples)) the original sounds (6)
* ((general person)) (6)
* sounds ((made by system)) (6)

ii. Objects are organised by ways of speaking

Notes on different ways of speaking:

Only really one way of speaking here: trying to make noises, work out what's going on, trying to match examples.

iii. Systematically itemise the actors (who are a subset of the objects)

Main actors:

* P24 (19) - was trying to work out what, did a standard beat, wasn't expecting those noises, thought own things were gonna be horrific; could do one example but not spot on, couldn't work out how to do that chimey sound; found the sounds a bit easier to recreate in Yg than Xg; liked Y sound better; would like to play around with it all a bit more

* ((general person)) (6) makes certain noises, ((system)) then makes strange noises which couldn't do with own mouth; has to learn to get a grip of this process

* ((system)) (3) makes noises/sounds; changes the tone of the noises you're saying

Bibliography

C. R. Abril. I have a voice but I just can't sing: a narrative investigation of singing and social anxiety. *Music Education Research*, 9(1):1–15, Mar 2007. doi: 10.1080/14613800601127494.

M. D. Alder, R. Togneri, and Y. Attikiouzel. Dimension of the speech space. *Communications, Speech and Vision, IEE Proceedings I*, 138(3):207–214, 1991.

J. E. Angus. The probability integral transform and related results. *SIAM review*, 36(4):652–654, 1994.

ANSI. *Acoustical Terminology*. Number S1.1-1960. American National Standards Institute, New York, 1960.

C. Antaki, M. Billig, D. Edwards, and J. Potter. Discourse analysis means doing analysis: a critique of six analytic shortcomings. *Discourse Analysis Online*, 1(1), 2003.

C. Arndt. *Information Measures*. Springer, 2001.

M. Artac, M. Jogan, and A. Leonardis. Incremental PCA for on-line visual learning and recognition. In *Proc. International Conference on Pattern Recognition (ICPR 2002)*, volume 3, pages 781–784, 2002. doi: 10.1109/ICPR.2002.1048133.

J.-J. Aucouturier and F. Pachet. Improving timbre similarity: how high's the sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

J.-J. Aucouturier and F. Pachet. Jamming with plunderphonics: interactive concatenative synthesis of music. *Journal of New Music Research*, 35(1): 35–50, Mar 2006.

P. Banister, E. Burman, I. Parker, M. Taylor, and C. Tindall. *Qualitative Methods in Psychology: A Research Guide*. Open University Press, Buckingham, 1994. ISBN 978-0335191819.

W. Barendregt, M. M. Bekker, D. G. Bouwhuis, and E. Baauw. Identifying usability and fun problems in a computer game during first use and after some practice. *International Journal of Human-Computer Studies*, 64(9): 830–846, 2006. doi: 10.1016/j.ijhcs.2006.03.004.

M. Barthet. From clarinet control to timbre perception. *Acta Acustica*, accepted.

J. W. Beauchamp. Synthesis by spectral amplitude and "brightness" matching of analyzed musical instrument tones. *Journal of the Acoustic Engineering Society*, 30(6):396–406, 1982.

J. Beirlant, E. J. Dudewicz, L. Györfi, and E. C. van der Meulen. Nonparametric entropy estimation: an overview. *International Journal of Mathematical and Statistical Sciences*, 6:17–39, 1997.

J. L. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975. doi: 10.1145/361002.361007.

A. Beygelzimer, S. Kakade, and J. Langford. Cover trees for nearest neighbor. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 97–104, New York, 2006. ACM Press. doi: 10.1145/1143844.1143857.

J. A. Bilmes. What HMMs can do. *IEICE Transactions on Information and Systems*, E89-D(3):869–891, 2006. doi: 10.1093/ietisy/e89-d.3.869.

J. A. Bilmes, J. Malkin, X. Li, S. Harada, K. Kilanski, K. Kirchhoff, R. Wright, A. Subramanya, J. A. Landay, P. Dowden, and H. Chizeck. The Vocal Joystick. In *Proc. IEEE Intl. Conf. on Acoustic, Speech and Signal Processing*, volume 1, pages 625–628, May 2006. doi: 10.1109/ICASSP.2006.1660098.

G. Bloothooft, E. Bringmann, M. van Cappellen, J. B. van Luipen, and K. P. Thomassen. Acoustics and perception of overtone singing. *Journal of the Acoustical Society of America*, 92(4):1827–1836, 1992. doi: 10.1121/1.403839.

J. O. Borchers. A pattern approach to interaction design. *AI & Society*, 15(4): 359–376, 2001. doi: 10.1007/BF01206115.

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth Statistics/Probability Series. Wadsworth Inc, 1984.

C. E. Brodley and P. E. Utgoff. Multivariate decision trees. *Machine Learning*, 19(1):45–77, 1995. doi: 10.1023/A:1022607123649.

P. M. Brossier. *Automatic Annotation of Musical Audio for Interactive Applications*. PhD thesis, Dept of Electronic Engineering, Queen Mary University of London, London, UK, Mar 2007. URL `http://aubio.piem.org/phdthesis/`.

N. Bryan-Kinns, P. G. T. Healey, and J. Leach. Exploring mutual engagement in creative collaborations. In *Proceedings of the 6th ACM SIGCHI Conference on Creativity and Cognition*, pages 223–232, New York, 2007. ACM Press. doi: 10.1145/1254960.1254991.

J. A. Burgoyne and S. McAdams. Non-linear scaling techniques for uncovering the perceptual dimensions of timbre. In *Proceedings of the International Computer Music Conference (ICMC'07)*, volume 1, pages 73–76, Copenhagen, Denmark, Aug 2007.

J. A. Burgoyne and S. McAdams. A meta-analysis of timbre perception using nonlinear extensions to CLASCAL. In R. Kronland-Martinet, S. Ystad, and K. Jensen, editors, *Sense of Sounds*, volume 4969/2009 of *Lecture Notes in Computer Science*, chapter 12, pages 181–202. Springer, Berlin, 2009. doi: 10.1007/978-3-540-85035-9_12.

M. J. Butler. *Unlocking the Groove: Rhythm, Meter, and Musical Design in Electronic Dance Music*. Indiana University Press, Bloomington, 2006.

W. Buxton and R. Sniderman. Iteration in the design of the human-computer interface. In *Proceedings of the 13th Annual Meeting, Human Factors Association of Canada*, pages 72–81, 1980.

A. Caclin, S. McAdams, B. K. Smith, and S. Winsberg. Acoustic correlates of timbre space dimensions: a confirmatory study using synthetic tones. *Journal of the Acoustical Society of America*, 118(1):471–482, 2005. doi: 10.1121/1.1929229.

A. Caclin, E. Brattico, M. Tervaniemi, R. Näätänen, D. Morlet, M.-H. Giard, and S. McAdams. Separate neural processing of timbre dimensions in auditory sensory memory. *Journal of Cognitive Neuroscience*, 18(12):1959–1972, Dec 2006. ISSN 0898-929X. doi: 10.1162/jocn.2006.18.12.1959.

A. Caclin, M.-H. Giard, B. K. Smith, and S. McAdams. Interactive processing of timbre dimensions: a Garner interference study. *Brain Research*, 1138: 159–170, 2007. doi: 10.1016/j.brainres.2006.12.065.

P. B. Callahan. Optimal parallel all-nearest-neighbors using the well-separated pair decomposition. In *Proceedings of the 34th IEEE Symposium on Foun-*

dations of Computer Science, pages 332–340, 1993. doi: 10.1109/SFCS.1993.366854.

S. K. Card, W. K. English, and B. J. Burr. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics*, 21(8):601–613, 1978. doi: 10.1080/00140137808931762.

S. K. Card, T. P. Moran, and A. Newell. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1983.

M. Casey. MPEG-7 sound-recognition tools. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):737–747, 2001. doi: 10.1109/76.927433.

C.-I. Chang, Y. Du, J. Wang, S.-M. Guo, and P. D. Thouin. Survey and comparative analysis of entropy and relative entropy thresholding techniques. *IEE Proceedings in Vision, Image and Signal Processing*, 153(6):837–850, Dec 2006. ISSN 1350-245X. doi: 10.1049/ip-vis:20050032.

E. Chávez, G. Navarro, R. Baeza-Yates, and J. L. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33:273–321, 2001. doi: 10.1145/502807.502808.

F. Chen, D. Lambert, and J. C. Pinheiro. Incremental quantile estimation for massive tracking. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 516–522. ACM New York, NY, USA, 2000. doi: 10.1145/347090.347195.

J. Chen, Y. Huang, Q. Li, and K. K. Paliwal. Recognition of noisy speech using dynamic spectral subband centroids. *IEEE Signal Processing Letters*, 11(2):258–261, 2004. doi: 10.1109/LSP.2003.821689.

J. Chowning. The synthesis of complex audio spectra by means of frequency modulation. *Journal of the Audio Engineering Society*, 21(7):526–534, 1973.

J. T. Chu. On the distribution of the sample median. *Annals of Mathematical Statistics*, 26(1):112–16, 1955.

J. E. Clark and C. Yallop. *An Introduction to Phonetics and Phonology*. Blackwell Textbooks in Linguistics. Blackwell, Oxford, 2nd edition, 1995.

N. Collins. On onsets on-the-fly: real-time event segmentation and categorisation as a compositional effect. In *Proceedings of Sound and Music Computing*, pages 20–22, Oct 2004.

N. Collins. *Towards Autonomous Agents for Live Computer Music: Realtime Machine Listening and Interactive Music Systems*. PhD thesis, University of Cambridge, 2006. URL `http://www.cogs.susx.ac.uk/users/nc81/thesis.html`.

N. Collins and J. d'Escriván, editors. *The Cambridge Companion to Electronic Music*. Cambridge University Press, 2007. ISBN 9780521688659.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience New York, 2nd edition, 2006.

M. Cross. The Zen of Screaming: Vocal instruction for a new breed. DVD, Warner Brothers, USA, 2007.

G. A. Darbellay and I. Vajda. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, 45(4):1315–1321, 1999. doi: 10.1109/18.761290.

M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1(3):131–156, 1997.

M. Davy, F. Desobry, A. Gretton, and C. Doncarli. An online Support Vector Machine for abnormal events detection. *Signal Processing*, 86(8):2009–2025, Aug 2006. doi: 10.1016/j.sigpro.2005.09.027.

E. de Bodt, M. Cottrell, and M. Verleysen. Statistical tools to assess the reliability of self-organizing maps. *Neural Networks*, 15(8-9):967–978, 2002. doi: 10.1016/S0893-6080(02)00071-0.

A. de Cheveigné and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111: 1917–1930, 2002. doi: 10.1121/1.1458024.

G. De Poli. Methodologies for expressiveness modelling of and for music performance. *Journal of New Music Research*, 33(3):189–202, 2004. doi: 10.1080/0929821042000317796.

G. De Poli and P. Prandoni. Sonological models for timbre characterization. *Journal of New Music Research*, 26(2):170–197, 1997. doi: 10.1080/09298219708570724.

B. Denby, T. Schultz, K. Honda, T. Hueber, J. M. Gilbert, and J. S. Brumberg. Silent speech interfaces. *Speech Communication*, 52(4):270–287, Apr 2010. doi: 10.1016/j.specom.2009.08.002.

K. Dickinson. 'Believe'? Vocoders, digitalised female identity and camp. *Popular Music*, 20(3):333–347, 2001. doi: 10.1017/S0261143001001532.

C. Diks and V. Panchenko. Rank-based entropy tests for serial independence. *Studies in Nonlinear Dynamics & Econometrics*, 12(1), 2008. ISSN 1558-3708.

C. Dobrian and D. Koppelman. The 'E' in NIME: musical expression with new computer interfaces. In *Proceedings of New Interfaces for Musical Expression (NIME)*, pages 277–282. IRCAM, Centre Pompidou Paris, France, 2006.

P. Domingos and M. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29(2–3):103–130, 1997. doi: 10.1023/A:1007413511361.

V. Duangudom and D. V. Anderson. Using auditory saliency to understand complex scenes. In *Proceedings of Eusipco 2007 (15th European Signal Processing Conference)*, pages 1206–1210, 2007.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, 2nd edition, 2000.

H. Edelsbrunner. *Algorithms in Computational Geometry*, volume 10 of *EATCS Monographs on Theoretical Computer Science*. Springer, Berlin, 1987. ISBN 978-3-540-13722-1.

K. A. Ericsson and H. A. Simon. *Protocol Analysis: Verbal Reports as Data*. MIT Press, Cambridge, MA, 1984.

L. Fabig and J. Janer. Transforming singing voice expression – the sweetness effect. In *Proceedings of the 7th International Conference on Digital Audio Effects (DAFx)*, Naples, Italy, 2004.

Z. Fang, Z. Guoliang, and S. Zhanjiang. Comparison of different implementations of MFCC. *Journal of Computer Science and Technology*, 16(6):582–589, 2001. ISSN 1000-9000. doi: 10.1007/BF02943243.

R. M. Fano. *Transmission of Information*, pages 57–59. MIT Press, Cambridge, MA, 1961.

S. Feld. Pygmy POP. A genealogy of schizophonic mimesis. *Yearbook for Traditional Music*, 28:1–35, 1996.

S. Fels. Designing for intimacy: creating new interfaces for musical expression. *Proceedings of the IEEE*, 92(4):672–685, 2004. doi: 10.1109/JPROC.2004.825887.

A. Figueiredo. Comparison of tests of uniformity defined on the hypersphere. *Statistics and Probability Letters*, 77(3):329–334, 2007. doi: 10.1016/j.spl.2006.07.012.

F. Fontana. Preserving the structure of the Moog VCF in the digital domain. In *Proceedings of the International Computer Music Conference (ICMC'07)*, volume 1, pages 291–294, Copenhagen, Denmark, Aug 2007.

J. Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1): 2–10, 1999. doi: 10.1007/s005300050106.

A. M. Fraser. Information and entropy in strange attractors. *IEEE Transactions on Information Theory*, 35(2):245–262, Mar 1989. ISSN 0018-9448. doi: 10.1109/18.32121.

C. J. Frisbie. Anthropological and ethnomusicological implications of a comparative analysis of Bushmen and African Pygmy music. *Ethnology*, 10(3): 265–290, 1971.

D. B. Fry. *The Physics of Speech*. Cambridge Textbooks in Linguistics. Cambridge University Press, 1996/1979.

O. Fujimura and D. Erickson. Acoustic phonetics. In W. J. Hardcastle and J. Laver, editors, *The Handbook of Phonetic Sciences*, Blackwell Handbooks in Linguistics, pages 65–115. Blackwell, 1999.

L. Fuks. *From Air to Music: Acoustical, Physiological and Perceptual Aspects of Reed Wind Instrument Playing and Vocal-Ventricular Fold Phonation*. PhD thesis, KTH, Stockholm, Sweden, 1998. URL `http://www.speech.kth.se/music/publications/leofuks/thesis/contents.html`.

R. Fukui. *TIPA Manual*, 2004. URL `http://www.ctan.org/tex-archive/fonts/tipa/tipaman.pdf`.

G. L. Gaile and J. E. Burt. *Directional Statistics*, volume 25 of *Concepts and Techniques in Modern Geography*. Geo Abstracts Ltd., 1980.

J. Gama. Functional trees. *Machine Learning*, 55(3):219–250, 2004. doi: 10.1023/B:MACH.0000027782.67192.13.

T. Ganchev, N. Fakotakis, and G. Kokkinakis. Comparative evaluation of various MFCC implementations on the speaker verification task. In *Proceedings of the 10th International Conference on Speech and Computer (SPECOM 2005)*, volume 1, pages 191–194, 2005.

M. Garcia. Observations on the human voice. *Proceedings of the Royal Society of London*, 7:399–410, 1854.

P. E. Garner and D. M. Howard. Real-time display of voice source characteristics. *Logopedics Phoniatrics Vocology*, 24(1):19–25, 1999. doi: 10.1080/140154399434526.

General Instrument. *GI AY-3-8910 Programmable Sound Generator datasheet*, 1979.

D. Gerhard. Pitch extraction and fundamental frequency: history and current techniques. Technical Report TR-CS 2003-06, Dept. of Computer Science, University of Regina, 2003. URL `http://www.cs.uregina.ca/Research/Techreports/2003-06.pdf`.

B. R. Gerratt and J. Kreiman. Toward a taxonomy of nonmodal phonation. *Journal of Phonetics*, 29(4):365–381, Oct 2001. doi: 10.1006/jpho.2001.0149.

O. K. Gillet and G. Richard. Automatic labelling of tabla signals. In *Proceedings of the 4th ISMIR Conference*, 2003.

R. Göb, C. McCollin, and M. F. Ramalhoto. Ordinal methodology in the analysis of Likert scales. *Quality and Quantity*, 41(5):601–626, 2007. doi: 10.1007/s11135-007-9089-z.

S. Grant, T. Aitchison, E. Henderson, J. Christie, S. Zare, J. McMurray, and H. Dargie. A comparison of the reproducibility and the sensitivity to change of visual analogue scales, Borg scales, and Likert scales in normal subjects during submaximal exercise. *Chest*, 116(5):1208–1217, 1999. doi: 10.1378/chest.116.5.1208.

J. M. Grey. Multidimensional perceptual scaling of musical timbres. *Journal of the Acoustical Society of America*, 61(5):1270–1277, 1977. doi: 10.1121/1.381428.

J. M. Grey. Timbre discrimination in musical patterns. *Journal of the Acoustical Society of America*, 64(2):467–472, 1978. doi: 10.1121/1.382018.

J. M. Grey and J. W. Gordon. Perceptual effects of spectral modifications on musical timbres. *Journal of the Acoustical Society of America*, 63(5):1493–1500, 1978. doi: 10.1121/1.381843.

L. Gu and K. Rose. Perceptual harmonic cepstral coefficients for speech recognition in noisy environment. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01)*, 2001.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, Mar 2003.

P. Hämäläinen, T. Mäki-Patola, V. Pulkki, and M. Airas. Musical computer games played by singing. In *Proceedings of the 7th International Conference on Digital Audio Effects (DAFx'04)*, Naples, 2004.

S. Handel and M. L. Erickson. Sound source identification: the possible role of timbre transformations. *Music Perception*, 21(4):587–610, 2004. doi: 10.1525/mp.2004.21.4.587.

H. M. Hanson. *Glottal Characteristics of Female Speakers*. PhD thesis, Division of Applied Sciences, Harvard University, 1995. URL `http://hdl.handle.net/1721.1/22393`.

S. Harada, J. O. Wobbrock, J. Malkin, J. A. Bilmes, and J. A. Landay. Longitudinal study of people learning to use continuous voice-based cursor control. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, pages 347–356. ACM New York, NY, USA, 2009. doi: 10.1145/1518701.1518757.

S. Harrison, D. Tatar, and P. Sengers. The three paradigms of HCI. In *SIGCHI Conference on Human Factors in Computing Systems*, San Jose, California, USA, 2007.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2001.

S. Hawkins and J. Midgley. Formant frequencies of RP monophthongs in four age groups of speakers. *Journal of the International Phonetic Association*, 35 (2):183–199, 2005. doi: 10.1017/S0025100305002124.

A. Hazan. Towards automatic transcription of expressive oral percussive performances. In *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI '05)*, pages 296–298, New York, NY, USA, 2005a. ACM Press. ISBN 1-58113-894-6. doi: 10.1145/1040830.1040904.

A. Hazan. Billaboop: real-time voice-driven drum generator. In *Proceedings of the 118th Audio Engineering Society Convention (AES 118)*, number 6411, May 2005b.

P. G. T. Healey, J. Leach, and N. Bryan-Kinns. Inter-play: understanding group music improvisation as a form of everyday interaction. In *Proceedings of Less*

is More — Simple Computing in an Age of Complexity,, Microsoft Research Cambridge, 2005.

E. H. Margulis and W. H. Levine. Timbre priming effects and expectation in melody. *Journal of New Music Research*, 35(2):175–182, 2006. doi: 10.1080/09298210600835042.

N. Henrich. Mirroring the voice from Garcia to the present day: some insights into singing voice registers. *Logopedics Phoniatrics Vocology*, 31(1):3–14, 2006. doi: 10.1080/14015430500344844.

J. Herre, E. Allamanche, and O. Hellmuth. Robust matching of audio signals using spectral flatness features. In *Proceedings of the Workshop on the Applications of Signal Processing to Audio and Acoustics (WASPAA-2001)*, pages 127–130, 2001.

P. Herrera, A. Yeterian, and F. Gouyon. Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques. In *Proceedings of the 2nd International Conference on Music and Artificial Intelligence*, pages 69–80, 2002. doi: 10.1007/3-540-45722-4_8.

P. D. Hill, B. W. V. Lee, J. E. Osborne, and E. Z. Osman. Palatal snoring identified by acoustic crest factor analysis. *Physiological Measurement*, 20(2):167–174, 1999. doi: 10.1088/0967-3334/20/2/306.

C. A. R. Hoare. Algorithm 63 (partition) and algorithm 65 (find). *Communications of the ACM*, 4:321–322, 1961. doi: 10.1145/366622.366642.

M. Hoffman and P. R. Cook. The featsynth framework for feature-based synthesis: design and applications. In *Proceedings of the International Computer Music Conference (ICMC'07)*, volume 2, pages 184–187, Copenhagen, Denmark, Aug 2007.

D. Hosseinzadeh and S. Krishnan. On the use of complementary spectral features for speaker recognition. *EURASIP Journal on Advances in Signal Processing*, 2008 (Article ID 258184, 10 pages), 2008. doi: 10.1155/2008/258184.

A. J. M. Houtsma and J. Smurzynski. Pitch identification and discrimination for complex tones with many harmonics. *Journal of the Acoustical Society of America*, 87(1):304, 1990. doi: 10.1121/1.399297.

D. M. Howard. The human singing voice. In P. Day, editor, *Killers in the Brain: Essays in Science and Technology from the Royal Institution*, pages 113–134. Oxford University Press, USA, 1999.

D. M. Howard and A. M. Tyrrell. Psychoacoustically informed spectrography and timbre. *Organised Sound*, 2(2):65–76, 1997. doi: 10.1017/S1355771897009011.

A. Hunt and M. M. Wanderley. Mapping performer parameters to synthesis engines. *Organised Sound*, 7(2):97–108, 2002. doi: 10.1017/S1355771802002030.

International Phonetic Association. *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press, 1999. ISBN 9780521637510. doi: 10.2277/0521637511.

International Telecommunication Union. Method for the subjective assessment of intermediate quality levels of coding systems (MUSHRA). Technical Report ITU-R BS.1534-1, International Telecommunication Union, 2003. URL `http://www.itu.int/rec/R-REC-BS.1534/en`.

P. Iverson and C. L. Krumhansl. Isolating the dynamic attributes of musical timbre. *Journal of the Acoustical Society of America*, 94(5):2595–2603, 1993. doi: 10.1121/1.407371.

A. Jain and D. Zongker. Feature selection: evaluation, application, and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158, Feb 1997. ISSN 0162-8828. doi: 10.1109/34.574797.

J. Janer. *Singing-driven Interfaces for Sound Synthesizers*. PhD thesis, Music Technology Group, Universitat Pompeu Fabra, Barcelona, 2008. URL `http://www.mtg.upf.edu/~jjaner/phd/Tesi_jjaner_online.pdf`.

J. Janer and M. de Boer. Extending voice-driven synthesis to audio mosaicing. In *Proceedings of the 5th Sound and Music Computing Conference (SMC)*, volume 4, Berlin, 2008.

T. Jehan. Event-synchronous music analysis/synthesis. In *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFx-04)*, pages 361–366, Naples, Italy, 2004.

K. Jensen. *Timbre models of musical sounds*. PhD thesis, University of Copenhagen, Denmark, 1999. URL `http://www.aaue.dk/~krist/TMoMS.pdf`.

K. Jensen. The timbre model (abstract). *Journal of the Acoustical Society of America*, 112(5):2238–2238, 2002.

J. B. Johnson. Thermal agitation of electricity in conductors. *Physical Review*, 32(1):97–109, 1928. doi: 10.1103/PhysRev.32.97.

A. Kapur, M. Benning, and G. Tzanetakis. Query-by-beat-boxing: music retrieval for the DJ. In *Proceedings of the Fifth International Conference on Music Information Retrieval*, pages 170–177, 2004.

W. Kelly. The adaptability of karaoke in the United Kingdom. In *Karaoke Around the World: Global Technology, Local Singing*, pages 83–101. Routledge, 1998.

M. G. Kendall and B. B. Smith. The problem of m rankings. *The Annals of Mathematical Statistics*, 10(3):275–287, Sep 1939.

C. Kiefer, N. Collins, and G. Fitzpatrick. HCI methodology for evaluating musical controllers: a case study. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, pages 87–90, 2008.

Y. E. Kim. *Singing Voice Analysis/Synthesis*. PhD thesis, Massachusetts Institute of Technology, 2003.

K. Kiviluoto. Topology preservation in Self-Organizing Maps. In *Proceedings of the International Conference on Neural Networks*, volume 1, pages 294–299, 1996. doi: 10.1109/ICNN.1996.548907.

M. Kob. Analysis and modelling of overtone singing in the Sygyt style. *Applied Acoustics*, 65(12):1249–1259, 2004. doi: 10.1016/j.apacoust.2004.04.010.

T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990. doi: 10.1109/5.58325.

T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, 3rd edition, 2001.

B. Kotnik, D. Vlaj, and B. Horvat. Efficient noise robust feature extraction algorithms for Distributed Speech Recognition (DSR) systems. *International Journal of Speech Technology*, 6(3):205–219, 2003. doi: 10.1023/A:1023410018862.

G. Kovačić, P. Boersma, and H. Domitrović. Long-term average spectra in professional folk singing voices: a comparison of the Klapa and Dozivački styles. In *Proceedings of the Institute of Phonetic Sciences, University of Amsterdam*, volume 25, pages 53–64, 2003.

J. Kreiman, B. R. Gerratt, K. Precoda, and G. S. Berke. Perception of supraperiodic voices (abstract). *Journal of the Acoustical Society of America*, 93(4):2337, Apr 1993. doi: 10.1121/1.406275.

J. Kreiman, D. Vanlancker-Sidtis, and B. R. Gerratt. Defining and measuring voice quality. In *Proceedings of From Sound To Sense: 50+ Years of Discoveries in Speech Communication*, pages 163–168. MIT, Jun 2004.

J. Krimphoff, S. McAdams, and S. Winsberg. Caractérisation du timbre des sons complexes. II. Analyses acoustiques et quantification psychophysique. *Journal de Physique IV*, 4(5):5, 1994.

C. L. Krumhansl. Why is musical timbre so hard to understand? In S. Nielzen and O. Olsson, editors, *Structure and Perception of Electroacoustic Sound and Music*, number 846 in Excerpta Medica, pages 43–53. Elsevier, Amsterdam, 1989.

C. L. Krumhansl and P. Iverson. Perceptual interactions between musical pitch and timbre. *Journal of Experimental Psychology: Human Perception and Performance*, 18(3):739–751, 1992.

J. Kybic. Incremental updating of nearest neighbor-based high-dimensional entropy estimation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'06)*, volume 3, pages 804–807, 2006. doi: 10.1109/ICASSP.2006.1660776.

P. Ladefoged and I. Maddieson. *The Sounds of the World's Languages*. Blackwell, 1996.

S. Lakatos. A common perceptual space for harmonic and percussive timbres. *Perception & Psychophysics*, 62(7):1426–1439, 2000.

D. Lang, M. Klaas, and N. de Freitas. Insights on fast kernel density estimation algorithms. Technical report, Department of Computer Science, University of Toronto, 2004. URL `http://www.cs.toronto.edu/~dstn/papers/empirical.pdf`.

P. Langley, W. Iba, and K. Thompson. An analysis of Bayesian classifiers. *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 223–228, 1992.

J. Laver. *The Phonetic Description of Voice Quality*. Cambridge Studies in Linguistics. Cambridge University Press, 1980.

E. G. Learned-Miller. A new class of entropy estimators for multi-dimensional densities. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03)*, volume 3, pages 297–300, Apr 2003. doi: 10.1109/ICASSP.2003.1199463.

E. G. Learned-Miller and J. W. Fisher, III. ICA using spacings estimates of entropy. *Journal of Machine Learning Research*, 4:1271–1295, 2003.

K. Lederer. The phonetics of beatboxing (undergraduate degree dissertation), 2005. URL `http://www.humanbeatbox.com/phonetics`.

C.-C. Lee, E. Mower, C. Busso, S. Lee, and S. Narayanan. Emotion recognition using a hierarchical binary decision tree approach. In *Proceedings of Interspeech 2009*, pages 320–323, 2009.

J. W. Lee, P. S. Jones, Y. Mineyama, and X. E. Zhang. Cultural differences in responses to a Likert scale. *Research in Nursing & Health*, 25(4):295–306, 2002. doi: 10.1002/nur.10041.

E. Lesaffre, D. Rizopoulos, and R. Tsonaka. The logistic transform for bounded outcome scores. *Biostatistics*, 8(1):72–85, 2007. doi: 10.1093/biostatistics/kxj034.

D. J. Levitin, S. McAdams, and R. L. Adams. Control parameters for musical instruments: a foundation for new mappings of gesture to sound. *Organised Sound*, 7(2):171–189, 2002. doi: 10.1017/S135577180200208X.

Y. Li and D. L. Wang. Detecting pitch of singing voice in polyphonic audio. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005)*, volume III, pages 17–20, 2005.

Y. Li, M. Dong, and J. Hua. Localized feature selection for clustering. *Pattern Recognition Letters*, 29(1):10–18, 2008. doi: 10.1016/j.patrec.2007.08.012.

P.-Å. Lindestad, M. Södersten, B. Merker, and S. Granqvist. Voice source characteristics in Mongolian "throat singing" studied with high-speed imaging technique, acoustic spectra, and inverse filtering. *Journal of Voice*, 15(1):78–85, 2001. doi: 10.1016/S0892-1997(01)00008-X.

A. Loscos and Ó. Celma. Larynxophone: using voice as a wind controller. In *Proceedings of International Computer Music Conference 2005*, Barcelona, 2005.

A. Loscos, P. Cano, and J. Bonada. Low-delay singing voice alignment to text. In *Proceedings of the ICMC*, 1999.

G. Lugosi and A. Nobel. Consistency of data-driven histogram methods for density estimation and classification. *Annals of Statistics*, 24(2):687–706, 1996. doi: 10.1214/aos/1032894460.

S. Mabry. *Exploring Twentieth-Century Vocal Music: A Practical Guide to Innovations in Performance and Repertoire*. Oxford University Press, USA, 2002.

D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

E. Maestre, J. Bonada, and O. Mayor. Modeling musical articulation gestures in singing voice performances. In *Proceedings of 121st Convention of the Audio Engineering Society*, San Francisco, CA, USA, 2006.

E. Maestre, R. Ramírez, S. Kersten, and X. Serra. Expressive concatenative synthesis by reusing samples from real performance recordings. *Computer Music Journal*, 33(4):23–42, 2009. doi: 10.1162/comj.2009.33.4.23.

M.-W. Mak, C.-H. Sit, and S.-Y. Kung. Extraction of speaker features from different stages of DSR front-ends for distributed speaker verification. *International Journal of Speech Technology*, 8(1):67–77, 2005. doi: 10.1007/s10772-005-4762-x.

T. Mäki-Patola and P. Hämäläinen. Latency tolerance for gesture controlled continuous sound instrument without tactile feedback. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 1–5, 2004.

R. L. Mandryk and M. S. Atkins. A fuzzy physiological approach for continuously modeling emotion during interaction with play technologies. *International Journal of Human-Computer Studies*, 65(4):329–347, 2007. doi: 10.1016/j.ijhcs.2006.11.011.

J. Markel. Digital inverse filtering – a new tool for formant trajectory estimation. *IEEE Transactions on Audio and Electroacoustics*, 20(2):129–137, 1972. ISSN 0018-9278.

J. Marozeau and A. de Cheveigné. The effect of fundamental frequency on the brightness dimension of timbre. *Journal of the Acoustical Society of America*, 121(1):383–387, 2007. doi: 10.1121/1.2384910.

S. Marsland. *Machine Learning: An Algorithmic Perspective*. Chapman and Hall CRC Machine Learning and Pattern Recognition. Chapman and Hall, 2009.

W. L. Martens and A. Marui. Predicting timbral variation for sharpness-matched guitar tones resulting from distortion-based effects processing. In *Proceedings of 118th Audio Engineering Society Convention*, number 6486, May 2005.

T. M. Martinetz, S. G. Berkovich, and K. J. Schulten. 'Neural-gas' network for vector quantization and its application to time-series prediction. *IEEE Transactions on Neural Networks*, 4(4):558–569, 1993. doi: 10.1109/72.238311.

D. C. Martins Jr, U. M. Braga-Neto, R. F. Hashimoto, M. L. Bittner, and E. R. Dougherty. Intrinsically multivariate predictive genes. *Journal of Selected*

Topics in Signal Processing, 2(3):424–439, Jun 2008. ISSN 1932-4553. doi: 10.1109/JSTSP.2008.923841.

S. McAdams, S. Winsberg, S. Donnadieu, G. De Soete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: common dimensions, specificities, and latent subject classes. *Psychological Research*, 58(3):177–192, 1995. doi: 10.1007/BF00419633.

S. McAdams, B. Giordano, P. Susini, G. Peeters, and V. Rioux. A meta-analysis of acoustic correlates of timbre dimensions (abstract). *Journal of the Acoustical Society of America*, 120:3275, 2006.

S. McCandless. An algorithm for automatic formant extraction using linear prediction spectra. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 22(2):135–141, 1974.

J. McCartney. Rethinking the computer music language: SuperCollider. *Computer Music Journal*, 26(4):61–68, 2002. doi: 10.1162/014892602320991383.

P. McLeod and G. Wyvill. A smarter way to find pitch. In *Proceedings of the International Computer Music Conference (ICMC'05)*, pages 138–141, Barcelona, Spain, 2005.

A. Meng. *Temporal Feature Integration for Music Organisation*. PhD thesis, Technical University of Denmark (DTU), 2006. URL `http://www2.imm.dtu.dk/pubdb/p.php?4502`.

F. Mitchell. Form and expression in the vocal works of Edgard Varèse. *Contemporary Music Review*, 23(2):71–103, 2004. doi: 10.1080/0749446042000204554.

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

P. Mitra, C. A. Murthy, and S. K. Pal. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):301–312, 2002. doi: 10.1109/34.990133.

D. F. Morrison. *Applied Linear Statistical Methods*. Prentice Hall, 1983.

S. K. Murthy. Automatic construction of decision trees from data: a multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2(4):345–389, 1998. doi: 10.1023/A:1009744630224.

J.-J. Nattiez. Inuit vocal games. In *The Canadian Encyclopedia*. Historica Foundation, 2008. URL `http://www.thecanadianencyclopedia.com/index.cfm?PgNm=TCE&Params=U1ARTU0001711`.

R. B. Nelsen. *An Introduction to Copulas*. Springer, New York, 2nd edition, 2006.

M. E. R. Nicholls, C. A. Orr, M. Okubo, and A. Loftus. Satisfaction guaranteed: the effect of spatial biases on responses to Likert scales. *Psychological Science*, 17(12):1027–1028, 2006. doi: 10.1111/j.1467-9280.2006.01822.x.

K. I. Nordstrom and P. F. Driessen. Variable pre-emphasis LPC for modeling vocal effort in the singing voice. In *Proceedings of the International Conference on Digital Audio Effects (DAFx-06)*, pages 157–160, Montreal, Quebec, Canada, Sep 2006.

D. A. Norman. *The Design of Everyday Things*. Basic Books, New York, 2002.

T. L. Nwe and H. Li. Exploring vibrato-motivated acoustic features for singer identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(2):519–530, 2007. doi: 10.1109/TASL.2006.876756.

H. Nyquist. Thermal agitation of electric charge in conductors. *Physical Review*, 32(1):110–113, 1928. doi: 10.1103/PhysRev.32.110.

N. Orio. Music retrieval: a tutorial and review. *Foundations and Trends in Information Retrieval*, 1(1):1–90, Nov 2006. doi: 10.1561/1500000002.

D. O'Shaughnessy. Interacting with computers by voice: automatic speech recognition and synthesis. *Proceedings of the IEEE*, 91(9):1272–1305, Sep 2003. doi: 10.1109/JPROC.2003.817117.

A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot. One microphone singing voice separation using source-adapted models. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 90–93, Oct 2005. doi: 10.1109/ASPAA.2005.1540176.

G. Papanikolaou and C. Pastiadis. Multiple dichotomies in timbre research. *Archives of Acoustics*, 34(2):137–155, 2009.

J. Paulus and A. Klapuri. Model-based event labelling in the transcription of percussive audio signals. In *Proceedings of the 6th International Conference on Digital Audio Effects (DAFX-03)*, 2003.

D. Pearce. Speech processing, transmission and quality aspects (STQ); distributed speech recognition; front-end feature extraction algorithm; compression algorithms. Technical Report ES 201 108, v1.1.3, European Telecommunications Standards Institute, Sep 2003. URL `http://webapp.etsi.org/workprogram/Report_WorkItem.asp?wki_id=18820`.

G. Peeters. A large set of audio features for sound description. Technical report, IRCAM, 2004.

I. Peretz and R. J. Zatorre. Brain organization for music processing. *Annual Review of Psychology*, 56(1):89–114, 2005. doi: 10.1146/annurev.psych.56.091103.070225.

T. J. Pinch and W. E. Bijker. The social construction of facts and artefacts: or how the sociology of science and the sociology of technology might benefit each other. *Social Studies of Science*, 14(3):399–441, 1984.

R. Polfreman. A task analysis of music composition and its application to the development of Modalyser. *Organised Sound*, 4(1):31–43, 1999. doi: 10.1017/S1355771899001053.

I. Poupyrev, M. J. Lyons, S. Fels, and T. Blaine. New Interfaces for Musical Expression. Workshop proposal, 2001. URL `http://www.nime.org/2001/docs/proposal.pdf`.

J. Preece, Y. Rogers, and H. Sharp. *Interaction Design*. Wiley, 2004.

D. Pressnitzer and D. Gnansia. Real-time auditory models. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 295–298, Barcelona, Spain, 2005.

M. Puckette. Low-dimensional parameter mapping using spectral envelopes. In *Proceedings of the International Computer Music Conference (ICMC'04)*, pages 406–408, 2004.

P. Pudil, J. Novovičová, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119–1125, 1994. doi: 10.1016/0167-8655(94)90127-9.

C. P. Quesenberry and F. L. Miller. Power studies of some tests for uniformity. *Journal of Statistical Computation and Simulation*, 5(3):169–191, 1977. doi: 10.1080/00949657708810150.

F. Questier, R. Put, D. Coomans, B. Walczak, and Y. Vander Heyden. The use of CART and multivariate regression trees for supervised and unsupervised feature selection. *Chemometrics and Intelligent Laboratory Systems*, 76(1):45–54, 2005. ISSN 0169-7439. doi: 10.1016/j.chemolab.2004.09.003.

L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993. ISBN 0-13-015157-2.

L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*. Prentice-Hall Englewood Cliffs, NJ, 1978.

A. K. Ramalingam and S. Krishnan. Gaussian Mixture Modeling of Short-Time Fourier Transform features for audio fingerprinting. *IEEE Transactions on Information Forensics and Security*, 1(4):457–463, 2006. doi: 10.1109/TIFS.2006.885036.

D. M. Randel. Drum set. In D. M. Randel, editor, *The Harvard Dictionary of Music*, page 256. Harvard University Press, 4th edition, 2003.

V. Rao and P. Rao. Vocal melody detection in the presence of pitched accompaniment using harmonic matching methods. In *Proceedings of the 11th International Conference on Digital Audio Effects (DAFx-08)*, pages 371–378, 2008.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

C. Roads. Introduction to Granular Synthesis. *Computer Music Journal*, 12(2): 11–13, 1988. doi: 10.2307/3679937.

S. T. Roweis. EM algorithms for PCA and SPCA. In *Advances in Neural Information Processing Systems (NIPS)*, pages 626–632, Denver, Colorado, 1998.

P. Salamé and A. Baddeley. Effects of background music on phonological short-term memory. *Quarterly Journal of Experimental Psychology Section A*, 41 (1):107–122, 1989. doi: 10.1080/14640748908402355.

A. Schoenberg. *Harmonielehre*. University of California Press, Berkeley, Los Angeles, 3rd edition, 1922.

M. R. Schroeder and B. S. Atal. Code-excited linear prediction (CELP): high quality speech at very low bit rates. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'85)*, pages 937–940, 1985.

B. Schuller, S. Steidl, and A. Batliner. The Interspeech 2009 emotion challenge. In *Proceedings of Interspeech 2009*, pages 312–315, Sep 2009.

D. Schwarz. Current research in concatenative sound synthesis. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 9–12, Barcelona, Spain, 2005.

S. R. Searle, G. Casella, and C. E. McCulloch. *Variance Components*. Wiley-Interscience, online edition, 2006. ISBN 978-0470316856. doi: 10.1002/9780470316856.

J. S. Seo, M. Jin, S. Lee, D. Jang, S. Lee, and C. D. Yoo. Audio fingerprinting based on normalized spectral subband centroids. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, volume 3, pages iii/213–iii/216, 2005. ISBN 1520-6149. doi: 10.1109/ICASSP.2005.1415684.

J. P. Shaffer. Multiple hypothesis testing. *Annual Review of Psychology*, 46(1): 561–584, 1995. doi: 10.1146/annurev.ps.46.020195.003021.

R. V. Shannon, F.-G. Zeng, V. Kamath, J. Wygonski, and M. Ekelid. Speech recognition with primarily temporal cues. *Science*, 270(5234):303–304, Oct 1995. doi: 10.1126/science.270.5234.303.

J. Sharpe. Jimmie Riddle and the lost art of eephing. `http://www.npr.org/templates/story/story.php?storyId=5259589`, 2006.

Shure Inc. *Shure SM58 user guide*, 2006. URL `http://www.shure.co.uk/products/microphones/sm58`.

A. F. Siegel. *Statistics and Data Analysis: An Introduction*. John Wiley & Sons Inc, New York, 1988.

D. Silverman. *Interpreting Qualitative Data: Methods for Analysing Talk, Text and Interaction*. Sage Publications Inc, 3rd edition, 2006.

Sinclair Research Ltd. *Sinclair Spectrum 128 Service Manual*, 1985. `ftp://ftp.worldofspectrum.org/pub/sinclair/technical-docs/ZXSpectrum128K_TechnicalManual.pdf`.

E. Sinyor, C. McKay, R. Fiebrink, D. McEnnis, and I. Fujinaga. Beatbox classification using ACE. In *Proceedings of the International Conference on Music Information Retrieval*, pages 672–675, 2005.

J. A. Sloboda, K. J. Wise, and I. Peretz. Quantifying tone deafness in the general population. *Annals of the New York Academy of Sciences*, 1060:255–261, 2005. doi: 10.1196/annals.1360.018.

D. Soto-Morettini. *Popular Singing: A Practical Guide to Pop, Jazz, Blues, Rock, Country and Gospel*. A & C Black, 2006. ISBN 978-0713672664.

T. Sporer, J. Liebetrau, and S. Schneider. Statistics of MUSHRA revisited. In *Proceedings of the 127th Audio Engineering Society Convention (AES 127)*, number 7825. Audio Engineering Society, Oct 2009.

J. Stepánek. Relations between perceptual space and verbal description in violin timbre. In *Proceedings of Acústica 2004*, number 077-S, Guimarães, Portugal, 2004.

D. W. Stewart, P. M. Shamdesani, and D. W. Rook. *Focus Groups: Theory and Practice*. SAGE Publications, 2nd edition, 2007.

B. H. Story, I. R. Titze, and E. A. Hoffman. The relationship of vocal tract shape to three voice qualities. *Journal of the Acoustical Society of America*, 109(4):1651–1667, Apr 2001. doi: 10.1121/1.1352085.

D. Stowell and M. D. Plumbley. Pitch-aware real-time timbral remapping. In *Proceedings of the Digital Music Research Network (DMRN) Summer Conference*, Jul 2007.

D. Stowell and M. D. Plumbley. Characteristics of the beatboxing vocal style. Technical Report C4DM-TR-08-01, Dept of Electronic Engineering, Queen Mary University of London, London, UK, 2008a. URL `http://www.elec.qmul.ac.uk/digitalmusic/papers/2008/Stowell08-beatboxvocalstyle-C4DM-TR-08-01.pdf`.

D. Stowell and M. D. Plumbley. Robustness and independence of voice timbre features under live performance acoustic degradations. In *Proceedings of the 11th Conference on Digital Audio Effects (DAFx-08)*, pages 325–332, Sep 2008b.

D. Stowell and M. D. Plumbley. Fast multidimensional entropy estimation by k-d partitioning. *IEEE Signal Processing Letters*, 16(6):537–540, Jun 2009. doi: 10.1109/LSP.2009.2017346.

D. Stowell and M. D. Plumbley. Timbre remapping through a regression-tree technique. In *Proceedings of Sound and Music Computing*, pages 45–50, 2010.

D. Stowell and M. D. Plumbley. Delayed decision-making in real-time beatbox percussion classification. *Journal of New Music Research*, in press.

D. Stowell and M. D. Plumbley. Cross-associating unlabelled timbre distributions to create expressive musical mappings. In *Proceedings of the 2010 Workshop on Applications of Pattern Analysis (WAPA2010)*, accepted.

D. Stowell, M. D. Plumbley, and N. Bryan-Kinns. Discourse analysis evaluation method for expressive musical interfaces. In *New Interfaces for Musical Expression*, pages 81–86, 2008.

D. Stowell, A. Robertson, N. Bryan-Kinns, and M. D. Plumbley. Evaluation of live human-computer music-making: quantitative and qualitative approaches. *International Journal of Human-Computer Studies*, 67(11):960–975, Nov 2009. doi: 10.1016/j.ijhcs.2009.05.007.

B. L. Sturm. Adaptive concatenative sound synthesis and its application to micromontage composition. *Computer Music Journal*, 30(4):46–66, 2006. doi: 10.1162/comj.2006.30.4.46.

J. Sundberg. Level and center frequency of the singer's formant. *Journal of Voice*, 15(2):176–186, 2001. doi: 10.1016/S0892-1997(01)00019-4.

C. Sutton. Transcription of vocal melodies in popular music. Master's thesis, Dept of Electronic Engineering, Queen Mary University of London, London, UK, 2006.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. doi: 10.1126/science.290.5500.2319.

A. Tindale, A. Kapur, G. Tzanetakis, and I. Fujinaga. Retrieval of percussion gestures using timbre classification techniques. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 541–545, 2004.

D. Tompkins. *How to Wreck a Nice Beach: The Vocoder from World War II to Hip-Hop*. Melville House, 2010.

V. Tyagi and C. Wellekens. On desensitizing the Mel-cepstrum to spurious spectral components for robust speech recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, volume 1, pages 529–532, 2005. ISBN 1520-6149. doi: 10.1109/ICASSP.2005.1415167.

G. Tzanetakis, G. Essl, and P. Cook. Automatic musical genre classification of audio signals. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 205–210, 2001.

R. M. van Besouw, J. S. Brereton, and D. M. Howard. Range of tuning for tones with and without vibrato. *Music Perception*, 26(2):145–155, 2008. doi: 10.1525/MP.2008.26.2.145.

O. Vasicek. A test for normality based on sample entropy. *Journal of the Royal Statistical Society. Series B (Methodological)*, 38(1):54–59, 1976.

V. Verfaille, U. Zölzer, and D. Arfib. Adaptive digital audio effects (A-DAFx): a new class of sound transformations. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1817–1831, Sep 2006. ISSN 1558-7916. doi: 10.1109/TSA.2005.858531.

J. D. Victor. Binless strategies for estimation of information from neural data. *Physical Review E*, 66(5):51903, 2002. doi: 10.1103/PhysRevE.66.051903.

H. L. F. von Helmholtz. *On the Sensations of Tone*. Dover, New York, NY, USA, 2nd edition, 1954/1877.

A. Vurma and J. Ross. Timbre-induced pitch deviations of musical sounds. *Journal of Interdisciplinary Music Studies*, 1(1):33–50, 2007.

M. M. Wanderley and N. Orio. Evaluation of input devices for musical expression: borrowing tools from HCI. *Computer Music Journal*, 26(3):62–76, 2002. doi: 10.1162/014892602320582981.

S. Wegener, M. Haller, J. J. Burred, T. Sikora, S. Essid, and G. Richard. On the robustness of audio features for musical instrument classification. In *Proceedings of the 16th European Signal Processing Conference (EUSIPCO)*, 2008.

D. L. Wessel. Timbre space as a musical control structure. *Computer Music Journal*, 3(2):45–52, 1979.

C. Wharton, J. Rieman, C. Lewis, and P. Polson. The cognitive walkthrough method: a practitioner's guide. In J. Nielsen and R. L. Mack, editors, *Usability Inspection Methods*, pages 105–140. Wiley, New York, 1994.

G. Widmer and W. Goebl. Computational models of expressive music performance: the state of the art. *Journal of New Music Research*, 33:203–216(14), Sep 2004. doi: 10.1080/0929821042000317804.

I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Technique*. Morgan Kaufmann, San Francisco, CA, USA, 2nd edition, 2005.

M. Wright, R. J. Cassidy, and M. F. Zbyszynski. Audio and gesture latency measurements on Linux and OSX. In *Proceedings of the International Computer Music Conference (ICMC 2004)*, pages 423–429, Miami, FL, USA, 2004.

I. Xenakis. *Formalized Music: Thought and Mathematics in Composition*. Pendragon Press, revised edition, 1992.

B. Xiang, U. V. Chaudhari, J. Navratil, G. N. Ramaswamy, and R. A. Gopinath. Short-time Gaussianization for robust speaker verification. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2002)*, volume 1, pages 681–684, 2002.

R. Xu and D. Wunsch II. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, May 2005. ISSN 1045-9227. doi: 10.1109/TNN.2005.845141.

D. Zangger Borch, J. Sundberg, P.-Å. Lindestad, and M. Thalén. Vocal fold vibration and voice source aperiodicity in 'dist' tones: a study of a timbral ornament in rock singing. *Logopedics Phoniatrics Vocology*, 29(4):147–153, 2004. doi: 10.1080/14015430410016073.

Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58, 2009. doi: 10.1109/TPAMI.2008.52.

L. C. Zhao, P. R. Krishnaiah, and X. R. Chen. Almost sure L_r-norm convergence for data-based histogram density estimates. *Theory of Probability and its Applications*, 35(2):396–403, Jan 1990. doi: 10.1137/1135057.