I declare that the material contained in this thesis is my own original work except where another source has been acknowledged.

Charles Patrick Martin
May 2016
Acknowledgements

A long list of friends and colleagues deserve acknowledgement for their assistance over the past three years.

I acknowledge the help of Henry Gardner, my primary supervisor. Henry’s encouragement affected all aspects of my research, from our earliest meetings planning an iPad ensemble to the final publications before submitting my thesis. His experience in HCI as well as music helped me define a clear path for my research and design experiments to test my digital musical instruments. I look forward to our future collaborations.

My other supervisors, Ben Swift and Alec Hunter, also provided great assistance. Ben’s broad knowledge of computing, music, and HCI has been inspirational. Alec’s work at the ANU School of Music led to a series of performance collaborations and helped recruit participants for my improvised performance studies. Michael Martin deserves special mention for his assistance in statistics, experiment design, and contributions to several of my published papers.

At the ANU, I also received assistance from Gary France, Alistair Rid-dell, Andrew Sorenson, Tim Brook, Martyn Jolly, Jean-Philippe Demarais, Craig Greening, Erin Helyard, and Tony Oates. I also thank Anders Åstrand and Bill Cahn for their support. I thank Maryann Martin for the thoughtful editorial advice she provided as I was preparing this thesis for submission.

Much of my work was motivated by musical performances and throughout this project I was joined in concert by many enthusiastic performers
who bravely played experimental iPad software in front of an audience. The members of Ensemble Metatone, Christina Hopgood, Jonathan Griffiths, and Yvonne Lam, deserve special mention for their hours of work in the rehearsal studio and on stage. In the United States, I collaborated on several performances with my colleagues from Ensemble Evolution, Maria Finkelmeier and Jacob Remington.

Other musicians who assisted with this research include Steve Fitzgerald, Jacinta Dunlop, Malcolm Newland, Jeremy Barnett, Chloë Hobbs, members of the ANU New Music Ensemble in 2014 and 2015, and the volunteer participants of my formal studies. I thank all these friends for their enthusiasm, creativity and energy.

Finally, I thank my family for their support and love. I especially thank my wife, Christina Hopgood, who has not only worked with me on artistic projects but has supported me through a hard final year of my studies.
Abstract

This thesis concerns the making and performing of music with new digital musical instruments (DMIs) designed for ensemble performance. While computer music has advanced to the point where a huge variety of digital instruments are common in educational, recreational, and professional music-making, these instruments rarely seek to enhance the ensemble context in which they are used. Interaction models that map individual gestures to sound have been previously studied, but the interactions of ensembles within these models are not well understood. In this research, new ensemble-focussed instruments have been designed and deployed in an ongoing artistic practice. These instruments have also been evaluated to find out whether, and if so how, they affect the ensembles and music that is made with them.

Throughout this thesis, six ensemble-focussed DMIs are introduced for mobile touch-screen computers. A series of improvised rehearsals and performances leads to the identification of a vocabulary of continuous performative touch-gestures and a system for tracking these collaborative performances in real time using tools from machine learning. The tracking system is posed as an intelligent agent that can continually analyse the gestural states of performers, and trigger a response in the performers’ user interfaces at appropriate moments. The hypothesis is that the agent interaction and UI response can enhance improvised performances, allowing performers to better explore creative interactions with each other, produce
better music, and have a more enjoyable experience.

Two formal studies are described where participants rate their perceptions of improvised performances with a variety of designs for agent-app interaction. The first, with three expert performers, informed refinements for a set of apps. The most successful interface was redesigned and investigated further in a second study with 16 non-expert participants. In the final interface, each performer freely improvised with a limited number of notes; at moments of peak gestural change, the agent presented users with the opportunity to try different notes. This interface is shown to produce performances that are longer, as well as demonstrate improved perceptions of musical structure, group interaction, enjoyment and overall quality.

Overall, this research examined ensemble DMI performance in unprecedented scope and detail, with more than 150 interaction sessions recorded. Informed by the results of lab and field studies using quantitative and qualitative methods, four generations of ensemble-focused interface have been developed and refined. The results of the most recent studies assure us that the intelligent agent interaction does enhance improvised performances.

**Keywords:**

mobile computer music, touch-screen performance, collaborative performance practice, human computer interaction, intelligent agent, networked interaction, percussion, artistic research, rehearsal-as-research.
Acknowledgements  v

Abstract  vii

Abbreviations  xv

1 Introduction  1
   1.1 Making music with iPads  3
   1.2 Studying Improvisation  5
   1.3 Percussionist-Centred Design  5
   1.4 Research Questions  7
      1.4.1 How might touch-screen instruments be used in ensemble improvisations by percussionists?  7
      1.4.2 How can a system of apps and agents be built that reacts to, and directs, ensemble improvisations?  8
      1.4.3 How does this system of apps and agents affect ensemble improvisation?  8
   1.5 Summary of Contributions  9
      1.5.1 Vocabulary of Percussive Touch-Screen Gestures  9
      1.5.2 Performance Tracking Agent  10
      1.5.3 A Repertoire of Touch-Screen Apps  10
      1.5.4 Study of expert performers with agent-app systems  10
### Contents

1.5.5 Study of non-expert performers with agent and direct ensemble interactions ............................................ 11

2 Ensemble Interaction in Computer Music .................................................. 13
  2.1 Models of DMI Interaction ........................................................................ 14
  2.2 Laptop Orchestras ..................................................................................... 17
  2.3 Mobile Music ........................................................................................... 19
    2.3.1 Mobility in Mobile Music ..................................................................... 20
    2.3.2 Development Tools for Mobile DMIs ................................................ 22
  2.4 Other Systems for Ensemble Improvisation .............................................. 22
  2.5 Interacting with Agent Systems ............................................................... 25
  2.6 Conclusions ............................................................................................. 27

3 Ensemble Metatone: A Touch-Screen Ensemble .......................................... 31
  3.1 Percussionist-Centered Design ................................................................ 33
  3.2 MetaTravels: A Prototype Touch-Screen Instrument .................................. 37
    3.2.1 Touch-Screen Interface ........................................................................ 39
    3.2.2 UI Features ........................................................................................ 40
  3.3 MetaLonsdale: A Networked Touch-Screen Instrument ........................... 41
    3.3.1 Syncing Harmony .............................................................................. 43
  3.4 Recording Touch-Screen Data .................................................................... 44
  3.5 A Research Rehearsal Series ..................................................................... 47
  3.6 Analysing the Rehearsal Process ............................................................... 54
    3.6.1 Taps ..................................................................................................... 56
    3.6.2 Swipes ................................................................................................ 57
    3.6.3 Combinations ...................................................................................... 58
    3.6.4 Volume Control ................................................................................... 58
    3.6.5 Relationship Between iPads and Percussion ...................................... 58
    3.6.6 Structuring an Improvisation ............................................................ 59
  3.7 Conclusions ............................................................................................. 59
  3.8 Artistic Outcomes ..................................................................................... 60
4 Tracking Ensemble Touch-Screen Performances 63
  4.1 Tracking Gestures ............................................ 64
    4.1.1 Characterising Percussive Gesture ..................... 67
  4.2 System Design .............................................. 69
    4.2.1 Gesture Classifier ...................................... 70
    4.2.2 Transition Matrices .................................... 73
    4.2.3 The Flux Measure ...................................... 77
    4.2.4 Identifying New Ideas .................................. 78
  4.3 A Repertoire of Touch-Screen DMIs ........................ 81
    4.3.1 BirdsNest .................................................. 82
    4.3.2 Snow Music ............................................... 84
    4.3.3 Singing Bowls ........................................... 86
  4.4 Preliminary Evaluation ....................................... 88
    4.4.1 Gesture Classifier Accuracy ............................. 89
    4.4.2 Computational Cost ..................................... 92
    4.4.3 Concert Performance .................................... 93
  4.5 Chapter Summary .............................................. 94

5 Evaluating Apps and Agents 97
  5.1 A Rehearsal-As-Research Methodology ...................... 99
  5.2 Experimental System ......................................... 102
    5.2.1 Gesture Classifier ...................................... 103
    5.2.2 Generating Fake Gestures ............................... 103
    5.2.3 iPad Apps ............................................... 105
  5.3 Experiment .................................................... 106
    5.3.1 Participants ............................................. 108
    5.3.2 Questionnaires ......................................... 109
  5.4 Results ....................................................... 110
    5.4.1 Survey Data ............................................. 110
    5.4.2 Qualitative Analysis .................................... 118
  5.5 Discussion .................................................... 119
  5.6 Summary ....................................................... 120
Contents

6 Refining Networked Interactions for Touch-Screen Apps 123
   6.1 PhaseRings ............................................. 124
      6.1.1 Sound Schemes ..................................... 126
      6.1.2 Compositions ....................................... 128
      6.1.3 Local and Agent Interactions ...................... 129
      6.1.4 Inter-App Connectivity .............................. 131
   6.2 Designs for Networked Interaction .................... 132
      6.2.1 App Features ....................................... 132
      6.2.2 Tonality ............................................ 134
      6.2.3 Sound Palette ...................................... 135
      6.2.4 Secondary Sounds .................................. 136
   6.3 Chapter Summary ......................................... 137

7 Comparing Agent-Control with a Networked User-Interface 139
   7.1 Related Work ........................................... 141
   7.2 System Design .......................................... 143
      7.2.1 App .................................................. 143
      7.2.2 Agent ............................................... 146
   7.3 First Experiment ......................................... 147
      7.3.1 Participants ......................................... 148
      7.3.2 Procedure ........................................... 148
      7.3.3 Questionnaires ..................................... 149
   7.4 Results ................................................... 150
      7.4.1 Survey Data ......................................... 150
      7.4.2 Post-hoc testing ..................................... 154
      7.4.3 Performer Preferences .............................. 156
      7.4.4 Performance Length ................................ 157
      7.4.5 New-Idea Messages ................................ 158
      7.4.6 Button Presses ...................................... 159
   7.5 Follow-up Study .......................................... 160
   7.6 Discussion ................................................ 164
   7.7 Summary .................................................. 167
## Contents

### 8 Preserving Touch-Screen Improvisations  
8.1 Curating the Improvised  
8.2 Towards a protocol for touch-screen musical performance  
   8.2.1 The Metatone Log Protocol  
8.3 Transcoding Performance Protocols  
   8.3.1 Gestural Scores  
   8.3.2 Visualisations  
8.4 Summary  

### 9 Conclusions  
9.1 Iterating App Designs  
9.2 Ensemble Director Agents  
9.3 Reflecting on the Research Questions  
   9.3.1 How might touch-screen instruments be used in ensemble improvisations by percussionists?  
   9.3.2 How can a system of apps and agents be built that reacts to, and directs, ensemble improvisations?  
   9.3.3 How does this system of apps and agents affect ensemble improvisation?  
9.4 Limitations and Future Work  
9.5 Final Remarks  

### Appendices  
A Software Releases  
   A.1 MetaLonsdale  
   A.2 BirdsNest  
   A.3 Snow Music  
   A.4 PhaseRings  
   A.5 Metatone Classifier  
   A.6 Response from Users  
B Documentation of Artistic Outcomes  
   B.1 Compositions  

---

xiii
Abbreviations

CMM Creative Music Making. 47–50, 54


DSP Digital Signal Processing. 22


GUI Graphical User Interface. 9, 11, 18, 19, 23, 37, 38, 40, 83, 87, 132, 135–137, 140, 145, 167, 187, 192–195, 203, 204

HCI Human Computer Interaction. 2, 9, 33, 35, 65, 68, 72, 88, 124, 140, 145, 186, 190, 198, 199

LOrk Laptop Orchestra. 17–20, 22–26, 99

LPN Local Performance Network. 14, 16, 28–30, 44, 63, 123–126, 129, 132, 135, 137, 139, 140, 176, 185, 186, 188, 192, 194, 195

MIDI Musical Instrument Digital Interface. 18, 24, 25, 172
Abbreviations

ML Machine Learning. 66, 67, 71, 73, 100

NIME New Interfaces for Musical Expression. 64–66

OSC Open Sound Control. 10, 44, 173–176

TM Transition Matrix. 70, 75–78, 80

WIST Wireless Sync-Start Technology. 143
Introduction

This thesis is about making ensemble music with touch-screen apps, and finding ways to use network connections between devices to support and extend the creative possibilities of group performances. This research seeks to evaluate how such apps are used by ensembles, how they affect the performers, and how they shape their musical output. The apps run on iPads which are posed as self-contained Digital Musical Instruments (DMIs) where users interact with touch-screens to directly control synthesised sounds. Importantly, these apps are designed to react not only to the individual performer who is using them, but to be aware of the ensemble context and update their interface and functionality in response to the collaborative interactions that take place in a group performance.

The following chapters describe six musical iPad apps and a server application developed during the course of this project to track performances with these apps. This system was initially developed through a practice-led artistic process where a group of expert percussionists were brought together to help define the creative possibilities of performing with a prototype app. This process led to the development of a successful artistic practice and prompted several new apps with a range of ensemble-interaction behaviours. The modes of performance that this group defined were also successfully replicated with other expert groups around the world. A second phase of this research focussed on one refined app that was introduced to groups with mixed instrumental experience. In addition to using these mu-
1. Introduction

Musical apps in natural musical environments such as rehearsing and performing public concerts, they were also examined in formal Human Computer Interaction (HCI) studies.

There are four important antecedent fields for the work in this project. The first is the crossover field that exists between computer music and HCI and deals with New Interfaces for Musical Expression (NIME) (Poupyrev et al., 2001). In this area, researchers are concerned with systems for controlling computer music in live performances, with much work already carried out in terms of examining performing with touch-screen devices. The second important field is collaborative creativity, which draws on work in psychology to understand how performers and others develop new creative work in groups, and why this is such an appealing and productive process. This area also connects with HCI research to support cooperative creative processes. Finally, the artistic goals of this project are supported by the fields of percussion performance and free-improvisation. Contemporary percussion practice encompasses performance on an unlimited number of sounding objects with a common gestural language and set of fundamental techniques. As will be described, this practice is very useful for characterising a performance practice on touch-screens. Non-idiomatic free-improvisation is a mode of performance where music is created in the moment of performance by the musicians themselves without adhering to any particular stylistic framework. While this form of music-making has connections with the free-jazz movement of the 1960s (Bailey, 1993), it is also highly connected with contemporary percussion groups of the 1970s that performed with an expanding array of instruments from cultures around the world (Cahn, 2005). Free-improvisation is also a performance mode focussed on ensemble performance, and is particularly open to the spontaneous creation of new groups where performers learn to perform with each other while improvising (Stenström, 2009). As such, free-improvisation is an ideal method for examining new ensemble-focussed musical instruments, with unfamiliar interfaces and for which the parameters of performance are not yet known.

In the remainder of this chapter, these motivating fields are examined in more detail and used to construct a series of research questions. The contri-
1.1 Making music with iPads

In recent times, mobile devices equipped with multi-touch displays, network connections, sensors, and powerful CPUs have proliferated everyday life, so much so that some have questioned the long term relevance of keyboard-and-mouse computing (Alba, 2015). Much computer music research has centered on developing new ways to control and perform digital sound with various sensor technologies; hence, the multiple affordances of sensor-packed mobile devices, combined with their portable and self-contained design, has made these devices a playground for developing new digital musical instruments (Tanaka, 2010). Examples such as Ocarina (Wang, 2014) and Orphion (Trump & Bullock, 2014) have also enjoyed commercial and critical success. Of particular interest to this thesis, however, is how these devices have been used in ensemble performance, and this will be further discussed in Chapter 2.

Even though smartphone and tablet ensembles have emerged worldwide, very few of the applications used by these groups sense and take advantage of the ensemble context in which they are used, with the functionality of these applications remaining identical whether they are used solo or in groups. However, while skilled free-improvisers can perform coherent solo works, the excitement of real-time, ensemble collaboration has made group performance (such as in the archetypal jazz combo band) the more popular medium. The excitement of group interactions in ensemble performance motivates the development of DMIs that can sense and react to these interactions, just as they react to the individual performers gestural interactions with a creative interface. As illustrated in Figure 1.1, it may be appropriate to consider ensemble interactions at a higher level than the individual
1. Introduction

Figure 1.1: DMIs map performer interactions to low-level performance elements such as notes. This is a sufficient model to deal with individual performances such as the left diagram. For collaborative performances, participants develop natural ensemble interactions. An ensemble DMI must deduce some of these interactions which could be mapped to higher-level performance elements such as musical harmonic progression.

interactions where performers might directly control notes and sound.

Can DMIs be developed that support or enhance this kind of collaborative creativity? In this thesis, a number of designs for ensemble-aware apps will be described and evaluated. In short, these apps establish network connections and share information about how they are used throughout a performance. As will be described in Chapter 6, this information goes far beyond the shared metronome functionality explored in some commercial apps. A server-based system that tracks how apps are played, in real-time, during a performance, will be described in significant detail in Chapter 4, while later chapters will evaluate the effect of this system on real performances.
1.2 Studying Improvisation

Free-improvised performance has been defined as the performance of music “without any restrictions regarding style or genre and without having predetermined what is to be played” (Stenström, 2009, p. vi). Developing DMI systems to support free-improvisation is a significant motivation for the work in this thesis. Improvisation, however, is also used as a research and training methodology. Bill Cahn, a member of the pioneering percussion group Nexus, writes that improvisation encourages “a deeper knowledge of the instruments and their sound-making possibilities” (Cahn, 2005, p. 3). Digital media theorist Aden Evens writes that when improvising with an unfamiliar instrument “the musician generates novel and surprising results even when applying familiar technique” (Evens, 2005, p. 153). Ensemble free-improvisation emphasises interactions between performers, with Mazzola and Cherlin (2009) pointing out that one of the free improviser’s primary roles is “to negotiate (while playing) with their fellow players every single item they bring into play . . . as if partaking in a dynamic and sophisticated game” (p. 7).

In Chapter 3, a characterisation of touch-screen performance is identified using the explorations in a series of improvised rehearsals and performances. In chapters 5 and 7, improvisation is used as a performance task for evaluating different app designs in formal lab-based studies. Systems developed in this research for capturing and tracking touch-screen performances can also be used to preserve and understand improvisations on mobile devices in great depth. In Chapter 8, a protocol will be described for capturing touch-screen performances. This protocol has been used to preserve a significant number of performances in detail — over the course of this project, more than 150 collaborative interaction sessions were recorded.

1.3 Percussionist-Centred Design

Percussion is a performance practice more defined by modes of interaction and performative gestures than by any particular instrument (Schick,
The conceptual separation of percussive performance technique from instruments matches a similar dichotomy in computer music, where control interface and sound synthesis components are separable and interchangeable. For this reason, percussionists are posed as *gestural explorers* in Chapter 3, where the unknown performance parameters of prototype interfaces are mapped out. Percussion training also emphasises chamber ensemble performance and improvisation; hence, a group of percussionists is ideal for trialling and evaluating prototype designs for enhancing ensemble interactions, as detailed in Chapter 5.

Given the extensive work with percussionists in prototyping potential apps, the design process in this thesis is said to be percussionist-centred, a term that will be more fully explained in Chapter 3. While this may seem to unnecessarily limit DMI designs to appeal to a small number of experts, it must be noted that percussion instruments are frequently viewed as highly accessible and intuitive. The same instruments seen in the preschool classroom are often played on the symphony orchestra stage. A percussionist-centred process, then, could be seen as expanding the potential pool of users, as percussionists expect low boundaries for interaction but broad possibilities for expression. This choice is vindicated in experiences with non-percussionists in Chapter 7 where a refined interface enabled exceptional levels of enjoyment and creative exploration.

Many of the artistic accomplishments documented in this work are related to the professional percussion community. My colleagues in Canberra enabled a series of percussion and iPad performances as well as the release of two albums of music for these instruments. In the USA, I was able to pursue further performances with Ensemble Evolution, with these works presented to the percussion community at the Percussive Arts Society International Convention in 2013 and 2014. Interest from these audiences encouraged the public distribution of apps developed in this project that have been incorporated into a number of performances unrelated to this research. A complete list of the artistic outcomes related to this work is in Appendix B.
1.4 Research Questions

The following chapters present several designs and implementations for a system of touch-screen musical instruments and server-based agents. This system will be investigated according to the following overarching research questions.

1.4.1 How might touch-screen instruments be used in ensemble improvisations by percussionists?

Free-improvisations tend towards the ephemeral and analysis of their internal structure is rare. However, computer-based interfaces can be instrumented so that interactions are logged for later analysis. In this project, protocols of interaction data are used alongside audio and video recordings of performances to understand in depth how a group of percussionists use the instrument designs that are presented. In Chapter 3, two app designs
1. Introduction

are used in a series of rehearsals by Ensemble Metatone, a research-focussed percussion group. Data from these sessions is used in a qualitative study to identify a vocabulary of percussive gestures these performers used to construct their improvisations.

1.4.2 How can a system of apps and agents be built that reacts to, and directs, ensemble improvisations?

While touch-screen DMIs have been popular on smartphones and tablets since these mobile devices were made available, instruments that are specifically designed for ensemble use and that react to ensemble interactions have been rare to non-existent.

In Chapter 2, a number of different paradigms for interacting with ensembles are described. At the simplest level, this involves synchronising the activity of certain features of the app with specific messages over a local network. A more advanced approach is to build an ensemble directing agent that can track the performers’ actions in real-time throughout a performance in order to build a model of the performance structure. These data and specific moments of interest that occur during the performance can be returned to the iPad interface. An iPad app, MetaLonsdale, that synchronises features across the ensemble is introduced in Chapter 3, while a server-based agent system is described in Chapter 4 that uses transitions between gestures to identify moments of peak change in performances. Three apps are subsequently described that react to this information in different ways, according to interaction paradigms of support, disruption and reward. In Chapter 6, the designs presented in this thesis for networked interaction between an ensemble of touch-screen performers are summarised.

1.4.3 How does this system of apps and agents affect ensemble improvisation?

The app and agent designs discussed throughout this thesis share the goal of enhancing ensemble interactions — a hallmark of free-improvised performance — and thus aim to improve performance outcomes. In chapters 5
1.5. Summary of Contributions

and 7, the designs and results of two formal HCI studies are discussed. These studies were designed to replicate the conditions of rehearsals of improvised music, where performers play a number of performances under different conditions. Both qualitative data from interviews and quantitative results from surveys and performance protocols are used to examine the effects of these musical systems on performers, and on their improvisations. In Chapter 5 three apps with supportive, disruptive, and rewarding responses to this data are compared in a formal HCI study. In Chapter 7 a refined version of this agent system is compared to a direct Graphical User Interface (GUI) system for proceeding through sets of notes in a touchscreen interface.

1.5 Summary of Contributions

The work in this thesis represents an examination of ensemble-focused DMIs of unprecedented scope and detail. Six touchscreen DMIs for iPad computers have been produced, four of which have been publicly released. A performance tracking system, which logs touchscreen performances as well as interacts as an intelligent agent, has been produced, significantly refined, and released as an open-source project. Over 150 interaction sessions with these instruments and agent system have been logged, and three formal studies have been conducted as well as a quantitative meta-analysis ranging over all recorded sessions.

1.5.1 Vocabulary of Percussive Touch-Screen Gestures

A vocabulary of touchscreen gestures used by a group of percussionists is identified in Chapter 3 from a qualitative study of their rehearsals and performances with two touchscreen instruments. Unlike the discrete command gestures used in other HCI research, these gestures are continuous and performative, which means that they can be used to describe the free-form touchscreen performances that take place in this research.
1. Introduction

1.5.2 Performance Tracking Agent

The touch-screen gestures are used to define a performance tracking agent system that classifies performers’ touch-data continuously and in real-time during an interaction session. This agent uses a novel transition-matrix approach to identify moments of peak gestural change that may correspond to the start of natural performance segments observed in improvisation. A protocol of Open Sound Control (OSC) messages is defined to record the interaction of the agent with current and future touch-screen apps. Chapter 4 describes this agent in detail, as well as the results of a preliminary evaluation that includes cross-validation of the gesture classification system and benchmarking with respect to the number of simultaneous iPad performers.

1.5.3 A Repertoire of Touch-Screen Apps

Following the experiences of developing and performing with two prototype touch-screen apps, three apps were revised to interact with the performance tracking agent: BirdsNest and Snow Music (previously used in performance); and Singing Bowls (newly created). The three apps were distinct from a sonic and user interaction perspective, but were also designed to interact with the agent in different ways. Three paradigms of agent-ensemble interaction are proposed that are based on concepts of support, disruption, and reward. Chapters 4 and 6 describe how these apps implement these ensemble-focussed interactions, which go far beyond the commercial standards for ensemble synchronisation during performance.

1.5.4 Study of expert performers with agent-app systems

Performance experiences with the repertoire of touch-screen apps suggested that the system could actually enhance collaborative improvisations, at least with a group of expert performers. To evaluate this assertion, a group of three performers participated in an intensive comparative study during one rehearsal session where 18 performances covering all combinations of the three apps and two versions of the agent were recorded. The results of
1.5. Summary of Contributions

this study, described in Chapter 5 strongly supported one interface, *Singing Bowls*, and its reward paradigm for interaction. However, the intelligent agent did not have a clearly significant advantage over a control agent based on randomly generated interventions.

1.5.5 Study of non-expert performers with agent and direct ensemble interactions

As a significant preference was established for the interface and agent interactions style of *Singing Bowls*, a refined interface was developed to investigate this instrument with new performers. The renamed *PhaseRings* app could operate with the indirect agent-interaction, or a direct ensemble-interaction via an onscreen GUI button. A two-way study, reported in Chapter 7, was performed with four groups of four performers. While the button interface was highly rated, the use of both interfaces simultaneously had the highest ratings. A follow up study comparing this simultaneous condition with a more subtle combination of the two interfaces supported the utility of the agent and demonstrated the emergence of distinct performance styles in the groups.
Ensemble Interaction in Computer Music

Since computer-generated electronic music first became possible in the 1950s, composers and performers have struggled to create musical works that are “live” in the same sense as acoustically produced works (Croft, 2007). At first, computer music was mainly limited to studio productions rendered to tape and live performance simply consisted of playing back recordings. Later, miniaturised and inexpensive computers enabled DMIs that could be brought onto the stage and performed live. One of the most exciting aspects of live musical performance is the on-stage interaction of multiple musicians. A response from computer musicians has been to co-opt readily available mobile computers, such as laptops, phones and tablets, to create ensemble-focussed instruments. Groups founded to perform with such instruments have embraced the new artistic practices, repertoire, and performance techniques that modern DMIs afford.

This chapter will discuss the recent history of ensemble performance with DMIs. Experimentation in this field is rich and varied, with some researchers taking traditional acoustic ensembles as a starting point, and others developing new collaborative experiences only made possible with computer-based instruments. Despite the significant interest in ensemble performance with DMIs, the majority of theoretical models dealing with these instruments focus on the individual interactions of performers. Two models for ensemble interaction will be discussed that have particular rel-
2. Ensemble Interaction in Computer Music

Figure 2.1: Rowe’s model for interactive music systems is divided into three stages: sensing, processing, and response. Simple DMIs could use one kind of sensor as input and then a simple mapping to a synthesised sonic response. More complex systems might have multiple dimensions of sensors and responses, including networked interactions, with a high level musical model in the processing stage.

Relevance for the DMIs implemented in later chapters: Weinberg’s (2005b) Local Performance Network (LPN), where instruments directly message each other during a performance, and the concept of an Ensemble Director Agent (EDA) (Pressing, 1990), where intelligent agent software tracks the performance and makes high-level suggestions and interventions in performers’ DMIs, much like a human conductor. Few examples of this latter model exist in the literature, and automating high-level control over performance structure is still considered by some authors to present an “open challenge” to researchers (Bown et al., 2013, p. 33). At the conclusion of this chapter, a framework for such a model of ensemble interaction will be proposed.

2.1 Models of DMI Interaction

DMIs are systems of (sometimes interchangeable) parts including sensors or input devices, sound generators, and a model for mapping interactions among these components (Jordà, 2005). Due to the potential complexity of these systems, and the potential to expand them with new technologies, models have emerged to understand their possible interactions. Rowe (1993, Chap. 1) defined interactive computer music systems as those where “behaviour changes in response to a musical input.” According to Rowe (1996, p. 51), these systems consist of three conceptual stages, illustrated in Figure 2.1: sensing of musical input, processing where such input is comprehended and mapped to potential output, and response, where sonic (or...
2.1. Models of DMI Interaction

Other) output is synthesised. This model characterises systems as simple as a electronic keyboard to those as complex as autonomous machine listening and improvisation systems.

Bongers (2000) classified DMIs with reference to the humans involved in providing musical input who could be performers in a traditional instrumental setting, audience members in an installation setting, for example, or both. Drummond (2009, p. 127) extended this classification to include systems unique to ensemble situations that could include “multiple performers with a single interactive system; and multiple systems interacting with each other and/or multiple performers.” This final and most complex class of systems is sufficient to describe true ensemble DMI interactions. In this case, multiple musicians each use a separate DMI as their performance instrument, and the DMIs interact not only with their performers but also with each other.

The interactions between individual performers and DMIs — sensing, processing, and responding — are well covered in previous literature (e.g., see Bongers (2000)). Despite recent interest in computer music ensembles, the unique possibilities of using DMIs in an ensemble situation have not been exhaustively investigated even though many examples exist, including The Hub (Gresham-Lancaster, 1998) and Sensorband (Bongers & Sensorband, 1998). Before examining current trends in ensemble performance with DMIs, it is useful to consider what shape this ensemble interaction could take.

First of all, the sensing and response stages of interaction may already be covered in the traditional, individual DMI model, so most of the ensemble-interactive parts of a system will occur in the processing stage. Instruments may share data that they have sensed and form a response based on the input of multiple, rather than individual performers. Rowe (2001, pp. 1-15) has described how interactive music systems could encode a kind of musicianship to understand the melody, harmony, and performance structure of a performance. The task of making musical sense of input could take significant processing effort (Rowe, 1996, p. 55). In electro-acoustic ensemble situations, the sonic output may not be well-characterised by traditional
2. Ensemble Interaction in Computer Music

![Diagram of Rowe's interaction model](image)

Figure 2.2: Rowe’s interaction model applied to a simple DMI running on a mobile device. The touch-screen is the only sensor; touches are mapped to abstract musical data (notes) in the processing stage, and finally synthesised for playback as the response.

music theory; hence, a system may be better directed towards identifying broader structural elements in the performers’ collective input.

One kind of ensemble interaction can be created by direct connections made between individual DMIs used by the ensemble so that the musical gestures of the performers are shared in some way to others in the group. Weinberg (2005b) has described such a scheme as a Local Performance Network (LPN) and has elsewhere emphasised that instruments can be arranged in interconnection topologies that accentuate particular ensemble dynamics, such as creating a group leader, or having each instrument depend on only one other in a loop (Weinberg, 2005a). An alternative to interconnection would be to have a centralised musical agent dedicated to managing ensemble-interaction. Pressing (1990) has suggested that in addition to the “intelligent instrument” role that musical agents tend to hold, other, non-sounding, roles could also be possible:

**The computer as musical director** — Here the computer monitors information put out by the performer(s) and other parts of the system and applies tests, which are used to decide whether to invoke interruptions to ongoing processes, change data transformations, correct errors, issue commands to performers, etc. (Pressing, 1990, p. 23)
While the DMIs focus on their individual performers' actions, an Ensemble Director Agent (EDA), much like an orchestra's conductor, could focus on an overview of the whole group's performance. The director agent's responses need not be at the low level of notes and sounds, but rather, high-level commands returned to the individual DMIs that affect their sense-process-respond models. This kind of interaction, rather than influencing any particular musical phrase, might shape the musical work as a whole. In the following sections, current examples of DMI ensembles and recent work in mobile-music and musical agent interactions will be discussed in terms of these conceptual models.

2.2 Laptop Orchestras

An important recent development in computer music performance has been the emergence of the Laptop Orchestra (LOrk) (Trueman, 2007) in the mid-2000s. LOrks aim to mimic some of the formalism inherent in the symphony orchestra, Indonesian gamelan, and other groups by providing a large ensemble of performers with similar individual laptops and software setups. In the original PLOrk (Princeton Laptop Orchestra) group, the equipment consisted of 15 “meta-instruments” (Trueman et al., 2006). Each meta-instrument consisted of a laptop computer, an audio interface, a hemispherical multi-channel loudspeaker typical of Trueman’s designs (Smallwood et al., 2009), various hardware controller interfaces, and an assortment of digital music software that was used to compose and perform the group’s works. Each performer in PLOrk had a self-contained musical instrument, from control input to sound diffusion. In these LOrks, the electronic sound is typically diffused through small speakers co-located with each performer and directly connected to their laptop system, rather than through large loudspeaker systems. This arrangement allows for a natural balance of the ensemble’s sonic output for both the audience and the performers on stage.

While LOrk hardware setups are usually fixed, the software usually varies between works and is considered to be the “composed” aspect of the musical work. The composer specifies how the performers should control
the instruments as well as the instrument’s sounds and interaction scheme. In fact, the works are often more focussed on defining an interaction scheme for the performers who then improvise their own parts within this framework (Smallwood et al., 2008). One interesting approach has been to use the control inputs of the laptop itself as the interface, rather than a Musical Instrument Digital Interface (MIDI) or other custom control device. Fiebrink et al. (2007) advised composers not to “forget the laptop” (p. 164), suggesting ways to use the keyboard, webcam, built-in microphone, trackpad, and network interface of the portable computers to control sound directly, rather than through a GUI. Those authors even suggested using the “sudden motion sensor” (p. 166), a feature intended to protect the spinning hard disks of 2007-era laptops, as a controller. *Smacking Music* (Fiebrink et al., 2007) asks the performers to strike their laptops to initiate synthesised sounds. Later laptop trackpads were able to sense multiple touch points and Schlei (2010) refined the possibilities for controlling synthesis processes with these everyday devices. Some LOrk performances have even focussed on collaborative live coding interfaces, such as *LOLC* (Freeman & Troyer, 2011; Lee et al., 2011) which merged coding with internet chat systems.

LOrks have often been formed as university courses where participants not only perform in the ensemble, but also research, compose, and develop new works for the group (Trueman et al., 2006). The phenomenon spread with notable LOrks at Stanford (Wang et al., 2009), University of Virginia (Bukvic et al., 2010), and Louisiana State University (Beck et al., 2011). By 2014, over 160 LOrks were in existence (Knotts & Collins, 2014). The participants in these groups have also been involved in managing the fleet of hardware (Bukvic et al., 2010) and developing specific distributions of music software such as *Pd-L2Ork*\(^1\). Over multiple seasons of rehearsal and performance, LOrks have developed a compositional repertoire and pedagogy (Wang, Trueman, et al., 2008). Non-academic groups such as Sideband (Britt et al., 2015) have pursued a performance practice involving a continuing pool of expert performers rather than the frequently changing membership of university groups.

\(^1\)Pd-L2Ork is available at [https://github.com/pd-l2ork](https://github.com/pd-l2ork)
2.3 Mobile Music

One of the most important contributions of LOrks has been to emphasise liveness and ensemble interaction in computer music performance. Laptop music performance is complicated by the containment of players’ performative gestures inside a GUI that is hidden from the audience (Collins, 2003). In response, LOrk composers have emphasised simple gestural controllers, such as Nintendo Wii Remotes, and interactions between ensemble members that mimic those present in acoustic chamber ensembles or orchestras. The self-contained meta-instruments that LOrks use have also enabled them to involve a large number of potential performers without computer music experience, such as school students (B. Sawyer et al., 2013).

2.3 Mobile Music

Simultaneously with the development of LOrks, the concept of mobile music has gained currency (Gaye et al., 2006). Broadly speaking, the field of mobile music covers research into all sound-creating portable devices, but of particular interest to the present work is how powerful mobile computers, such as smartphones and tablets, have been co-opted as musical instruments for ensemble performance. D. A. Williams (2014) reported the experiences of an educational tablet ensemble and concluded that the mobile devices suggest exploratory and collaborative modes of music making. In a study of music teachers’ experiences, P. Riley (2013) noted that touchscreen instruments are easy for beginners and appropriate for improvisation or composition activities. The many affordances of mobile devices, such as their multiple sensors, convenient form-factors, and growing cultural importance make them ideal for experimenting with multi-modal DMI interaction (Tanaka, 2010). The affordances of a mobile DMI can be perceivable (Norman, 1988), such as piano keys displayed on a touchscreen, or hidden affordances (Gaver, 1991; Gibson, 1977), such as sounds triggered by movement or location. Smartphone ensembles have emerged that, like LOrks, explore a combination of composition with instrument design and software development to create new music. The results of such mobile music investigations are now used in the studio and on the concert stage (M. Jenkins,
2. Ensemble Interaction in Computer Music

In the early 2000s, mobile devices became powerful enough to perform similar real-time sound synthesis as had been possible on PCs for some years. Geiger’s PDa (Geiger, 2003) port of the Pd (Pure Data) computer music environment (Puckette, 1997b) to Linux-based, handheld computers proved that this was possible. Later work with this system set out basic mappings for touch-screens to control digital instruments (Geiger, 2006). In turn, pdPod adapted PDa to run on hacked Apple iPods (Kaltenbrunner, 2007), one of the most popular mobile devices of this era. While the processing power of these devices was very limited by modern standards, the potential for them to be used as standalone digital musical instruments was recognised. Eide’s (2008) Slättberg instrument enclosed an iPod into a kind of noise music hurdy-gurdy (Kirn, 2008), where a crank activated the iPod’s built in touch-wheel interface. Schiemer and Havryliv (2007) created the Pocket Gamelan system where the loud internal speakers and compact size of mobile phones was used to create instruments that could be swung around performers’ heads for a microtonal, doppler-shifting effect.

2.3.1 Mobility in Mobile Music

The portability of mobile devices also suggested musical experiences that rely on location and many such locative mobile DMIs have been introduced (Kirisits et al., 2008). Gaye et al. (2003) described Sonic City, a wearable rig of microphone, headphones, sensor interface and a miniature laptop that would play music in response to data collected at the user’s location. Tanaka and Gemeinboeck (2008) used GPS and digital compass sensors to direct groups of smartphone-wearing participants on “missions” (p. 179), while their location and sounds and images from the devices’ microphones and cameras are used to drive an audio-visual artwork. The Ocarina iPhone app was designed to connect users across the world in a melodic social network. Snippets (up to 30 seconds) of users’ performances were uploaded and could then be browsed, played back, and “hearted” by other users (Wang, 2014).

Just as the fixed meta-instrument setups in LOrrks had enabled the de-
2.3. Mobile Music

Figure 2.3: Stanford’s MoPhO (Mobile Phone Orchestra) performing with iPhones and hand-mounted speakers at NIME 2010 in Sydney, Australia.

Development of repertoires of laptop orchestra works, the many affordances of smartphones suggested their use in ensembles where DMI design occurred in parallel with collaborative performance (Wang, Essl, & Penttinen, 2008). The emergence of multi-touch screens has emphasised explorations of expressive touch user interfaces which can be rapidly configured into many kinds of interface paradigms (Oh et al., 2010). Wang et al. (2014) have described a repertoire of works developed for Stanford’s MoPhO (Mobile Phone Orchestra), while Essl and Rohs (2009) previously used the input affordances of mobile phones as a starting point for an overview of musical interaction with these devices.

Mobile music ensembles have pursued several means of diffusing the sound from their portable instruments. One approach has been to use mobile devices to control synthesis process on a centralised computer connected to large loudspeakers, as used in Swift’s (2013) Viscothèque. Another approach used by Stanford’s MoPhO group (Oh et al., 2010) and University of Michigan’s Mobile Phone Ensemble (Essl, 2010) involved using small battery-powered speakers mounted on each player’s hands. A MoPhO performance in Figure 2.3 shows that this approach allows the performers to
move freely in a concert hall although at significant cost to acoustic power.

2.3.2 Development Tools for Mobile DMIs

A number of development frameworks have emerged for rapidly creating mobile music applications by allowing easier access to synthesis algorithms, and the sensors and touch-screens of mobile devices. Bryan et al. (2010) designed the MoMu toolkit, an Objective-C development framework, while MobileFaust enabled Faust Digital Signal Processing (DSP) code to be integrated into mobile apps (Michon et al., 2015). libpd\(^2\) allowed Pure Data to be used on several platforms to build the audio components of an application, while native frameworks are used for everything else (Brinkmann, 2012; Brinkmann et al., 2011).

An alternative to developing native apps appeared with the release of RjDj (Sterling, 2008), an iPhone app that could play back interactive computer music scenes developed in the Pure Data environment. Customised RjDj scenes have been used by artists for musical performances; for example, Tanaka’s (2010) four hand iPhone performances. Similar systems, such as urMus (Essl & Müller, 2010) and TC-11 (Schlei, 2012), have allowed mobile DMIs to be designed within an app. MobMuPlat made use of libpd in a mobile application that can open Pure Data files directly on the user’s device for rapid creation of mobile DMIs (Iglesia, 2013). Ensemble works using MobMuPlat such as Iglesia’s (2014) *Reality Denied Comes Back to Haunt* have now been performed by Sideband and other laptop ensembles.

2.4 Other Systems for Ensemble Improvisation

Although a minority of the repertoire for LOrks and mobile ensembles has been strictly composed, these types of ensembles tend to emphasise improvised performances from “sonic sculptures, to structured and free improvisations” (Wang et al., 2014, p. 457). This may be partly due to the compositional effort going towards developing new DMIs, interfaces, and

\(^2\)libpd is available online at: https://github.com/libpd
ensemble interactions rather the structure and content of the musical work. An emphasis on improvisation is also prevalent in other exploratory computer music ensembles and interfaces outside of the LOrk domain.

One approach has been to produce distributed interfaces where each performer controls one part of a broader computer music environment. This was the case in Riddell’s *HyperSense Complex* (Riddell, 2005) where three performers improvised using finger-mounted flex-sensors connected to a common synthesis system. This work followed other interfaces for finger sensors, in particular Laetitia Sonami’s *Lady’s Glove* (Bongers, 2007) and other works that appropriated Mattel’s 1989 PowerGlove, a gestural version of the controller for the Nintendo Entertainment System (Riddell, 2005). HyperSense Complex stood out by posing the ensemble as a crucial part of the work; the computer musical environments were designed for six hands improvising, not two. Riddell (2005) commented that an initial process with a simple mapping allowed the performers “to understand how we would work together” (p. 126), this enabled works with more complex ensemble dynamics that, for instance, required “two performers to trigger events along transparent cyclic rhythmic structures while the third controlled effects” (p. 127).

Another theme in ensemble improvisation with computer instruments has been to have a common interface that multiple users are able to control simultaneously. The *Daisyphone* (Bryan-Kinns, 2004), is such a system for ensemble improvisation over a computer network. Multiple participants have access to a common GUI where musical events are marked on a circular sequencer space with a continually moving play-head. The users can add and change musical events together and thus collaboratively arrange a continually evolving musical work. A similarly collaborative interface is the *Reactable*, developed by Jordà (2008). The Reactable consists of a circular tabletop projection surface along with tangible objects representing components of a music synthesis system. Analogously to patching a modular synthesiser, Reactable performers place objects on the table and bring them into close proximity to create chain of signal generators, controllers and effects. In what Jordà (2003) calls a “sonigraphical” design, the Reactable
2. Ensemble Interaction in Computer Music

displays the signal connections between tangible objects with oscilloscope-trace-like links that illustrate the audio signal at each stage of the patch. The Reactable’s underlying software and hardware system, reacTIVision (Kaltenbrunner & Bencina, 2007), can not only distinguish between tangible objects using a series of visual markers, but recognise their location, rotation, and finger touches on the table surface. The finger touches and rotation are used by the Reactable to set parameters for each tangible synth object. As the Reactable’s interface is an arrangement of real-world objects, multiple users simply collaborate by constructing one or more signal chains together, and are free to manipulate each other’s settings. Xambó et al.’s (2013) study of collaborative Reactable improvisations revealed complex negotiations between shared and personal space, as well as the peer learning that occurred while jamming with this interface.

One final model of ensemble practice in computer music is where performers develop their own instrument in order to participate in the group performances. In the late 1970s, The League of Automatic Music Composers pioneered interconnected computer music systems in the San Francisco Bay Area (Bischoff et al., 2007). They performed with individually developed systems using early KIM-1 PCs and shared musical information over serial connections to create ensemble-interactive works (Bischoff et al., 1978). In the 1980s, members of this group formed The Hub, and continued to create idiosyncratic shared-data compositions using newer MIDI standards (Gresham-Lancaster, 1998). A very recent example of this concept is Rosli et al.’s (2015) feedback ensemble that demonstrates an almost perverse instrumental connection by providing an instrumented audio feedback network between the performers, each of whom are asked to provide their own musical device with an audio input and output. During the performance, the feedback network dynamically maps audio through and among the participants in a variety of changing topologies. Similarly to a LOrk stage setup, a sound output for each performer is diffused locally through a personal speaker; however, the actual sound output can be strongly affected by those sounds produced before them in the currently active signal mapping.
2.5 Interacting with Agent Systems

All of these systems have emphasised performances with an ensemble collocated in both space and time, however, it has been acknowledged that computer supported cooperative work (CSCW), can occur remotely, or asynchronously (Johansen, 1988). Networked systems enable performances split between multiple locations, for example, SoundWIRE’s performance of Terry Riley’s *In C* involved acoustic and laptop orchestras at venues in California and China (Cáceres et al., 2008). Shifting musical collaboration across time, Smule’s *Guitar!* app allows users to play along with performances previously recorded by others (Smule Inc., 2013).

2.5 Interacting with Agent Systems

In LOrks and other ensembles employing new interfaces for musical expression, intelligent agents have often been used in a variety of roles. Agents were initially conceived as intelligent software that could independently carry out operations on behalf of a user and have often been envisioned as communicating with users using natural language (A. Kay, 1984). Jordà (2005) has observed that computer-based DMIs can be imbued with a level of intelligence that enables them to interact with human performers along a continuum of independence, from reactive instrument to fully autonomous ensemble member. Towards the autonomous end of this spectrum, DMIs become musical agents.

Agent systems that participate in performances have been explored in a variety of contexts. Lewis (2000) designed the *Voyager* system in 1986 as an improvisation partner for a human performer, which both reacts to MIDI transcriptions of the musician’s performance and generates its own performances independently. The *Cypher* system developed by Rowe (1992) uses multiple agents that listen and respond to either composed or improvised input from live musicians. Eigenfeldt and Kapur (2008) described a system of agents that control an ensemble of twelve robotic percussion instruments. A. Martín et al. (2011) designed a toolkit for creating musical agents that would act as improvisation partners for musicians on other instruments; here, the toolkit would observe a human performance on an in-
2. Ensemble Interaction in Computer Music

terface for adjusting parameters in electronic instruments and automatically build a statistical model to imitate the performance. Performing agents can also be modelled on particular musicians, such as the model of Stevie Ray Vaughan’s performance described by Vempala and Dasgupta (2007). Other agents have been designed to “continue” a human performance by analysing and imitating their style of playing (Pachet, 2003). In recent times, many systems featuring and advocating for agent performers have been presented at Musical Metacreation workshops (Bown et al., 2013). These events have led to the development of a taxonomy of musically metacreative behaviours that define these behaviours in order of increasing autonomy (Eigenfeldt et al., 2013).

As previously mentioned, an autonomous agent could assume roles other than that of a performer. While Pressing (1990) suggested that an agent could be an ensemble director, such agents have not been explored to the same depth and extent as performing agents. The contribution of ensemble director agents has tended towards low-level roles such as time-keeping, where, like a metronome, they send clicks to other agents or DMIs to keep a synchronised beat. Trueman (2007) has described works for PLOrk where one laptop system is designated to be the conductor and supplies a metronome signal to the other laptops over a network. In these works, the conductor laptop, operated by a human performer, was also used to control other high level aspects of the performance, such as the “volume of particular sections within the orchestra” (p. 176). This arrangement has also been used with mobile devices, d’Alessandro et al. (2012) describe digital choir performances where a central conducting interface communicates with an ensemble of iPhone players. Smallwood et al. (2008) questioned whether a LOrk conductor should be human or “a program operating over the network” (p. 15), but the latter case is not explored except as a metronome. Eigenfeldt and Kapur (2008) described a conductor agent that directed ensembles of performing agents by providing a metronome pulse and communicating global parameters, such as the density of notes, from a user interface. These examples of ensemble director agents certainly play a valuable musical role, but they aren’t designed to automatically react to the
ongoing musical output of the group like the improvising agents in *Voyager* or *Cypher*.

The role of director in an ensemble involves decision making and interaction at a high musical level as opposed to the low level of individual notes and phrases that performance agents must deal with. High level decisions may include aspects of performance such as large-scale determination of tempo, dynamics, balance, texture, tonality, and orchestration, all of which contribute to the structural content of a musical work. However, agent control of musical structure has been largely unexplored in the literature with some authors stating that “large-scale musical structure is something that is the most difficult for an artist to delegate to a system” (Eigenfeldt et al., 2013, p. 47). A reason for this difficulty might be the lack of technical foundations for automating high-level musical decisions. Another explanation could be that musicians and researchers prefer to control high-level aspects of their systems. This is often the case with ensembles of agent-based performers where the creator takes on a role of conductor or live-composer while agents make low-level decisions. As observed by Eigenfeldt et al. (2013, p. 47), “Artists within this nascent field will always want to create works of which they feel are artistically strong, and may feel the need to retain interactive control of their systems.” In fact, delegating long term structure to agent systems has been identified as an open challenge in computational creativity (Bown et al., 2013).

### 2.6 Conclusions

In this chapter, the history of ensemble interaction in computer music has been briefly reviewed. Early systems from the late 1970s were idiosyncratic and hand-built from the available computers of that era (Bischoff et al., 1978). Performers hacked both software and hardware as part of their artistic process. More recent ensembles of laptop and mobile device performers have used off-the-shelf hardware, and well-understood development tools. These more recent groups have often been organised as music courses in academic institutions where they have engaged with many performers and
2. Ensemble Interaction in Computer Music

Figure 2.4: Integrating an LPN (Local Performance Network) of mobile DMIs into the interactive music system model. Network connections are additional sensors and response outputs, while the processing stage alters the mapping of touch to sound due to these interconnections.

composers to generate a wide variety of repertoire. The multiple affordances of mobile devices, and their increasing popularity in educational, recreational, as well as professional music-making, justifies further investigation of the concept of a mobile music ensemble.

Ensemble-focussed DMIs have featured network functionality to allow communication between performers or shared control over a common interface. There are, however, only a few instances in the literature where such network features are examined to the same depth as the interfaces, mappings, and synthesis outputs of individual DMIs. In this chapter, two architectures for connecting an ensemble of DMIs have been discussed which would be particularly applicable to a mobile music ensemble: Local Performance Networks (LPN) and Ensemble Director Agents (EDA). In a Local Performance Network (LPN), DMIs connect to each to share information
during the performance. In this architecture, illustrated in Figure 2.4, the individual DMIs must include networked communication in their internal model of Sensing, Processing, and Response. A mobile-music DMI featuring LPN interactions will be introduced in Chapter 3.

A more sophisticated model for ensemble-focused systems would be to include an Ensemble Director Agent (EDA) in an ensemble. While using an EDA has been theorised, their implementations have often been more simplistic than agents designed to perform as improvisation partners or generative musicians. The requirements for an EDA are similar to Rowe’s interactive music system, and a design model is shown in Figure 2.5. The EDA must listen to the performers in the group in some way; it must process the input to form a high-level model of the performance; it must decide when to interact according to a model of ensemble musicianship; and its response should be an inaudible signal to the performers and DMIs. An
2. Ensemble Interaction in Computer Music

EDA would have responsibility for the large-scale structure of a performance rather than the low-level notes. The agent should be aware of broad changes in the ensemble’s performance as a whole and not necessarily of the performers’ individual notes and sounds. When the agent responds, it should be to influence the long term shape of the performance, or to encourage or discourage certain ensemble behaviours.

A design for an EDA according to these requirements will be described in Chapter 4. In later chapters, its behaviour in performances with ensembles of human musicians will be examined. Experiments will be described that compare different designs for EDA interaction with LPN interactions. As mentioned previously, the idea of large-scale structural interactions requires technical foundations. What should such an agent be listening for? When and how should it respond? In the next chapter, a process of developing an artistic practice with a mobile-device ensemble will be described in order to characterise and understand the possibilities of these performances and provide a framework for a true ensemble-interactive agent.
Ensemble Metatone: A Touch-Screen Ensemble

This chapter describes the first prototype touch-screen instruments developed for this project, and the earliest study examining their use. To explore and identify how the instruments could be used in an ensemble setting, a group of expert performers was brought together to improvise new iPad music in a series of recorded rehearsals and to develop a well-defined artistic practice. This rehearsal series and subsequent concerts are subjected to a qualitative study in this chapter to identify a vocabulary of touch-gestures that is used in later chapters to automatically track touch-screen performances. The study also yields new insight into how the new instruments complemented the ensemble’s existing experience in percussion and suggests how iPad-only groups can function. Parts of this chapter have previously been published in three peer-reviewed papers: “Exploring Percussive Gesture on iPads with Ensemble Metatone” (C. Martin et al., 2014a), “MetaTravels and MetaLonsdale: iPad Apps for Percussive Improvisation” (C. Martin et al., 2014b), and “Making Improvised Music for iPad and Percussion with Ensemble Metatone” (C. Martin, 2014a).

It is generally accepted that the performer is the most important stakeholder in developing new DMIs (O’Modhrain, 2011), and evaluations of DMIs usually focus on how they are used by performers. It follows, then, that an ensemble-focussed DMI should be evaluated by an ensemble. To this end, in February 2013, a call was put out to form an ensemble of
3. Ensemble Metatone: A Touch-Screen Ensemble

Figure 3.1: Ensemble Metatone in an early rehearsal with the MetaTravels app. From left to right: Charles Martin (CM), Jonathan Griffiths (JG), Christina Hopgood (CH), and Yvonne Lam (YL).

experienced percussionists in Canberra specifically to perform with iPad instruments developed in this research. The author, Charles Martin (CM), and three percussionists who had studied at the ANU School of Music, Christina Hopgood (CH), Yvonne Lam (YL), and Jonathan Griffiths (JG) joined the group, which was called Ensemble Metatone.

The goals of Ensemble Metatone were twofold. First, to use the iPad instruments in a series of recorded rehearsals, not just to evaluate the instruments themselves, but to research the process of performing with touchscreen instruments in general as distinct from the percussion instruments with which the performers had trained. The second goal was to prepare music for iPad and percussion to be presented at professional concerts and recorded for album releases, and to pursue an artistic practice specialising in touch-screen instruments and percussion.

It should be noted that the research discussed in this chapter was practice-led (Candy, 2006) and that the author participated in the artistic process as well as designing the apps that were used. In creative arts
3.1. Percussionist-Centered Design

research it is a well-established methodology for a practitioner/researcher to examine their own practice, particularly in “new media and music” (H. Smith & Dean, 2009, p. 8). In HCI, the methodology in this chapter could be called autobiographical design. While this process does not prove generalisable results, it allows an in depth examination of long-term usage of a system at the early stages of development (Neustaedter & Sengers, 2012).

This chapter concerns the early part of Ensemble Metatone’s process of artistic development as the group first encountered a touch-screen DMI and started to develop a style of improvised music to interrogate the affordances of this instrument. Two iPad apps used by the group in this period will be discussed in this chapter: MetaTravels, a prototype app developed for the first rehearsals; and MetaLonsdale, which, along with a more focussed artistic idea, contained the first networked features used with the group. In the next section, the motivations for using percussionists to explore a new touch-screen DMI will be discussed. Later sections describe the design of the apps and the results of the rehearsal and performance study.

3.1 Percussionist-Centered Design

Percussion is a musical practice more defined by the methods of interacting with instruments rather than the instruments themselves. Percussionists perform by “striking, scraping, brushing, rubbing, whacking, or crashing any and practically every available object” (Schick, 2006, p. 5). Blades (1992) discussed the earliest percussion instruments, that is, idiophones, where the body of an instrument creates the sound, rather than an air column or string. He divided them by their method of interaction (p. 36): “shaken”, “stamping” (played with the hands or feet), “scraped”, “concussion” (two parts struck together), and “struck” (with a stick or non-sounding implement). These descriptions match taxonomies of modern percussion instruments (e.g., G. Cook (1997)) and focus on the mode of interaction with the instruments rather than their physical design.

Percussionists are accustomed to exploring non-traditional objects to create music and they use these percussive gestures to coax wide varieties
Ensemble Metatone: A Touch-Screen Ensemble

of timbres and musical gestures from simple instruments. For example, performers of Xenakis’s (1975) *Psappha* or Feldman’s (1965) *King of Denmark* must design their own multi-instrument setup to fit each composer’s specifications. To meet the requirement for metal instruments, for example, a performer might find a car’s suspension spring, a saw-blade or create a unique object from scratch. For percussionists, free-improvisation is often a process of gestural exploration, discovering new sounds from traditional and non-traditional instruments and responding to other sounds in an ensemble. This percussive approach to investigating the hidden affordances of new instruments can be applied to touch-screen computers which can also be struck, scraped, and rubbed with fingers and hands.

While it is well established that popular touch-screen devices can be used to make music, mainstream creative frameworks for their use are limited (tapping virtual piano keys in the Apple Inc. (2011) *Garageband* app, for instance). The percussive affordances of these devices motivates an exploration of their use in a modern percussion ensemble to establish more varied modes of interaction. A process where such exploration is used as part of the development of a DMI could be termed *percussionist-centred* design.

New touch-screen DMI designs have often focussed on engaging novice or non-musical users, such as Smule’s *Magic Fiddle* iPad app (Wang et al., 2011), or Ren et al.’s (2012) tabletop virtual instruments. In contrast, the iPad instruments described in this research were developed to harness the existing exploratory skills of professional percussionists. The musical works subsequently developed were targeted towards audiences of experimental and contemporary classical music.

The field of computer music abounds with creative examples of touch-sensor-based instruments, but the goals of this project placed constraints on what could practically be used within the percussion group; the touch-screen devices needed to be self-contained, durable, and easy to provision with software updates. Many tablet computing devices meet these requirements, but Apple iPads were chosen as the computer instrument in the ensemble because of their widespread adoption and established develop-
Figure 3.2: JG’s iPad set on a stick tray with a variety of percussion instruments and mallets. iPads fit well into the requirements of a percussion ensemble and were easily adopted by Ensemble Metatone.

ment frameworks for musical applications. Even though iPad touch screens have drawbacks such as a lack of pressure sensitivity or physical feedback, their physical dimensions mimic that of some simple percussion instruments such as the woodblock or tambourine.

A potential limitation of using iPads for percussion performance is the latency between screen touches and output sound. Ng et al. (2012) note that commercial touch-screen systems have a display latency somewhere between “50-200ms” (p. 453) while Michon et al. (2015) report a touch-to-sound latency of 45ms on an iPad 2 and just 36ms on a more powerful iPhone 5 (p. 397). While very low latency (< 10ms) is desirable in an interactive music system (Wright & Brandt, 2001), Mäki-Patola and Hämäläinen (2004) found that playing style affects the detection of latency and suggest that the type of music and nature of the sound affect latency tolerance. Despite the iPad’s limitations, the study described in this chapter shows that they
can be played musically and expressively and that touch-to-sound latency appears to be tolerable for ensemble improvisation.

HCI studies of gestures on touch screens have been conducted for tasks such as activating a shortcut in a smartphone (Bragdon et al., 2011; Ouyang & Li, 2012), manipulating virtual objects on a table-based interface (Hinrichs & Carpendale, 2011) and controlling a video performance (Hook et al., 2013). Many of these studies have characterised gestures that emerged as part of users’ interactions with a touch interface. In a similar way, the work described in this chapter examines touch-screen gestures that emerge when iPads are introduced into a modern, free-improvisation percussion ensemble. Qualitative analysis of a series of the group’s rehearsals and discussions reveals a vocabulary of new gestures invented by the musicians. These gestures were used by the musicians to creatively interact with, and expand the power of, two specially-designed iPad percussion apps. Unlike other studies such as the work of Wobbrock et al. (2009), the gestures observed in the present research are generally two-handed and combine many touches over a number of seconds to express sustained musical ideas.

Two apps were produced for the group’s initial projects: MetaTravels and MetaLonsdale (shown in Figures 3.3 and 3.4), each becoming associated with an improvised musical work of the same name. Both apps used the same percussion-inspired interaction scheme allowing access to pitched percussion sounds and field recordings. The majority of the iPad screen was a performance surface and there were few graphical UI elements. Tapping the screen produced short sounds at a pitch determined by the location of the tap. Swiping triggered continuous field recordings with the velocity of the swipe directly mapped to the volume of the field recording. Both apps featured simple delay functions that repeated tapped notes, and switchable auto-play features that algorithmically produced background sounds. These features were activated by switches in the user interface. A button on both apps allowed the performer to shuffle the available sounds. The apps allowed logs of performances to be captured by sending records of each touch event to a server across a local network. In the following sections, the designs of these apps will be discussed before reporting the results of
studying the group’s early rehearsals and performances.

3.2 MetaTravels: A Prototype Touch-Screen Instrument

*MetaTravels* was developed to be the main instrument for Ensemble Metatone, and was intended to contain sufficiently expressive sonic possibilities to support long improvisations without relying on other instruments or software. This goal set the app apart from the author’s previous touch-screen DMI, *Snow Music*, which was designed to complement acoustic percussion instruments (C. Martin, 2012a). MetaTravels allowed performance with a range of field recordings and sampled percussion sounds in a free-form touch screen area. Most of the screen was taken up by an image background, while four GUI switches on the lower edge of the screen could be used to control special features in the interface.

Touch interaction with MetaTravels was divided between two fundamental touch events exposed in Apple’s iOS operating system (Apple, 2015). `touchDown` events are triggered every time the user touches the screen and these events triggered a sound with a percussive envelope\(^2\). `touchMoved` events are triggered when a user has already touched the screen and moved the touch point. These events were used to control a continuous sound

---

\(^1\) *Snow Music* was developed as part of the author’s masters research where a process of integrating mobile music devices into an existing free-improvised percussion practice was subjected to ethnographic investigation (C. Martin, 2012b). The mobile devices comprised smartphones, tablets, and a DIY vibraphone microphone system (C. Martin, 2013). The DMIs studied at that time took a supportive role in the performances and did not include any ensemble-focussed features. MetaTravels features similar “percussive” control over field recordings to *Snow Music*, but was developed from scratch to support different types of sounds (including pitched samples) and the collection of touch-data. The quantitative research methodology introduced in later chapters and the focus on technology to support ensemble interaction distinguishes this thesis from the earlier ethnographic research by the author.

\(^2\) In audio synthesis, an envelope is the shape of the waveform used to control the amplitude of a sound, i.e., envelopes shape continuous sounds into separate notes. A percussive envelope has a short attack, no sustain, and a long decay, like the sound of striking a large gong.
3. Ensemble Metatone: A Touch-Screen Ensemble

![Image](image.png)

**Figure 3.3:** Screenshot of the *MetaTravels* app running under iOS 7. The white circle represents the user’s touch point while red circles are looped touches. Four switches at the lower left corner control algorithmic background sound generators and a looping feature, and a button shuffles the sounds available to the player from the app’s palette. The text at the bottom of the screen shows the status of the logging functionality. A video demo of MetaTravels is available at: [http://dx.doi.org/10.5281/zenodo.51814](http://dx.doi.org/10.5281/zenodo.51814)

with amplitude proportional to the speed of the moving touch. Apple’s `UIPanGestureRecognizer` class was used to keep track of the movements of a single touch, so that one continuous sound and many taps could be performed simultaneously.

MetaTravels and all other apps developed for Ensemble Metatone were developed in Objective-C using Apple’s iOS frameworks. Audio synthesis was performed with `libpd` (Brinkmann et al., 2011), a C library version of the Pure Data (Puckette, 1997a) audio programming language. Pure Data programs are generally developed and operated in performance through a GUI interface, `libpd` allows these files to be opened and operated programmatically through an Objective-C interface (among other possibilities) and
3.2. MetaTravels: A Prototype Touch-Screen Instrument

for the synthesis audio unit to be managed in the same way as other Core Audio components in iOS. For the apps in this thesis, the synthesis components were composed in the Pure Data GUI, but operated programmatically through libpd in performance.

3.2.1 Touch-Screen Interface

The free-form playing area of MetaTravels allowed performers to tap to create notes anywhere on the screen and there were no markings in the interface to indicate the volume or pitch of sounds. MetaTravels used a radial scheme for assigning pitch to tapped notes. Taps in the centre of the screen were assigned the lowest pitch (MIDI note 30) with pitches increasing linearly as the distance from the centre increased with the highest notes located in the corners of the iPad screen (MIDI note 94). Pitches of notes were quantised to integer MIDI notes, corresponding to equal-temperament, chromatic pitches. Volume\(^3\) was assigned according to the vertical distance of taps from the bottom of the screen. The volume ranged from 25 at the bottom of the screen to 121 at the top.

This scheme for assigning MIDI notes to on-screen taps is quite different from those available in commercial touch-screen music apps. These usually favour either skeuomorphic onscreen models of keyboards and other instruments, as is the case with Garageband (Apple Inc., 2011), or grid models for abstract musical interfaces, as with Mugician (Fielding, 2010). The Bloom app by Eno and Chilvers (2008) is a rare exception with an unmarked 2-axis interface for producing notes. The vertical axis in Bloom determines pitch from a scale and the horizontal axis determines volume.

MetaTravels was loaded with a selection of field recordings made in Sweden and Japan as well as samples of an almglocken (tuned cowbell), singing bowl, and three different cymbals. Taps on the screen played either field recordings, instrument samples, or both at the same time. Moving touches always played field recordings. The particular samples that were played by taps and moving touches in the app could not be directly specified

\(^3\)Velocity in MIDI parlance, with a seven bit range from 0 (silence) to 127.
by the user, but a GUI reset button was present on the screen which would shuffle between the samples as well as increment through the three tap-modes (field recordings, samples, and both).

The rationale for the reset button and the unmarked pitch interface was to eliminate menus and configuration screens from the app in favour of an exploration model of performance. Rather than selecting sounds and pitches before playing them, the user would have to experiment by playing the app and shuffling the sounds only if they were not satisfied with the current selections. This model emphasised listening and interaction with the interface rather than operating a menu-heavy user interface as is present in much music software.

3.2.2 UI Features

MetaTravels featured four UI switches in the on-screen interface that could be used to access special features in the performance. Three of these switches toggled algorithmic sound generators that would continually produce a backing soundscape of notes from three different algorithmically produced sound sources. Each of the three sound generators played sparse clusters of notes — short sections of field recordings or instrumental samples with percussive envelopes applied. These sound generators were similar to those that had previously been implemented in the Snow Music app, and shown in previous research to help structure performances (C. Martin, 2012b).

The final UI switch controlled a looping function. This feature repeated each note tapped by the performers after a delay of five seconds between five and fifteen times. The actual looping time and the position of the tapped notes was designed to change slightly after each repeat. The loop time would increase between 0 and 0.05 seconds after each repeat, and the touch point would move towards the centre of the screen.

The musical result of these changes in looped notes was that a short melody would undergo noticeable sonic changes as it was repeated. The first few repeats would be recognisable with only subtle changes in rhythm.
3.3 MetaLonsdale: A Networked Touch-Screen Instrument

and pitch. After a few repeats, the notes would play at a much lower pitch (as the touch-points move towards the centre of the screen), and the rhythm would have become quite different. This process would continue until the looped notes exhausted their assigned number of loops. In this way, the looping feature worked differently to typical digital delay effects that repeat sounds in the same rhythm at each delay with a volume that reduces after each repeat at a constant rate.

A final important feature of MetaTravels was its ability to log touch interactions to a computer on the local network during a performance. This feature was crucial to the research outcomes of Ensemble Metatone’s performances, and is more fully discussed in Section 3.4 (p. 44).

3.3 MetaLonsdale: A Networked Touch-Screen Instrument

Soon after Ensemble Metatone began their early rehearsals, MetaLonsdale (shown in Figure 3.4) was developed for a site-specific performance in the Everything Nothing Projects gallery in Lonsdale St Traders, Canberra, a pop-up shopping area. In preparation for the performance, field recordings and photographs were taken in the shopping area and street and samples were taken from a variety of percussion instruments. The earliest version of MetaLonsdale was simply a clone of MetaTravels with these new audio resources replacing those used in the earlier app and a random image from Lonsdale St Traders as the app background. The three algorithmic audio generators in MetaTravels were simplified to one in MetaLonsdale, with this feature named *autoplay*.

An important new feature of MetaLonsdale was the limitation of pitches to diatonic scales, rather than the chromatic scale available in MetaTravels. An Objective-C model of common musical scales\(^4\) was created that would allow screen positions to be easily mapped to pitches from a required scale. A sequence of scales and root keys was chosen for MetaLonsdale: F mixolydian, F# lydian, and C lydian #5, and performers could progress through

\(^4\)The ScaleMaker class currently supports the major modes, melodic minor modes, whole-half (octatonic) scale, and whole tone scale.
Figure 3.4: Screenshot of the MetaLonsdale app. The app is broadly similar to MetaTravels but contains audio material focussed on cafés of Lonsdale St Traders. The reset button (labelled “sounds”) is in the lower right of the interface along with a text label showing the active scale for pitched sounds. A demo of this app is available at: http://dx.doi.org/10.5281/zenodo.51818

this sequence by tapping the reset button. This allowed performances to have a sense of harmonic progression as the scale changed during improvisation. Rather than advance one scale with each tap, MetaLonsdale only changed scale in response to 25% of button presses. The app did, however, shuffle the sound material with every reset tap, as with MetaTravels. A text label in the app showed the scale that was currently activated. When the app advanced to a new scale, the image backdrop was changed.

The premiere performance with MetaLonsdale took place at the Everything Nothing Projects gallery in Lonsdale St Traders in July 2015 with the app used in a duo between Charles Martin and Christina Hopgood. In this performance, although there was no connection between the scales on each of the two iPads, the scales happened to be almost always at the same point in the progression as the performers tapped the reset button at roughly the same frequency. Anecdotally, the performance was satisfying
3.3. MetaLonsdale: A Networked Touch-Screen Instrument

Figure 3.5: Christina Hopgood and Charles Martin performing the premiere improvisation with MetaLonsdale at Everything Nothing Projects gallery, Lonsdale St Traders, Canberra. Image © R. Thomson 2013, reproduced with permission.

and the performers arrived at a natural conclusion for the improvisation after both iPads had completed one cycle through the scales, returning to the starting tonality.

3.3.1 Syncing Harmony

In the Everything Nothing Projects performance, both iPads serendipitously ended up on the same scale most of the time. However, it was clear that this would not always be the case, particularly in improvisations with more performers who may reset the sounds at different rates. To create performances where the app’s harmonic progression was clearly communicated to the audience, a networked functionality was developed for MetaLonsdale to synchronise changes to the scale. This was designed so that when any member of the ensemble triggered a scale change on their iPad, the scale on the rest of the ensemble’s iPads would change as well.
To accomplish this syncing feature, the apps automatically advertised themselves as offering a Bunker service on a local WiFi network and browsed the network to find the other iPads present in the ensemble. In this way, the apps formed an LPN of connections, and were able to synchronise aspects of their interface during the performance. Whenever an app updated its scale, it sent an OSC message to each of the other apps in the ensemble which updated their interface as well. This system was also used to mirror other interface changes across the ensemble. Each tap of the reset button and each change in switch state was communicated to the whole ensemble of iPads. While scale messages were always applied, the other apps randomly choose to react to messages for looping, autoplay, and shuffling the sounds 20% of the time. This partial synchronisation of features was designed to assign and reassign ensemble members into musical sections as the performers found their instruments matching up functionality and sounds throughout each performance.

3.4 Recording Touch-Screen Data

An important, research-focussed feature of MetaTravels and MetaLonsdale was the ability to simultaneously record touch-screen interactions of each member of the ensemble on a local server. This was initially accomplished through messages sent in OSC format (Freed & Schmeder, 2009) to an application written in the SuperCollider computer music environment (McCartney, 2002), and in later rehearsals by OSC-Logger, a purpose-built, standalone Mac OS X application. The collected information included touch data from the performers’ free-form gestures on the screen, changes to looping and soundscape switches, activations of the reset button, as well as values returned by the iPads’ 3-axis accelerometers, sampled at 10ms intervals.

This information constituted an extremely detailed recording of the performances although, at the time, the later uses of the data were unknown. As will be discussed at greater length in Chapter 8, the data format was influenced by the TUIO protocol (Kaltenbrunner et al., 2005), as well as
3.4. Recording Touch-Screen Data

Figure 3.6: Raw data from the first Ensemble Metatone rehearsal in April 2013. This early data-collection format simply saved the text representation of SuperCollider OSC message objects, the time received, and the IP address it was received from to a file. A more refined protocol is described in Chapter 8.

formats used by Swift (2012) to record collaborative iPhone performances. Following Swift’s (2012) findings that accelerometer data, which records the angle at which the device is held as well as movements of the device, could provide an important differentiator for iPhone performances, this data was collected in early rehearsals. However, it soon became clear that with iPads, as opposed to iPhones, the performers were more likely to rest the device on the floor, their legs, or a table, than to hold it continually. As a result, the accelerometer data was generally static and so it was not collected in later sessions.

The initial goal of the touch-data logging system was to have a complete record of the performers’ interaction with the touch screens. At first, little
Figure 3.7: Prototype touch visualisation from an early Ensemble Metatone rehearsal in April 2013. Each coloured dot represented the touches of a different iPad player, with the dots fading out over several seconds so that touch gestures become visible.

thought was given to keeping the data in an archival format so early recordings are idiosyncratic, consisting of time in seconds since the recording was started, a text representation of each OSC message captured, and the IP address of the sender device (see Figure 3.6). Logged representations of DMI performances have been used to create animated visual representations that lead to deeper insights into these collaborative interactions (Brown, 2010). Touch-data from Ensemble Metatone’s rehearsals was similarly animated with the Processing programming environment (Reas & Fry, 2006), to create animated visualisations of all performers’ touches. Although the early visualisations, such as the one shown in Figure 3.7, were rudimentary, they were an important aspect of the corpus of rehearsal data discussed in the next section.
3.5 A Research Rehearsal Series

The first series of rehearsals of Ensemble Metatone were treated as research sessions designed to examine the emergence of “expressive interactions” (Hook et al., 2011, p. 1265) in the performers’ use of the new iPad instruments. The goal of these rehearsals was to collaboratively develop a method of improvised musical performance, rather than to follow an ensemble director’s instructions or a musical score. This type of non-hierarchical musical collaboration to develop a musical work has been previously documented by Hayden and Windsor (2007), who noted that “live improvised group decisions” (p. 33) could be used to create musical structures. The rehearsal process was modelled on Cahn’s (2005) Creative Music Making (CMM) process, which is a method for teaching non-idiomatic musical performance using free-improvisation. In a CMM session, performers improvise together freely in small groups following only two rules:

Figure 3.8: The Listening Space studio at the ANU School of Music (as set for study sessions in April 2015). The four corners of the room contain large loudspeakers (hidden) that were used to monitor the sound of each iPad. Recordings were made from a control room through the upper right door.
3. Ensemble Metatone: A Touch-Screen Ensemble

![System diagram for Ensemble Metatone’s iPad setup in early rehearsals and concerts. Each iPad’s headphone sound output was routed to an independent loudspeaker as well as to a multi-track audio recording system. The iPad apps also sent details of touch interactions to a logging application over a WiFi network.](image)

**Figure 3.9:** System diagram for Ensemble Metatone’s iPad setup in early rehearsals and concerts. Each iPad’s headphone sound output was routed to an independent loudspeaker as well as to a multi-track audio recording system. The iPad apps also sent details of touch interactions to a logging application over a WiFi network.

**Rule 1:** Performers may play (or not play) anything they wish on any available instrument of their own choosing — there are no mistakes.

**Rule 2:** Performers should listen as deeply as possible to themselves and to the other performers, but it’s important that it be perfectly clear to all participants — players and listeners alike — that there is no penalty for breaking this rule (Cahn, 2005, p. 35).

There is no time-limit for free improvisations in CMM and, as Cahn writes, the “music comes to an end when all of the players have individually decided to stop playing” (p. 41). This means that even if all but one performer has decided to stop, the performance will continue while at least one is still playing. Such a situation is often controversial within the group but can also lead to a “deeper or an entirely new avenue of musical content” (Cahn, 2005, p. 41) if other performers start playing again.
Rule 2 of CMM emphasises *listening* during performance, but the rehearsal practice of CMM also involves listening to recordings. Cahn has suggested that each rehearsal be recorded and immediately played back to the performers. Even though the performers had only just experienced the improvisation, this playback stage would give them the chance to hear the whole ensemble context without focussing on their individual contribution. A third stage of CMM consists of *questioning* periods, where the session facilitator leads a discussion about the improvisation played by the performers including comparisons to earlier performances. These discussions could directly follow the performance or take place after the playback of the recording.

In rehearsals with Ensemble Metatone, the author acted as a performer and facilitator in the sessions and discussions happened during and after the playback. While listening to each recorded performance, the performers were able to articulate their musical intentions and discuss the structure of the pieces in context in a manner similar to video-cued recall (Costello et al., 2005). After each performance had finished playing, the discussion continued and expanded to cover the whole performance.

The location of the rehearsals had a significant impact on the CMM process. All of Ensemble Metatone’s rehearsals took place in the Listening Space studio at the ANU School of Music. This studio was purpose built for computer music research with surround-sound or ambisonic playback. The room is roughly square, with four large loudspeakers installed behind acoustic panels in each corner as well as a cubic array of eight small loudspeakers. Two other studios are connected to the Listening Space: a small recording booth, and an editing suite with a computer workstation and a stereo monitoring system. Patch panels in each room and a patch bay allow the loudspeakers to be connected to devices in any room. This arrangement meant that the performers’ sound-output could be taken directly from the iPads via the headphone jacks, routed to the editing suite where it was

---

5The Listening Space studio was originally constructed for the Australian Centre for the Arts and Technology where much research into ambisonic performance took place throughout the 1990s (Worrall, 1998).
recorded by the computer, and simultaneously played back through the large monitors (a signal diagram is shown in Figure 3.9). The performers were arranged in the Listening Space such that each player’s sound was dispersed by the loudspeaker directly behind them. In the playback part of the rehearsals, the sounds were dispersed in exactly the same way, so the performers could pick out their individual sounds as well as move around the space to hear the performance from the other players’ perspectives.

As the rehearsals were intended not just to develop music but to be used as research material, all Ensemble Metatone rehearsals were video recorded, often with a camera positioned over the performers to capture some of their touch movements as well as their ensemble communications. In addition to direct line recordings of the iPad headphone outputs, a room microphone captured comments during improvisations and discussions that took place afterwards.

Ensemble Metatone’s rehearsal process started in April 2013 as soon as the first prototype of MetaTravels and the touch-data recording system were functioning. Four CMM rehearsal sessions using MetaTravels, each lasting several hours, were held prior to the first performance in May 2013. Two improvisation paradigms were explored over these sessions: iPads only, and iPads and percussion (shown in Figures 3.10a and 3.10b). In the iPad-only configuration, the performers used the MetaTravels app for the whole session. In the iPad and percussion sessions, each performer was allowed to choose their setup of percussion instruments to use in addition to the MetaTravels app. In these sessions, there was no particular requirement for the performers to use all parts of their setup, but in recordings they could be seen spending time on different instruments, including the app, and even using the app with one hand while playing instruments with the other.

The choice of (non-iPad) instruments was free for each performer; they were limited to a selection that was practical in terms of expressive possibilities but would also fit into the Listening Space studio. In the first iPad and percussion rehearsal, shown in Figure 3.10b, CM chose a vibraphone and various types of mallets, YL chose a set of graduated woodblocks and one octave of crotales, CH chose a setup of ceramic pots and plates, and JG
3.5. A Research Rehearsal Series

(a) Ensemble Metatone in an iPad-only rehearsal (2013-04-27). All performers were using the MetaTravels app.

(b) Ensemble Metatone in an iPad and percussion rehearsal (2013-05-04). While everybody used the same app, the performers were free to choose their own instrumental setup.

Figure 3.10: Two paradigms of improvisation were explored in the rehearsal series: iPads alone (Figure 3.10a), and iPads with percussion setups (Figure 3.10b). A video overview of these sessions is available at: http://dx.doi.org/10.5281/zenodo.51815
### 3. Ensemble Metatone: A Touch-Screen Ensemble

<table>
<thead>
<tr>
<th>Date</th>
<th>Type</th>
<th>Instrumentation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013-04-20</td>
<td>Rehearsal</td>
<td>iPads</td>
<td>First Ensemble Metatone session (<em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-04-21</td>
<td>Rehearsal</td>
<td>iPads</td>
<td>Metatone trio session (<em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-04-27</td>
<td>Rehearsal</td>
<td>iPads</td>
<td>Metatone quartet session (<em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-05-04</td>
<td>Rehearsal</td>
<td>iPads/percussion</td>
<td>Metatone quartet session (<em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-05-14</td>
<td>Performance</td>
<td>iPads/percussion</td>
<td>Metatone quartet perform at Canberra International Music Festival (<em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-07-07</td>
<td>Performance</td>
<td>iPads</td>
<td>Metatone duo perform at Everything/Nothing Projects, (<em>MetaLonsdale</em>)</td>
</tr>
<tr>
<td>2013-08-03</td>
<td>Rehearsal</td>
<td>iPads/percussion</td>
<td>Rehearsal for Metatone Research Concert (<em>MetaLonsdale</em> and <em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-08-03</td>
<td>Performance</td>
<td>iPads/percussion</td>
<td>Metatone Research Concert, (<em>MetaLonsdale</em>), (<em>MetaTravels</em>) - live recording released as <em>Ensemble Metatone</em>.</td>
</tr>
<tr>
<td>2013-10-06</td>
<td>Performance</td>
<td>iPads/percussion</td>
<td>Metatone Trio perform at Electrofringe festival (<em>MetaLonsdale</em> and <em>MetaTravels</em>)</td>
</tr>
<tr>
<td>2013-10-25</td>
<td>Performance</td>
<td>iPads</td>
<td>Metatone perform at Revenant Media event at ANU School of Art Gallery (<em>MetaLonsdale</em>)</td>
</tr>
<tr>
<td>2013-11-21</td>
<td>Performance</td>
<td>iPads/percussion</td>
<td>Trio performance for JG’s graduation recital</td>
</tr>
</tbody>
</table>

Table 3.1: A listing of Ensemble Metatone’s activities in 2013 including studio rehearsals and public performances.
3.5. A Research Rehearsal Series

Figure 3.11: Ensemble Metatone performing with iPads and percussion instruments at the end of the initial research rehearsal process. In this concert, MetaLonsdale was used for an iPad-only performance, and MetaTravels was used for a percussion/iPad improvisation. Video and interaction data recordings from this concert are available at: http://hdl.handle.net/1885/101467

chose a number of drums and cymbals. As the performers needed to have their hands free to pick up sticks as well as perform on the touch-screens, they placed the iPads either on percussion tables (as shown in Figure 3.2), or directly on larger instruments such as the vibraphone. This configuration of instruments evolved over the rehearsal series and initial concerts with Ensemble Metatone. The performers thoughts regarding how the iPads fit in with their percussion setups are discussed in Section 3.6.

Ensemble Metatone’s first performance was conducted in the iPad and percussion configuration; however, the introduction of MetaLonsdale suggested a return to iPad-only performance. As discussed earlier, following an initial duo concert with MetaLonsdale, the app was equipped with a system of network messages between the performers’ iPads to achieve synchronous harmonic motion throughout the performances. In August 2013, the group rehearsed with this system and used it in an iPad-only performance at a
“Research Concert” along with an iPad and percussion improvisation using MetaTravels. This concert, pictured in Figure 3.11, was designated to be the culmination of the research in this chapter and a milestone in Ensemble Metatone’s development. Unlike the previous rehearsals, the Research Concert was performed on stage with audience members present from the experimental arts community in Canberra. The concert was audio recorded, and several angles of video were captured. A recording of the concert was later released (see Appendix B.3.1).

In addition to the performance at the Research Concert, two interviews were conducted. The first, directly after the concert, was an open-ended interview with audience members and musicians present about their perceptions of the performance practice that had been developed. The second interview, held the following day, was an unstructured discussion among the members of Ensemble Metatone concerning the techniques they had developed for performing with the iPads and how they had used them to shape the performances over the whole process of rehearsals and concert. The discussion also covered their ideas for future directions for the group. These interviews, and other discussions recorded during the ensemble’s rehearsals, formed the basis of the qualitative research project detailed in the next section. A full listing of Ensemble Metatone’s activities in 2013 is shown in Table 3.1.

3.6 Analysing the Rehearsal Process

Ensemble Metatone’s research rehearsal process was conducted to develop an artistic practice with the prototype touch-screen DMIs, as well as to build a body of knowledge about how such instruments could be used in ensemble performances. This research was reported in C. Martin et al. (2014a). Data collected from Ensemble Metatone’s rehearsal process up to the Research Concert was subjected to a qualitative analysis to address this question. The data consisted of video and multi-channel audio recordings of each CMM rehearsal session, touch-data collected during the sessions, and audio of the final two interviews. Touch-data from the sessions was
3.6. Analysing the Rehearsal Process

animated (as shown in Figure 3.7) and synchronised with the audio and video recordings so that all angles of video could be viewed simultaneously with the performer’s touch motions.

The rehearsals and interviews formed a corpus of over six hours of video as listed in Table 3.2. The discussions contained in this corpus were transcribed and coded using thematic analysis (Braun & Clarke, 2006) to highlight themes in the rehearsal process. In thematic analysis, codes are associated with sections of the data to represent meaningful features. These codes are then linked together through themes to understand the phenomena under examination. For this study, the codes followed the following three themes:

1. The performers’ articulations of gestures they used on the iPads.
2. The performers’ understanding of the iPads in relation to percussion instruments.
3. How the iPad instruments influenced the structure of the ensemble improvisations.

Understanding gestures was the most important goal so two themes followed the fundamental touch-gestures available in MetaTravels and Meta-Lonsdale: taps, producing single sounds, and swipes producing continuous sounds. Two further themes were used for combination gestures and other gestures that could not be categorised. The other two aspects of the rehearsal process were represented by one theme each.

The interview material associated with the gestural codes revealed that the performers had used many more fine-grained varieties and combinations of these gestures than had been anticipated in the design of the apps. This vocabulary of gestures embraced the affordances of the iPad instrument but also reflected the group’s training and experience with percussion. The data also revealed that the performers refined the relationship between the iPads and percussion instruments, and discovered how these instruments could be used to structure improvisations. In the following sections, findings related to each of these themes will be discussed with reference to the coded data.
3. Ensemble Metatone: A Touch-Screen Ensemble

<table>
<thead>
<tr>
<th>Date</th>
<th>Type</th>
<th>Duration</th>
<th>Video</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013-04-20</td>
<td>iPad only</td>
<td>1H16M</td>
<td><a href="http://hdl.handle.net/1885/101466">http://hdl.handle.net/1885/101466</a></td>
</tr>
<tr>
<td>2013-04-20</td>
<td>iPad only</td>
<td>40M</td>
<td><a href="http://hdl.handle.net/1885/101473">http://hdl.handle.net/1885/101473</a></td>
</tr>
<tr>
<td>2013-04-21</td>
<td>iPad only</td>
<td>46M</td>
<td><a href="http://hdl.handle.net/1885/101472">http://hdl.handle.net/1885/101472</a></td>
</tr>
<tr>
<td>2013-04-27</td>
<td>iPad only</td>
<td>1H</td>
<td><a href="http://hdl.handle.net/1885/101471">http://hdl.handle.net/1885/101471</a></td>
</tr>
<tr>
<td>2013-05-04</td>
<td>iPad/percussion</td>
<td>54M</td>
<td><a href="http://hdl.handle.net/1885/101470">http://hdl.handle.net/1885/101470</a></td>
</tr>
<tr>
<td>2013-05-04</td>
<td>iPad/percussion</td>
<td>1H8M</td>
<td><a href="http://hdl.handle.net/1885/101469">http://hdl.handle.net/1885/101469</a></td>
</tr>
<tr>
<td>2013-08-04</td>
<td>interview</td>
<td>29M</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2: The corpus of rehearsal and interview recordings used in this study. Excerpts of each of rehearsal have been released online (C. Martin, Hopgood, Griffiths, & Lam, 2016).

3.6.1 Taps

By their own admission, the pitched percussion sounds that could be created with taps on the iPad were well understood by the performers. They used the iPads to create interpretations of these sounds throughout the rehearsal process.

*Fast Taps*

The performers used fast tapping and the delay function to create a continuous, pitched sound. Fast Taps were generally not used in precise rhythms but in a percussive single stroke roll where continuous fast attacks created the illusion of a continuous sound (G. Cook, 1997). When this was performed with low pitches, the attacks blended together and created an “organ” or “bass” sound which was used as a continuous low-pitched drone. High-pitched Fast Taps were a very distinctive sound that encouraged imitation among the group while Fast Taps used on field recordings created soft underlying textures. The performers commented on the experience of filling up the delay function’s buffer with fast taps: “setting it going and just watching it for a while” (JG).
3.6. Analysing the Rehearsal Process

**Slow Taps**

Very slow taps were associated with low-pitched sounds from the performers and were used with and without the delay function to create a slow but measured rhythm.

**Random Phrases**

YL demonstrated a “capricious” sound using multiple fingers in quick succession to tap all over the iPad screen. Unlike the fast and slow tap, these random phrases grouped a small number of taps into a short gesture.

3.6.2 Swipes

Swiping, moving a finger across the iPad screen, produced continuous field recording sounds with volume directly controlled by the velocity of the movement. The performers used this control over volume (or dynamics in musical terminology) to express rhythm and to accompany performers using other gestures.

**Short Swipes**

Short swipes had a distinct beginning and end and were usually performed in a straight line across the iPad screens. The end of short swipes was often more striking than the beginning due to the sudden stop in sound when the performer lifted their finger from the screen. Performers emphasised this contrast by accelerating their movement at the end of the gesture producing a loud sound and then silence. This gesture was rhythmically clear; as CH remarked, it was: “more defined, I can make rhythms that way”.

**Swirls**

A continuous swirling sound could be achieved by moving a finger continually back and forth across the iPad screen. While large circular motions could be made slowly or quickly to form soft or loud dynamics, the performers often used small swirls for softer sounds and larger swirls for accents.
3. **Ensemble Metatone: A Touch-Screen Ensemble**

and loud continuous sounds. This gesture mimics the techniques used when playing snare drum with wire brushes where the brushes are swept across a rough drum-skin in patterns of large and small swirls to create rhythms (G. Cook, 1997). A very slow movement across the iPad screen was used to produce a very quiet but continuous sound in a long, focussed gesture.

3.6.3 **Combinations**

Two gestures were discussed that used combinations of taps and swirls. In the first, a small swirl in one hand was used as an underlying sound for phrases of taps in the other. The second combination-gesture used taps with a distinctive pitched sound followed by a short swipe.

3.6.4 **Volume Control**

The iPads’ volume controls were used performatively throughout the improvisations. While the performers would use the hardware volume control to balance their sound with the rest of the group, they also used it to fade continuous sounds in and out, especially those created with the delay function. Here, CH used quick changes to the mute volume setting to produce a sense of space and rhythm in her phrases.

3.6.5 **Relationship Between iPads and Percussion**

Performers commented that the iPad instruments afforded less control over sound than percussion instruments and, in response, they developed gestures that emphasised rhythmic playing and dynamics. The performers did, however, acknowledge features of the iPad instruments that were not available in percussion instruments. The field recordings were “something we can’t replicate” (JG) and the network feature that matched the scale of pitched notes across the ensemble “made everyone sound more cohesive” (CH). When matched with percussion instruments, the iPad became “just another instrument” (CH) in multi-instrument setups. In this situation, CH commented that she “really wanted to play the iPad with sticks,” but also acknowledged that she would sometimes play her regular percussion
instruments with her fingers. In this way, the gestural vocabulary of iPad performance had influenced her approach to the percussion setup.

3.6.6 Structuring an Improvisation

Rather than using the app to create melodies and rhythmic accompaniments, the performers acknowledged that they were, instead, creating “textures” (YL, JG) or layerings of timbrally diverse sounds. While performers frequently imitated each other, responding with similar gestures, they also developed a sense of their own style over the rehearsals; for example, YL used “the ideas I was doing yesterday.” They also had moments of introversion: for instance, “I wasn’t listening to anybody, just having fun” (JG).

The performers were conscious of creating an aesthetically pleasing structure in their performances and were more successful after a number of improvisation sessions together. At the end of the series of rehearsals, YL commented that “we’re really developing a sense of . . . motion. There’s definitely parts of it where . . . we’re definitely now in a new section.” Given the ease of creating continuous sounds with the iPads, the performers appreciated moments of silence or “space” (YL) in between musical ideas. When discussing the goal of playing longer performances, the performers noted they would need to continue developing confidence pursuing particular musical ideas. YL commented that “we’d have to be comfortable sitting on one idea for longer” while JG suggested that the process involved “getting used to . . . the really gradual shift” of musical ideas.

3.7 Conclusions

The main contribution of the rehearsal study was a vocabulary of expressive gestures that emerged from the group to communicate musical ideas and to overcome some of the perceived limitations of the apps. The performers borrowed from their percussion backgrounds to create distinctive gestures using one or both hands: fast, slow and random phrases of taps; short swipes; fast and slow swirls; combinations of swirls and taps; single
handed extension of taps with a swipe; and combinations of these gestures
with the iPad’s volume control. The performers used these gestures to ex-
press rhythm and dynamics and to explore timbres from outside the design
intentions of the apps. While the organ sound of continually delayed fast
taps was not foreseen in the design of the MetaTravels and MetaLonsdale,
this became a key motif in performances. This, and the use of other novel
sounds, demonstrates that the performers were able to uncover hidden af-
fordances of the mobile DMIs.

The group’s understanding of how the iPads fit into their existing per-
cussive artistic practice was also investigated, as well as how they used the
iPads to give structure to an improvisation. While the performers felt the
touch-screen provided less control than percussion instruments, they used
the non-percussive sounds from the apps to augment the traditional percus-
sion in their setups. Even though they were equipped with identical iPad
instruments, the performers developed individual styles over the rehearsal
process. In each improvisation they were conscious of carefully pacing their
performance and using space between phrases and ideas. The group used
ensemble interactions such as imitation as well as introverted explorations
of particular gestures.

This study has demonstrated that expressive gestures can be used to
expand the power of simple musical touch-screen apps and to create comp-
pelling performances. These gestures will be used in later chapters to con-
struct an agent that tracks performers’ gestural progression as a basis for
interacting with ensembles. Following the performers’ experiences and dis-
coversies with MetaTravels and MetaLonsdale, revised app designs will be
introduced that more clearly suggest the percussive gestures. In addition
to performances with Ensemble Metatone, these apps will also be used by
non-expert performers to develop improvised musical works.

3.8 Artistic Outcomes

The rehearsal and performance series described in this chapter successfully
leveraged the percussionists’ expert knowledge to create coherent, impro-
3.8. Artistic Outcomes

vised musical works with the two iPad apps. Ensemble Metatone’s Research Concert in August 2013 was recorded and released as a digital-audio album, *Ensemble Metatone*, which represented the group’s artistic practice up to that time. This artistic outcome is further described in Appendix B.3.1 and the performance and interaction data for the series of rehearsals and concert have also been released⁶ (C. Martin, Hopgood, Griffiths, & Lam, 2016). The same set of works was performed in two subsequent events in October 2013: the Electrofringe festival in Newcastle (Figure 3.12a), and at a media art event, Revenant Media, at the ANU School of Art Gallery (Figure 3.12b). Performing at these two curated events cemented the group’s artistic practice with the two new instruments and also exposed this practice to a significant audience from the artistic communities of Canberra and Newcastle.

It is important to note that the software development effort in creating MetaTravels and MetaLonsdale was matched by artistic effort on the part of Ensemble Metatone in creating new ways to use these instruments in improvisation. In fact, the two activities complemented each other; new directions in improvisation were afforded by newly refined features in the app that were demanded by the performers, and so on. If it has not already become clear, this cycle will become a major theme of how the musical systems in this thesis were developed.

⁶These data are available at: http://dx.doi.org/10.5281/zenodo.51595
3. Ensemble Metatone: A Touch-Screen Ensemble

(a) Ensemble Metatone performing at Electrofringe 2013, at Hunter St TAFE, Newcastle.

(b) Ensemble Metatone performing at Revenant Media, at the ANU School of Art Gallery, Canberra.

Figure 3.12: The works developed in the research rehearsal series went on to be performed at a number of curated artistic events after the August 2013 concert.
In Chapter 2, various models were introduced for designing DMIs that are *ensemble-focussed*. Two of these models, LPN (Local Performance Networks) and EDA (Ensemble Director Agents), were identified as being particularly appropriate for the mobile music instruments under examination in this thesis. While the interconnected network messages of the MetaLonsdale app described in Chapter 3 were an example of an LPN, an EDA required more theoretical development. What should such an agent be listening for, particularly in improvised performances that may not be well characterised by the pitches and rhythms of typical music theory? When and how should the agent respond? In Chapter 3, Ensemble Metatone’s improvised touch-screen performances were characterised in terms of continuous percussive gestures rather than specific notes. In this chapter, a method for automatically classifying performances in terms of these gestures will be developed, as will a method for segmenting such performances by “new idea” moments, when performers spontaneously change their playing as they move through sections in an improvisation. These analyses define a model of machine musicianship that is appropriate for tracking ensemble performances with the touch-screen apps discussed here and elsewhere in this thesis. The analyses can be deployed in a real-time application that returns information to the performers’ DMIs and constitutes an EDA. Such a system will be described in this chapter and a preliminary evaluation of the system will also be per-
4. Tracking Ensemble Touch-Screen Performances

formed. The work presented in this chapter has been previously published in the 2015 International Conference on New Interfaces for Musical Expression under the title “Tracking Ensemble Performance on Touch-Screens with Gesture Classification and Transition Matrices” (C. Martin, Gardner, & Swift, 2015).

Several iPad apps have been designed to interact with an EDA by updating their user interface in response to the agent’s messages. The aim of this app-agent interaction was to present an “interface-free interface” to the performers, where the EDA-determined musical direction is used to adjust pitches, effects, and sonic-material available to them. Three apps that each have different paradigms for interaction with the agent will be introduced in this chapter: Snow Music supports the performers with complementary sounds when they continue certain gestures; Singing Bowls rewards the performers with new pitches and harmonies when they explore different gestures together; and BirdsNest disrupts performers who stay on certain gestures too long with changes in the app’s sound and features.

A preliminary evaluation of the EDA’s touch-gesture classifier has demonstrated a 97% level of accuracy under cross-validation with formally collected training data. Time profiling for a typical performance has shown that the system should scale for use in live concerts with up to 25 performers. In this chapter, experiences with the three apps over several concerts will be reported that demonstrates that the range of iPad apps provides performers with opportunities to develop styles of gestural and musical interaction, both with the agent and each other. This preliminary evaluation will be expanded with a more rigorous user study in Chapter 5.

4.1 Tracking Gestures

A gesture is a motion of the body that contains information. (Kurtenbach & Hulteen, 1990, p. 310)

The concept of gesture is frequently presented in studies of musical performance and is often considered to be central to the New Interfaces for
4.1. Tracking Gestures

Musical Expression (NIME) field. While the above quote from Kurtenbach and Hulteen (1990) is a succinct expression of the concept from an HCI perspective, “musical gestures” can refer to several different concepts, even within the proceedings of the NIME conference (Jensenius, 2014). Cadoz and Wanderley (2000) described two kinds of musical gesture in a survey of the term’s use in HCI and music: Effective gestures are those that are used to control a musical instrument, while ancillary gestures are not involved in creating sound; rather, they are used to communicate to other musicians or simply to emphasise the unfolding music. A more abstract meaning of musical gesture that does not fit into the typical HCI understanding is to refer to “motion-like qualities in the perceived sound,” or even in the musical
instructions of a score (Jensenius, 2014, p. 218).

Ancillary musical gestures can be captured and harnessed as an extra dimension of computer musical control, as demonstrated by Caramiaux et al. (2012) for clarinet, and for percussion with the Radio Drum instrument (Schloss, 1990) or with computer vision methods (Lai, 2009). Effective gestures can also be tracked by sensors attached to instruments, such as bow sensors (Young, 2002), brass valve sensors (L. Jenkins et al., 2013), or the piezoelectric pickups of electronic drums (Tindale, 2007). For the touch-screen mobile devices in the present research, a large amount of data about the performers’ effective gestures could readily be collected from the touch-screens, so extra sensors were not necessary.

Researchers in the NIME field have suggested that it is often easier to collect gestural data than it is to interpret and respond to it musically (P. Cook, 2001). An important recent trend has been to apply powerful Machine Learning (ML) algorithms to such problems so that many dimensions of sensor data may be mapped to much simpler continuous or discrete changes in the synthesis output of a performance interface. For example, Fels and Hinton (1995) used neural networks to map multiple hand sensors to a speech synthesiser. Caramiaux and Tanaka (2013) have provided an overview of machine learning from a DMI designer’s perspective, distinguishing between regression and classification tasks, and reviewing available tool-kits. Fiebrink et al. (2009) used the WEKA machine learning toolkit (Garner, 1995) to create the Wekinator system, designed to allow DMI designers to quickly train ML processes with examples of gestures, map the output to a synthesis environments or other musical software, and evaluate the results on-the-fly through performance. Other tool-kits and libraries have emerged that integrate with computer music environments, such as the SARC EyesWeb Catalogue (Gillian et al., 2011), ml.lib (Bullock & Momeni, 2015), and a library by B. D. Smith and Garnett (2012). Another approach to tracking performances is to extract features from live audio streams, as was done by Hsu (2007) to track improvisations by live instruments.

An important distinction to make is that these examples have typically
4.1. Tracking Gestures

Figure 4.2: An excerpt from Burtner’s (2011) *Syntax of Snow* for solo glockenspiel and bowl of amplified snow. The composer defines a vocabulary of gestures for interacting with the snow with one hand represented by symbols below a regular staff for notes on the glockenspiel. (Score excerpt © M. Burtner 2010, reproduced with permission.)

used ML methods to track the musical gestures of individual performers and to map them to synthesised sonic responses. In the mobile DMIs presented in Chapter 3, the individual sound synthesis responses had already been mapped using existing touch-screen tracking methods provided by the iOS operating system. In this chapter a system will be presented that classifies the gestures of a mobile-music ensemble simultaneously and continuously analyses the whole ensemble’s behaviour. One approach for analysing performer behaviour is to construct transition matrices of changes between a set of musical states that characterise that performance. This approach was first described by Swift et al. (2014) in their analysis of live coding protocols. In the present work, this transition matrix approach will be further developed for real-time gestural analysis of touch-screen ensembles.

4.1.1 Characterising Percussive Gesture

While traditional musical notation specifies sonic outcomes — pitch, articulation and rhythm — it is possible to compose music by specifying gestures used for interacting with instruments. For percussionists, where gestures are transported across a variety of instruments, this has been used to no-
tate music performed on unconventional objects. de Mey’s (1987) *Music de Tables* is written for three percussionists who perform on the surfaces of regular tables; here, de Mey defines a vocabulary of notation for gestures that are used with standard rhythmic notation in the score. Burtner’s (2011) *Syntax of Snow* asks the solo performer to play a glockenspiel with one hand and a bowl of snow with the other. The score sets out a complex scheme of gestures for “playing” the snow, with a pair of symbols (see Figure 4.2) for each gesture, representing the type of gesture as well as hand position in the bowl. Some of the gestures in this score (e.g., “touch with finger”, “swish with palm”, “draw line”) could generalise to other instruments and to touch-screens.

It is notable that many of the gestures indicated in Burtner’s (2011) score could be interpreted as being continuous rather than ceasing after following the instruction. For example, “fingers tapping” should probably be interpreted not as one or two taps but as a continual tapping until the performer reaches the next instruction. In HCI research, gestures on touch-screens are frequently characterised as having a short and finite expression such as the “unistroke” gestures described by Wobbrock et al. (2007). These gestures are usually designed to execute a command in software (e.g., double tap to open a menu) rather than to create an artistic expression. For this reason, characterisations of touch gestures that already exist in the HCI literature are unsuitable for characterising performative touch gestures that mainly consist of continuous interactions.

In Chapter 3, a vocabulary of continuous gestures was identified that was used by expert percussionists on the MetaTravels and MetaLonsdale iPad interfaces. These results have been used to construct an agent that observes performers’ touch-screen interactions in real-time and classifies them as a sequence of gestural states. Free-improvised ensemble musical performances can be considered as sequences of musical sections segmented by moments where the group spontaneously moves to explore a new musical idea (Stenström, 2009, pp. 58–59). The agent estimates the occurrence of new musical ideas across the ensemble by calculating a measure, flux, on the transition matrix of these gesture states. The gestural states and
4.2 System Design

Figure 4.3: The performance architecture of the system of server-based agent and iPad apps. Each iPad connects automatically to the server over WiFi. All touch interactions are logged and classified into gestures by the server which returns individual gesture and ensemble “new-idea” events throughout the performance. Each iPad’s sound is projected from a loudspeaker via the iPad’s headphone jack.

Identifications of “new idea” moments are returned to the DMIs in the ensemble. As the agent responds with general data about the ensemble’s state, it is up to the DMI apps to respond in some way. In Section 4.2, the design of the agent will be described in detail and in Section 4.3 three apps will be described that encode a variety of responses to agent interactions. The agent will be subjected to a preliminary evaluation in Section 4.4.

4.2 System Design

The EDA developed in this research, Metatone Classifier, consists of a Python application that can run on a laptop computer in the performance venue, or on a remote server. The agent is generally operated as a server
4. Tracking Ensemble Touch-Screen Performances

process from the command line, but a simple UI for Apple OS X has been developed that allows the server to be monitored during research performances. The agent interacts with specially designed iPad apps that are used by the ensemble as performance instruments, but has no sound output capabilities itself. During performances, the ensemble’s iPad apps connect to the server over a WiFi network using Bonjour (zero-configuration networking) provided by the `pybonjour` module (see Figure 4.3). Once connected, the iPad apps send logs of each touch event to the agent using the OSC message format\(^1\) (Freed & Schmeder, 2009). Once touch events have been sent to the agent, it begins to analyse the performance and return information to the performers’ iPads at a rate of once per second. The analysis is performed in two stages: first, each performer’s recent touches are classified into a gesture class which is returned to their iPad; second, gesture transitions from the whole ensemble are compiled into a Transition Matrix (TM) which can be analysed to measure the state of the whole ensemble. This information is then sent to every iPad in the performance. In the following sections, the rationale and construction of the agent will be described.

4.2.1 Gesture Classifier

Metatone Classifier classifies gestures by calculating descriptive statistics from each performer’s touch data using a sliding window of five seconds duration. The statistics are shown in Figure 4.1, and include frequency of movement, frequency of touch starts, mean location of touches, standard deviation of touch location, and mean velocity. These statistics are similar to those used in touch-interface performance applications such as the TUIO protocol (Kaltenbrunner et al., 2005). Similar feature vectors were calculated by Swift (2012) in post-hoc analysis of musical smartphone improvisations and were found to distinguish between musicians’ personal styles of performance.

\(^1\)The format of these logs and the OSC messages are discussed in more detail in Chapter 8.
4.2. System Design

<table>
<thead>
<tr>
<th>#</th>
<th>Label</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>centroid_x</td>
<td>Mean X position</td>
</tr>
<tr>
<td>2</td>
<td>centroid_y</td>
<td>Mean Y position</td>
</tr>
<tr>
<td>3</td>
<td>std_x</td>
<td>S.D. of X position</td>
</tr>
<tr>
<td>4</td>
<td>std_y</td>
<td>S.D. of Y position</td>
</tr>
<tr>
<td>5</td>
<td>freq</td>
<td>Total number of touch messages</td>
</tr>
<tr>
<td>6</td>
<td>movement_freq</td>
<td>Total number of moving touch messages (velocity ≠ 0)</td>
</tr>
<tr>
<td>7</td>
<td>touchdown_freq</td>
<td>Total number of touch-down messages (velocity = 0)</td>
</tr>
<tr>
<td>8</td>
<td>velocity</td>
<td>Mean velocity</td>
</tr>
</tbody>
</table>

Table 4.1: The feature vector of descriptive statistics used in Metatone Classifier to classify touch gestures.

Gestures are identified from these features vectors by a Random Forest classifier (Breiman, 2001) from Python’s scikit-learn package (Pedregosa et al., 2011). This particular ML algorithm was selected from the numerous approaches available for classification tasks due to its proven utility in Swift’s (2012) smartphone performance analysis. In that research, Random Forests was shown to outperform alternative algorithms such as Naive Bayes or Support Vector Machines in tasks with similar touch data.

The Random Forest classifier in the present system was used to identify gestures from the vocabulary of nine continuous touch-screen gestures shown in Table 4.2. These gestures were chosen from the vocabulary characterised in Section 3.6 (p. 54). Some of those gestures, such as random phrases (see §3.6.1, p. 56), were deemed too idiosyncratic to include in this classification, but others, such as short swipes were divided into swipes that are fast and regular, and those that specifically accelerate to produce an emphatic rhythm. Three kinds of swirls were included to cover the small and large swirls, as well as the very slow, soft swirls, observed in the original characterisation. Interaction with the iPad volume control was not included as this was not included in the touch-screen data. The nine
4. Tracking Ensemble Touch-Screen Performances

<table>
<thead>
<tr>
<th>#</th>
<th>Code</th>
<th>Description</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>N</td>
<td>Nothing</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>FT</td>
<td>Fast Tapping</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>ST</td>
<td>Slow Tapping</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>FS</td>
<td>Fast Swiping</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>FSA</td>
<td>Accelerating Fast Swiping</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>VSS</td>
<td>Very Slow Swirling</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>BS</td>
<td>Big Swirling</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>SS</td>
<td>Small Swirling</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>C</td>
<td>Combination of Swirls and Taps</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 4.2: The nine touch-screen gestures that Metatone Classifier is trained to identify during performances. These were influenced by the characterisation developed in Section 3.6.

gestures included “nothing”, which, although not strictly a touch gesture, represents the “space” that was discussed as an important component of free-improvisation structure (§3.6.6, p. 59).

The gestures listed in Table 4.2 are continuous gestures, rather than the time-delimited command gestures that are typical of HCI research. So how much time is required to distinguish each of these gestures? Intuitively, if the window is too short, it may be difficult to distinguish multiple classes of gesture, for instance, between swirls and swipes or slow and fast taps. A very long window could contain multiple kinds of gestures, which might be classified too frequently as combinations. As mentioned above, five seconds of touch-data was chosen as the classification window for Metatone Classifier by an early process of trial and error. This choice might also be justified on psycho-acoustic grounds. Five seconds has been considered to be an upper bound for perception of local sound objects in music (Godøy et al., 2010), so it would appear to be a sensible choice for a low-level gesture classification. Another timing parameter is the frequency of sampling touch-data for gestural classifications. Early post-hoc analysis sampled every five seconds; however, trial-and-error experimentation found that sampling at
one second intervals (while retaining a five second window) provided a more responsive system for real-time application and better captured slight local variations in the performance.

As with many ML algorithms, Random Forests requires explicit training with known examples of each class that it is expected to identify. The earliest prototype of Metatone Classifier used data from one of the Ensemble Metatone performances described in Section 3.5 (p. 47). This data was classified by hand every five seconds directly from the performance video and touch-data animation. The time-stamped classifications were then matched with the touch-data recording to produce labelled feature vectors for each class. While this training data suggested that gesture classification was possible, it suffered from inaccuracies due to matching the video with the touch-data and the relative frequency of particular gestures. Since then, two improved sets of training data have been collected in a studio session and using a computer conducted survey. The accuracy of these datasets will be discussed in Section 4.4.1.

Metatone Classifier’s gesture identification system has been used in a post-hoc context to generate gestural scores of performances such as the one illustrated in Figure 4.1 (p. 65). In graphical form, a human viewer can easily break up an ensemble performance into sections and identify ensemble behaviours. In a real-time context the gesture classifications are further analysed by Metatone Classifier to identify the start of new segments in the performance automatically, as will be discussed in the next section. The agent also immediately sends identified gestures back to the performers’ iPad apps, which can potentially respond to sequences of similar gestures with supportive functionality.

4.2.2 Transition Matrices

Metatone Classifier automatically assesses the state of the whole ensemble and responds to some of the structural information that can be intuitively gained from a gestural score. When viewing such a score, it is often not the individual gestural locations that are interesting, but the trajectories
between states over a number of samples, particularly when multiple performers transition from state to state together.

The probabilities of state transitions can be used to characterise and model sequences of discrete states. These models have been used to study melodic sequences in music (Simonton, 1994), and to define Markov processes (Sericola, 2013) for composing melodies algorithmically (Ames, 1989). In the present research, transition probabilities have been used to characterise the performance of a whole ensemble rather than just individual musicians, and to sample similar information as can be seen at a glance in the graphical scores. Before describing how Metatone Classifier uses this information, a transition model for gestural performances will be briefly motivated.

**Markov Chains and Transition Probabilities**

Given a set of gestures $G$, each musician’s activity in a performance consisting of $N$ gesture instances (or states) can be represented as a sequence:

$$X = (X_1, \ldots, X_N) \quad X_i \in G$$  \hspace{1cm} (4.1)

To examine transitions between gestures we can consider the sequence $X$ as a Markov chain, and calculate its stochastic transition matrix (Sericola, 2013, chap. 1). The stochastic transition matrix, $P$, for a Markov process with $m$ states is defined to be an $m \times m$ matrix:

$$p_{ij} = \Pr(X_{t+1} = j | X_t = i) \quad 1 \leq i, j \leq m$$  \hspace{1cm} (4.2)

such that the transition probability from state $i$ to state $j$ is given by the entry in the $i^{th}$ row and $j^{th}$ column.

Given a dataset of gestures, the stochastic transition matrix for a musician’s performance can be defined in terms of known transitions. Let $N_{ij}$ be the number of times that state $i$ was followed by state $j$ in $X$. The maximum likelihood estimator of $P$ is then

$$p_{ij} = \frac{N_{ij}}{\sum_j N_{ij}}$$  \hspace{1cm} (4.3)

The matrix $P$ is a right stochastic matrix, that is, each row sums to 1.
4.2. System Design

Normal Transition Matrices

Stochastic matrices calculated for Markov models are useful for generating new states, as one might do for an algorithmic composition. Metatone Classifier, however, is designed only to analyse an existing sequence of states, rather than generate new ones, so a slightly different formulation is required.

Taking the sequence $X$ from Equation 4.1, let the entries of the *normal* transition matrix $Q$ be defined as follows:

$$q_{ij} = \Pr(X_{t+1} = j \cap X_t = i)$$ (4.4)

Each entry in $Q$ is the probability that any particular transition in the sequence will be from state $i$ to state $j$. This is distinct from the broader probability in Equation 4.2 that given state $i$, state $j$ will follow. Similar to the stochastic matrix, the entries in $Q$ can be calculated by a maximum likelihood estimator:

$$q_{ij} = \frac{N_{ij}}{\sum_{i,j} N_{ij}}$$ (4.5)

This kind of transition matrix is called *normal* as it is normalised with respect to the element-wise 1-norm so that $\|Q\|_1 = 1$ where

$$\|Q\|_1 = \sum_{i,j} |q_{ij}|$$ (4.6)

Finally, the transition matrix that characterises the whole ensemble’s activity is the average of matrices for each performer. This is a concise way of characterising the gestural transition behaviour of the ensemble rather than individual performers. Such ensemble TMs are used in Metatone Classifier as this software is designed to analyse group behaviour.

An important advantage of normal TMs over stochastic TMs is that every element will have a value, even if some states are not found in a particular sequence. In Metatone Classifier, the vocabulary of states, $G$, has been defined; however, it is not certain that a given sequence $X$, and each sub-sequence, will contain every member of $G$. For example, if a sequence does not contain state $i$, $N_{ij} = 0$ for all states $j$, and Equation 4.3 would be undefined for $p_{ij}$. However, as long as a sequence contains at
4. Tracking Ensemble Touch-Screen Performances

Figure 4.4: The heat-map plot of a 15-second transition matrix calculated from a studio improvisation with four performers. This matrix shows movement in between Taps and Swipes gesture groups and steady performance of swirls and combinations, the flux of this transition matrix is 0.25.

least two states and, thus, one transition, Equation 4.5 will be defined for any \( i \) and \( j \), and a normal transition matrix can be calculated, even if the majority of its elements are zero.

It is reasonable to ask whether these gestural sequences are well-characterised by the first-order Markov property, that is, that every state transition depends only on the one previous state. While this model would certainly not account for all structure present in a performance, it is sufficient to represent short-term changes in gestural behaviour over the ensemble. In Metatone Classifier, TMs are only calculated for short 15 second sections of the performance, which would contain very little long-term planning.

An example of a transition matrix derived from a 15 second section of a real performance is shown in Figure 4.4. Even though Metatone Classifier identifies nine distinct gestures, TMs are calculated for a reduced set of five gesture groups. The mapping from gesture classes to groups is shown in the right-most column of Table 4.2 and the five groups are: Nothing, Taps, Swipes, Swirls, and Combinations. This means that each transition
matrix is 5 × 5 rather than 9 × 9. Reducing the number of classes in these matrices has the effect of emphasising more significant gestural transitions in these matrices, for example, those from slow taps to big swirls, and de-emphasising frequent small changes, such as slow taps to fast taps.

4.2.3 The Flux Measure

To compare the ensemble activity between sections of the performance, we derive a high level quantity, called flux, which measures how often the musicians change gesture. The TM can be interpreted as a description of trajectories through the set of gestures, $G$. One style of moving through this space is in a segmented fashion, where a musician will spend long periods performing one gesture, only occasionally changing to another. At the other end of the spectrum is a more frantic approach where a musician jumps frequently between gestures, never dwelling on any particular gesture for too long.

Mathematically, we can discriminate between these and intermediate styles of interaction by calculating the ratio of the sum of off-diagonal elements in the TM with the sum of all elements of the matrix. This measure can be defined for the TM $Q$ as follows:

$$\text{flux}(Q) = \frac{\|Q\|_1 - \|\text{diag}(Q)\|_1}{\|Q\|_1}$$

(4.7)

where $\|Q\|_1$ is the element-wise 1-norm of the matrix $Q$ defined in Equation 4.6 and $\text{diag}(Q)$ is the vector of the main diagonal entries of $Q$.

If, as in Metatone Classifier, transition matrices are normal so that $\|Q\|_1 = 1$, flux can be expressed simply as:

$$\text{flux}(Q) = 1 - \text{tr}(Q)$$

(4.8)

where the trace $\text{tr}(Q)$ is the sum of the diagonal entries of $Q$.

The flux measure returns a value in the range $[0, 1]$. It will return 0 when all non-zero elements of the matrix are on the main diagonal, that is, the performers never change gesture. Flux will return 1 when no performer stays on the same gesture for two states in a row. Flux is small (closer to 0)
when the ensemble rarely changes gesture, and large (closer to 1) when the
performers change gesture frequently and is, therefore, a measure of how
often an ensemble changes from state to state.

4.2.4 Identifying New Ideas

Metatone Classifier is designed to identify moments of peak flux in the
performance that might correspond to performers exploring a new gestural
idea. The agent reports such moments to the performers’ iPads so that
they can update their functionality in response. A plot of flux values for
TMs calculated for a 15-second rolling window is shown against a gesture
score in Figure 4.5. It is clear from this plot that the broad segmentations
in the performance, due to performers moving to different gestures almost
simultaneously, is mirrored by increases in flux over a number of measure-
ments. Metatone Classifier searches for such increases by comparing values
of flux over adjacent 15-second TMs.

In this implementation, each second, Metatone Classifier computes the
ensemble TMs of the two previous 15-second windows of the performance,
and calculates their flux. When the most recent flux measurement exceeds
the immediately previous measurement by a given threshold, then the sys-

tem reports a new-idea message back to the performers’ iPads. An initial
process of trial and error with post-hoc analyses of performances led to a
threshold of 0.15 being chosen as most representative of genuine new-idea
moments. In later performances and experiments, however, threshold val-
ues up to 0.3 were found to be more practical. It is possible that a variable
threshold could be used to identify new-idea segments of different magni-
tude, possibly catering to a wider variety of performance styles; however,
this idea was not explored in the present implementation.

The process of identifying a new-idea is illustrated in Figure 4.6. An
identified new-idea moment has been marked with a vertical red dashed line

2This measurement only considers increases in flux; however, identifying decreasing
patterns is an alternative approach that could be considered in future research.
3From this point forward, new-idea will refer to this circumstance and message trig-
gerated by Metatone Classifier.
Figure 4.5: Flux values for 15-second transition matrices plotted below a gesture-score for an iPad improvisation. Structurally interesting moments in the performance, e.g. where performers spontaneously and simultaneously move to new gestures, seem to match increases in flux. The x-axis shows time in 24-hour format (hh:mm).
Figure 4.6: Metatone Classifier calculates transition matrices over 15-second windows of gesture classifications. Two adjacent windows are compared to track how flux has changed over 30 seconds of the performance.

in the upper part of this figure. The figure also shows the relevant TMs with their 15-second windows marked. The first TM, with a flux value of 0.067, is almost completely static with mainly self-transitions from Taps to Taps. The second TM shows movement from Combinations to Taps, Swirls to Swipes, and Swipes to Swirls and Combinations with a flux value of 0.244. The difference between the second flux and first is 0.177 and thus satisfies the 0.15 threshold for a new-idea message. This identification appears to be visibly justified by the gesture score which shows a marked movement of all three performers to swipes, swirls, and combination gestures around the trigger point.

Because it is possible that a single new-idea would be captured by several sequential measurements, the iPad apps include a rate-limiting function that will ignore messages arriving more frequently than once per minute. Metatone Classifier also continually sends the gesture classifications back
4.3 A Repertoire of Touch-Screen DMIs

Three iPad apps, *BirdsNest*, *Snow Music*, and *Singing Bowls*, were designed to interact with Metatone Classifier. The design of these apps directly followed the prototypes MetaTravels and MetaLonsdale, discussed in Chapter 3, and they also have sound material designed in Pure Data and integrated into the app using *libpd*, with the remaining components designed in Objective-C. Like the prototype apps, BirdsNest, Snow Music, and Singing Bowls featured simple percussion-inspired schemes for mapping touch to sound in free-form touch areas. Tapping the screen produced a short, percussive sound while swiping or swirling produced continuous sounds with a volume proportional to the velocity of the moving touch point. Sound output from the apps was via the iPads’ headphone output which could be dispersed through a mixer and PA system or directly connected to powered speakers.

Even though the performers’ low-level touch interactions with the three

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gesture Classification Window</td>
<td>5s</td>
</tr>
<tr>
<td>Gesture Classification Frequency</td>
<td>1Hz</td>
</tr>
<tr>
<td>Transition Matrix Window</td>
<td>15s</td>
</tr>
<tr>
<td>New-Idea Threshold</td>
<td>0.15–0.3</td>
</tr>
</tbody>
</table>

Table 4.3: These timing and threshold parameters were determined by trial and error for Metatone Classifier. A range of new-idea threshold values may be appropriate for identifying genuine changes in musical direction.
4. Tracking Ensemble Touch-Screen Performances

apps are similar, the apps have been designed to respond to gesture classifications and new-idea messages according to three quite different paradigms: BirdsNest has been designed to be disruptive, Snow Music to be supportive and Singing Bowls to be rewarding. Each paradigm uses the same gesture classifications and new-idea messages to emphasise a different aspect of ensemble interaction among the performers. An important aspect of these agent-app interactions is that they do not reduce the users’ freedom to perform sounds on the screen; rather, they are concerned with operating higher-level functions in the apps, such as activating looping or generative performance features, or selecting new sounds. In MetaTravels and MetaLonsdale, such functions were accessed through GUI buttons and switches, but in the agent-interactive apps, the functions can be automatically adjusted throughout the performance by the EDA. In the following sections, each of the three apps and their agent-interaction paradigm will be described in detail.

4.3.1 BirdsNest

BirdsNest allows musicians to create a sonic journey through a Scandinavian forest using field recordings and percussive samples. Performers who linger too long on a particular gesture will find that the app disrupts them by shuffling their sounds, or changing the features of the app. BirdsNest’s interface is shown in Figure 4.7 and is closely related to MetaLonsdale. In fact, this app was originally adapted from MetaLonsdale for a Sounds of the Treetops performance at the Percussive Arts Society International Convention (PASIC) in November 2013. BirdsNest was later integrated into Ensemble Metatone’s performances using the Metatone Classifier EDA.

The sonic material in BirdsNest is composed to allow each performer to create a journey through fields, a forest, up into the trees, and finally to a bird’s-eye vantage point of the whole landscape below. Visually this journey is communicated through a series of background images and collections.

---

4This performance was by Ensemble Evolution, a different group from Ensemble Metatone discussed in Chapter 3.
4.3. A Repertoire of Touch-Screen DMIs

Figure 4.7: Screenshot of the BirdsNest app that features field-recordings and images from a forest in Northern Sweden. Performers create a sonic journey through this forest by exploring the available sound material from the forest floor to a bird’s-eye vantage point. A video demo of BirdsNest is available at: http://dx.doi.org/10.5281/zenodo.51819

of bird sounds and field recordings from the location as well as sampled percussion instruments, such as wood block and xylophone, which complement this musical idea. Unlike MetaLonsdale, which had a focus on a sequence of scales, BirdsNest is based around this evocative series of sound colours. Each scene has a palette of sound material of which only a few sounds are available to each player. The sounds are triggered by tapping and swirling on the backdrop image. The interface has a GUI button that performers can use to shuffle sounds available to them from the available palette. It also has a looping function controlled by a switch where tapped notes are repeated approximately every five seconds for a limited number of times with increasingly randomised pitch and rhythm. Another switch controls an autoplay function where field-recordings from the sound palette are generatively triggered as a backing soundscape.

As mentioned, BirdsNest is designed to disrupt the musicians’ perfor-
4. Tracking Ensemble Touch-Screen Performances

mance to discourage performers from staying on any one gesture for too long. Each iPad keeps track of previous gestures used by the performer, and when a performer stays too long on a particular gesture, the app switches one of the features on or off, or it spontaneously changes the palette of sounds. Although the agent interactions may disrupt the performer, it is possible to reset these features or continue to explore the palette of sounds using the user interface elements. Users and the agent can both affect features inside each forest scene; however, moving between each forest scene relies on new-idea messages from the agent, and these changes occur simultaneously across the ensemble.

The agent interactions in BirdsNest are designed to encourage performers to continuously explore new gestural material and interact with one another throughout the performance. While it does not prevent performers from staying on the same gesture continuously, it actively challenges them by changing their instrument, confronting them with new sonic interactions.

4.3.2 Snow Music

Snow Music aims to emulate a bowl of snow, allowing performers to manipulate recordings of snow being squished, smeared, stomped and smashed. The app supports the performers by generating complementary sounds along with their snow-improvisations. The vocabulary of gestures is linked to a library of these sounds which evolve in response to new-idea messages from Metatone Classifier.

Snow Music is actually a revised version of a prototype app developed for the author’s previous research project that took place in Piteå, Sweden (C. Martin, 2012a, 2012b). The original app was used in conjunction with percussion instruments and was only intended to be used as a small part of a larger setup. The extensive revisions were focussed on refining the interface so that the app could be used by itself in improvisations, and on integrating the supportive interaction with the Metatone Classifier agent.

As with previous apps, the performers are free to improvise in any way they wish with the snow samples that are available in Snow Music, and
the app includes a palette of snow sounds and generative accompaniment sounds. However, the app is designed so that performers can only experience these new sounds or textures by interacting with the snow sounds, not by activating UI elements. The app watches for runs of similar gestures and activates extra sounds that support the player’s intent. For instance, a run of tapping gestures causes the app to layer the snow sounds with a glockenspiel sound, while continuous swirling activates a generative backdrop of melodic bell sounds. Three switches in the corner of the app (see Figure 4.8) inform the player of the presence of each of the Bell, Snow, and Cymbal, backdrop sounds, and when generative sounds are triggered, the notes are animated with coloured circles in the playing area. These supportive sounds are switched off when the performer explores other gestures.

In the case of Snow Music, new-idea messages shuffle the snow samples available to the player and change the pitches used in the supportive sounds.
4. Tracking Ensemble Touch-Screen Performances

Figure 4.9: The minimalist user interface of the Singing Bowls app. The available notes are divided by circles. When the performer activates a note by tapping or swirling, it pulses with a colour given by the pitch. A video demo of Singing Bowls is available at: http://dx.doi.org/10.5281/zenodo.51821

While the presence of the supportive sounds are shown on the screen with UI switches and animations, the performers are not able to control them directly with UI elements. Although this app appears to have a limited selection of sounds available to the player, the interaction with the agent challenges the individual and the whole ensemble to fully explore a range of touch-gestures together. The aim is to support mindful exploration with a range of complementary musical elements.

4.3.3 Singing Bowls

Singing Bowls, shown in Figure 4.9, is a ring-based interface for interacting with bell samples. Unlike the range of sounds available in the other Metatone apps, Singing Bowls allows the performers to interact with sounds generated from one bell sample. The app does, however, allow more expressive interactions with this sound where different kinds of touch gestures modu-
late the sound in a variety of ways. While the app allows the users to access a limited number of notes, it *rewards* adventurous ensemble improvisation with new selections of pitches generated from a harmonic sequence.

Similarly to MetaLonsdale and BirdsNest, Singing Bowls is based around a series of harmonically related musical scales. At any one time, a selection of three to ten pitches is available on the screen with each pitch represented by a ring and a text label. This pitch “setup”, or collection of pitches available on screen, is generated from the currently active scale. Although the performers have the same active scale, and thus the same harmonic location, the pitch setups are generated independently by each iPad so that they each have a unique melodic space to explore.

Tapping on a pitch ring triggers a short note at that pitch. A swipe generates a continuous sound at the pitch where the swipe began. The continuous sounds are generated using an overlapping sample looper as detailed by Puckette (2007, §2.6) that loops grains of up to 100ms duration from the bell sample. The looping parameters (apart from pitch) are randomised after each note to create subtly different sound colours. The app can play one continuous note and up to four tapped notes simultaneously. To emphasise the different varieties of swirl and swipe gestures that the performers in Ensemble Metatone use, the app continuously calculates the velocity, direction, and position of moving touches which affect two synthesis parameters. The angle of the velocity vector of a moving touch point is used to modulate the pitch allowing different kinds of vibrato to be created with different movement patterns. The distance of the moving touch from the original touch point controls the degree to which a “crystal reverb” effect is applied to the continuous note, giving the performers a way to control timbre in real time.

The performers are given visual feedback when playing Singing Bowls. When tapped or swirled, each pitch pulses a particular colour following a system inspired by de Maistre’s (1917–1919) artworks that relate music and colour. When swirling, the intensity of the colour and speed of pulsing is connected to the performer’s touch velocity and position.

Unlike BirdsNest, the Singing Bowls app has no GUI elements to change
sound or scale. Instead, the app waits for new-idea messages from the Meta-
tone Classifier agent, and then *rewards* the performers with a new setup.
This reward will give each performer a new number and selection of pitches
that may be drawn from the next scale in the harmonic progression. Intu-
itively, performers who have moved to a new musical section may appreciate
the new melodic possibilities of a different set of pitches or scale. This in-
teraction is also designed to encourage ensembles to perform adventurously,
as a group that stays within a limited gestural space will be stuck with the
same notes throughout the whole performance.

Singing Bowls is hard-coded to choose two setups each from a sequence
of three scales, making six setups available before returning to the beginning
of the cycle. Since the iPads all receive new-idea messages together, they are
always in the same place in the sequence and the performers have a sense of
cohesive harmonic progression. Because each setup is created generatively,
the performers continue to see new notes whenever a new-idea message is
triggered, even if the ensemble repeatedly advances through the whole cycle
of scales.

### 4.4 Preliminary Evaluation

This section describes a preliminary evaluation of the Metatone Classifier
agent focussed on its practicality for use in musical performances. Three
aspects of the system are examined:

1. The gesture classification system is subjected to cross-validation typ-
ical of machine learning evaluations.

2. The computational complexity of the agent is examined to show that
it is sufficiently fast to use in real performances.

3. Performance experiences are referenced to give some context to how
the apps and agents have been used in practice.

Much more thorough and rigorous HCI investigations of this system will be
conducted in chapters 5 and 7.
4.4. Preliminary Evaluation

<table>
<thead>
<tr>
<th>#</th>
<th>Description</th>
<th>N</th>
<th>Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2013 gestures 5s window</td>
<td>98</td>
<td>0.915</td>
<td>0.08</td>
</tr>
<tr>
<td>2</td>
<td>2013 gestures 1s window</td>
<td>486</td>
<td>0.942</td>
<td>0.032</td>
</tr>
<tr>
<td>3</td>
<td>2014 gestures 1s window</td>
<td>532</td>
<td>0.973</td>
<td>0.022</td>
</tr>
</tbody>
</table>

Table 4.4: Three sets of training data have been used with Metatone Classifier and were compared using cross-validation. This table shows the number of feature vectors in each data set, mean accuracy and standard deviation.

Figure 4.10: Screen capture from the proof-of-concept gesture collection performance. The video was used to match up feature vectors calculated from touch-data with the correct gestures.

4.4.1 Gesture Classifier Accuracy

The training data used for the gesture classifier in Metatone Classifier was evaluated using standard cross-validation methods from machine learning. Three sets of training data were available for a comparative evaluation: a proof-of-concept set of example gestures with feature-vectors calculated on 5 second windows, the same set using a rolling 5-second window at one-second intervals, and a production set captured following a formal procedure. The proof-of-concept gesture data was collected by matching video analysis of a performance by the author (shown in Figure 4.10) of examples of each gesture on one of the iPad apps together with the logged touch data.
4. Tracking Ensemble Touch-Screen Performances

Figure 4.11: The test harness used to collect training data for Metatone Classifier. This software asks a user to input each gesture for one-minute in a randomised order with timed breaks in between and adds the target gesture to the touch-data log so that the training set can be produced automatically.

The production set was collected using a survey application, written in the Processing environment (Reas & Fry, 2006), that is shown in Figure 4.11. This application instructed the performer to play each gesture on an iPad app in randomised order for one minute with a 20 second break in between each gesture. The nothing gesture was performed by not touching the iPad screen. As the application annotated the touch-data with the currently displayed gesture, the data could be processed automatically. Waiting periods in between gestures were removed as were ambiguous feature vectors at the beginning and end of each example gesture to retain highest quality training data. As with the proof-of-concept data, the production set of training data was performed by the author in one session using the survey application.

The three classifiers were evaluated using stratified 10-fold cross validation which was performed 10 times on each training set, producing 100 estimates of accuracy for each classifier. The results of these validation tests are shown in Figure 4.12. A one-way ANOVA procedure revealed a signif-
4.4. Preliminary Evaluation

![Box-plots of the cross validation results using the three sets of training data. The most recent training data was most successful.](image)

Figure 4.12: Box-plots of the cross validation results using the three sets of training data. The most recent training data was most successful.

Significant effect of training set on accuracy with $F(2, 297) = 31.7, p < 0.001$. Paired Bonferroni-corrected t-tests confirmed significant ($p < 0.05$) differences between the three sets of training data with the newest set producing a mean accuracy of 0.973 with standard deviation of 0.022. This level of accuracy is consistent with that reported in other systems that recognise touch command gestures (Wobbrock et al., 2007). It is notable that the production training set produced a significantly more accurate classifier even though the number of example gestures was only 9.5% higher. The improvement was more likely due to the quality of data collected using the survey application.

It should be noted that this evaluation has only examined the consistency and quality of the training data, and not the performance of the whole Metatone Classifier system under real-world circumstances. Fiebrink et al. (2011) recommend that machine learning models used in the Wekinator system should be subjected to “direct evaluation” by users who observe the model’s behaviour in response to real-time input. Direct evaluation of Metatone Classifier is considered in the investigations in chapters 5 and 7. It would also be possible in future revisions of the system to use the formal procedure to capture and compare training data from other performers who may use a broader range of gestural variations.
4. Tracking Ensemble Touch-Screen Performances

![Graph showing the relationship between number of performers and mean time for one analysis.](image)

**Figure 4.13:** Measurements of computational complexity suggests that the time to complete one classification step increases linearly with the number of iPads.

### 4.4.2 Computational Cost

The computational cost of the agent was profiled using the [line profiler](http://github.com/rkern/line_profiler) Python module during performances with zero to four iPad performers. The test system ran Apple OS X on an Intel Core i7-2720QM 2.2GHz processor. With four iPads, the most common configuration used in this thesis, the classification and analysis function that is triggered once per second took a mean time of 0.158s to complete. The major components of this function were the calculation of feature vectors for the performers (0.06s), the Random Forest classifications (0.032s), and the calculation of transition matrices (0.049s).

The mean time for the classification and analysis function to complete had a significant ($p < 0.001$) linear relationship with the number of iPads performing, as illustrated in Figure 4.13. We can estimate from the linear model that on the test system this function could take 0.038s per iPad plus 0.0085s overhead. With this estimation, an ensemble of around 25 iPads could be an upper-bound for analysis in the desired one-second timeframe (with similar hardware). Although this would be sufficient for the membership of most institutional computer music ensembles, the ubiquity of mobile devices such as the iPad suggests that performances of much
larger ensembles could be explored in future.

This computational profiling suggests that extensions to the analysis performed in Metatone Classifier might be possible without impacting the potential for larger ensembles. While the computational cost of performing gesture classification and generating transition matrices would increase with the size of an ensemble, the cost of measures on the transition matrices will not, since these have a fixed size. Other matrix measures may reveal different aspects of the ensemble’s musical behaviour and could probably be incorporated without a significant cost in computation.

4.4.3 Concert Performance

Metatone Classifier was premiered in concert performance with the three apps described in this chapter in March 2014. In between March and August 2014, the system was used in several live performances by Ensemble Metatone and other ensembles as well as in an installation context with small groups of visitors interacting with the apps and agent. While the most common performance configuration included four iPad performers, some concerts have featured two to seven players. Concerts with this system have generally consisted of about 45 minutes of music with free-improvised as well as composed works. Performers generally all play on the same app for any particular piece and the agent was used for every piece in the concert.

While audiences have responded with great interest to the wide variety of sounds produced by the different apps, performers have reported being more engaged by the different modes of interaction available to them, particularly when performing free-improvisations. As discussed in the previous section, while the fundamental interface for sound-creation is similar on the three apps, they each have different features and different modes of interaction with the agent. Performers have reported that they responded to the different apps by developing different strategies for drawing out their favourite sounds through gestural interaction with the agent and the other

\footnote{The author participated in these performances.}
4. Tracking Ensemble Touch-Screen Performances

<table>
<thead>
<tr>
<th>Date</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>14-03-17</td>
<td>Performance</td>
<td>You Are Here festival concert.</td>
</tr>
<tr>
<td>14-05-05</td>
<td>Performance</td>
<td>Touch &amp; Tone Concert, Boston.</td>
</tr>
<tr>
<td>14-08-14</td>
<td>Performance</td>
<td>Colour Music Exhibition Opening.</td>
</tr>
<tr>
<td>14-08-24</td>
<td>Performance</td>
<td>Colour Music Performance.</td>
</tr>
</tbody>
</table>

Table 4.5: A listing of early performances with Metatone Classifier, and the apps described in this chapter (not including rehearsals or lab study sessions).

performers. In doing so, they developed their own personal styles and attempted to replicate their favourite moments from rehearsals (where certain circumstances may have triggered a new-idea message or a change of sound palette). These reported experiences confirm that the system of apps and agent is practical for real-world performances.

4.5 Chapter Summary

In this chapter, a novel EDA and a repertoire of three contrasting apps have been presented. The Metatone Classifier agent tracks a whole ensemble’s touch-screen musical performance by classifying gestures, calculating transition matrices, and identifying new ideas. The three iPad apps use messages from this agent to support, disrupt, and reward gestural explorations in collaborative improvised performances. The agent uses a novel flux measure to determine change points in the group’s activity and to make calculated, real-time interventions to the iPad interfaces. Results have been presented of an evaluation of this agent’s gesture classifier where three training sets of data were compared using cross-validation. Computational profiling has been performed on the gesture classification and ensemble tracking algorithms in this system to estimate an upper bound for ensemble size given the desired timeframe of one analysis per second. The use of this system
in a series of concerts confirms that it is practical for use in exploratory mobile-music performances.

While performances with this system have typically included four iPad players, the profiling results suggest that the current agent could scale up to around 25 players before classifications would be delayed. However, since the agent’s interactions are based on assumptions of 5-second windows for touch gestures, and 15-second windows for transition matrices of ensemble interactions, it is possible that timely response from the agent is not critical and that a longer analysis cycle would be sufficient. This compromise may be required to cater for very large iPad ensembles.

The evaluation of the gestural classifier revealed that training data collected under controlled conditions had produced a significantly more accurate Random Forest classifier even though the size of the training set was similar to that previously collected. This result justifies the extra effort required to design a system for automatically and accurately capturing training data rather than the previous manual analysis of video-recorded gestures. This system could be used in future to collect training data from other performers or to expand Metatone Classifier to support other vocabularies of gestures. Transition matrix measures other than flux could also be added to track different aspects of the ensemble performance. One possibility would be to apply flux to transitions that are weighted by their independence. For example, taps to swirls could count as a more significant change than taps to combinations.

The initial reception to the repertoire of Metatone apps and the Metatone Classifier agent suggest that an EDA can have a positive impact on performances, and particularly on free improvisation. Live performances can be a stressful and complicated environment to study directly. In particular, it is difficult to collect quantitative data from musicians who are preoccupied with the practical challenges of performing. Chapter 5 describes a lab-based methodology modelled on rehearsal processes that was used with experienced iPad performers to measure the effects of this system of apps and agent. In Chapter 7, the methodology is extended to a study of beginner iPad performers.
Evaluating Apps and Agents

This chapter describes a formal evaluation of the system of touch-screen musical instrument apps and server-based computational agent described in Chapter 4. As described in that chapter, Metatone Classifier, an Ensemble Director Agent (EDA), was developed for use in group free-improvisations. This EDA tracks ensembles by identifying the individual touch-gestures of each performer and uses them to estimate the occurrence of new-idea moments across the whole ensemble. These new-idea moments appeared to correspond to the emergent changes in musical direction that are often described as giving free-improvisations a segmented structure (Stenström, 2009, pp. 58–59). Three musical apps, Singing Bowls (SB), BirdsNest (BN) and Snow Music (SM), were developed that react to the agent by updating their interfaces in real-time. Each of the three apps encoded a different behavioural model of interaction in ensemble improvisation: A reward model (SB) gave performers access to new notes at each new section of the performance, a disruption model (BN) interrupted performers who stayed on one gesture for too long, and a support model (SM) played complementary sounds when performers focussed on individual gestures.

As described in Section 4.4.3 (p. 93), this system of apps and EDA had been used in prototype form in a number of rehearsals, performances, and even an installation setting. These experiences provided anecdotal evidence that the agent interventions were accurate, occurred at appropriate times, and could enhance improvisations. It also appeared that the different app
5. Evaluating Apps and Agents

designs, and their responses to agent interactions in particular, influenced the performances in different ways. The formal experiment reported in this chapter was designed to make some of these anecdotal conclusions precise by comparing performances under different app and agent conditions in a controlled environment.

To expose the effect of Metatone Classifier’s ensemble tracking functionality, a second EDA was developed to direct the ensemble with the same set of signals, but with these signals generated randomly from a statistical model. This generative EDA was posed as a control condition in the experiment. The three iPad apps constituted a main effect for app interface and were able to respond to both the ensemble-tracking agent and the generative agent. The participants in this study were three of the members of Ensemble Metatone. By the time of the experiment, these touch-screen musicians had more than a year of experience improvising on the iPad apps presented in this thesis in addition to their professional music training.

In a methodology that combined a balanced experimental design with rehearsal processes, this group performed a series of 18 improvisations on all combinations of the three iPad interfaces and the two agents. Quantitative analyses were performed on the survey ratings from the musicians, and on the number of agent messages sent during performances. A qualitative analysis compiled from interviews supplemented the findings.

The results of this study support the effectiveness of the ensemble-tracking EDA, although the source of agent interventions was seen as less important than how the apps responded. The app condition was found to have a significant main effect on the performer’s responses to several questions, including the quality and level of creativity in performances. Singing Bowls and its reward interaction model showed the most positive response with the performers actively seeking out interaction with the agent when using this app. The performers articulated problems with the other two apps while still finding ways to use them in interesting improvisations and their responses were used to redesign the apps for later performances. Following a review of methodologies for evaluating similar DMI systems in Section 5.1,

1The author was omitted from this testing group.
changes made to the system of apps and agents for the evaluation will be described in Section 5.2. Section 5.3 will describe the experimental design, and the results will be analysed and discussed in Section 5.4. The work in this chapter has previously been published as “Music for 18 Performances: Evaluating Apps and Agents with Free Improvisation” in the Australasian Computer Music Conference (C. Martin, Gardner, Swift, & Martin, 2015).

5.1 A Rehearsal-As-Research Methodology

As discussed in Chapter 2, many DMI designers have investigated ways to use new computer music instruments in ensemble performance. A pertinent question for the present chapter is how such ensemble-DMIs can be evaluated. Ensemble Metatone was formed in the spirit of a Laptop Orchestra, where multiple performers use identical hardware and software in musical performance. The use of the EDA, Metatone Classifier, and the wish to engage in a formal process of evaluation sets this group apart from the other LOrks described in Section 2.2. In the broader field of computer music, a range of methodologies have been explored for evaluating DMIs in the lab and on stage. In this section, some of these methodologies will be reviewed, to motivate a rehearsal-as-research study for evaluating the present system of apps and agents.

O’Modhrain (2011) argued that there are multiple stakeholder perspectives that could be considered in evaluating a DMI, including audiences, performers, designers, and manufacturers. The most important of these stakeholders, however, are the performers as they are “the only people who can provide feedback on an instrument’s functioning in the context for which it was ultimately intended” (p. 34). For improvised music, like that performed by Ensemble Metatone, this is particularly important. Improvised performers are responsible not only for translating musical intentions into sound with the DMI, but for creating these intentions as well.

Many researchers have relied on performers to evaluate their own interactions with a DMI. While working with end-users of an interactive machine learning system for performing music, Fiebrink et al. (2011) relied on the
performers’ evaluations to improve the ML model. The users evaluated these systems repeatedly across a number of novel criteria to iteratively improve the DMI, and this “direct evaluation” was found to have more utility than a typical cross-validation approach for ML systems. Gurevich et al. (2012) examined the performer’s perspective to identify styles and skills that emerge when multiple participants engage with very simple electronic instruments. Their study took a qualitative approach supported by grounded theory (Corbin & Strauss, 2008). The participants were asked to practice each day for one week with the instrument, and then to perform a two-minute solo improvisation which was followed by an interview. Stowell et al. (2009) conducted similar interviews with individual performers directly after a solo performance, but also interviewed the participants as a group.

The processes that lead to successful musical performances are often long-term and can consist of multiple cycles of rehearsals, jam sessions and performances. Over these longer-term processes, different types of collaborations can develop between performers (Hayden & Windsor, 2007). In DMI research, the performer’s perspective on the DMI may also develop over rehearsal and performance processes. The DMIs themselves may also be changing due to design refinements occurring in parallel with performances. Gelineck and Serafin (2012) argued for qualitative longitudinal evaluations that go beyond “first impressions” to capture these evolving perspectives by interviewing performers after several weeks of experience.

Longitudinal DMI evaluations often use a variety of data sources such as video of performances, data protocols from interaction sessions, as well as interviews. These studies often use ethnography (Krüger, 2008) to address and draw conclusions from these data. A long-term ethnographic study of the Reactable table-top surface observed collaborative and constructive processes in video footage of improvised performances (Xambó et al., 2013). Ethnographic techniques have also been used to study natural rehearsal and development processes such as Unander-Scharin et al.’s (2014) Vocal Chorder project, where an autobiographical design process transitioned into an interface developed for other performers. Indeed, the research presented
5.1. A Rehearsal-As-Research Methodology

in Section 3.6 used such a methodology to characterise Ensemble Metatone’s performance practice.

An important advantage of longitudinal studies for ensembles is the ability to observe and account for changes in the collaboration over time. Cahn (2005, pp. 37–38) has observed the strong learning effect present in new improvisation ensembles where members overcome initial inhibitions to test the limits of new-found musical freedom with “severe departures from normal music making.” This phase is followed by a plateau of thoughtful free-improvisation where “listening and playing come into more of a balance.” Many DMI studies, including work in this thesis, examine performers’ early interactions with a DMI. In contrast, the performers in Ensemble Metatone had already developed an active artistic practice and had experience with the DMI, so there would not be dramatic changes in between performances due to this learning effect.

Given the experience and training of the participants in the present study, the practice phase used by Gurevich et al. (2012) and Gelineck and Serafin (2012) was deemed to be unnecessary. A more appropriate model is direct evaluation (Fiebrink et al., 2011) where the performers evaluate their experience after each improvisation. Eisenberg and Thompson (2003) have shown that observers can assess improvisations via a Likert-style rating scale; however, the present study focussed on the performers’ own evaluations using a similar rating scale survey. The structure of the study was modelled after a typical rehearsal where repeated performances under differing musical conditions occur in a single session. In a research setting, this process can be used to assess several replicates of a number of experimental conditions. The methodology used in this study where performers directly assess multiple performances in one session could be termed rehearsal-as-research.

As Ensemble Metatone had already established a performance practice and were experienced with regard to the demands of a rehearsal environment, they were able to test the six experimental conditions with 18 improvisations in one session. Although this number of performances would not be unexpected in a professional rehearsal, it is an unusual number of tests
5. Evaluating Apps and Agents

Figure 5.1: A system diagram of the agent software interacting with an ensemble of iPad instruments. In the test condition, touch messages are classified as gestures by a Random Forest classifier, while in the control, gestures are generated from a statistical model disconnected from the performers’ current actions.

for a single DMI evaluation session.

5.2 Experimental System

The designs of the Metatone Classifier agent system and the iPad apps used in this study were documented at length in Chapter 4. A formal evaluation, however, demanded some additions to this system in the interest of exposing the effects of the agent and app design on the performers and their improvisations. In particular, a *generative* version of Metatone Classifier was developed that directed the ensemble using randomly generated signals, rather than by tracking the ensemble gestures. The architecture of the system of agents and apps used in this formal study is illustrated in 5.1.
5.2. Experimental System
and will be described in the following sections.

5.2.1 Gesture Classifier
As reported in Section 4.2 (p. 69), the Metatone Classifier agent receives touch-data from the performers’ iPads and classifies this data at a rate of once per second using a Random Forest classifier. The gesture classes are taken from a vocabulary of continuous touch-gestures, described in Section 3.6 (p. 54), which characterises percussive improvisation on touch-screens. The identified gestures are sent back to the performers’ iPads, and also stored by the agent and continuously analysed for new-idea moments in the ensemble improvisation.

The gesture classifier can be considered to be the sensing part of the Metatone Classifier agent, as all decisions regarding signals to be sent back to the performer iPads are based on the classified gesture data. In this experiment, the Metatone Classifier agent with an unchanged gesture classifier is designated as the classifying agent (CLA). Another agent, described below, replaces the classification step and effectively decouples the agent’s behaviour from the performers’ actions.

5.2.2 Generating Fake Gestures
In order to evaluate the effect of the gesture classifying agent (CLA) on performances in this experiment, a contrasting generative agent (GEN) was developed to produce fake agent responses. The GEN agent was designed to be used as a control in the study and replaces the gesture classifier with a random process for producing a sequence of gesture classes. Figure 5.1 shows that subsequent parts of the agent software that analyse the ongoing performance and return signals to the iPads remain the same between both agents. This means that the GEN agent records and reports fake gestures and fake new-idea messages to the iPads in the same way as with the CLA agent.

To build this control agent, a live touch-screen performance by Ensemble Metatone was analysed with the classification system and the resulting
sequence of states was used to construct a first-order Markov model. The concept of using a Markov model to generate data is a common design pattern in computer music and Ames (1989) has described how it can be used to algorithmically compose melodies or vary other musical parameters. In the present research, the model was used to generate fake gesture classifications similar to the gestural output of Ensemble Metatone. A plot of the transition-matrix for this Markov model is shown in Figure 5.2. As this model was used to generate new states, not to analyse an existing sequence, it was calculated as a right-stochastic matrix according to Equation 4.3.

The gesture states returned by this model are statistically similar to the states identified by the classifying agent, but decoupled from the performers’ actual gestures. This meant that the generative agent could be used as a control in the experiment to expose the effect of an intelligent mediation of live performance. Although the generative agent was used as a control in this way, such a design could have genuine artistic merit and be used in performance as a generative EDA.
5.2. Experimental System

Figure 5.3: The three apps used in this study, from left to right: BirdsNest (BN), Singing Bowls (SB), and Snow Music (SM).

5.2.3 iPad Apps

Three different iPad apps were chosen from the repertoire previously introduced in Chapters 3 and 4 which had been routinely used by Ensemble Metatone. The three apps, BirdsNest (BN), Singing Bowls (SB), and Snow Music (SM), are shown in Figure 5.3. While these apps shared the ability to create sound with free-form touch improvisation, their different sound material and contrasting designs for interaction with the agents made them three distinct instruments. The different responses to agent interactions were of particular interest in the present study as previous performance experiences had suggested that the musicians were developing different strategies for performing with the three apps.

As described in Section 4.3, BirdsNest was designed to disrupt the musicians’ performance. Based on gesture feedback from the agent, the app would watch for runs of identical gestures and then switch on looping and autoplay features in the user interface in order to prompt new actions by the performers. New-idea messages were used to randomise the sounds available to the user from a palette of sample and pitched material.

Snow Music used a supportive paradigm. The app would watch for sequences of similar gestures and activate extra layers of complementary sounds. For instance, the app would support a run of tapped snow sounds by layering the taps with glockenspiel notes while a backdrop of generative
5. Evaluating Apps and Agents

Table 5.1: The experiment schedule showing the balanced ordering of apps and agents. The experiment was performed in one session divided by breaks into six groups of three five minute performances.

<table>
<thead>
<tr>
<th>Set</th>
<th>Performance 1</th>
<th>Performance 2</th>
<th>Performance 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>orientation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>SM, CLA</td>
<td>BN, GEN</td>
<td>SB, CLA</td>
</tr>
<tr>
<td>2</td>
<td>BN, CLA</td>
<td>SB, GEN</td>
<td>SM, GEN</td>
</tr>
<tr>
<td>3</td>
<td>SB, CLA</td>
<td>SM, CLA</td>
<td>BN, GEN</td>
</tr>
<tr>
<td>4</td>
<td>SB, GEN</td>
<td>BN, CLA</td>
<td>SM, GEN</td>
</tr>
<tr>
<td>5</td>
<td>BN, GEN</td>
<td>SM, CLA</td>
<td>SB, CLA</td>
</tr>
<tr>
<td>6</td>
<td>SM, GEN</td>
<td>SB, GEN</td>
<td>BN, CLA</td>
</tr>
<tr>
<td>7</td>
<td>interview</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

bell melodies would be layered on top of the sound of continuous swirling gestures. When the performer moves on to other gestures, the supportive sounds were switched off. New-idea messages in Snow Music changed the pitches of the supportive sounds and the snow samples available to the performer. While the actions of the supportive sounds were shown on the screen, the performers were not able to control them directly.

Finally, the Singing Bowls app rewarded the player’s exploration of gestures with new pitches and harmonic material. This app only allows the performer to play a limited number of pitches at a time. When the ensemble’s performance generated a new-idea message, the app rewarded the players by changing the number and pitches of rings on the screen. The pitches are taken from a sequence of scales so that as the performers explore different gestures together, they experience a sense of harmonic progression.

5.3 Experiment

The present experiment took the form of a lab-based study under controlled conditions. Although analogous to a rehearsal call for professional musicians in its length and artistic intent — a performance of this ensemble actually
5.3. Experiment

took place some four weeks later at an art exhibition — the research intent of this experiment meant that it was quite an unusual rehearsal from the musicians’ perspective.

In the experiment, two agents (the classifying agent: CLA, and generative agent: GEN) were crossed with three iPad apps (BirdsNest: BN, Singing Bowls: SB, and Snow Music: SM) to obtain the six independent conditions. The ensemble were asked to perform improvisations limited to five minutes each and to immediately fill out questionnaires after each improvisation. It was determined that 18 of these sessions would fit into a standard three-hour rehearsal session which allowed for three trials of each of the six independent conditions.

The entire rehearsal was divided into six sets of three performances (see Table 5.1) preceded by an orientation and followed by an open-ended interview. In the orientation, the performers played with each app with the agent switched off, and then on, to highlight the effects that the agent could have in each app interface. In each set, the musicians used each app once and the order of apps was permuted between sets in a balanced design following E. J. Williams (1949) to offset local learning effects. Successive performances with each app alternated between the two agents. The experiment was blinded insofar as the performers were aware that two agents were under investigation but were not made aware of the difference between them or of which agent was used in each performance.

The experiment took place in an acoustically treated recording studio (see Figure 5.4). The performers were seated in the recording room while the two experimenters were present in a separate control room. The experiment was video recorded with two angles\(^2\) which allowed the performers’ faces and screens to be seen. The sound of each iPad was recorded from the headphone output in multi-track recording software\(^3\) and simultaneously diffused through large monitor speakers behind the performers. Audio from a microphone directly in front of the ensemble as well as from a microphone

---

\(^2\)The video recorders were a GoPro HERO 3+ Black and a Zoom Q2HD both set to record in 1920*1080 resolution.

\(^3\)The audio recording was made through a Presonus Firepod interface in Apple Logic Studio.
5. Evaluating Apps and Agents

Figure 5.4: The ensemble setup for the lab study shown from one of two camera angles. Each performer's sound was dispersed through a large loudspeaker directly behind them and simultaneously recorded.

In front of the experimenter was also recorded to capture discussion during the experiment and during the post-session interview. In each performance session all touch-interaction messages from the three performers’ iPads were recorded (even though only the CLA agent made use of this information), as were the messages returned to the performers by the agents.

5.3.1 Participants

The participants in this study (Performer A, Performer B, and Performer C) were members of Ensemble Metatone and, as such, had previous experience using the apps and agents under investigation. All three participants were professional percussionists and had worked together previously in educational and professional contexts. The fourth member of this ensemble (Experimenter A) was also the designer of the apps and agents but did not participate in the performances in this study. A second researcher (Experimenter B) assisted with running the study. The two experimenters were also experienced musicians. The three performers were chosen to partici-
Table 5.2: The questionnaire filled out by performers after each improvisation consisted of these seven questions. Each was answered on a 5-point Likert-style scale (*terrible, bad, neutral, good, excellent*).

<table>
<thead>
<tr>
<th>Q#</th>
<th>Question Text</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>How would you rate that performance?</td>
<td>quality</td>
</tr>
<tr>
<td>Q2</td>
<td>How would you rate the level of creativity in that performance?</td>
<td>creativity</td>
</tr>
<tr>
<td>Q3</td>
<td>How did the agent’s impact compare to having it switched off?</td>
<td>agent impact</td>
</tr>
<tr>
<td>Q4</td>
<td>How well were you able to respond to the app’s actions?</td>
<td>response to app</td>
</tr>
<tr>
<td>Q5</td>
<td>How well were you able to respond to the other players’ actions?</td>
<td>response to group</td>
</tr>
<tr>
<td>Q6</td>
<td>How was the app’s influence on your own playing?</td>
<td>app influence (self)</td>
</tr>
<tr>
<td>Q7</td>
<td>How was the app’s influence on the group performance?</td>
<td>app influence (group)</td>
</tr>
</tbody>
</table>

5.3.2 Questionnaires

At the end of each performance, the performers filled out written surveys consisting of the questions listed in Table 5.2. The responses were given on a five point Likert-style scale\(^4\) (*terrible, bad, neutral, good, excellent*). The two experimenters present during the lab study were also surveyed on Question 1.

\(^4\)The specific response labels for each question are listed in §D.1
5. Evaluating Apps and Agents

Ordinal rating scale questionnaires have previously been used in creativity and musical performance research to assess multiple facets of solo, free-improvisations on piano keyboards (Eisenberg & Thompson, 2003) and jazz improvisations on wind instruments (D. T. Smith, 2009). The format and content of the questionnaire in the present study followed these evaluations of improvised performance in evaluating overall quality, creativity, and ensemble interaction. Previous informal evaluations reported in Chapter 4 had suggested that these aspects of improvised performance were affected by agent-interactions. The questionnaire also included specific questions targeting the perceived impact of the agents and the changes that they caused in the apps.

5.4 Results

In the following sections the data collected in the study session will be analysed and discussed. This corpus consists of 57 minutes of interviews, 92 minutes of performances, 32.2MB of touch and interaction data as well as the experimenters’ notes. The quantitative data from surveys and agent-app interactions will be addressed before considering the interview responses.

5.4.1 Survey Data

The survey responses from each question were analysed separately using univariate two-way repeated measures analysis of variance (ANOVA) procedures to determine the significance of main and interaction effects of the two independent variables (app and agent). Post-hoc Bonferroni-corrected paired t-tests were used to assess significant variations between each of the six experimental conditions and the three apps. This is a standard procedure for significance testing used in human-computer interaction studies (Lazar et al., 2010, §4.5.3, §4.4.2). The analysis was performed in R following procedures outlined by Field et al. (2012, §12.5).

The distribution of the responses to the survey questions are shown in Figure 5.5 with responses divided by app and agent. In these box-plots, the middle of the box indicates the median of the data, while the bottom
5.4. Results

<table>
<thead>
<tr>
<th>Question</th>
<th>BN</th>
<th>SB</th>
<th>SM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1: quality</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Q2: creativity</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Q3: agent impact</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Q4: response to app</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Q5: response to group</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Q6: app influence (self)</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Q7: app influence (group)</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure 5.5: Box-plots of responses to each question in the survey. Q1 includes responses from the two experimenters and three performers, while Q2-7 were answered only by the performers. Acronyms refer to the experimental conditions as follows BN — BirdsNest, SB — Singing Bowls, SM — Snow Music, GEN — generative agent, and CLA — classifying agent.
Figure 5.6: Performances ordered by the mean response to all questions. The error bars show standard error and the grand mean is shown as a dashed horizontal line. The distribution of apps is striking with SB eliciting the highest ratings. Two thirds of the CLA agent performances appear above the mean.
and top edges of the box show the first and third quartiles respectively (McGill et al., 1978). The whiskers show the extent of data within 1.5 times the interquartile range from each edge. Data outside of the whiskers is taken to be an outlier and marked with a dot. Results from five of these seven questions (1,2,4,6,7) were found to be significant and will be considered below in detail. The other questions (3,5) were not significantly affected by the change of apps and agents. The normal variations of musical interactions in between five minute performances may have affected these questions more than the independent variables.

**Mean Response**

Figure 5.6 shows the mean response to all questions, yielding a holistic overview of the results. For the apps, this figure shows that, in general, Singing Bowls was rated higher than Snow Music, which in turn was rated higher than BirdsNest. For the agents, performances with the classifying agent were, in general, more highly rated than those with the generative agent, with six of the nine classifier performances appearing above the grand mean.

**Performance Quality and Creativity**

Questions 1 and 2 of the survey concerned the overall level of quality and creativity in each performance; the distribution of responses to these questions are shown in Figure 5.7. Considering all responses to Question 1 in the survey (including the two experimenters), the app used had a significant effect on the perception of quality in the performances ($F(2, 8) = 5.006, p < 0.05$). The main effect of the agent and the interaction effect were not found to be significant. Bonferroni-corrected paired $t$-tests revealed that, without considering the agent, performances with the Singing Bowls app

---

5Frigge et al. (1989) outlined several implementations of the box-plot used in statistical software. The details of the implementation for the plots produced for this thesis can be found in the **ggplot2** R package documentation (Wickham, 2013).
5. Evaluating Apps and Agents

![Box plots of Q1: quality and Q2: creativity ratings by app and agent.]

**Figure 5.7:** Distribution of performance quality (Question 1) and creativity (Question 2) ratings by app and agent. For both questions, the app had a significant effect on ratings.

were of significantly higher quality than with Snow Music ($p = 0.04$) and BirdsNest ($p = 0.002$).

A significant main effect of the app was also observed on the performers’ ratings of the level of creativity in their performances ($F(2, 4) = 8.699, p < 0.05$). Bonferroni-corrected paired t-tests only showed that performances of Singing Bowls with the generative agent were rated as significantly more creative ($p < 0.05$) than Snow Music with the generative agent and BirdsNest with either agent.

**Responding to the App’s Actions**

The performers were surveyed on how well they were able to respond to changes in the app interfaces caused by the agent (Question 4). Although the agent and app worked together as a system to change the interface, on the survey this was called “the app’s actions” as from the performers’ perspective, they were only aware of changes in the app.

A box plot of the results are shown in Figure 5.8. There was a significant effect of the app ($F(2, 4) = 13.32, p < 0.05$) and a significant interaction effect between agent and app ($F(2, 4) = 7.75, p < 0.05$). The effect of the agent was of borderline significance ($F(1, 2) = 16, p = 0.0572$). Post-hoc Bonferroni-corrected pairwise t-tests revealed that the performers were
5.4. Results

Figure 5.8: Distribution of ratings of individual performers’ responses to the agents’ actions (Question 4). The main effect of the app and an interaction effect between app and agent were found to be significant.

able to respond to the Singing Bowls app in combination with the classifying agent significantly better than for the other two apps with either agent ($p < 0.05$) but with only borderline significance against the Singing Bowls app with the generative agent ($p = 0.11$). These tests revealed that when using the BirdsNest and Snow Music apps and the classifying agent, performers reported that they were better able to respond to the app’s actions than with the generative agent, although significance was borderline ($p < 0.1$).

**App/Agent Influence**

Questions 6 and 7 both relate to the influence of the app and agent on the performance with the former asking about impact on the individual and the latter on the group. By uni-variate ANOVA, the effect of the app on the performers’ own playing was found to be significant ($F(2, 4) = 137.2, p < 0.01$). The effect of the agent on the group performance was of borderline significance ($F(1, 2) = 16, p = 0.0572$).

A multivariate ANOVA on both outcomes showed significance only for the app’s effect ($F(2, 4) = 4.238, p < 0.05$). These results suggest that although the app interface was the most important factor in the participants’ perceptions of their own playing, the agent was a more important factor
5. Evaluating Apps and Agents

Figure 5.9: Distribution of responses about the influence of the app and agent on the performers’ own playing (Q6), and the group’s playing (Q7). In Q6, the app had a significant effect, while in Q7 the agent appears to have been more important.

Figure 5.10: Distribution of the number of new-idea messages sent during performances. The GEN agent failed to match the CLA agent’s behaviour, but the app also had an effect.

when considering the group.

New-Ideas

As discussed in Section 5.2, the generative agent produced randomised gesture classifications based on a statistical distribution derived from a previous performance of the iPad ensemble. It was hoped that this agent would act as a control in the experiment by producing a similar number of new-
idea messages but at times that did not correlate with activity in the live performance. However, from Figure 5.10, it is clear that the classifying agent produced more new-idea messages than the generative agent. This difference can be investigated by treating the number of new-idea messages as a dependent variable.

A two-way ANOVA showed that only the effect of the agent on the number of the new-idea messages was significant ($F(1, 12) = 24.19, p < 0.001$). Although the app’s effect on new-ideas was not found to be significant, the number of new-ideas generated with Singing Bowls and the classifying agent appears higher than with Snow Music or BirdsNest (Figure 5.10). This suggests that the musicians may have performed more creatively with Singing Bowls, cycling through numerous gestures as an ensemble.

Figure 5.11 shows the performers’ responses to all questions against the number of new-ideas in each performance. A linear model of responses for each agent suggests that ratings decline as the generative agent produced more new-idea messages, while ratings increase as the classifying agent produced more messages. This may suggest that, for the generative agent, more changes due to new-idea messages annoy the performers as they do not necessarily correspond to their actions. For the classifying agent, large
numbers of new-idea messages may have been triggered by particularly creative and engaged performances which elicited higher ratings. While the generative agent did not produce the same numbers of new-idea messages as the classifying agent, if it had, the performers’ responses may have been more negative.

5.4.2 Qualitative Analysis

Video recordings were made of the orientation briefing, the 18 performances, and the post-experiment interview. Thematic analysis (Braun & Clarke, 2006) of a transcription of the interview revealed how the performers’ experiences were shaped by the three apps and their interaction with the agents. This qualitative analysis was used to direct a redesign of two of the apps leading up to a concert performance by the ensemble four weeks after the experiment.

From the interview data, and confirming the analysis of the Likert data for Question 1 and 2, the performers were most satisfied with the Singing Bowls app. It was noted that Singing Bowls was “familiar as an instrument . . . responds as we’re used to” (Perf. B) and that the “time went quickly in performances” (Perf. B). The ensemble noticed some performances (with the generative agent) where the Singing Bowls app “didn’t change at all” (Perf. C). Rather than being discouraged, the performers tried actively to “get it to respond by copying and mimicking and getting everybody to change” (Perf. A). Because of this positive reception, Singing Bowls was deemed a success and its design was not revisited before the concert performance.

In marked contrast to Singing Bowls, performances with Snow Music felt like they “went on forever” (Perf. A), and suffered from a lack of structure and motivation to keep playing with the smaller palette of snow sounds. The performers suggested that the “use of space” (i.e. silence) in Snow Music performances could improve quality and allow focus on particular sounds. The interaction with the supporting sounds was described as “lovely” (Perf. A) and “would play some really good stuff” (Perf. C). In response to these
comments, design revisions were made to add to the palette of snow sounds and to refine the synthesis system to be more expressive. A sequence of harmonies was added to the pitched sounds in this app to encourage the group to continue exploring throughout the performance.

BirdsNest performances suffered the lowest ratings from the performers who felt annoyed and “isolated” (Perf. A) by the disruptive interaction between the app and agent and found it “really hard” (Perf. C) to use creatively. While the app’s sounds were “pretty” (Perf. A) it was “hard to have that flow of ideas between people” (Perf. C). It was noted that BirdsNest was “less similar than an instrument” (Perf. B) than the other apps and that the sounds were “long . . . and the change in pitch is . . . less perceptible” (Perf. C). Following these comments, BirdsNest was extensively revised for later concerts. The autoplay feature and the disruptive control of the looping function were removed. A sequence of images with corresponding scales and subsets of the sound palette was devised to form a compositional structure for performances. The sounds button was retained to refresh the palette of sounds for each scene, but, as with Singing Bowls, movement through the compositional structure depended on new-idea messages. The synthesis system for playing bird samples was refined to play sounds of varying, but usually much shorter, length.

The qualitative analysis suggested that, from the performers’ point of view, the source of the agent’s interventions (either responding to their gestures or generated from a model) was not as important as the way that the apps responded to these interventions. The rewarding paradigm used in Singing Bowls was the most successful in engaging the performers’ interest. It was notable that with Singing Bowls the performers sought out agent interactions, particularly when the agent did not respond as was the case with the generative agent.

5.5 Discussion

The primary statistical limitation of the present study was the small number of participants surveyed. Only three participants with very specialised
5. Evaluating Apps and Agents

skills were surveyed, so any generalisation of their responses is limited. The goal of this study was not to evaluate performances by inexperienced players but by practised iPad musicians with an important stake in the quality of the instruments they use. The participants were an expert iPad ensemble with extensive performance experience and, as a result, more experimental conditions and more improvisations could be examined than would be feasible with beginners. Given the strong preference for the Singing Bowls app, future studies with more participants may be warranted that focus only on this app to reduce the number of required trials. There is also potential for other control conditions with more dramatic contrasts to the classifying agent, for example, an agent that sends completely random or no gesture messages to the apps. It is also possible that more sophisticated statistical models, such as a generalised linear mixed-effects model (Breslow & Clayton, 1993), could reveal further insights into the survey data captured in this study.

The multi-track audio and video recordings of the 18 improvised performances and corresponding touch gesture data were important outcomes of this study and these are available online\(^6\). Other studies have used detailed logs of improvisations as a basis for analyses of keyboard (Dean et al., 2014; Gregorio et al., 2015; Pressing, 1987) and live-coding (Swift et al., 2014) performances. Further investigation of these experiment protocols could lead to new understandings of the structure of touch-screen improvisation and point to potential improvements in the gesture-classifying agent’s ability to track such performances.

5.6 Summary

This chapter described the evaluation of a system for ensemble touch-screen musical performance including two server-based agents and three iPad apps. One agent classified performers’ gestures to track new ideas while the other generated similar messages from a statistical model. The three iPad apps

---

\(^6\)Performance recordings for this study have been published at [http://dx.doi.org/10.5281/zenodo.51570](http://dx.doi.org/10.5281/zenodo.51570) (C. Martin, Hopgood, Lam, & Griffiths, 2016)
used the responses from these agents to support, disrupt, and reward gestural exploration in collaborative improvised performances.

This system was investigated through its use in real musical performances in a formal, order-balanced study that considered surveys, interviews, and interaction data by an expert iPad ensemble with 14 months of experience. The participants’ high skill level allowed a novel experimental design to be introduced where a total of 18 performances over six conditions were recorded and evaluated in one rehearsal session.

The different apps were found to have a significant main effect on the performers’ perception of improvisation quality and creativity, how well they were able to respond to interface changes, and the app’s influence on individual playing. The main effect due to the agent was found only to have borderline significance on the app’s influence on the group performance and the performers’ ability to respond to interface changes. However, this question did reveal a significant interaction effect between the app and agent conditions.

While significant effects due to the agent were somewhat elusive, investigating data logs of the performances revealed that the generative agent produced significantly fewer new-idea messages than the classifying agent. This shortcoming of our system may have limited the contrasting effects of the two agents. Modelling the performer responses with respect to the number of new-idea messages suggests that performances with many classified new-ideas were rated highly, but frequent generated new-ideas may have had a negative impact on ratings.

From the results of this study, it can be concluded that the design of an agent’s interaction with a creative interface can make or break a performer’s positive perception of this interaction; this design can also limit or enhance dynamic and adventurous playing. While experienced performers can create high quality, creative performances mediated by agents of many designs, connecting the agent to their actions seems to have a positive effect on how they respond to interface changes and their perception of the group performance. Rewarding users for their collaborative exploration was found to be an especially engaging paradigm for supporting creativity. The idea
5. Evaluating Apps and Agents

of disrupting performers’ flow to encourage more creative interaction was roundly rejected in both quantitative and qualitative results.

This study has been a snapshot of the participants’ ongoing artistic practice, with recommendations from the performers taken into account in updates to the apps for subsequent performances. In Chapter 6, design guidelines for agent-app interactions will be further articulated. Given the success of Singing Bowls in the present evaluation, a new app, PhaseRings, will be introduced that extends the interface and agent-interaction to a range of sounds and composition possibilities. In contrast to the experienced group discussed in this chapter, PhaseRings and Metatone Classifier will be further investigated with groups of new iPad performers in Chapter 7.
Refining Networked Interactions for Touch-Screen Apps

Working with software-based musical instruments presents unique opportunities for enhancing ensemble performance by sharing information among the networked instruments. This is particularly pertinent for improvised performance, where ensemble creativity and performance quality are closely related to a feeling of synchronisation between the performers. While touch-screen musical instruments are becoming commonplace in music education and professional performance, software design for such instruments has yet to take group performance into account to truly support and extend musicians’ existing creative practices.

So far in this thesis, four apps, MetaLonsdale, BirdsNest, Singing Bowls, and Snow Music, have been presented that were designed specifically for performing free-improvised ensemble music. These apps have employed a number of designs for sharing information across the ensemble of devices using network interactions that are intended to assist the performers to create coherent musical structures and to encourage them to explore a wider range of sonic ideas in their improvisation. The apps have used the two network architectures presented in Chapter 2 and illustrated in Figure 6.1, a LPN, and an EDA. While the earlier MetaLonsdale used an LPN approach, BirdsNest, Snow Music, and Singing Bowls were designed to interact with the Metatone Classifier EDA. The study reported in Chapter 3 (§3.6, p. 54), suggested that MetaLonsdale’s LPN features did enhance performances and
the HCI study that was presented in Chapter 5 found that the Singing Bowls app was the most successful at engaging the performers’ creativity. These results justify further investigation of the interface used in Singing Bowls, and the LPN, as well as EDA approaches to ensemble interaction.

In this chapter, a new app, PhaseRings, will be presented that integrates the most successful elements from both LPN and EDA interactions with the annular interface of Singing Bowls. This app continues to focus on simple percussive interactions, but includes many more features than previous apps. These features, and the design motivations behind them, will be described in detail. In the latter part of this chapter, the network interaction designs of all the apps presented in this thesis will be further characterised by the musical data that is affected, and the performance events that trigger these interactions. These designs will be compared with respect to performance feedback reported in previous chapters leading to the conclusion that the syncing strategies are successful in stimulating cohesive and adventurous performances, but that the automatic changes in user interface can present musical challenges to the performers. Even though these challenges can disrupt musical flow, they may be one of a range of interactions that form part of the group mind experience. Some of the work in this chapter has been previously published as “That Syncing Feeling: Networked Strategies for Enabling Ensemble Creativity in iPad Musicians”, in the Proceedings of CreateWorld 2015 Conference (C. Martin & Gardner, 2015).

6.1 PhaseRings

The formal evaluation performed in Chapter 5 and several performance and installation experiences had demonstrated that the annular design of Singing Bowls was an appealing and expressive interface for performers and non-performers alike. However the app was limited to a single bell sound, one sequence of scales from which the pitch-rings were generated, and required ensemble interactions with Metatone Classifier to advance through different pitch setups. A new app, PhaseRings, was designed to address
6.1. PhaseRings

Figure 6.1: The touch-screen apps have used two network architectures: LPN (above), where the apps connect to each other to share information, and EDA (below), where the apps connect to a server-based agent that mediates the performance. Either or both of these models can be used to sync various kinds of musical data during performances.

Figure 6.2: PhaseRings shares Singing Bowls’ annular interface with rings representing pitches that can be tapped, swirled, or swiped. A stepper control allows the user to access different setups, and a gear icon allows access to a configuration menu. The current scale is displayed in a text label. A video demo of PhaseRings is available at: http://dx.doi.org/10.5281/zenodo.51822
these limitations by providing a range of sound options that might appeal to a broader musical audience and new performers. Most importantly to the discussion in this chapter, PhaseRings included both EDA and LPN interactions that controlled progression through setups of notes. This allowed PhaseRings to be used in a variety of ensemble situations including those where Metatone Classifier might not be available.

PhaseRings included a settings menu in the main interface where users could directly control the sound of the app, and the particular scales used for constructing the ring setups. Following iPad design conventions, access for the menu is via a small gear-shaped icon and the menu appears as a popover pane directly above the ring interface as shown in Figure 6.3. This is a dramatic revision of the design specifications set out earlier in this thesis (Section 3.2.1) where an in-app menu was framed as being in opposition to natural musical exploration of an app’s capabilities. Compromising on this specification allowed the menu in PhaseRings to provide a straightforward way of accessing particular sounds, rather than exploring all possible sounds. This was seen as more useful to users outside of Ensemble Metatone and the other research participants who had used Singing Bowls.

Some features of the PhaseRings app relied on new capabilities of Apple’s iOS operating system that had recently become available. Apps were allowed to respond to touch radius in addition to location and velocity of touches. This parameter roughly captures the size of the user’s touch, and can be controlled by pressing finger more or less firmly into the screen, or using the flat or tip of a finger or thumb. Following other apps such as Orphion (Trump & Bullock, 2014), touch radius was mapped to the volume of tapped sounds in PhaseRings, and the timbre of some synthesised sounds.

6.1.1 Sound Schemes

PhaseRings includes seven sound schemes that are configurable through the menu shown in Figure 6.4. Five of these schemes (Singing Bowls, Gongs, Crotales, Terracotta Pots, and Marimba) are produced using percussive samples featured in apps previously described in this thesis. These sounds
are controlled similarly to the Singing Bowls app; taps are mapped to tuned playback of the sample with a percussive envelope, swirls control the volume of a continuous sound produced using granular synthesis of the sample material.

Two new sounds are produced via well-known pure synthesis processes. The “Phase Synthesis” sound is produced via a phase modulation algorithm as presented by Puckette (2007, pp. 141–145). Unlike the relatively sweet sound of the percussive samples, the parameters for the Phase Synthesis sound were chosen to produce a complex, noisy timbre that is far removed from the fundamental frequency of the carrier sine wave. Tapping with a larger touch radius results in an even more distorted sound, and the angular velocity of swirled sounds also adjusts the modulation. The second pure synthesis sound is a modelled string sound produced using the Karplus and Strong (1983) algorithm. In this case, sustained sounds are replaced with a tremolo, or rapidly triggered notes. The timbre of the string synthesiser
6. Refining Networked Interactions for Touch-Screen Apps

Figure 6.4: PhaseRings’ sound schemes include sampled sounds of several percussion instruments such as gongs, almglocken, marimba and crotales. Two pure-synthesis sounds are also included: a modelled string sound, and a phase synthesis sound.

is also variable, through touch radius for tapped sounds, and the angular velocity of swirled or swiped sounds.

6.1.2 Compositions

Similarly to Singing Bowls, PhaseRings presents the performer with pitches from one of three scales in each setup displayed on the screen. The scales used in this sequence, however, can be selected from four preset sequences and one fully configurable sequence. Each of these sequences of three scales is called a composition in PhaseRings. When performing with the app, six pitch setups based on the three scales are available to the player, either through the GUI buttons, or via the EDA reward interaction first used in Singing Bowls. When a composition is loaded, two setups of pitches are generated from each scale, with a minimum of three and maximum
of eleven pitches. These setups are retained while proceeding through the
cyclic sequence of setups until another composition is loaded.

Each of the preset compositions simply consists of three scales chosen by
the author and a performer who collaborated on the latter design process. A
custom composition preset allows the user to choose each scale, and its root
pitch, so that tonalities can be chosen for the users’ own performances. The
menus for these options are shown in Figure 6.5. The scale options follow
the scales integrated into the ScaleMaker class discussed in section 3.3, and
includes the major, and melodic minor modes, octatonic, and whole-tone
scales.

6.1.3 Local and Agent Interactions

PhaseRings integrates the reward paradigm for EDA interaction introduced
in Singing Bowls. When connected to Metatone Classifier during an ensemble
performance, the app responds to new-idea messages by advancing to
the next pitch setup in its sequence. However, the app also includes an LPN
paradigm similar to that first introduced with MetaLonsdale. The stepper
(plus and minus) buttons at the bottom of the screen allow the user to
move forward or backwards through the six available pitch setups. In en-
semble performances, PhaseRings automatically connects to other apps on
the same WiFi network and communicates any setup changes to the other
connected apps. When Metatone Classifier is also running, the EDA and
LPN are both available simultaneously.

This hybrid approach to ensemble DMI design in PhaseRings enabled
a high degree of flexibility in staging performances with the app. In some
performances situations (for example, a short demonstration in a seminar),
it is less practical to set up a laptop running Metatone Classifier on stage.
In this case, an LPN performance could be held with a battery-powered
WiFi hub carried onto the stage with the iPads.

Some performances demanded EDA interactions, but not LPN, such
as improvisation workshops with new performers. To accommodate this
situation, the presence of the stepper control in the PhaseRings interface
Figure 6.5: Menu options for choosing composition (top left), including a custom composition that can be defined by the user (top right). The custom composition consists of three scales defined by a base note (bottom left), and a tonality (bottom right). Two pitch setups are generated from each of these scales by choosing a randomised selection of between three and eleven notes.
6.1. PhaseRings

Figure 6.6: The PhaseRings interface background changes colour from grey to black to indicate that the app has connected to a Metatone Classifier server.

was configurable, but only through a “performance start” message sent by Metatone Classifier. Performance start messages consist of two arguments: type, which indicates the interface configuration and whether EDA interactions should be active, and composition, which indicates which of the preset compositions (as listed in 6.1.2) should be active in the performance. This message is sent to PhaseRings apps as soon as they connect to the Metatone Classifier agent and ensures that all apps in the performance have the same composition loaded and the same interface setup. Ensuring that all apps in a performance have connected to Metatone Classifier is indicated by changing the background colour of the interface (as shown in Figure 6.6) and this enables the ensemble members to quickly check that their apps are ready for performance.

6.1.4 Inter-App Connectivity

After PhaseRings’ initial release, it was reviewed by users and on blogs specialising in mobile music practice (see Section A.6, p. 209). While this feedback was positive, users frequently requested that the app support systems for inter-app connectivity, so that users can, for example, play sounds in PhaseRings, apply effects in another app, and record in a third, all on the
same device. To meet this requirement, the app was revised to support the AudioBus system (Audiobus Pty. Ltd., 2012), Apple’s built-in inter-app audio system, and to send and respond to internal MIDI messages. These enhancements were not related to the research questions addressed in this thesis; however, they are noted to emphasise that PhaseRings was intended for a broader audience than previous apps, including individual musicians rather than ensembles.

6.2 Designs for Networked Interaction

The apps presented in this research, MetaLonsdale, BirdsNest, Snow Music, Singing Bowls, and PhaseRings, have each been designed to exchange high-level musical information over a network during ensemble performances without disturbing the performers’ fundamental control over their own music-making. The designs for networked interaction can be characterised by the network architecture that is required, the trigger for the DMI interface to change, and the musical data that is affected. The connections between these three aspects are illustrated in Figure 6.7. The two networked architectures were the EDA, with a centralised server, and the distributed LPN with direct connections between each app. Three triggers have been explored in the Metatone apps, user-interface actions by the performers, new-idea triggers from the Metatone Classifier EDA, and gesture-runs — sequences of identical gestures returned by the Metatone Classifier to a particular iPad. The musical data affected by these triggers were tonality, sound palette, app features, and secondary sounds. In the following sections, these designs will be summarised with respect to these four kinds of musical data.

6.2.1 App Features

MetaLonsdale and BirdsNest included network interactions that change instrument functions under musicians’ fingers. Both of these apps had three GUI controls: switches for the looping and autoplay features, and a button to shuffle the available sounds. The two apps, however, used different
Figure 6.7: The networked interaction designs can be characterised by their Architecture, Trigger, and the Musical Data they affect. This diagram illustrates these connections.
triggers and network architectures to exchange this information during performances. MetaLonsdale implemented selective syncing of these features between the performers. When one performer turned looping or autoplay on, or shuffled their sounds, a message was sent to every iPad in the group which then choose whether to make a change in response. The rationale behind this interaction was to stimulate the performers to try new ideas in tandem with others in the group. Changes to these features in BirdsNest were triggered by runs of identical gestures. If, for instance, a performer tapped continually, the app might switch on looping. This design was intended to discourage performers from staying on the same touch gestures for too long by changing the configuration of the app interface. If the performers appreciated the change, they could leave it switched on, but they were also free to change the interface back to the way they wanted it.

Both of these strategies for changing the app functionality might feel disruptive to the performers, but the musical challenge of working with the new interface configuration could also be fun. In the study reported in Section 3.6 (p. 54), one of the performers commented on MetaLonsdale: “sometimes it will throw you a curve ball and you’re stuck with something you don’t want and you have to find a way of making something meaningful of it.” On the other hand, the performers were conscious of how their exploration in MetaLonsdale might affect others: “I thought I was a bit of a bossy boots . . . because I kept pressing the change sounds button and then everyone would change.” The EDA-mediated disruption in BirdsNest was less successful than that in MetaLonsdale, and left performers feeling isolated by the changes in the interface. It could be that without the connection to other performers’ explorations, the disruption proved to be an annoying and unwelcome intrusion. This interaction was disabled in BirdsNest following the study reported in Chapter 5.

6.2.2 Tonality

MetaLonsdale, Singing Bowls, and PhaseRings have all featured cyclic progressions of scales that form a harmonic context for the performers’ impro-
6.2. Designs for Networked Interaction

Visions. Syncing changes in scale across the group’s iPads in these apps create cohesive harmonic progressions across ensembles and natural segments in their improvisations. MetaLonsdale used direct LPN connections between the iPads triggered by the GUI sounds button to make sure that whenever one iPad changed scale, the others followed. In Singing Bowls, changes in pitch setup and through the sequence of scales were triggered by new-idea messages from Metatone Classifier. In Section 5.4.2 (p. 118), the performers reported that they worked together to trigger new-idea messages so that they could access new notes and this interaction was the most successful in that study.

Singing Bowls had a focussed design where only EDA interactions were allowed to change the interface. This certainly prevents performers from cycling too quickly through the setups and Ensemble Metatone’s performances have shown that this can result in high-quality improvisations. The performers’ direct interactions with the GUI button in MetaLonsdale also seemed to be used musically to positive effect. PhaseRings was designed to update its tonality and pitch setups in response to both the user-interface and new-idea triggers. The stepper button shown in Figure 6.2 allows performers to directly navigate through the pitch setups which affected the other performers’ iPads as well. The new-idea interaction from Singing Bowls can be configured to be simultaneously active and PhaseRings is the only app in this research where the same musical data is affected by two different networked triggers.

6.2.3 Sound Palette

BirdsNest and Snow Music allowed the performers to play with one sound from a palette of recorded material. In Snow Music, these sounds were all field-recordings of snow being manipulated with hands and feet, while in BirdsNest, they consisted of samples of birds, percussion instruments, and forest field recordings. Both of these apps used new-idea messages from Metatone Classifier to trigger changes in their sound palettes. Snow Music’s interaction was very simple; only one snow sample was available to
the user at a time, and a new-idea message triggered the selection of a new one from the library. In BirdsNest, a sound-palette corresponds to each of the four forest scenes in the app. The performers can used the sounds button to explore within each palette, while new-idea messages trigger a change to the next palette in the sequence.

The evaluation in Section 5.4 (p. 110) suggested that the simplistic design in Snow Music resulted in boring performances from the musicians’ perspective, but this was resolved to some extent by adding more snow samples to the palette. BirdsNest, too, was more successful after the study due to refinements to its sound palettes. The possibility of exploration within a sound palette using a direct GUI control seemed to be desirable.

6.2.4 Secondary Sounds

Snow Music was equipped with a number of secondary sounds that could be layered with sounds controlled by the user during performances. These consisted of three algorithmic sound generators (bells, snow, and cymbals), and glockenspiel and cymbal layers over the performer-controlled snow sounds. Unlike the autoplay feature in BirdsNest and MetaLonsdale, the performers had no control over these sounds, which were entirely triggered by gesture-runs detected from Metatone Classifier. Even though the secondary sounds were triggered by the same mechanism as the disruptive app features in BirdsNest, they did not interrupt the performer’s control over the primary snow sounds in the app and so had a supportive function. The use of gesture-runs to trigger secondary sounds seemed to be much gentler than triggering app features and was more highly rated in the Chapter 5 evaluation.

These secondary sounds relied on gesture classifications from the Metatone Classifier EDA, but did not rely on the ensemble context. It would be possible to implement gesture classifications inside the app for solo performances. Alternatively, future additions to Metatone Classifier could track gesture-runs over the whole ensemble or subsets of performers. The algorithmic generators that perform the secondary sounds could also be con-
6.3 Chapter Summary

The apps discussed in this chapter include various strategies for enhancing improvised ensemble performance where musical data was synchronised across a group’s iPads. This information was shared using an LPN of connections between iPads, as well as connections to the Metatone Classifier EDA. The apps sync the scales available to the performers across the ensemble, advance through palettes of sound material, randomly match the app’s functions, and control secondary backing sounds. These musical data are affected by different triggers in the five apps; GUI buttons and switches are the simplest and most easily implemented in an LPN, but the ensemble-tracking Metatone Classifier EDA afforded tracking new-idea moments, and runs of identical gestures. Having implemented and tested these designs, PhaseRings, which contains both LPN and EDA interactions, was introduced in this chapter.

So do these syncing strategies help the ensemble reach the group mind state? The tonality and app feature syncing in MetaLonsdale had a notable effect on the cohesiveness of performances, but the musicians were conscious of the “bossy boots” effect that their own explorations had on other players. BirdsNest’s disruptive response to gesture-runs ended up isolating users, but the secondary sounds in Snow Music were received more warmly. An ensemble-wide implementation of the gesture-runs trigger could be more effective at bringing the ensemble together. In Singing Bowls, posing access to new notes and tonalities as a reward for interesting ensemble interaction has proved to be successful and so was also included in PhaseRings along with GUI buttons to access the various pitch setups. Even though “curve ball” situations can frustrate performers, disruption of the musical status quo appears to help create stimulating and challenging performances in
some cases. One musician remarked that when playing MetaLonsdale he tended to “stick on the same thing for 20 minutes so it’s good that you changed it.” Changes in the app interface, then, can be positive, but it is a matter of when, as in MetaLonsdale, and why, as in the well-received reward interface changes in Singing Bowls.

The network interaction designs implemented in the five apps discussed in this chapter seem to have allowed Ensemble Metatone to extend their practices in a way that existing acoustic or electronic instruments cannot. R. K. Sawyer (1999) pointed out that the emergence of creativity in collaborative performances cannot be fully explained by individual analysis of the performers. Based on the experiences reported in this chapter, a corollary to this statement could be that systems for computer supported group creativity must take group interactions into account. That is, the system must enhance the communications, negotiations, and game-play that mark free-improvisations in order to extend the musicians’ feeling of group flow.
Comparing Agent-Control with a Networked User-Interface

Novel DMIs involving laptops (Lee et al., 2012), mobile phones (Oh et al., 2010), tablets (Iglesia, 2014), or DIY instruments (Rosli et al., 2015) frequently have musical ensembles as a focus of their performance practice. Chapter 2 introduced a range of ensemble interactions that have been built into ensemble DMIs. Two of these interactions, a Local Performance Network (LPN) and an Ensemble Director Agent (EDA), were later implemented in a variety of iPad apps. PhaseRings, which was described in Chapter 6, included the possibility of using both or either architecture to control progression through a sequence of interfaces. The EDA that is used with PhaseRings was described in Chapter 4 and could be termed an intelligent listener. This agent intervenes only occasionally in a musical performance. It was designed to listen continuously and occasionally nudges human performers by making suggestions via re-configurations of their interfaces.

The experiences and studies reported in chapters 4 and 5 suggest that there is a compelling rationale for the design of such an agent and that it assisted with a successful artistic practice. However, the question still remains whether an intelligent listener EDA really does improve free-improvised performance. On the negative side, the agent may broadcast UI updates to performers at inappropriate times, leading to a disruption of their musical focus and concentration. It could be that leaving users in control of the
group’s UI updates via an LPN-shared interface is preferable to agent interaction. This chapter describes a formal HCI study where PhaseRings is used by inexperienced iPad performers to investigate this question. Experimentally, the goal was to compare the effects of the agent (the Agent-Control factor), against a direct-interaction version of the same interface where a GUI button was accessible to all performers in the ensemble (the GUI-Control factor). These interfaces were tested individually, removed entirely (as a control condition), and combined, forming a $2 \times 2$ factorial design with four conditions: Button, Agent, Nothing, and Both.

The difficulties of systematically evaluating creativity support tools, particularly in performing arts practices, are well documented (e.g., Shneiderman (2007)). The present study follows the approach of other HCI evaluations of musical interfaces in both surveying performers as well as analysing objective data from formally-structured ensemble performances (Bryan-Kinns & Hamilton, 2012). Although it has been noted that the use of measures such as time-to-completion of musical tasks (Wanderley & Orio, 2002) might not be appropriate for studies of creative interfaces (Stowell et al., 2009), in this chapter, time-to-completion was able to be used as an objective measure because of the conceptual association of a long improvisation with a deeply engaging one. In the first study, data was collected from 16 improvisational ensemble performances in an order-balanced experiment across four different ensembles, each with four performers.

The results show that while performers preferred GUI control over their interface, and rated these performances more highly in terms of technical proficiency, complexity, creativity and quality, they improvised for significantly longer under the agent conditions. These findings motivated the development of a refined version of the iPad interface utilising both the agent and GUI controls in a mixed-initiative design. The new design was shown in a follow-up study to support better ensemble interactions and improvisation structure, as well as higher quality and more enjoyable performances. This research offers evidence that an intelligent listener agent and a networked GUI both improve aspects of improvised music-making.

The research in this chapter has been published as “Intelligent Agents
7.1 Related Work

Free-improvised ensemble music, where there are no restrictions on style and no pre-determination of the music that will be played, has a significant history of enquiry focussed on how these open-ended collaborations lead to the spontaneous creation of structured music (Stenström, 2009). Although related to free-jazz, non-idiomatic free improvisation has few boundaries and is often a process of exploration and discovery of new performance methods and sounds (Bailey, 1993). Such improvisations can be modelled as a sequence of non-overlapping event clusters that each contain the exploration of a particular musical idea (Pressing, 1988). The group interactions that lead to the emergence of a coherent structure over the performance are often considered to be a marker of skill in improvisation (R. K. Sawyer, 2006). Borgo (2005) emphasised the concept of “group mind” in ensemble improvisation, where a state of creative flow among the group leads to seemingly effortless interaction. Mazzola and Cherlin (2009) defined free
jazz performances in terms of interactions where performers must “nego-
tiate” each musical decision and engage in a “dynamic and sophisticated
game” (p. 7) with their ensemble. In fact, this emphasis on collaborative
creativity makes free-improvisation an ideal part of musical education and
has been adopted in pedagogies such as Cahn’s (2005) Creative Music Mak-
ing.

For novel DMIs, ensemble improvisation is frequently a focus of the
performance practice. The Daisyphone allows a group to collaboratively
compose looping phrases using a shared compositional workspace (Bryan-
Kinns, 2004). Jordà’s (2010) Reactable allows a group of performers to
collaboratively manipulate synthesis processes with tangible objects on a
tabletop interface. Rosli et al.’s (2015) feedback ensemble asks performers
to create their own musical device and connect to an instrumented audio
feedback network.

Laptop Orchestras pioneered the practice of large computer music en-
sembles with identical hardware and software and a repertoire of composi-
tions that frequently included improvisation (Smallwood et al., 2008) and
collaborative networked interfaces (Lee et al., 2012). Mobile music per-
formance using mobile devices such as smartphones and tablets takes ad-
vantage of the many sensors, touch screens and small form-factors of these
devices (Tanaka, 2010). Mobile music apps are now common in professional
music production (M. Jenkins, 2012), in educational performance settings
(D. A. Williams, 2014), as well as in research-focussed ensembles (Oh et al.,
2010; Wang, Essl, & Penttinen, 2008).

There are also many commercially available apps for mobile music per-
formance which, like Orphion (Trump & Bullock, 2014), Ocarina (Wang,
2014), and Magic Fiddle (Wang et al., 2011), frequently emphasise novel
multi-touch and sensor interactions that allow performers direct control
over synthesis parameters and the performance of melody. However, the
majority of commercial apps ignore possible ensemble contexts of their use
and do not make use of network connectivity between performers’ devices.
The most common approach has been to synchronise rhythmic information
between sequenced music processes on multiple devices such as in Korg’s
Wireless Sync-Start Technology (WIST) (Korg Inc., 2011), which is similar to synchronisation available using the MIDI standard’s timing clock (MIDI Manufacturers Association, 1996).

In this chapter, networked, ensemble-focussed features in a mobile music app will be compared. The experimental features in PhaseRings use direct, indirect (Shneiderman & Maes, 1997), and mixed-initiative (Horvitz, 1999) interaction models, while the Metatone Classifier EDA sits between agents that participate in and merely conduct ensemble performances. As was discussed in Chapter 2, such ensemble tracking agents have been theorised since the 1990s but, as examples and evaluations of their use are rare, the parameters for agent-performer interactions are not yet fully defined.

The evaluation of PhaseRings and Metatone Classifier in this chapter uses a refined version of the rehearsal-as-research methodology defined in Section 5.1 (p. 99). This methodology follows other research in including both quantitative and qualitative approaches (Stowell et al., 2009), and focusing on performers (O’Modhrain, 2011). The methodology follows Bryan-Kinns and Hamilton’s (2012) use of quantitative preference surveys as well as Likert-scaled ratings to allow performers to compare multiple similar interfaces.

7.2 System Design

The system used in this study included the PhaseRings app running on Apple’s iPad platform and the Metatone Classifier EDA. This software was discussed in detail in chapters 4 and 6 and will be briefly reviewed below with a focus on the elements investigated in the experiment.

7.2.1 App

PhaseRings (see §6.1, p. 124) was the only iPad app used in this study. For this experiment, a simplified version of the app was produced where the settings menu was disabled and different versions of the interface could be remotely activated by the experimenter. As previously described, PhaseRings consists of an annular interface for triggering sounds using simple per-
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.2: The architecture of the performance system. Performers use the iPad app to improvise with a selection of notes, while an agent on the server continually classifies and tracks the ensemble’s gestural changes. Under the GUI-Control condition, a UI button is visible on the screen to update the interface with new notes and sounds, while under the Agent-Control condition these changes are made by the agent.

cussive touch gestures. Each ring in the app’s interface represents a single pitch, tapping the ring will trigger a short sound, while a moving touch plays a continuous sound. In this way, the app allows direct control over basic musical concepts of pitch and rhythm in a percussive interface similar to other instruments well-suited to improvisation by beginners.

As described in Section 6.1.2 (p. 128), PhaseRings displays a limited number of pitch rings on each performer’s screen. All of these notes are taken from the same musical scale, so that the ensemble can play concordant notes, but are generated independently so that performers have different
7.2. System Design

Figure 7.3: A screenshot of the experimental version of PhaseRings prepared for the formal study. Tapping the GUI button (at the bottom of the screen) updates the interface with new rings, notes and sounds on all performers' iPads. A video demo of this interface is available at: http://dx.doi.org/10.5281/zenodo.51801

sets of pitches. Performances with PhaseRings are segmented in time by moments where the performers' interfaces are simultaneously updated with a new set of notes. For the experimental version of the app, this update also randomised the sound scheme in the app (see §6.1.1, p. 126), resulting in a different timbre.

These interface updates are triggered by two different mechanisms, roughly corresponding to direct user interface and indirect agent interaction models in HCI (Shneiderman & Maes, 1997). In the direct case, a GUI button is present in the touch interface for each performer. For the present experiment, this was simplified from the stepper controller (shown in Figure 6.2, p. 125) to a single button shown on the screen in Figure 7.1. When one player activates this button, the app interface is updated for all performers in the group. The presence of this GUI element is a factor in the experiments (GUI-Control). In the indirect case, interface updates are triggered by new-idea messages from the Metatone Classifier EDA (see §4.2.4, p. 78)
which update all performers’ interfaces simultaneously.

7.2.2 Agent

The agent triggers an interface update when it calculates that the amount of gestural change in the ensemble has exceeded a predetermined threshold. These moments can correspond to natural segments of ensemble improvisations, and so are appropriate times to intervene in the performance. As described in Section 4.2 (p. 69), the agent performs gestural classification on each performer’s touch data, once per second, to produce a sequence of gestural states. Transitions between these states are summed to calculate a transition matrix (TM) for the ensemble, calculated over a sliding 15-second window of touch data. The flux metric on TMs is used to determine the current rate of gestural change within the group. A sharp increase in this measure can indicate that the ensemble is shifting to a new musical idea. In response, the agent sends a new-idea message which updates the interface on the iPad apps with a new set of notes or sounds.

This response is posed as reward to the ensemble for exploratory behaviour. Intuitively, if a group has changed musical idea, they may appreciate new sounds to work with. This interaction results in a musical experience similar to “structured improvisation” or “game pieces”, except with an ensemble-tracking agent director, rather than a human. In Chapter 5, this agent, and the new-idea reward interaction, was compared with a design that generated random interface updates. That study suggested that the ensemble-tracking agent was more positively received than the generative agent.

In the present system, the agent runs as a server application on a laptop on the local network, to which the iPad apps connect automatically at the start of each performance. The architecture of this system is shown in Figure 7.2. The agent also logs all the touch data during the improvisations, as well as interactions with the button interface and new-idea messages. The presence of the agent in the performance was a factor (Agent-Control) in the controlled experiments. The nature of the interface changes was the same
7.3 First Experiment

The first experiment took place over four 90-minute sessions spread across two weeks. The venue was a recording studio equipped with a quadraphonic speaker system and a control room for controlling and recording the sessions and monitoring the performers. This studio setting allowed multi-track audio and video recordings to be made of the whole session.

In each experimental session, all of the participants played in four separate improvisational group performances, each with a different interface update regime: agent-controlled interface changes (Agent), button-controlled interface changes (Button), both agent- and button-controlled changes (Both), and no changes — a static interface for the whole performance (Nothing). These conditions correspond to the combinations of the two top-level factors, GUI-Control and Agent-Control. The groups were exposed to these

<table>
<thead>
<tr>
<th>Group</th>
<th>Performance 1</th>
<th>Performance 2</th>
<th>Performance 3</th>
<th>Performance 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Agent</td>
<td>Both</td>
<td>Nothing</td>
<td>Button</td>
</tr>
<tr>
<td>2</td>
<td>Nothing</td>
<td>Agent</td>
<td>Button</td>
<td>Both</td>
</tr>
<tr>
<td>3</td>
<td>Button</td>
<td>Nothing</td>
<td>Both</td>
<td>Agent</td>
</tr>
<tr>
<td>4</td>
<td>Both</td>
<td>Button</td>
<td>Agent</td>
<td>Nothing</td>
</tr>
<tr>
<td>5</td>
<td>Both</td>
<td>Fade</td>
<td>Both</td>
<td>Fade</td>
</tr>
<tr>
<td>6</td>
<td>Fade</td>
<td>Both</td>
<td>Both</td>
<td>Both</td>
</tr>
</tbody>
</table>

Table 7.1: Schedule for the experiment showing the counter-balanced ordering of conditions. Each group participated in one session including an induction, the four performances, and a debrief interview. Sessions 5 and 6 were subsequently added as a follow-up study with a revised interface.

under both the GUI-Control and Agent-Control factors and these systems could be switched on or off at the start of each performance. This was by design, to facilitate the two-factor experiment described in the next section where all combinations of Agent-Control and GUI-Control are compared: Button, Agent, Both and Nothing.
configurations in counter-balanced orders (shown in Table 7.1) chosen by taking a random order and applying Bradley’s (1958) balancing procedure. This was done to counteract any immediate sequential effects as well as the ensemble learning effects that are known to be exhibited by new groups of improvisers (Cahn, 2005).

7.3.1 Participants

Participants were recruited for this study through posters and presentations at a university music school. While the studies presented in chapters 3 and 5 had focussed on percussionists with a high level of iPad performance experience, one aim of the present study was to examine a broader group of users who may not have had experience with computer music or percussion. Twenty-five respondents were asked to select available times through a web-interface, and 16 participants (7 female, 9 male) were invited to attend one of four sessions based on availability and order of response. 14 participants were university students with 11 of these studying music. Three of the participants had previously performed with PhaseRings and were distributed into different ensembles.

7.3.2 Procedure

Each session began with a 20 minute induction during which the participants were introduced to the app and the experimental procedure and were given the chance to try each of the experimental conditions. The participants also filled out an entry survey to record demographic information and their experience levels in electronic music and improvisation. In each of the four performances, the experimenter remotely activated all of the iPad apps to indicate the start of one of the sessions and monitored the participants from the control room.

To give each condition a fair trial and to ensure the sessions did not run over time, the participants were asked to improvise for at least seven minutes but no longer than 11 minutes in each performance. After seven minutes, the performers were free to stop individually when they wished
and the performance was considered to be finished when all performers had stopped playing. If the performance ran longer than 11 minutes, the performance was halted by turning off the speaker system. The seven-minute mark was indicated to the participants by two remote controlled stage lights positioned in the studio. These lights faded to green to indicate the start of the performance and faded to blue at seven minutes to indicate that the participants could finish whenever they wanted. Participants were not aware that the time-to-completion was being used as a metric for this study.

7.3.3 Questionnaires

At the end of each improvisation, the participants were asked to fill out a survey of twenty-four Likert-style questions to record their ratings of various aspects of the performance on nine-point scales. The questionnaire was divided into five sections to evaluate aspects of the improvisations, similar to those defined by Eisenberg and Thompson (2003), and was significantly more fine-grained than the survey used in Chapter 5 (§ 5.3.2, p. 109). The sections were: technical proficiency, musical interaction, musical structure, creativity, and performance quality. The questions for the survey are listed in Table 7.2. The response to each question was given by a nine-point scale formed from unnumbered check boxes. A label was given at the extreme ends and in the centre. In the results section below, the responses are labelled from one to nine. All but one of these questions used a scale of the same orientation: very little or very bad was at 1, neutral at 5, and very much or very good at 9. Question 12, regarding the length of the performances, had labels indicating much too short at 1, perfect at 5, and much too long at 9.

At the end of the entire session, participants filled out another survey in which they were asked to choose which condition they most preferred over seven aspects of the performances. The questions for the preference survey are shown in Table 7.3. The responses to this preference survey were used

\[\text{The specific anchors of the scale for each question are listed in §D.2.1.}\]
7. Comparing Agent-Control with a Networked User-Interface

as a starting point for an unstructured discussion with each ensemble which lasted about 10 minutes.

7.4 Results

The data gathered in this study included coded survey results, performance protocols that include records of touch interactions, button presses, and agent interactions, as well as recordings of the performances and post-session interviews.

7.4.1 Survey Data

The survey data (shown in box-plots in Figure 7.4) was analysed using an Aligned Rank Transform (ART) procedure followed by a two-way mixed-effects ANOVA to assess significance. As the survey data is ordinal, rather than interval-scaled, the assumptions of the classical within-groups factorial ANOVA procedure may be violated. The ART procedure has been recommended as the most appropriate procedure for factorial HCI studies with ordinal dependent variables, such as Likert-scaled responses (Wobbrock et al., 2011). Analysis was performed in R using the ARTool v0.9.5 package (M. Kay & Wobbrock, 2015). Post-hoc analysis was performed using Bonferroni-corrected paired t-tests.

The ANOVA procedures showed that the presence of the UI button in the touch-screen interface (GUI-Control) had a significant main effect on nine questions in the survey. A detailed overview of the ANOVA results is shown in Table 7.4 and the distribution of responses for these questions is shown in Figure 7.5. When the button was present in the interface, performers reported higher personal proficiency, that the app had a more positive influence on their personal performance, and that they presented more new ideas to the group. They rated the complexity of the performances as higher and also rated their personal creativity, the creativity of others in the group, and the overall creativity more highly. Finally, they rated the quality of their personal contribution and the overall quality of
<table>
<thead>
<tr>
<th>Q#</th>
<th>Question Text</th>
<th>Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>How much did you focus on particular touch gestures in that performance?</td>
<td>gestural focus</td>
</tr>
<tr>
<td>Q2</td>
<td>How much did you explore a range of touch gestures?</td>
<td>gestural exploration</td>
</tr>
<tr>
<td>Q3</td>
<td>How would you rate your technical proficiency using the app in that performance?</td>
<td>technical proficiency</td>
</tr>
<tr>
<td>Q4</td>
<td>How much did the app impede your performance?</td>
<td>app impediment</td>
</tr>
<tr>
<td>Q5</td>
<td>How much did the app enhance your performance?</td>
<td>app enhancement</td>
</tr>
<tr>
<td>Q6</td>
<td>How much did you interact musically with the other performers?</td>
<td>personal interaction</td>
</tr>
<tr>
<td>Q7</td>
<td>How much did the other performers interact musically with you?</td>
<td>group interaction</td>
</tr>
<tr>
<td>Q8</td>
<td>How well were you able to respond to the other musicians’ actions?</td>
<td>response to ensemble</td>
</tr>
<tr>
<td>Q9</td>
<td>How well were you able to respond to the app?</td>
<td>response to app</td>
</tr>
<tr>
<td>Q10</td>
<td>How would you rate the overall level of musical interaction among the ensemble?</td>
<td>overall interaction</td>
</tr>
<tr>
<td>Q11</td>
<td>How would you rate the complexity of that performance?</td>
<td>complexity</td>
</tr>
<tr>
<td>Q12</td>
<td>How appropriate was the length of that performance?</td>
<td>length</td>
</tr>
<tr>
<td>Q13</td>
<td>How would you rate the app’s influence on your own playing?</td>
<td>app influence (self)</td>
</tr>
<tr>
<td>Q14</td>
<td>How would you rate the app’s influence on the ensemble performance?</td>
<td>app influence (group)</td>
</tr>
<tr>
<td>Q15</td>
<td>How would you rate the overall musical structure of that performance?</td>
<td>overall structure</td>
</tr>
<tr>
<td>Q16</td>
<td>How much did you present new musical ideas to the others in the ensemble?</td>
<td>self new-ideas</td>
</tr>
<tr>
<td>Q17</td>
<td>How much did you take on and develop musical ideas first presented by the others in the ensemble?</td>
<td>group new-ideas</td>
</tr>
<tr>
<td>Q18</td>
<td>How would you rate your personal creativity in that performance?</td>
<td>self creativity</td>
</tr>
<tr>
<td>Q19</td>
<td>How would you rate the other performers’ creativity in that performance?</td>
<td>group creativity</td>
</tr>
<tr>
<td>Q20</td>
<td>How would you rate the overall creativity in that performance?</td>
<td>overall creativity</td>
</tr>
<tr>
<td>Q21</td>
<td>How would you rate the quality of your contribution to that performance?</td>
<td>personal contribution</td>
</tr>
<tr>
<td>Q22</td>
<td>How would you rate the quality of the other performers’ contribution in that performance?</td>
<td>group contribution</td>
</tr>
<tr>
<td>Q23</td>
<td>How would you rate the overall quality of that performance?</td>
<td>overall quality</td>
</tr>
<tr>
<td>Q24</td>
<td>How enjoyable or unpleasant was that performance?</td>
<td>enjoyment</td>
</tr>
</tbody>
</table>

Table 7.2: Questions from the survey administered after each performance in the study. Each question was answered on a 9-point Likert-style scale.
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.4: Box-plots of responses to each question in the survey.
7.4. Results

<table>
<thead>
<tr>
<th>Q#</th>
<th>Question Text</th>
<th>Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>Which method do you think resulted in the best performance?</td>
<td>overall preference</td>
</tr>
<tr>
<td>Q2</td>
<td>Which method do you think resulted in the most creative performance?</td>
<td>most challenging</td>
</tr>
<tr>
<td>Q3</td>
<td>Which method do you think resulted in the best performance structure?</td>
<td>easiest</td>
</tr>
<tr>
<td>Q4</td>
<td>Which method do you think resulted in the best musical interaction in the ensemble?</td>
<td>best musical interaction</td>
</tr>
<tr>
<td>Q5</td>
<td>Which method did you find easiest to perform with?</td>
<td>best structure</td>
</tr>
<tr>
<td>Q6</td>
<td>Which method did you find most challenging to perform with?</td>
<td>most creative</td>
</tr>
<tr>
<td>Q7</td>
<td>Which method of triggering change in the app did you prefer?</td>
<td>best performance</td>
</tr>
</tbody>
</table>

Table 7.3: Questions from the preference survey administered at the end of each session. Each question was answered by choosing one of the experimental conditions.

The survey data did not show any significant main effect due to the Agent-Control factor.

Overall, these results suggest that the performers felt their proficiency and creativity were enhanced by the presence of the button in the interface (whether in the Button or in the Both conditions). This may have been due to the extra personal performance options, or the feeling of control that they had when interacting with the button. The main effect for GUI-Control was in spite of the potential disruption caused by one performer pressing the button and changing everyone’s interfaces; as discussed below, at times there were a very large number of these button presses during a performance.
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.5: Distribution of results for questions where a significant main effect for GUI-Control was found. “Nothing” tended to attract the lowest ratings, and “Both” the highest.

7.4.2 Post-hoc testing

Post-hoc Bonferroni-corrected paired t-tests\(^2\) confirmed that Button-only performances were rated as more complex (Q11) than Agent-only performances ($p = 0.05$). Performers thought that they presented more new ideas to the group (Q16) in Both performances than in Agent-only performances ($p = 0.04$) and Nothing performances ($p = 0.06$). They regarded their personal creativity (Q18) to be significantly higher in Both performances than Agent-only ($p = 0.02$) and they also perceived their personal contribution (Q21) to be better in Both than in Agent-only performances ($p = 0.05$).

\(^2\)These tests were performed with R's `pairwise.t.test` function which produced the adjusted $p$-values appearing here.
7.4. Results

Table 7.4: Questions showing a significant main effect for the presence of the button in the instrument interface (GUI-Control). The performers felt their proficiency, creativity and contribution to performances was enhanced when the button was present.
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.6: Distribution of the performers’ overall preferences from the exit survey. The preference for the button-only condition is apparent, especially when performers considered the easiest interface and musical interaction. The Nothing condition was considered to be the most challenging.

7.4.3 Performer Preferences

After completing performances with each of the four interfaces, the performers were asked to pick one performance condition that ranked highest across seven aspects of the performances. The results shown in Figure 7.6 convincingly demonstrate that the Button-only condition was the preferred choice across all questions except for “most challenging” and “most creative”. Chi-squared tests were used to determine how significantly the distribution of responses had deviated from chance. The questions about which condition made it easiest to perform ($\chi^2(3, N = 16) = 9, p < 0.05$) and which condition resulted in the best musical interaction ($\chi^2(3, N = 16) = 12, p < 0.01$) had significant results where performers favoured the Button-only condition. Performers rated the Nothing condition most frequently as the most challenging to use, and this question also showed a significant deviation ($\chi^2(3, N = 16) = 14, p < 0.01$).
7.4. Results

Figure 7.7: Performer-distinguished improvisation lengths by overall interface condition. The effect of Agent-Control was highly significant with longer performances under Agent and Both conditions.

7.4.4 Performance Length

As previously discussed in Section 7.3.2 (p. 148), performances in this study had a minimum duration of seven minutes and a maximum duration of 11 minutes. The precise length of the performance was defined as the time from the first sound (green lights) to the time when all performers had ceased playing. An alternative performance length was defined for each performer in the group; the start was given by the first performer’s first touch and the end for each performer is given by their own final touch. The distribution of these performer-lengths by experimental condition is given in Figure 7.7.

By treating this performer-length as a dependent variable and modelling using a two-way within-groups ANOVA procedure, a highly significant main effect due to Agent-Control was found in the lengths of performances ($F(1, 15) = 85.4, p < 0.001$). Post-hoc Bonferroni-corrected paired t-tests showed that the Agent, Button, and Both performances were longer than the Nothing performances ($p < 0.01$), but that Agent-only performances were not significantly longer than Button-only performances.

This result is at odds with the survey results that showed a significant main-effect for GUI-Control. It may be that even though performers stated that they preferred having the button present in the interface, in Agent and
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.8: In all performances, the agent tracked touch-gestures of the ensemble and attempted to identify new-idea moments, but these were only used by the app under the Agent and Both conditions. The count of new-ideas was highest in the Nothing and Both conditions, with a significant interaction between GUI-Control and Agent-Control effects.

Both performances they ended up more deeply engaged in the improvisation, leading to longer performances. Observations from video recordings of the sessions (as seen in the video figure) showed that performers appeared to be deeply engaged in every single performance in this experiment. The post-session discussions suggested that performers were often unaware of the relative length of performances.

7.4.5 New-Idea Messages

Under the Agent and Both conditions, the iPad app interfaces would be updated with new notes and sounds in response to new-idea messages sent from the agent software. However, the agent was still actively classifying new-idea instances in the Button and Nothing conditions, even though they had no effect on the interface. This means that the number of new-idea messages can be used as a dependent variable that measures how much the performers interacted with each other in the way that the agent was designed to measure.

Figure 7.8 shows the number of new-idea messages sent in performances under each experimental condition. While one might expect more new-ideas
7.4. Results

to be produced when the interface was actively responding to the new-idea messages, curiously this was not always the case. The Nothing and Both conditions seem to have higher numbers of new-idea messages than either the Agent or Button only conditions. A two-way within-groups ANOVA procedure shows a significant interaction effect ($F(1, 3) = 11.7, p < 0.05$) between the GUI-Control and Agent-Control factors and no significance for either main effect. It may be that under the Nothing condition, where there were no interface updates, the performers focussed more on their ensemble interactions. Conversely, in the Button-only condition, the button may have distracted from performing with varied touch gestures, however this may be a novelty effect that does not continue in real-world performance. The Both condition appears to have encouraged more new-idea interactions than the Agent condition. While having both interfaces active may have allowed the most natural ensemble interactions, it could be that the Agent-only interface updates confused or annoyed the performers and the resulting disruption to their interactions was seen in the lower number of new-idea messages.

7.4.6 Button Presses

During Button and Both performances, the performers were able to trigger iPad app interface updates at any time during the performances. Interaction logs recorded during the performance allowed us to see which performers pressed the button at which times. The median number of button presses in one performance per performer was 3.5; however, the maximum was 457! Of the sixteen participants, three had pressed the button more than 100 times in a single performance. Observation of the performance recordings shows that these “button maniacs” had used the button not to find new notes in the iPad app interface or to segment the performance harmonically, but to create a unique sound where all performers’ interfaces rapidly changed synthesis parameters and pitch.

Button mania was not anticipated in the design of the app, which had no mechanism to limit the frequency of button presses. For the participants,
7. **Comparing Agent-Control with a Networked User-Interface**

and the button maniacs in particular, the button may have represented not just one of two ways to segment performances, but a new musical device. In Button and Both performances, this device could be used to create a unique sound that was not available in Nothing and Agent performances. This interpretation may explain why performers felt more technically accomplished when the button was present — they were not only in control of interface updates, but able to create an entirely new sound.

### 7.5 Follow-up Study

The results of the first study in this chapter suggest that both the GUI-Control and Agent-Control factors had an effect on different aspects of the performance. Direct group interactions, via the button, affected the performers’ perceptions of the performance, while the agent may have helped them to achieve a deeper engagement resulting in longer performances. A follow-up study was then conducted to compare the Both condition with a new mixed-initiative interface that interlaced the dynamics of the Agent-Control and GUI-Control conditions. Under the new “Fade” condition, the button was normally not accessible to the performers. When the agent sent a new-idea trigger, the button was faded into view and became accessible. If a performer tapped the button, it would update the interfaces as normal and then fade away. If performers did not tap the button, it would stay visible and they could use it when they wished. In this way, the Fade condition prevented the button from being activated repeatedly, while allowing the performers to have the final say over interface changes.

The follow-up study was conducted with an identical procedure to the first study. Two groups of four participants took part; seven of the participants had been in the first study and the last one had been a performer in earlier performances and rehearsals with the same app. The two groups performed two replications of the two conditions for a total of four improvisations in balanced order (see sessions 5 and 6 in Table 7.1). The same survey was used after each performance and a preference survey with the same questions was used at the end of the session. As before, each session
Figure 7.9: Box-plots of responses for each question in the follow-up study.
7. Comparing Agent-Control with a Networked User-Interface

Figure 7.10: Distribution of results for the follow-up study where a significant effect of the interface condition was found. In all of these questions, Fade was rated more highly than Both.

Figure 7.11: Preference surveys from the follow-up study. The performers were roughly split on most questions; however, Both was seen as easier and Fade as more challenging.
concluded with a group interview. It turned out that two of the button maniacs from the first study were in one of these groups.

Again, an ART and one-way mixed effects ANOVA procedure was performed on the results of the final two performances in the follow-up study. Table 7.5 shows questions with significant main effects for the change in interface and Figure 7.10 shows the distribution of survey responses for these questions. These results show that the performers perceived an improvement in their ensemble interaction with the new Fade interface, as well as improvements in the musical structure and overall quality and enjoyment of the improvisations.

The results of the preference survey in Figure 7.11 show the participants roughly split in preference for the two interfaces, with none of the questions leading to significant differences related to the conditions. The Both condition was seen as an easier interface than the Fade condition, where particular types of ensemble interaction were required in order for the agent to present the button to the performers.

While performance length had been an important discriminating dependent variable in the original study, the two conditions in the follow-up had no significant effect on the length of improvisations, despite the Fade condition giving rise to the longest performance in each session. In the post-
7. Comparing Agent-Control with a Networked User-Interface

<table>
<thead>
<tr>
<th>Question</th>
<th>F</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>6. How much did you interact musically with the other performers?</td>
<td>26.82</td>
<td><em>p &lt; 0.01</em></td>
</tr>
<tr>
<td>7. How much did the other performers interact musically with you?</td>
<td>8.8407</td>
<td><em>p &lt; 0.05</em></td>
</tr>
<tr>
<td>10. How would you rate the overall level of musical interaction among the ensemble?</td>
<td>10.924</td>
<td><em>p &lt; 0.05</em></td>
</tr>
<tr>
<td>15. How would you rate the overall musical structure of that performance?</td>
<td>9.7391</td>
<td><em>p &lt; 0.05</em></td>
</tr>
<tr>
<td>23. How would you rate the overall quality of that performance?</td>
<td>8.7576</td>
<td><em>p &lt; 0.05</em></td>
</tr>
<tr>
<td>24. How enjoyable or unpleasant was that performance?</td>
<td>5.8973</td>
<td><em>p &lt; 0.05</em></td>
</tr>
</tbody>
</table>

Table 7.5: ART ANOVA results for questions showing significance for the effect of the different interfaces in the follow-up study. A one-way ART and mixed-effects ANOVA was performed on the results of the final two performances by each group.

performance interviews, the performers reported that they were not aware of how long the improvisations had been. It is likely that, at least for these two groups, both conditions supported an optimal length of improvisation.

7.6 Discussion

Over both studies, both in the qualitative survey responses and in the interviews, the participants expressed a great deal of enthusiasm for the experience. This once again demonstrates the ability of a simple iPad interface, judiciously augmented with some ensemble-wide interface dynamics, to give rise to real musical interaction. As discussed in Section 7.1, evaluating interfaces for open-ended collaborative creativity is a difficult task. It is encouraging to note that attempts with these studies to perform systematic, controlled evaluations of different aspects of the interface did not
lead to a lifeless, unmusical experience for the performers. The results of the two studies show that no particular interface had a clear-cut advantage over the others. The only interface that was rejected by the participants was the Nothing condition, even though some commented that the limited interface had forced them to “make do” and that they enjoyed trying to find creative ways to use their setup.

It was notable that the original and follow-up studies showed significant effects in different parts of the survey. In the original study, the GUI button interface resulted in significant effects for technical proficiency, the participants’ personal performance, improvisation complexity, and creativity. In the follow-up sessions, the revised, mixed-initiative interface achieved significant effects for group interaction, performance structure, and enjoyment. Both studies showed a significant effect on the question about overall performance quality.

The effect of Agent-Control on performance length was a particularly notable and surprising outcome of the first study. This objective measurement detected an effect that was not picked up by the subjective surveys. Significantly, the performers seemed to have little awareness of the relative length of performances and the results for the survey question on length (Q12) are clustered around “Just Right” (median of 5), with Agent-only and Button-only conditions being very slightly “too long” (median of 5.5). This suggests that, in this study, performance length could be related to the performers’ engagement with their interface and the ensemble improvisation. This engagement could be due to the agent improving the quality of the collaborative experience. Alternatively, the agent may have allowed the performers to focus more on their individual playing, improving their state of creative flow (Csikszentmihalyi, 1991). Finally, it could be that the performers were simply waiting for new notes under Agent-Control and that such delayed gratification could actually be an irritation. This aspect of the study could be further investigated in future work to determine whether performance length can be used to predict levels of engagement, creative flow, and interface satisfaction, and to refine how this measure is collected in other rehearsal-as-research studies. The existing data could
also be analysed with more sophisticated mixed-effects models or a Cox proportional hazards regression model (Andersen & Gill, 1982).

The performers generally used the button relatively sparingly throughout performances; however, the two “button maniacs” among them used the button far more than was anticipated in the design. Once they started this behaviour, they effectively dominated the performance, as the other performers’ interfaces behaved in the same way. While this behaviour was chaotic, it also added a unique sonic aspect to some of the performances and demonstrates the creative exploration at play in the sessions. It remains to be determined whether this unexpected use of the button could be a long-term feature of PhaseRings performances or a novelty effect. Although it should be noted that “button mania” also occurred in two follow-up sessions with the Both condition, a longitudinal study of rehearsals and performances with this interface would be required to understand more about this phenomenon.

In the two follow-up sessions, most of the performers picked the Fade condition as their favourite. However, they suggested in their interviews that both interfaces, Fade and Both, could be interesting and useful in performance situations. While the constant presence of the button allowed the finest direct control over the performance, the agent interaction in the Fade condition encouraged the performers to make the most of their current interface without rushing to new melodic options. Future revisions to PhaseRings’ design could enhance the contrast between these modes of performance. Specific performances could be then created that exploit the benefits of each interface.

Overall, 24 performances were recorded in the two studies encompassing a wide variety of musical explorations (some of which can be seen in the video figure) despite all using the same minimal app. Some groups tended towards the ambient and arrhythmic, while others developed a strong pulse and explored metric ideas and melodic motifs. In the interviews, the musicians told us that they had enjoyed the improvisations, the collaborative interactions that took place, and using the iPad app. This view is confirmed in the survey results which were mostly in the upper half of the rating scale,
even for the Nothing condition. The positive reception to the app, and the wide range of stylistic possibilities the participants discovered, suggests that future artistic performances with any of the conditions could be rewarding. The four interface conditions from the first study have since been used in an educational context in a high-school music class as the stimulus for engaging with different styles of free-improvisation. The ability of PhaseRings to operate under different interaction paradigms while keeping the same simple performance interface makes it very useful for such a setting and future work could examine its educational utility.

7.7 Summary

The results of the two studies in this chapter lead to the conclusion that networked interfaces with direct manipulation of the group performance space via a GUI button, and indirect manipulation via an EDA, give rise to different styles of interaction with touch-screen musical instruments and improve free improvisation in a musical ensemble. This study enabled the identification of precisely which aspects of performance were impacted by the experimental conditions. In the first study, performers rated direct interactions more highly when considering their technical proficiency and the complexity, creativity, and overall quality of performances. However, the indirect agent interaction resulted in significantly longer improvisations, which could indicate a higher level of creative engagement with the app and the ensemble. This is a particularly notable, and surprising, result, as the length of performances was not easily perceived by the performers, but detected with a quantitative measure. The simplest condition, where no changes occurred in the interface, was broadly rejected by the participants while the performances using both button and agent were broadly supported.

The results of the first study led to the design of a mixed-initiative interface for PhaseRings that interlaced the behaviour of the button with the agent. In this new condition, the EDA was able to enable a GUI button at appropriate moments in the performance. Performances with this new con-
7. Comparing Agent-Control with a Networked User-Interface

dition received significantly higher ratings for *structure, group interaction, enjoyment* and *overall quality*, than the simpler Both condition. However, the performers acknowledged that both of these interfaces had the capacity to direct the improvisation by encouraging particular ways of playing.

Overall, it can be concluded that exploiting the networked capabilities of mobile touch-screen devices in apps for musical performance can have significant effects on how these performances are perceived and carried out, and that they can improve group interaction, creativity, and length of engagement in improvisations. The research-as-rehearsal methodology of controlled, quantitative studies of free-improvisation, including measuring session length, has been shown to assist with DMI design leading to a refined interface. This research suggests that further designs for intelligent listener agents, networked user interfaces, and instruments that emphasise particular ensemble interactions could be useful in musical training, recreational music making, and on the concert stage.
Preserving Touch-Screen Improvisations

As an artistic experience, musical performance is fleetingly temporal and the history of music abounds with technologies and traditions to record musical performances in some format to be archived, understood, and developed. Traditions of musical recording, from notation, to the phonograph, to digital audio, have contributed to the ability of performers to create new musical works and to the place of music in the cultural landscape. Western classical music is often defined by the score (Davies, 1991), and popular music by the song or the recorded version (Kania, 2006), but the practice of free improvisation can defy the identification of a canonical musical work. When performing with DMIs, however, musicians have the opportunity to record a musical work in an extremely detailed form by capturing a log of their interactions as well as audio or video recordings. The log itself can serve as a kind of score, but it also affords the creation of other representations of the performance. These can give curators of free-improvisation new perspectives on the musical works, and give improvisers ways to develop and preserve their artistic practice.

In this chapter, a protocol is presented for automatically documenting performances of free-improvised music made on touch-screens. Rather than particular notes and rhythms, the focus of traditional musical notation, the protocol records detailed touch movements in absolute time and abstract percussive touch gestures identified in Section 4.2 (p. 69). The protocol can
serve as an archival format and addresses some of the issues with curating improvised music. The logs also satisfy Manovich’s (2002) principles of new media objects; their numerical representation allows them to be automatically varied and transcoded into new artefacts giving rise to new characterisations, analysis, and appraisal of performances. Algorithmically generated visualisations and gestural scores created from the logs lead to new perspectives on performances and feed back into the process of developing an improvised practice.

The work presented in this chapter has been accepted for publication as “A percussion-focussed approach for preserving touch-screen improvisation” in Curating the Digital: Spaces for Art and Interaction, Springer Series on Cultural Computing (C. Martin & Gardner, 2016).

8.1 Curating the Improvised

Although the field of improvised performance has a developing theoretical background and many highly-regarded practitioners, particular artistic expressions tend toward the ephemeral. In this chapter, an expanded method of documenting improvised music will be described that encodes not just the sounds made in the performance space, but performers’ music-making gestures and ensemble interactions. It is widely recognised that both musical works and new media artefacts can have a number of interacting representations (Rinehart, 2007). Musical works might be directed by a score; might be “thick” or “thin” depending on the freedom of interpretation afforded the performers; be represented in live performance, studio recordings, or by computer generated renderings; and may be composed or improvised (Davies, 2005). Combinations of these representations are often collected together to form an archive of a musical work.

Free-improvised music, where performers do not follow a set musical structure, is usually preserved using only audio and video recordings. While the improvised solos of famous jazz musicians are often transcribed, this is uncommon for free-improvised ensemble performances. Audio recordings of free-improvised performances capture the sonic results of the perform-
ers’ explorations of musical gestures and interactions with other ensemble members but the original performance gestures are lost. For performances with electro-acoustic instruments, audio-recordings can be inadequate for detailed analysis as it is often difficult to discern which musician is creating each sound.

Although it is rare for free-improvisations to be subjected to an internal musical analysis, unlike other forms of music including jazz improvisations, some characterisations of the internal structure of these performances have been published. Pressing (1988) has developed a model of improvisation that divides performances into a series of non-overlapping events separated by trigger points during the performance that initiate each event. Stenström (2009, pp. 286–289) proposed a terminology for free ensemble improvisation, including concepts such as “transitions” between musical ideas and “attractors” such as a steady pulse that encourage similar playing from other performers. Nunn (1998, Chap. 4) similarly argued that a “segmented form” divided by “transitions” is the fundamental structure of free improvisation. When improvised music is performed on touch-screen instruments, a log of touch-interactions captured during the performance can supplement traditional recordings and take the place of a musical score. While scores are generally used for composition, their use as documentation for new media artworks has been acknowledged (MacDonald, 2009). Such a log would also satisfy Manovich’s (2002) principles for a new media artwork. In particular, the log of touch-interactions is variable, forming the basis for derivative artworks that also represent aspects of the original performance.

Borgo (2006) has drawn parallels between the swarm-like collaboration in free-improvised performances and the community collaborations that define the field. This community of practitioners, listeners, and concert organisers are the curators of the free-improvised music world. In this chapter, it is argued that the ability to thoroughly document touch-screen movements of improvising ensembles enhances the ability of such a community to develop this artistic practice through archiving, replaying, and re-synthesising performances. Sections 8.2 and 8.3 will describe an approach to recording improvised touch-screen performances and transforming such transcriptions.
8. Preserving Touch-Screen Improvisations

into animations, gestural scores, and new artworks. These multiple representations of musical performance allow the curation and analysis of an emerging improvised practice.

8.2 Towards a protocol for touch-screen musical performance

Musical data has been abstracted from the temporality of performance for centuries since the development of musical notation. Mechanical instruments such as music boxes and barrel organs, developed in the 18th century (Fowler, 1967), first allowed music to be programmed rather than performed. More refined mechanical instruments such as the reproducing piano, which appeared at the turn of the 20th century (Kapur, 2005), allowed performances to be automatically transcribed and converted into a paper piano roll that could be replayed many times. All of these technologies have had an impact in the study of music as well as its performance. Musical notation enabled the field of musicology, where the musical score has traditionally been privileged as the canonical representation of a musical work. The piano-roll performances of many famous pianists were made before audio recording was widespread and have been used to analyse their performance styles.

An important antecedent of the present research was the MIDI protocol (MIDI Manufacturers Association, 1996), developed in the late 1970s to connect electronic music controllers, such as electronic versions of keyboards, drums, and wind instruments, as well as new musical interfaces such as THE HANDS (Waisvisz, 1985) with electronic syntheisers or digital recording systems. While this standard was intended as a control interface for live performance or recording, it was subverted by digital artists and researchers who recognised that the MIDI trace of a musical performance could be used for other purposes. MIDI was originally designed to be used with a physical serial connection; however, virtual MIDI connections are commonly used to connect multiple pieces of software and to other computers over a network (Lazzaro & Wawrzynek, 2004).

While the success of MIDI is ongoing, the semantics of the protocol is
mostly restricted to a keyboard-and-note perspective on musical data. The
typical MIDI interactions are note on and note off messages, each of which
contain a pitch and dynamic (volume) value. Changing parameters while a
note is playing can be achieved by simultaneously sending one of a limited
number of continuous control messages while the note is held. In an effort
to develop a semantics-free format for musical control that better reflected
the needs of modern computer music systems, OSC (Freed & Schmeder,
2009) was developed. This standard defines a message format but with
the specific content of the messages up to the application developer. The
flexibility of OSC has contributed to its success not just in computer music,
but in professional applications such as show control (Schmeder et al., 2010),
although it is not commonly used in commercial electronic instruments.

Some have attempted to define protocols using OSC to standardise inter-
action with certain types of interface. TUIO is one such protocol designed
for tabletop interfaces where fiducial markers and finger touches can be
tracked (Kaltenbrunner et al., 2005). Unlike MIDI, TUIO does not define
the purpose of messages but communicates only information about basic
components that the designers expected would be common to most table-
top interfaces. The TUIO protocol sends groups of messages together that
encompass the state of the whole tabletop interface. Most importantly, one
set message is sent for each object on the surface that has changed position.
A set message includes identification and position information about the
object being tracked as well as pre-calculated data such as velocity, accel-
eration, and rotation. These data simplify the requirements for software
receiving and interpreting TUIO.

As described in Section 8.2.1 below, the protocol for logging touch-screen
performances in this thesis needed to capture the fundamental interactions
occurring on the touch-screen, not how these interactions are interpreted
by the application currently running on the device. In Apple iOS devices,
data collected from the multi-touch digitiser in front of the screen is inter-
preted by the operating system which keeps track of individual touches and
divides the incoming data into events (Apple, 2015). Software developers
can implement a set of callback functions to access these events individ-
8. Preserving Touch-Screen Improvisations

- (void)touchesBegan:(NSSet *)touches
  withEvent:(UIEvent *)event;
- (void)touchesMoved:(NSSet *)touches
  withEvent:(UIEvent *)event;
- (void)touchesEnded:(NSSet *)touches
  withEvent:(UIEvent *)event;
- (void)touchesCancelled:(NSSet *)touches
  withEvent:(UIEvent *)event;

Figure 8.1: Callback methods for accessing touch events in Apple iOS (Apple, 2015). The apps in this thesis log each touchesBegan, touchesMoved, and touchesEnded event.

ually. So-called UIEvents track the state of touches on the screen - they “begin”, “move”, “end”, and may be “cancelled” if they are misrecognised (Figure 8.1). For the purposes of designing software for free-form touch improvisation, only the first three states are of interest. The touch-data objects described by these events have a record of their current as well as previous location on the screen. A value proportional to instantaneous velocity of moving touches can be easily calculated by finding the length of the vector from the previous location to the new.

8.2.1 The Metatone Log Protocol

Section 3.4 (p. 44) described how, starting with Ensemble Metatone’s earliest rehearsals, a protocol was developed to capture touch-screen information from each performer’s iPad which was sent to a central server using the OSC data format. As further apps and the Metatone Classifier EDA were developed, this protocol was extended to document the performers’ gestural and ensemble states and app-to-app interactions sent between iPads. A complete listing of the OSC messaging scheme is given in Table 8.1. When the Metatone Classifier receives one of these OSC messages, it assigns a timestamp and records it to a text file for later analysis. Each line of the text file is written in the CSV (comma separated values) format. Although the
8.2. Towards a protocol for touch-screen musical performance

<table>
<thead>
<tr>
<th>App to Server Messages:</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>/metatone/online</td>
<td>device</td>
</tr>
<tr>
<td>/metatone/touch</td>
<td>device</td>
</tr>
<tr>
<td>/metatone/touch/ended</td>
<td>device, X, Y, velocity</td>
</tr>
<tr>
<td>/metatone/switch</td>
<td>device</td>
</tr>
<tr>
<td>/metatone/app</td>
<td>device, name, position</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Server to App Messages:</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>/metatone/classifier/gesture</td>
<td>device, gesture type</td>
</tr>
<tr>
<td>/metatone/classifier/ensemble/event/new_idea</td>
<td>device, measure value</td>
</tr>
<tr>
<td>/metatone/classifier/ensemble/state</td>
<td>type, value 1, value 2</td>
</tr>
</tbody>
</table>

Table 8.1: Scheme for OSC messages from the Metatone iPad apps. The touch and touch ended messages record touch screen interactions directly from the iOS operating system (see Figure 8.1). The “device” parameters are unique identifiers for each iPad in the ensemble.

different aspects of the performance are recorded using different numbers of parameters, these can be trivially separated or reorganised by filtering through the unique OSC address of each type of message.

The iPad apps send messages to the server in response to three of the four touch-events in iOS, touchesBegan, touchesMoved, and touchesEnded. Both the beginning and movements of touches are recorded using a /touch message recording the iPad’s app-level unique device ID. The velocity of touchesMoved messages is recorded while touchesBegan messages are distinguished by having a velocity of zero. A /touch/ended message is used to record touchesEnded events. touchesCancelled messages are ignored.

As discussed in sections 3.2.1 (p. 39), 3.3 (p. 41), and 4.3 (p. 81), some of the iPad apps featured a small number of button and switch UI elements that were used in performances to activate looping functions and algorithmically-generated backing sounds, and to change the timbre and pitch of sounds available through the touch interface. The performers’ interactions with these elements are recorded using /switch messages which
record the iPad device ID, the name of the UI element and its new state. During performances, the iPad apps send messages to the other apps performing over LPN connections. These messages are also copied to and logged by Metatone Classifier using the /app OSC address.

Metatone Classifier also sends information back to the performers’ iPads during performances as described in Section 4.2 (p. 69). The protocol for performance logging includes three types of OSC messages sent from the server to the iPad apps: gesture classifications are returned to the iPads each second with /gesture messages; whenever a new musical section is detected, the server sends an /event/new_idea message to all connected iPads; and other measures of the ensemble state are returned to the iPads using the OSC address, /state.

The scheme for logging touch-interactions (see Table 8.1) was chosen to study the process of improvising with iPad instruments and not necessarily for replaying performances. Other aspects of the touch-screen state, such as unique identifiers for each touch point, are not tracked, nor are the exact pitches available on screen for each player. Multi-tracked audio recordings of performance are deemed to be sufficient record of the particular sounds created during the performance while the touch protocols store the details of performers’ interaction with the instruments — something the audio recording cannot achieve. While the protocols were created for research purposes, the CSV storage format allows these logs to be easily transformed into alternative representations of performances. The next section will describe these new outputs, and how they not only aid in understanding the improvised performances, but serve as representative artefacts along with audio and video recordings.

8.3 Transcoding Performance Protocols

The touch-screen performance protocols use a CSV format to record all touch events, UI interactions, and app-to-app communications that occur during improvised performances. These recordings are mutable new-media objects and, as suggested by Manovich (2002), by transcoding these objects
8.3. Transcoding Performance Protocols

“the logic of a computer can be expected to significantly influence the traditional cultural logic of media” (pp. 63–64). This section is concerned with ways in which these performance protocols can be transcoded into new representations of the improvisations, both in real-time, during a performance, and afterwards. Gestural classifications of each performer’s touches lead to graphical scores and animated visualisations provide new perspectives on the ensemble interaction. These representations not only form important archival documents of performance but feed into the ongoing artistic practice of touch-screen improvisation.

8.3.1 Gestural Scores

As described in Section 4.2.1 (p. 70), each second during performances, Metatone Classifier analyses the previous five seconds of touch-data from each connected iPad and identifies the performers’ current touch gesture. Applying this gestural classification scheme to touch-screen performances results in a new representation of the performance, that is, a time series of gesture classes at one-second intervals for each performer in the ensemble. Although this time series does not contain the details of each performer’s interactions, it can still serve as a kind of musical score for performances, albeit a non-traditional one.

When a graphical plot is created of such a time series, the score starts to bear resemblance to the time-space graphical scores of contemporary classical music. In the gesture-score plots of Figure 8.2, each performer’s gestures are represented by a different line. These gesture-scores reveal much about the structure of improvisations that is difficult to appreciate from temporal representations like audio and video recordings. In Figure 8.2, it is clear when all performers are focussed on the same broad gesture groups as their lines are on close levels of the plot. When one performer breaks out for a solo idea, their line moves up or down, or might move to the Nothing (n) level if they have stopped playing. Sometimes the performers split into sub-ensembles, exploring separate groups of gestures. As was argued in Section 4.2.4 (p. 78), moments of increased gestural change appear to segment
the improvisations into musical ideas.

Since these gestural scores are so helpful in understanding, at a glance, touch-screen improvisations, they serve as more useful archival documents of performances than, for example, still photographs of the stage setup. Rather than recording the location and setting of the performers, gesture-scores store a high-level view of the performers’ actions throughout a whole performance. Gesture-scores could also be used to create new musical performances. An ensemble of touch-screen musicians could simply read a gesture-score, as they would a graphical score, and play through the sequence of gestures. Alternatively, a computerised score display with a synchronised cursor could be used similar to Hope and Vickery’s (2015) Score-Player system. While such performances would probably not retain the sonic characteristics of the source improvisation, they would include similar musical interactions between the members of the ensemble and identical moments of structural change. By selectively recording and regenerating multiple versions of a transcribed gesture score, a touch-screen ensemble could curate a repertoire of works from improvised source material. This process has precedent in contemporary classical music; the composer Stuart S. Smith recalls allowing “muscle memory to create the initial gesture while letting my mind/ear polish the gesture, refining cliches out of the picture” (S. S. Smith & Goldstein, 1998, p. 190).

8.3.2 Visualisations

Visual representations of DMI interactions have been previously used to gain new insights into performance practices (Brown, 2010). To understand the structure of touch-screen performances in this research, a similar method has been developed for translating touch and gesture protocols into animated visual representations of the performance that can accompany audio and video recordings. Two animations are typically produced for each research-focused performance using the Processing programming language (Reas & Fry, 2006). The first is an animated version of the gesture-scores discussed in Section 8.3.1, with an integrated play-head that indicates the
Figure 8.2: Gesture scores for two improvisations: Ensemble Metatone performing with Snow Music on 2014-08-24 (left), and a quartet of research participants with PhaseRings on 2015-05-07 (right). Each line represents on player’s gestural classifications (See §4.2.1, p. 70); vertical lines represent moments when a new-idea message was triggered by the Metatone Classifier EDA. The x-axis shows time in 24-hour format (hh:mm).
current position in the performance.

The second animation visualises the logs of performers’ touches over each performance, a refined version of the visualisation described in Section 3.4 (p. 44). This program reads a captured log file and renders an animation of all four players’ touches overlaid in the space of one iPad screen with the different players distinguished by colour. Each touch point is represented by a circle that fades away allowing different types of swirl, swipe, and tap gestures to be distinguished at a glance. The software also draws a date and time-stamp in each frame as well as text notification of app and switch messages. This touch animation presents an entirely new view of the performance which was not visible to the performers or audience on the day. As all the touch movements are layered in one performance area it is immediately clear when performers mimic each other, form sections, or experiment with a new musical idea.

While the two visualisations can be viewed separately, they are most useful when synchronised with the audio and video recordings of performances and displayed together. Figure 8.3 shows stills from hybrid visualisations of three different performances. Each shows an alternative arrangement of visualisations from performance protocols and multiple camera-angles of video. While the video component of these hybrid visualisations preserve the context of the concert and the stage interactions and communications of the performers, the visualised touches and gestures have proven to be much more useful from a research standpoint. With the gesture-score, performance interactions can be seen across time, and with touch visualisation, gestures can be seen across the ensemble. In this way, the hybrid visualisation becomes a kind of “Seeing Space” for improvised touch performances where multiple dimensions of the performance can be examined simultaneously (Victor, 2014).

It is possible that the visualisations described here could also be used in a live performance context where they could be projected onto large screen or superimposed onto the performers’ touch-screens. A prototype touch visualisation has been developed for Metatone Classifier and has been used as a backdrop projection in some performances, but the implications and
Figure 8.3: Stills from hybrid visualisations of three Ensemble Metatone performances showing video, touch animations, and gesture-scores. From top to bottom, *MetaLonsdale* in Canberra, August 2013, *Study in Bowls* in Canberra, March 2014, and *Touch and Tone* in Boston, May 2014.
affordances of this addition to the performance have not yet been fully explored.

8.4 Summary

A protocol for logging free-improvised touch-screen musical performances has been presented in this chapter. This has been implemented in Metatone Classifier and has been used to record more than 150 collaborative interaction sessions by Ensemble Metatone and other groups so far\(^1\). This protocol records all touch interactions during each performance as well as network interactions between the performers’ apps and the server software. The protocol is a much more appropriate record of touch-screen performances than conventional systems, such as MIDI, because each touch-interaction maps directly to sound, and because network interactions are key to the ensemble improvisations in this research. The archive of performance protocols encodes aspects of the improvisations that are not accessible in traditional recordings and that align with theoretical models of free-improvised music. The performance logs in this archive are also new-media objects satisfying Manovich’s (2002) principles for new media and can be transcoded into multiple alternative representations of the performance. Two such representations were presented in this chapter: graphical gesture-scores, and animations that afford the viewer a new performer-centric perspective on the touch improvisations. These visualisations have been used in other chapters to help understand the nature of touch-screen improvisation, and the gesture-scores could even be used as the stimulus for future performances.

These logs have enabled quantitative analysis of interactions between apps and agents as discussed in chapters 5 and 7, and the duration of the performances in Chapter 7. Now that logs have been collected from a very large number of collaborative interactions, it is natural to ask whether these data can support further quantitative and qualitative insights into the nature of these touch-screen performances. Future investigations could

\(^1\)The archive of interaction logs is available online at http://dx.doi.org/10.5281/zenodo.51710
apply the transition analysis techniques used in Metatone Classifier to these records and may be able to differentiate numerically between different styles of playing. It may also be that new artworks can be created by re-synthesising or remixing records from the archive.
Conclusions

In this thesis, a system of touch-screen DMIs and Ensemble Director Agents (EDAs) have been described that have been designed to enable and enhance collaborative musical performances, particularly those that are free-improvised. The creation of these DMIs has been motivated by a desire to implement a system that allows the performers to retain freedom over low-level notes, rhythms, and sounds, and to assist them with high-level structural decisions, such as synchronously progressing through a harmonic sequence and altering the timbre of sounds. While this kind of interaction has been theorised before, analysis of the literature has revealed that most examples of network-assisted performance have focussed on simple interactions, such as networked metronome signals. The systems presented in this work have taken two broad approaches: direct interactions with the DMI interface that are communicated to all instruments in the ensemble via a Local Performance Network (LPN), and indirect interactions with an Ensemble Director Agent (EDA). The agent system listens to the performance by tracking touch-data from all performers simultaneously, classifying it according to a novel vocabulary of musical touch gestures, processing this data using transition matrices, and responding with messages at the start of new segments that it identifies in the ongoing performance. As it tracks low-level touch data, this system can be used with multiple apps, each of which encode different kinds of reactions to agent responses in their own interface.
9. Conclusions

Successive generations of these apps and agents have been investigated in a practice-based methodology mixing HCI style analysis with free-improvisation and rehearsal processes. A percussion/iPad quartet, Ensemble Metatone, was formed specifically to develop a performance practice with early apps, and this process yielded a high-level characterisation of touch-gestures and iPad improvisation. This group later participated in a formal study of three apps and two agents as well as many public performances that resulted in two recorded releases. The formal study revealed that one interface, Singing Bowls, captured the performers’ creativity with particular success. This interface was refined as PhaseRings and further investigated with new performers from a variety of backgrounds. An experimental PhaseRings interface was developed to specifically study the contrast between direct networked interaction, and indirect agent interaction with four ensembles of four performers each. A follow-up study with a mixed-initiative interface out-performed the simpler interface on several aspects of the performance as rated by the participants. In the next two sections, the designs of the apps and agents will be briefly summarised before reflecting on the research questions set out in Chapter 1.

9.1 Iterating App Designs

Four generations of app design have been described in this thesis. These designs, shown in Figure 9.1, range from the prototype apps developed for early rehearsals with Ensemble Metatone, to the second generation of PhaseRings designed in response to the formal user study.

MetaTravels was developed for Ensemble Metatone’s earliest rehearsals and was designed to enable performances with iPads as the primary instrument. This app also introduced networked logging functionality where touch data from the performance was recorded on a server. MetaLonsdale broadly improved the musical interactions in MetaTravels, introducing a sequence of scales to provide harmonic motion in performances. The data-sharing functionality in MetaLonsdale, accomplished using an ad-hoc LPN, evoked a range of responses from performers who noted that it resulted in
9.1. Iterating App Designs

Figure 9.1: The four generations of app design discussed in this thesis. Each generation was subjected to a formal study of improvised performances that informed later design iterations.

performances that were more cohesive, but that it reduced their individual control.

Three apps, *BirdsNest*, *Snow Music*, and *Singing Bowls*, were developed specifically to interact with the Metatone Classifier EDA. Of these apps, *BirdsNest* was most directly related to MetaLonsdale in the previous generation, *Snow Music* was also a revision of a previous app, but *Singing Bowls* was completely new. This generation of apps was marked by a wider range of interface and interaction ideas. Initially these were developed to afford varied performances with multiple apps producing different kinds of improvisation, but the differing designs also suggested the formal study discussed in Chapter 5.

*PhaseRings* was designed to integrate the best aspects of *Singing Bowls* into a more configurable app that was suitable for wider distribution. GUI controls were added to control the synthesis and harmonic series parameters.
9. Conclusions

An LPN system similar to MetaLonsdale was also implemented to allow cohesive ensemble performances without the Metatone Classifier EDA. This app was made available for free on the iTunes App Store and also introduced to a wider variety of musicians through improvisation sessions and a formal study comparing networked ensemble interactions in the app.

A final version of PhaseRings was developed after the user study that combined the EDA new-idea reward interaction with the LPN button interface. This mixed-initiative interface combined agent suggestions with direct user interactions in a way that had not been previously explored. The follow-up study supported the utility of this new design.

Figure 9.2: Three generations of EDA designs were used in this thesis. The earliest simply logged performance data, while the first version of Metatone Classifier interacted with performers via gesture classifications and new-idea messages. The final iteration could configure apps for different types of performances and could run on a remote web-server.
9.2 Ensemble Director Agents

A theme of this thesis has been to design and implement EDA systems that seamlessly track and interact with an ensemble of touch-screen performers, providing a positive effect on their improvisation through gentle and appropriate direction. The Metatone Classifier EDA has been described throughout this work. The development of this agent software occurred in three distinct stages as shown in Figure 9.2.

In the first step, a recorder which would log touch-data messages in early Ensemble Metatone rehearsals was created. The first iteration of this recorder was a script written in the SuperCollider computer music language. This was soon replaced by a Mac OS X application, *OSC-Logger*. The performances tracked by these programs provided an early corpus of data for developing the gesture classifier and transition matrix systems.

*Metatone Classifier* was built from the ground up in Python as a true EDA which would both record and interact with a touch-screen ensemble through gestural classifications and new-idea calculations. A *generative* version of this EDA was also developed that provided similar messages to the touch-screen apps but sourced from a statistical model rather than the touch-screen data. While this agent was generally operated from a command-line interface, a Mac OS X interface was also developed to monitor performances more easily.

The final version of Metatone Classifier was used in the formal study of PhaseRings and later performances. This version allowed the PhaseRings app to be configured remotely via performances type messages and included many refinements to the ensemble tracking calculations for increased reliability. The communication system with the apps was also extensively revised to communicate over web-sockets rather than with UDP messages. This resulted in increased reliability over WiFi networks as well as the ability to run the EDA on a remote web-server. While the capacity for connecting an ensemble of remote performers was not explored in this thesis, it remains an exciting direction for further study.
9. Conclusions

9.3 Reflecting on the Research Questions

The research questions set out in Section 1.4 (p. 7) framed the touch-screen DMI investigation in this thesis in terms of ensembles, rather than individual performers. As such, the DMI and EDA software summarised in the previous section were all designed for use by groups of musicians improvising together. To evaluate the effect of these new instruments on performers, a research methodology that follows rehearsal and improvisation training techniques was employed. These processes were refined over the successive formal studies in this thesis, resulting in the rehearsal-as-research HCI methodology described in Section 5.1 (p. 99). The findings of these studies will now be summarised with reference to the research questions.

9.3.1 How might touch-screen instruments be used in ensemble improvisations by percussionists?

Chapter 3 described Ensemble Metatone’s earliest encounters with two touch-screen DMIs. As evidenced by a series of public performances and recordings, this group was able to pursue a successful creative practice of improvisation with these instruments. The group’s early rehearsals and discussions were subjected to qualitative analysis in Section 3.6 (p. 54) to characterise their performances in terms of a vocabulary of percussive touch gestures. Unlike command touch-gestures commonly seen in HCI, these expressive gestures were continuous. Although it was acknowledged that the DMIs were more difficult to control than acoustic instruments, the performers developed these touch-gestures to overcome this limitation. Once the performers had established a range of techniques, they felt their improvisations became more successful.

This characterisation of a set of touch-screen music techniques became central to later investigations of performances. A distilled vocabulary of gestures was developed in Section 4.2.1 (p. 70) that could be automatically identified from logs of touch-screen interaction data. Gestural-scores of performances could now be produced that enabled intuitive analysis of ensemble interactions in performances at a glance. The ability to visually
9.3. Reflecting on the Research Questions

inspect and compare improvised performances by touch gesture allowed an unprecedented level of insight into these free-improvisations.

From these gesture-scores and from performance experience, it became evident that touch-screen free-improvisations might be segmented into sequences of musical ideas. Detecting such new ideas requires consideration of the whole ensemble’s musical trajectory simultaneously, and in Section 4.2.2 (p. 73), a method for analysing transitions between gestures over the ensemble was described. The novel flux measure was formulated to discriminate between the transition matrices of different sections of performance. Comparing the flux of adjacent performance sections automatically revealed some of the segmented structures of performance, as reported in Section 4.2.4 (p. 78).

The work in this thesis has investigated touch-screen performance at three different levels: the individual performers’ micro-gestures resulting in the characterisation of touch-screen performance; performers’ gestural trajectories revealed by the gesture classification and gesture-scores; and the overall ensemble structure of performances as measured by transition matrices and flux. Just as the investigations have ranged from the micro- to macro-structure of performances, they have also been conducted from a qualitative standpoint that considered performers’ discussions and feedback, and a quantitative one that interrogated the touch-data that constitutes the most objective record of these performances. In this thesis, these contributions have formed an important basis for developing and examining the ensemble-focussed DMI s that are the subject of the following two questions.

9.3.2 How can a system of apps and agents be built that reacts to, and directs, ensemble improvisations?

The iPad-based DMI s described in this thesis were all designed primarily for ensemble rather than solo performance. All except the earliest (Meta-Travels) featured mechanisms such that each performer’s interface reflected the actions of other performers in some way. In Chapter 2, two approaches to ensemble DMI design were identified in the established literature: Local
9. Conclusions

Performance Networks (LPNs), where DMIs connect to each other to share information during performances, and Ensemble Director Agents (EDAs), where a software agent tracks the ensemble and sends messages to direct them in some way. MetaLonsdale used an LPN approach to partially sync interactions across the ensemble and reflected earlier work in this direction. The Metatone Classifier software, by contrast, was a novel implementation of an EDA, a concept that had been theorised but had received little practical attention. Metatone Classifier continually tracked the ensemble performance using gesture classifications, transition matrices, and flux calculations to identify new-idea moments. This information was returned to the performers’ apps that were then responsible for responding in some way to the unfolding group interaction.

Following the early development of Metatone Classifier, various app designs were implemented to interact with it and, accordingly, to address the present research question. The broadest experimentation occurred in the repertoire of three iPad apps that were introduced in Section 4.3 (p. 81). Three different schemes for EDA interaction were defined with disruptive, supportive, and rewarding paradigms, respectively. The networked designs of these apps were further characterised in Chapter 6, where each was described by the network architecture used (EDA or LPN), the source of an ensemble-based trigger for interface response (user-interface, new-ideas, or gesture-runs), and the musical interface elements that were affected (app features, tonality, sound palette, or secondary sounds).

Also presented in Chapter 6 was the PhaseRings app, which was a revision of the highly successful Singing Bowls. Where Singing Bowls was entirely focussed on Metatone Classifier interactions, PhaseRings was open-ended with LPN features similar to MetaLonsdale and with a menu system for solo users. This app is unique in having the possibility of both EDA and LPN interactions affecting same aspects of the interface. The two triggers, GUI interactions and agent new-idea messages, were recontextualised in Chapter 7 in terms of direct and indirect interactions. The user-study informed another iteration of PhaseRings in which both of these interactions were interlaced in a mixed-initiative interface; agent recommendations
enabled a GUI button that users could tap — or not tap — as they wished. Although this idea was not pursued further than the follow-up study, it appears that mixed-initiative interfaces of this complexity could be successful targets for future investigations.

The spectrum of potential ensemble-facing DMI designs seems to be very broad indeed. Not all of the designs presented have proved to be creatively fruitful, and some, like the disruptive BirdsNest interface, were rejected by performers. The performers acknowledged, however, that interruptions in the interface sometimes provided a stimulus for pursuing new ideas. It could be that app designers can “compose” different ensemble interactions with the interactions in touch-screen apps. In an era when touch-screen DMIs appear to be proliferating in educational, recreational, and professional music making, designing ensemble interactions into these apps may become increasingly important.

9.3.3 How does this system of apps and agents affect ensemble improvisation?

The results of three formal studies concerning ensemble-focussed DMIs have been reported in this thesis. These studies have been designed to mirror natural rehearsal and improvisation processes that would not only be comfortable for the participants, but would leverage their experience and curiosity as creative professionals. Starting with the study reported in Chapter 5, quantitative data was collected from surveys that address different aspects of improvisation, as well as the objective data of performance protocols. These data allow conclusions to be drawn about precisely which aspects of performance are affected by the apps, agents, and interfaces under investigation.

The formal study of Ensemble Metatone reported in Chapter 5 compared three app designs interacting with a performance-tracking EDA, and a generative EDA. The apps featured differing musical designs, as well as different paradigms for responding to agent signals. The study was focussed around the performers’ perceptions of each improvisation, which were recorded in
written surveys with seven questions. The different apps were found to have had a significant effect on the perception of performance quality, creativity, how users were able to respond to interface changes, and the apps’ influence on personal performance. The main effect of the agent was only found to have borderline significance on how well users could respond to interface changes and the app’s influence on the group performance, but this latter question also showed a significant interaction effect due to both app and agent. This study found that the Singing Bowls interface and the interaction paradigm of reward was most positively perceived and justified further investigation of this interface in later chapters. The effect of the agents was not found to be statistically significant even though the classifying agent appeared be connected with higher ratings (see Figure 5.6 (p. 112)). The interface had a much stronger effect on the results. While this study revealed some of the effects of the networked DMIs, the small sample size limited the generalisability of the results and the six conditions made interpretation difficult. Later studies sought to revise the experimental design to identify findings with more clarity.

A wider range of effects was identified in the later study reported in Chapter 7 using the PhaseRings app. In this study, the two triggers for interface updates in PhaseRings, LPN-connected GUI interactions and EDA new-idea messages, were compared and crossed in a two-way, rehearsal-as-research design. The sixteen participants who took part represented a much wider range of musical experience than had previously been encountered in this research. As in the previous study, surveys were administered after each improvisation; however, these revised surveys contained 24 questions covering technical proficiency, musical interaction, musical structure, creativity, and performance quality. The survey results show significantly higher ratings due to the GUI interface in nine of the questions, including technical proficiency, performance complexity, and personal contribution. This interface clearly sparked creative investigation as shown by the “button maniac” behaviour exhibited in some groups. The EDA-controlled interface was found to have a significant effect on the duration of improvisations. Since the performers were generally unaware of the duration of their
improvisations, this objective measure could be related to their engagement and creative flow.

The revised, mixed-initiative “fade” interface was developed for PhaseR-nings in response to the results, and directly compared with simultaneous EDA and LPN interactions in a follow-up study. The new design was found to outperform the simpler interface in six questions, including performance structure, group interaction, enjoyment, and overall quality. Performance duration did not show a significant effect for the new interface in this experiment. In these two studies, the networked interfaces were shown to enhance improvisation, and the findings revealed the specific aspects of performances that were supported by each system. The introduction of performance duration as a dependent variable in this study revealed the impact of the agent which had previously been elusive. The findings confirmed that the contrasting designs for networked ensemble interaction can nudge performers into differing styles of improvisation. The study offers some proof that although direct GUI controls make the performers feel in control, indirect EDA interactions can actually hold the performers’ attention for longer.

It should be noted that although some experimental conditions appeared to perform better than others, all of the performances recorded in study sessions appeared to be of high quality. The participants frequently remarked on their enjoyment of the experience of improvising under different conditions and the experimenters found that the performances were eminently listenable. This bodes well for an underlying theme of this thesis: that ensemble performance with touch-screen apps is a valid and artistically rewarding pursuit. The formal studies have provided evidence that supporting ensemble interactions with sensitively designed network connections between interfaces really can enhance the performance, and the performers’ experience.

9.4 Limitations and Future Work

This thesis has focussed on creating and evaluating a system of apps and agents for ensemble performances, however, other approaches for designing
9. Conclusions

such a system are also possible. This research has emphasised percussive interactions with touch-screen devices but other vocabularies of touch-gesture interaction inspired by other instrumentalists and even other percussionists could also be discovered. Metatone Classifier could be augmented to understand multiple vocabularies of touch-gesture or variations on the vocabulary identified with Ensemble Metatone. It may even be possible to handle a different vocabulary for each individual in an ensemble.

Similarly, Metatone Classifier is not necessarily limited to analysing gestures on touch-screens. The same techniques could be applied to improvised performance with a variety of interfaces and types of sensor. The system could be extended with multiple classifiers that interpret input from other interfaces. It seems likely that other measures of transition matrices that complement or extend the flux measure could reveal other aspects of improvised interaction that have not been examined in this work. This research has focussed on one approach for applying machine learning to a musical ensemble, however this could be expanded in future to measure different kinds of ensemble performances. These could include composed works, ensembles with acoustic instruments, and even other art-forms or multimedia performances.

All the performances in this research took place with participants collocated in space and time, however, there is significant potential for an EDA to be used to connect remote musicians. Whether such a system can provide an enhanced sense of ensemble performance in such situations is an open question for future investigation. Such an EDA could also be trained on the signals it detects during an ensemble performance and switch to a generative mode for later individual practice. In this way, performers could overcome barriers of distance and time to replicate live ensemble improvisations.

The studies in this thesis have been designed according to a rehearsal-as-research methodology. While this methodology has been refined over the three studies, there is further scope for it to be developed and expanded. Other styles of rehearsal, including for non-improvised works, could be incorporated into the methodology, and it could be used with other musi-
9.5 Final Remarks

In this thesis, new network-connected DMIs have been developed to support and extend the creative possibilities of ensemble performance. Performances with these new instruments have been studied and characterised leading to an understanding of touch-screen improvisation from low-level touch-data up to ensemble interactions. A range of approaches to ensemble interaction were implemented in the six touch-screen apps described in this research, leading to a design framework for network interactions in touch-screen DMIs. Most importantly, however, these apps have been deployed in an ongoing artistic practice. The process began with Ensemble Metatone’s activities, where prototype apps were refined through studio rehearsals and public performances. Later apps were introduced to a much wider range of users in educational situations such as the ANU New Music Ensemble, public installations, workshops such as at the Electrofringe festival (see Figure 9.3), and through public release on Apple’s iTunes App Store. A desire to study the effects of these DMIs led to the development of a rehearsal-
as-research HCI methodology which was applied in a study of Ensemble Metatone, and to four ensembles of new iPad musicians. The findings of these studies helped to make the effects of these instruments precise and offer evidence of the creative enhancement that is possible when DMIs are designed to interact with not just the performer, but the entire ensemble.

This work has touched on a variety of motivating fields: computer music, improvised performance, and percussion. Navigating through these fields, this research has led to the establishment of a new artistic practice, the design of a broader variety of tools, and the expanded use of revised apps by new performers. Although this story has been self-contained, all of its motivating fields are relatively young. Many other networked DMI designs may be possible with effects on ensemble improvisations that are yet to be
discovered. Ensemble free-improvisation with touch-screen instruments is a form of music-making that is marked by its accessibility to new performers and also by its artistic potential in the hands of experienced musicians. It seems very likely that this area will continue to provide much research material for the fields of computer music and HCI, as well as rewarding artistic experiences for performers and audiences alike.
Appendices
A.1 MetaLonsdale

MetaLonsdale is a collaborative musical instrument originally created for a performance at Everything Nothing Projects at Lonsdale St Traders in Canberra, Australia. The app provides a free-form touch area that can be tapped and swiped to perform field recordings from this venue as well as an array of percussive sounds. GUI switches control a looping function and
A. Software Releases

Figure A.1: MetaLonsdale running on an iPad Air. A demo of this app is available at: http://dx.doi.org/10.5281/zenodo.51818

an autoplay feature that triggers clusters of quiet sounds. The sounds and tonality of the instrument can be shuffled by pressing the “sounds” button.

The app was originally created as a revision of MetaTravels, the prototype app used in Ensemble Metatone’s early rehearsals. Improvements in MetaLonsdale included a refined user interface and a sequence of predefined scales for pitched sounds that provided a sense of harmonic motion in performances. A network feature enables multiple devices running MetaLonsdale on the same WiFi network to find each other automatically and to control each other’s tonality, looping, and autoplay functions.

Following the premiere performance, MetaLonsdale was integrated into Ensemble Metatone’s performance and research practice in 2013 and the app was subsequently released on the iTunes App Store.

A.2 BirdsNest

BirdsNest is an iPad-instrument recalling the forest sounds of a Northern Swedish summer. Performers can tap and swipe in the free-form touch area
to make music with bird calls, percussion sounds, and field recordings. GUI switches control autoplay and looping features that add layers of sounds to the user’s improvisation.

BirdsNest was originally developed for performances by Ensemble Evolution in November 2013 in Boston and Indianapolis, USA. In this context, the app was used alongside a collection of “found sounds”, branches, toys, and other unconventional instruments, in an ensemble improvisation.

The app was revised to interact with Metatone Classifier in 2014 for a performances and formal studies with Ensemble Metatone. The app was first released in the iTunes App Store in early 2014, and a second version was released including design revisions informed by Ensemble Metatone’s activities in October 2014.

A.3 Snow Music

A prototype version of Snow Music was released on the iTunes App Store in 2012 and had been used in research published prior to the present thesis.
A. Software Releases

Figure A.3: The latest version of Snow Music running on an iPad Air. A video demo of Snow Music is available at: http://dx.doi.org/10.5281/zenodo.51820

(C. Martin, 2012a, 2012b). Following the development of MetaLonsdale and BirdsNest, Snow Music was extensively revised in 2014 to support performances where it was used by ensembles as their only instrument and to interact with the Metatone Classifier agent. This interaction was evaluated in the study detailed in Chapter 5.

Snow Music allows performers to interact percussively with field recordings of snow sounds collected in Piteå, Sweden in February 2012. The app also features three backing soundscapes that can be controlled by the switches in the corners of the screen. The 2014 revision refined the musical interaction and synthesis systems to allow more expressive performance with a wider range of snow and ice sounds. Visual feedback was also added for the soundscapes, that is, fading blue and white circles that indicate when the generators had played a note.
A.4 PhaseRings

PhaseRings (Figure A.4) is an annular interface for performing expressive music with touch gestures. Each ring on the screen represents a different pitch; these rings can be tapped and swirled to create combinations of long and short notes. The angle and size of taps changes the timbre and volume of these notes. While this interface originated with the Singing Bowls app, PhaseRings introduced a generative composition system, seven expressive sound schemes, and a system of menus for user configuration. The sound schemes included percussive sampled sounds, such as marimba and singing bowls, as well as phase and string synthesis sounds.

Three compositions of pitches are included in the app, but the user can define their own custom composition by choosing a sequence of root pitches and scales. This flexibility has allowed PhaseRings to be used by musicians to add sounds to their own musical works. PhaseRings is compatible with the AudioBus and inter-app audio systems for making internal audio and MIDI connections on iOS devices and can thus be used as a sound source.

Figure A.4: PhaseRings running on an iPad Air. A video demo of PhaseRings is available at: http://dx.doi.org/10.5281/zenodo.51822
A. Software Releases

Figure A.5: Metatone Classifier running on Apple OS X.

for other apps that record or process audio.

PhaseRings was released on the iTunes App Store in November 2014 to coincide with the Percussive Arts Society International Convention (PASIC). Since its release, PhaseRings has been adopted by other musicians for their own projects, some of which are listed in Section A.6.

A.5 Metatone Classifier

Metatone Classifier is a server application written in Python that logs, classifies, and interacts with ensembles of performers using the Metatone apps. The software can be operated from a command line or using an Apple OS X GUI interface. Binary releases for OS X and source code for Metatone Classifier are freely available from http://metatone.net and http://github.com/cpmpercussion/MetatoneClassifier (C. Martin, 2014b).

In this thesis, Metatone Classifier has generally been executed on a local server, but it can also run on a remote server over the internet to
A.6. Response from Users

Some of the apps presented here have been adopted by other musicians for their performance activities unrelated to this research. Warren Hyer, a percussionist from the Central Ohio Symphony, composed a quintet performance called “The Slight Sounds of Nature” that included BirdsNest and PhaseRings among other instruments. This work was performed as part of that orchestra’s activities in 2015.

PhaseRings, in particular, generated an enthusiastic response from the iPad music community after it was released in November 2014. Reviews of the app were posted on app review websites such as The Sound Test Room (Woods, 2014), and iOS Mars (Garland, 2015). Several users from these communities composed works using PhaseRings along with other apps; a selection of these are listed below:

- Aris Lanaridis (2016). *We.* [https://soundcloud.com/aris-lanaridis/we-1](https://soundcloud.com/aris-lanaridis/we-1)
B

Documentation of Artistic Outcomes

B.1 Compositions

Three scored compositions were produced over the course of this project. These works were composed to expand the repertoire available for concerts using the iPad apps detailed above, to explore performances with larger ensembles that were not comfortable with improvisation, and also to experiment with the gesture tracking interactions used with Metatone Classifier. The score for each composition is open-ended with respect for what particular app should be used and only roughly specify the pitch of notes; however, the gestures used to perform each note are indicated precisely. The scores also do not specify the size of ensemble that is required, and are flexible with respect to the length of performances and, in some cases, the particular rhythms that should be played. The reduced reliance on notation in these pieces follows conventions in minimalist and open-ended percussion works of the 20th century, as well as my previous percussion compositions for Ensemble Evolution and my contributions to the Sounds from the Treetops album (C. Martin et al., 2013). While the non-specific notation means that the performer is given broad freedom to interpret, and even to improvise, this is appropriate given the experimental nature of the digital musical instruments used in the performance. These three compositions are self-published and available for download at http://charlesmartin.com.au/compositions.
B. Documentation of Artistic Outcomes

Figure B.1: An excerpt from the score for Gesture Study No. 1. The three staves indicate three different performers and each bar is repeated around eight times. The articulation markings above the notes indicate the touch-gesture that should be used to play it. Open circles denote swirls, staccato (closed circles) indicates taps, and tenuto (horizontal lines) show that swipes should be used.

B.1.1 Gesture Study No. 1 for iPad Ensemble in three parts

*Gesture Study No. 1* was originally composed as a study for the Singing Bowls app, a prototype that was ultimately released as PhaseRings. This work was written specifically for a performance at the 2014 You Are Here Festival in Canberra. The original intention of this work was to demonstrate the ensemble-tracking interactions of the Metatone Classifier agent in a larger ensemble than had participated in earlier improvised performances. The score calls for three groups of iPad performers all using the same app.

As the PhaseRings app does not allow access to a full range of notes, and since the set of displayed notes is different for each player, *Gesture Study No. 1* does not specify precise pitches. Rather, each part is written on a three line staff, with pitches designated as high, middle, and low according to the line they are closest to. Melodic motion is indicated by notes changing vertical location, but the players are not intended to strictly perform this melodic shape.

The work is divided into four sections each lasting roughly 2’30”; each section is divided into five bars of 6/4 time all of which are intended to be repeated around 8 times. This open-ended repetition follows the convention set down in Reich’s (1973) *Music for Pieces of Wood*, and T. Riley’s (1964) *In C*, where each bar indicates an indefinitely looped section of music. In *Gesture Study No. 1*, as in *In C*, each performer is allowed to choose when
to move to the next bar, as long as they do not stray too far from the other performers in the group and start each new section together.

The most precise aspects about the score for *Gesture Study No. 1* are the rhythms written in the score and the notations for the touch gesture that should be used to perform each note. While an early version of the work used custom graphic notations to indicate gestures, a simpler set of standard notations was used in revised versions. Standard articulation markings, small symbols directly above the note to which they refer, are appropriated as gestural notations: Staccato (a closed circle) indicates tapped notes, tenuto markings (a horizontal line) shows swiped notes, and an open circle denotes swirls. Combinations of these gestures can occur with multiple notes in one part having different gestures (as occurs in the middle part of Figure B.1). Since the *nothing* gesture is easily indicated with rests, all five of the reduced gesture classes introduced in Chapter 4 can be indicated with this simple notation.

The precise gestural indications allow the composition to include interactions with the gesture-tracking Metatone Classifier agent. While the performers use different gestures inside each section, at the transition between sections all performers suddenly change from one broad gesture class (e.g. swirls) to another (e.g. swipes). These transition points are designed to trigger new-idea messages from the Metatone Classifier server and to change the notes available to the performers on screen.

The premiere performance of *Gesture Study No. 1* occurred at Ensemble Metatone’s concert at Canberra Museum and Gallery in March 2014. This concert featured seven performers playing the work, and was also the premiere performance with the Singing Bowls app and Metatone Classifier software. Later performances have taken place in trios and quartets including at the ANU Drill Hall Gallery, Boston’s Outpost 186 experimental music venue, and at the Percussive Arts Society International Convention in Indianapolis. In March 2015, a sextet from ANU New Music Ensemble also performed the work.
B. Documentation of Artistic Outcomes

Figure B.2: Excerpt from Gesture Study No. 2. The gestural vocabulary is identical to the first gesture study. The notes in each bar are intended to be repeated until the time indication at the next bar.

B.1.2 Gesture Study No. 2 for iPad Ensemble

*Gesture Study No. 2* was written for three or more iPad performers using any Metatone iPad app. This piece uses the same gestural notation as *Gesture Study No. 1*, but employs a different method for creating the rhythmic and gestural structure within the work. While the first gesture study had a strict rhythmic framework with bars looped a fixed number of times, this second work asks the players to play a simple rhythmic idea for a set amount of time. The work is four minutes in length, each bar represents a 15 second section, and the performers are divided into three groups. Throughout the work, gestural figures filter through the three parts and the performers play rhythmic loops of different lengths so that complex polyrhythmic patterns emerge.

The work was written for the ANU New Music Ensemble, where the time-space writing style had previously been explored in performances for toys and other unconventional musical objects. This mode of writing allows the group to explore simple gestural ideas with a very low barrier of entry, which is ideal for working with brand new iPad performers. *Gesture Study No. 2* was premiered with the New Music Ensemble at ANU Open Day in August 2014 and later revised for performance at an iPads in Percussion Ensemble session at PASIC 2014 and digital publication.

B.1.3 Correspondences for iPad/Percussion Ensemble

*Correspondences* was written for Ensemble Metatone’s *Colour Music* concert in August 2014. While the group had previously used percussion setups
in improvised performances, the venue for this concert was not appropriate for moving large instruments, so this work was developed to use “suitcase” setups of small percussion instruments that the performers could bring from their personal collections.

The work has only one part that any number of performers can play in unison. Each performer is instructed to use four small percussion instruments of their choice as well as one iPad app which should be the same across the group. Two staves are used for the iPad and percussion parts and the performers use one hand for iPad and the other for the instruments. The idea of this work is to compare the sonic output of similar percussive gestures on touch-screen apps and the small instruments. In the first part of the work, the apps and instruments have almost identical notation and sound together. A second section uses long swirled sounds on the iPad apps to accompany slow melodies of iPad sounds. The final section advises the performers to play any instrument and any iPad sound for each note. Correspondences was premiered at the Colour Music concert and later performed at PASIC 2014.
B. Documentation of Artistic Outcomes

![Figure B.4: Two setups from the premiere performance of Correspondences. The performers were asked to choose instruments that, along with an iPad, would fit on one percussion table.]

B.2 Performances

2013-05-14 *MetaTravels* at Canberra International Music Festival: DRUMatiX with guests JB Smith and Charles Martin, ANU School of Music.

2013-05-28 *Nordlig Vinter for Vibraphone and iOS devices* at NIME2013, KAIST, Daejeon, Republic of Korea.

2013-07-07 *MetaLonsdale* at Everything/Nothing Projects Exhibition Opening, Lonsdale St Traders, Canberra.

2013-10-06 *MetaTravels* and *MetaLonsdale* at Electrofringe 2013, Hunter St TAFE, Newcastle.

2013-10-25 *MetaLonsdale* at Revenant Media / Theremin ’75, ANU School of Art Gallery.

2013-11-07 Ensemble Evolution — GroupMuse Living Room Concert. Featured *BirdsNest* improvisation for iPad and percussion trio. Jamaica Plain, MA, USA.
B.2. Performances

2013-11-08 – 2013-11-10 Ensemble Evolution *Sounds from the Treetops* concert series at Arnold Arboretum, MA, USA. Included *BirdsNest* for iPad and percussion trio.


2014-03-17 You Are Here festival Canberra — premiere of *Gesture Study No. 1* for iPad septet and Ensemble Metatone using the Metatone Classifier in a public performance.

2014-03-20 ANU School of Music: Alec Hunter and Friends concert of improvised music.


2014-05-05 Touch and Tone Experimental Percussion Concert, Boston — performances of *Study in Bowls, MetaLonsdale* and *BirdsNest* for iPad trio.

2014-06-07 Performance with the ANU New Music Ensemble.


2014-08-14 Ensemble Metatone at Drill Hall Gallery’s Colour Music exhibition opening.

2014-08-24 Ensemble Metatone’s *Colour Music* concert at the ANU Drill Hall Gallery.

2014-08-30 *Gesture Study No. 2* with the ANU School of Music Experimental Music Studio at ANU Open Day.

2014-08-31 Performance with the ANU New Music Ensemble at Drill Hall Gallery.
B. Documentation of Artistic Outcomes

2014-09-03 Works for spatialised keyboards, percussion, and viola da gamba by Helyard/Hunter/Martin ensemble at ANU School of Music (Erin Helyard, Alec Hunter, Charles Martin).

2014-09-07 Helyard/Hunter/Martin concert at Drill Hall Gallery.


2015-03-14 ANU New Music Ensemble at the Art Not Apart festival, New Acton, Canberra.

2015-03-21 Gesture Study No. 1 with the ANU New Music Ensemble at the You Are Here festival.

2015-03-21 ANU School of Music Experimental Music Studio performance (Charles Martin, Alec Hunter, Johannes Luebbers) at the You Are Here festival.

2015-07-10 Coordinator of the DIY Synth Workshop at the ANU School of Music.

2015-07-21 Avoidance Behaviour: A Film with Live and Pre-recorded Soundtrack (Charles Martin, Alec Hunter, Elyse Howe) at the Ainslie Arts Centre.

2015-07-24 DIY Synth Group at Collected Resonances: Session 1 at the ANU School of Music.

2015-11-20 iPad recording session with ANU Experimental Music Studio to record Gesture Study No. 2 and a long form improvisation with the PhaseRings app.

2015-11-28 iPad duo performance Andromeda is Coming (Charles Martin, Alec Hunter) at Electrofringe, 107 Projects, Sydney.
B.3 Recordings

A significant part of the research in this thesis centred on performances which, although research-motivated, were part of an ongoing artistic practice. Two of Ensemble Metatone’s performances in Canberra were recorded and released as digital-only albums on Bandcamp. These releases not only represent the artistic achievements of this group but also the research outcomes. In particular, the second release Colour Music relies on the repertoire of refined iPad apps and the Metatone Classifier agent which could not have been developed except for the significant research efforts of the group over two years. These recordings are also, as far as we are aware, the only albums so far released by a touch-screen ensemble, and two of only a few live touch-screen music performances to be formally released.

B.3.1 Ensemble Metatone

Ensemble Metatone is a self-titled release of the group’s research performance August 2013. The recording consists of two tracks and encapsulates the performance practice and technology developed up to this point in the ensemble’s development. The first track is an iPad improvisation using the MetaLonsdale app including networked performance syncing features, and the second is an improvisation with the MetaTravels app and percussion setups chosen by the performers.

The cover art was designed by Benjamin Forster using touch-interaction protocols captured during the performance and represents an alternative perspective on the artistic outcomes of the projects. The background texture of the cover is made up of an excerpt of text from the MetaLonsdale recording, while the typographical marks in the foreground are the structural parts of the OSC message format of that text with the meaningful data removed.

This album was released in March 2014 simultaneously with a series of performances using the apps. The liner notes and track listing are as

---

1 Ensemble Metatone can be found online at: https://charlesmartin.bandcamp.com/album/ensemble-metatone/
B. Documentation of Artistic Outcomes

Figure B.5: Album cover for Ensemble Metatone. Designed by Benjamin Forster, the album art incorporates touch data from the performances and typographical symbols that structure that data.

follows:

*Ensemble Metatone*

Ensemble Metatone plays improvised and experimental music using Charles Martin’s iPad-instruments and percussion. Recorded live in Canberra at the ANU School of Music 2013-08-03, this concert followed a series of rehearsals-as-research where the performers thoroughly explored two iPad app-instruments through improvised performance.

In these contemplative works, the group performs on iPads with a vocabulary of touch-gestures inspired by their percussive
backgrounds and, conversely, uses acoustic percussion instruments to mimic and augment the apps’ sound-palettes.

**MetaLonsdale**

First created for a gallery opening in Canberra’s Lonsdale St Traders, the app-instrument for MetaLonsdale blends field recordings from the area’s cafes and shops with the textures of tuned percussion. While the four players improvise their parts, network connections between the iPads challenge their instincts by changing the sounds and effects under their fingers while matching their harmonic structure.

**MetaTravels**

An improvised conversation between four percussionists armed with their own selection of instruments and the MetaTravels app, this work sees the four performers reacting to each other’s musical gestures and setting off on tangents of exploration. Taking the affordances of the app as a starting point, the percussionists warp and slice the app’s field recordings, mimicking and contrasting these sounds with their acoustic instruments.

**Credits**

**Charles Martin** — vibraphone, iPad, and app design

**Christina Hopgood** — crotales, terracotta pots, and iPad

**Jonathan Griffiths** — snare drum, cymbals, and iPad

**Yvonne Lam** — bass drum, cymbals, and iPad

**B.3.2 Colour Music**

Following Ensemble Metatone’s performance at the ANU School of Art gallery in October 2013, the group was invited by Tony Oates to perform
as part of the Colour Music exhibition in August 2014 (Oates & Maloon, 2014). Similarly to the research concert the previous year, this performance came after several months of intensive rehearsals, recording sessions, and revisions to the iPad apps, and the concert included three improvisations on the interfaces evaluated at length in Chapter 5. The concert also included two notated compositions for iPad ensemble and Cage’s (1943) *Trio for seven woodblocks*, from the suite *Amores*. This work was performed on woodblocks, as well as adapted for the iPad touch-screens. The trio from *Amores* represents an extremely productive period in the short history of contemporary percussion performance where works that would set the course of conservatory training in percussion were composed. Performing this work, and *Correspondences*, where percussive gestures are com-
pared across touch-screens and small percussion instruments, was intended to bring touch-screen computers into the culture of exploration that has been so successful for percussion.

The selection of works highlights Ensemble Metatone’s artistic direction in 2014. From an initial focus on developing an improvised practice the group moved towards developing a repertoire for touch-screen performance and works that could be performed in collaboration with other groups. The concert was recorded and released digitally as *Colour Music* in 2015\(^2\). The liner notes from the online release are as follows:

*Colour Music*

Interweaving percussive gesture with electronic sound, *Colour Music* is a selection of works that seek to redefine how music can be made with touch-screen computers. The seven tracks include the results of a formal study of improvisation on three different iPad interfaces and notated compositions for iPad and percussion. This collection of Ensemble Metatone’s performances on touch-screen interfaces mirrors the formal explorations of percussion instruments initiated by composers such as John Cage in the first half of the 20th Century, an investigation that is made explicit by the group’s performance of Cage’s Trio for 7 woodblocks, first on the original blocks, and then on iPads.

The tracks were recorded live at the ANU Drill Hall Gallery during the *Colour Music* exhibition. The catalogue for this exhibition emphasises that it features visual artists who made connections between pictorial form and aspects of sound, music, and movement (Oates & Maloon, 2014). By improvising with the touch-screen interfaces of the iPads, Ensemble Metatone actively participates in this process; the apps’ visual interfaces are informed by percussion gesture and sonic affordance, but these

\(^2\)Colour Music can be found online at: [http://charlesmartin.bandcamp.com/album/colour-music](http://charlesmartin.bandcamp.com/album/colour-music)
B. Documentation of Artistic Outcomes

interfaces also inform the individual and ensemble performances.

*Gesture Study No. 1*

Gesture Study No. 1 is an open-ended exploration of touch-screen gestures for an ensemble of iPad performers. Over four sections, the performer improvise specific notes within notated rhythmic and gestural frameworks to emphasise the affordances of the iPads’ touch-screen interfaces. The first notated iPad work written for Ensemble Metatone, it was premiered at the You Are Here festival, Canberra in March 2014 with an expanded ensemble of seven players. In this recording, the performers used the PhaseRings app.

*BirdsNest*

An improvisation with the BirdsNest app, this track is a journey through a Northern Swedish forest in the bright evenings of summer. The app tracks the performers’ touch gestures, moving them through a sequence of soundscapes from the forest floor to high in the trees with percussive sounds and bird-song. The sound-world presented in this app was captured during a site specific project at the iconic Treehotel in Harads, Sweden.

*Snow Music*

The Snow Music app was designed in 2012 for percussion and iPad performances with field recordings of snow and ice from Piteå, Sweden. In 2014, Ensemble Metatone undertook a project to revise the design to interact with their ensemble performances. In this improvisation, the app tracks the performers’ gestures and responds by playing supportive background sounds. Throughout the performance, the app layers glockenspiel sounds over the performers’ snowy explorations.
**Singing Bowls**

This improvisation with Singing Bowls comes at the end of a process throughout 2014 to explore composition, and improvisation with the app. Singing Bowls is inspired by Tibetan prayer bowls, which can be struck or rubbed to produce a continual ringing sound. The app presents each performer with a selection of notes from the same scale, so that they can perform independent melodic parts in the same harmonic space. Throughout the performance, the app tracks their new gestural ideas, rewarding them with new notes and harmonies.

**Amores (John Cage, 1943) for blocks and iPads**

Much of contemporary percussion practice can be traced back to Cage’s landmark compositions such as *Amores* from 1943. Over two tracks, Ensemble Metatone perform the trio from Cage’s Amores on the original woodblocks, and then on the PhaseR-ings iPad app. Representing the creative possibilities of simple instruments, this work serves as a benchmark for Metatone’s computer music explorations.

**Correspondences**

Correspondences is a study in gesture across iPad screens and small percussion instruments. Written for an ensemble of flexible size performing in unison, each player chooses a set of four small percussion instruments to contrast with a musical iPad app that is common to the whole ensemble. Members of Ensemble Metatone use the BirdsNest app along with selections of instruments from their own collections. Triangles, woodblocks, prayer drums, and small cymbals are all present in the portable and playful setups.
B. Documentation of Artistic Outcomes

Credits

Charles Martin — percussion, iPad, and app design
Christina Hopgood — percussion and iPad
Jonathan Griffiths — percussion and iPad
Yvonne Lam — percussion and iPad

Special Thanks: Tony Oates and the ANU Drill Hall Gallery
C

Publications and Presentations

The following works were published and presented during my PhD candidacy. Some publications form the basis of work presented in this thesis and have been cited where they appear.

C.1 Peer-reviewed publications


C. Publications and Presentations


• Martin, C., Gardner, H., Swift, B., & Martin, M. (2016). Intelligent agents and networked buttons improve free-improvised ensemble...
C.2 Conference Presentations and Seminars


C.2 Conference Presentations and Seminars

- NIME 2013 poster, “Performing with a Mobile Computer System for Vibraphone”.

- ICMC 2013 poster, “Integrating Mobile Music with Percussion Performance Practice”.

- NIME 2013 performance, *Nordlig Vinter for vibraphone and iOS devices*.

- PASIC 2013 ensemble showcase concert, Ensemble Evolution — *Sounds from the Treetops* featuring BirdsNest for iPad and percussion trio.

- ACM CHI 2014 Curating the Digital workshop presentation, “Preserving Musical Performance on Touch-Screens”.

- ACM CHI 2014 presentation, “Exploring Percussive Gestures on iPads with Ensemble Metatone”.

- ACM CHI 2014 Interactivity installation, “iPad Apps for Percussive Improvisation”, a performance setup of 4 iPads was installed in the exhibition for attendees to use in short improvisations.

- ACMC 2014 presentation, “Exploring Percussive Gesture on iPads with Ensemble Metatone”.

229
C. Publications and Presentations

- PASIC 2014 clinic, “iPads in Percussion Ensemble”
- NIME 2015 poster, “Tracking Ensemble Performance on Touch-Screens with Gesture Classification and Transition Matrices”.
- ACMC 2015 abstract presentation, “Gesture Study and Correspondences: Composing for Percussive iPad Apps”.

C.3 Open Data Publications

This appendix contains the survey questions and rating scales used in chapters 5 and 7.

D.1 App and Agent Study Questionnaire

Each question in the app and agent study described in Chapter 5 was answered on a five-point Likert-style scale.

1. How would you rate that performance? (terrible, bad, neutral, good, excellent)

2. How would you rate the level of creativity in that performance? (terrible, bad, neutral, good, excellent)

3. How did the agent’s impact compare to having it switched off? (much worse, worse, same, better, much better)

4. How well were you able to respond to the app’s actions? (very badly, badly, neutral, well, very well)

5. How well were you able to respond to the other players’ actions? (very badly, badly, neutral, well, very well)

6. How was the app’s influence on your own playing? (very bad, bad, neutral, good, very good)
D. Questionnaires

7. How was the app’s influence on the group performance? (very bad, bad, neutral, good, very good)

D.2 Agent-Control and Networked UI Study Questionnaire

The study described in Chapter 7 included a questionnaire given after each performance and post-session preference survey.

D.2.1 Performance Questionnaire

The performance questionnaire included 24 questions that were answered on 9-point Likert-style scales. These scales included labels for the extreme and middle points.

Technical Proficiency

1. How much did you focus on particular touch gestures in that performance? (very little, neutral, very much)

2. How much did you explore a range of touch gestures? (Very Little, Neutral, Very Much)

3. How would you rate your technical proficiency using the app in that performance? (very bad, neutral, very good)

4. How much did the app impede your performance? (very little, sometimes, very much)

5. How much did the app enhance your performance? (very little, sometimes, very much)

Musical Interaction

6. How much did you interact musically with the other performers? (very little, sometimes, very much)
7. How much did the other performers interact musically with you? (very little, sometimes, very much)

8. How well were you able to respond to the other musicians’ actions? (very badly, average, very well)

9. How well were you able to respond to the app? (very badly, average, very well)

10. How would you rate the overall level of musical interaction among the ensemble? (very bad, neutral, very good)

Musical Structure

11. How would you rate the complexity of that performance? (very simple, neutral, very complex)

12. How appropriate was the length of that performance? (much too short, perfect, much too long)

13. How would you rate the app’s influence on your own playing? (very bad, neutral, very good)

14. How would you rate the app’s influence on the ensemble performance? (very bad, neutral, very good)

15. How would you rate the overall musical structure of that performance? (very bad, neutral, very good)

Creativity

16. How much did you present new musical ideas to the others in the ensemble? (very little, neutral, very much)

17. How much did you take on and develop musical ideas first presented by the others in the ensemble? (very little, neutral, very much)
D. Questionnaires

18. How would you rate your personal creativity in that performance? (very poor, average, very good)

19. How would you rate the other performers’ creativity in that performance? (very poor, average, very good)

20. How would you rate the overall creativity in that performance? (very poor, average, very good)

Performance Quality

21. How would you rate the quality of your contribution to that performance? (very bad, neutral, very good)

22. How would you rate the quality of the other performers’ contribution in that performance? (very bad, neutral, very good)

23. How would you rate the overall quality of that performance? (very poor, neutral, very good)

24. How enjoyable or unpleasant was that performance? (very unpleasant, neutral, very enjoyable)

D.2.2 Post-session Survey

Each question in the post-session survey was answered by selecting a preference out of one of the experimental conditions.

1. Which method do you think resulted in the best performance?

2. Which method do you think resulted in the most creative performance?

3. Which method do you think resulted in the best performance structure?

4. Which method do you think resulted in the best musical interaction in the ensemble?
D.2. Agent-Control and Networked UI Study Questionnaire

5. Which method did you find easiest to perform with?

6. Which method did you find most challenging to perform with?

7. Which method of triggering change in the app did you prefer?


References
References


238


References


References


References


References


References


250


References


References


254


References


Swift, B. (2013). Chasing a feeling: Experience in computer supported jamming. In S. Holland, K. Wilkie, P. Mulholland, & A. Seago (Eds.),
References


References


