A Percussion-Focussed Approach to Preserving Touch-Screen Improvisation

Charles Martin and Henry Gardner

Abstract Musical performances with touch-screen devices can be recorded by capturing a log of touch interactions. This object can serve as an archive or as a basis for other representations of the musical work. This chapter presents a protocol for recording ensemble touch-screen performances and details the processes for generating visualisations, gestural classifications, and graphical scores from these logs. Our experience of using these new representations to study a series of improvised ensemble performances with iPad-based digital musical instruments leads us to conclude that these new-media artefacts allow unique insights into ensemble interactions, comprehensive archiving of improvised performances, and the potential for re-synthesis into new performances and artworks.

1 Introduction

As an artistic experience, musical performance is fleetingly temporal and the history of music abounds with technologies and traditions to grab hold of musical performances, recording them in some format to be archived, understood, and developed. All of our 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 our cultural landscape.

While western classical music is often defined by the score [9], and popular music by the song or the recorded version [18], the practice of free improvisation often defies the identification of a canonical "musical work". In free-improvised ensemble

Charles Martin
Research School of Computer Science, The Australian National University, Canberra, Australia
e-mail: `charles.martin@anu.edu.au`

Henry Gardner
Research School of Computer Science, The Australian National University, Canberra, Australia
e-mail: `henry.gardner@anu.edu.au`

music, all decisions about the notes to play are made in the moment by individual musicians with the work ending only when all performers have stopped playing [7]. With each performance equally identifiable as a different work, the output of free-improvisation ensembles is often represented as audio recordings with liner notes that document changes in personnel or instrumentation. These notes are frequently the main indicator of difference between works. Such archives of free-improvised performances can be time-consuming to browse, due to the temporal nature of the recordings, and difficult to analyse, particularly when performers use similar-sounding electro-acoustic instruments. When performing with computer-based instruments, 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 also affords the creation of other representations of the performance giving curators of free-improvisation new perspectives on the musical works and the improvisers, themselves, ways to develop and preserve their artistic practice.

In this chapter, we present a protocol for automatically documenting performances of free-improvised music made on touch-screen computers. Our protocol, and the touch-screen instruments with which it is used, are designed from a percussionist-centred perspective. Rather than particular notes and rhythms, the focus of traditional musical notation, our protocol records detailed touch movements in absolute time and abstract touch gestures identified during the performance by our computer software. Given that ensemble interactions are one of the most interesting aspects of free-improvised music [4], our protocol is designed to connect to multiple touch-screen devices engaged in ensemble performance over a local network or the internet.

We argue that our protocol can serve as an archival format and addresses many of the issues with curating improvised music. We also argue that these recorded logs satisfy Manovich's principles of new media objects [22]; their numerical representation allows them to be automatically varied and transcoded into new artefacts allowing new characterisations, analysis, and appraisal of performances. Statistical analysis of the logs can abstract the form of improvisations away from particular instruments and players, providing a broader understanding than can be gained from audio recordings. Algorithmically generated visualisations and gestural scores created from the logs leads to new perspectives on our performances and feeds back in to the process of developing an improvised practice.

The protocol and techniques we describe in this chapter bring improvised musical interactions onto an equal footing with other artforms that are more easily catalogued and curated in spaces for art and interaction. We envision that the gestural scores, visualisations, and other artefacts derived from our performance logs could allow improvised musical interactions at touch-screens to be curated in a much more significant way than is possible with acoustic instruments.

We will describe our protocol for capturing touch interactions and how it has developed from previous schemes for capturing and controlling musical performances. We will also present the classification and visualisation tools used for analysing and comparing performances documented with our protocol. Our protocol and these tools were developed as part of a series of research projects in touch-screen musical performance and we will discuss how they have influenced the research and artistic outcomes of these projects. In particular, we will discuss the experiences of using these systems with Ensemble Metatone, a free-improvisation percussion group that participated in a longitudinal study over eighteen months performing with our touch-screen instruments.

2 Percussive Improvisation on Touch-Screens

The development of our touch-screen instruments and performance logging protocol was a percussionist-centred process. Performances and rehearsals of Ensemble Metatone, a free-improvising percussion ensemble, were conducted to explore and evaluate instruments. Our performance logging protocol was developed over this process to document and analyse these activities. The same instruments and tools were introduced to other experienced percussionists for subsequent performances and have only been used with non-percussionists relatively recently. Before detailing the protocol and instruments, it is worth considering why percussive improvisation is a useful process for characterising musical interaction.

Percussion is an artistic practice defined by an approach to interaction rather than by any particular instrument that percussionists play. Percussionists perform by "striking, scraping, brushing, rubbing, whacking, or crashing any... available object" [38]. Blades [3] discusses the earliest percussion instruments, idiophones, where the body of an instrument creates the sound, rather than an air column or string. He divides them by their method of interaction: "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 instruments (such as Cook [8]) and focus on the mode of interaction for the instruments rather than their physical design.

Modern percussionists are accustomed to exploring non-traditional objects to create music and use these percussive gestures to coax wide varieties of timbres and musical gestures from simple instruments. Performers of Xenakis' *Psappha* [46] or Feldman's *King of Denmark* [12] must design their own multi-instrument setup to fit the 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. This percussive approach to investigating new instruments can be applied to touch-screen computers which can also be struck, scraped, and rubbed with fingers and hands.

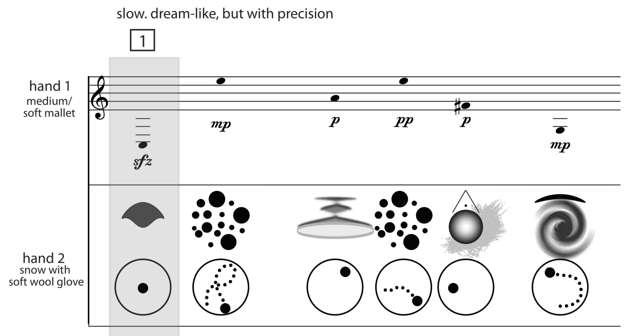

Fig. 1 An excerpt from Matthew Burtner's *Syntax of Snow* [6] for solo glockenspiel and bowl of amplified snow. The composer defines a vocabulary of gestures for interacting with the snow with one hand represented by symbols below a regular staff for notes on the glockenspiel. (Score Excerpt © M. Burtner 2010, reproduced with permission.)

2.1 Composing with Gestures

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 is a popular way of notating music for particularly unconventional instruments. Thierry de May's *Music de Tables* [31] is written for three percussionists who perform on the surfaces of regular tables. de May defines a vocabulary of notation for gestures that are used with standard rhythmic notation in the score. Burtner's *Syntax of Snow* [6] 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 1) for each gesture, representing the type of gesture as well as hand position in the bowl.

Although Burtner's vocabulary of gestures is specific to the particular case of performing with snow, some of the gestures (e.g "touch with finger", "swish with palm", "draw line") could generalise to other instruments and to touch-screens. For documenting performances on touch-screens it is necessary to characterise a vocabulary of gestures that is common to many different types of instruments that can be implemented on touch-screen devices and can express a wide variety of playing styles.

It is notable that many of the gestures indicated in Burtner's 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 Human Computer Interaction (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. [45]. 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 which mainly consist of continuous interactions.

2.2 Free-Improvised Performance

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" [41]. This definition is typical of many, but it frames free-improvisation subtractively: improvisation is "regular" performance minus restrictions. To understand what is really exciting about free-improvisation as a mode of artistic expression and as a methodology for researching unfamiliar interactions, we must look at what free-improvisation *adds* to performance. Bill Cahn, member of the pioneering percussion group, Nexus, writes that improvisation encourages "a deeper knowledge of the instruments and their sound-making possibilities" [7]. 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." [11]

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 [35] 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 proposes 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 [41]. Nunn [33] similarly argues that a "segmented form" divided by "transitions" is the fundamental structure of free improvisation.

Much free-improvised music is performed by ensembles of performers rather than solo artists [41]. In fact, definitions of free-improvisation have emphasised this aspect with Mazzola writing 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." [30]

2.3 Ensemble Metatone and Improvising iPad Ensembles

In the present research, an improvising percussion group was not only used to create new music but to explore and evaluate touch-screen instruments running on Apple iPads. Ensemble Metatone was brought together in Canberra, Australia, to study the process of performing free-improvised music starting with a prototype app and,

Fig. 2 Ensemble Metatone performing *MetaLonsdale* at the ANU School of Art Gallery in October 2013 (left to right: Jonathan Griffiths, Charles Martin, Christina Hopgood, Yvonne Lam).

through a process of iterative design, eventually presenting concerts with multiple apps. The members of the group (including one of the authors) were all highly qualified in classical percussion and had significant experience as improvisers.

Ensemble Metatone undertook a series of studio rehearsals throughout 2013 to develop a repertoire of free-improvised music with our iPad apps [24]. Using a process of "creative music making" [7] where improvisations are followed by critical listening and discussion, the performers developed a vocabulary of touch interactions inspired by their percussion training. Notably, they also discovered novel sounds from the instruments that could be created with unusual or atypical interactions and were not foreseen by the app designer [26]. The performers of Ensemble Metatone settled on a combination of iPad and acoustic percussion instruments allowing them to choose from a wide palette of sound colours in their performances. The initial series of rehearsals was followed by a recorded research concert with a live audience, and a series of performances at experimental art events. The recording of Metatone's research concert was released as a digital album in March 2014 [29].

Other percussion performers were also invited to work with our touch-screen apps in a variety of improvised and semi-composed performances in Australia and the USA. In 2014, a number of Metatone apps were made available for free in the Apple iTunes App Store and the authors are aware of several performances using the apps unrelated to our work.

In 2015, the Metatone apps were used as part of the educational activities of the New Music Ensemble at the ANU School of Music including rehearsals and performances and in activities with high-school students. In this setting participants had a range of instrumental experience so iPads were used as the only instrument. A formal study was conducted with four iPad quartets, including members of the New Music Ensemble and volunteers from the local music community, who performed several improvisations with different combinations of software features.

The majority of these rehearsals and performances were audio and video recorded and also documented using our touch-screen performance protocol. To date, we have archived over 100 performances of our iPad apps using these tools, forming a highly comprehensive corpus of work for studying improvisation on touch screens. In the following sections, we will describe forms of analysis and derivative works that can be generated from this archive of performance documentation.

2.4 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. We advocate an expanded method of documenting improvised music 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 [37]. Musical works might be directed by a score; might be "thick" or "thin" depending on the 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 [10]. 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 extremely uncommon for free-improvised ensemble performances. Audio recordings of free-improvised performances capture the sonic results of the performers' 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.

When improvised music is performed on touch-screen instruments, a log of touch-interactions captured during the performance can supplement traditional recordings and could 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 [21]. Such a log would also satisfy Manovich's principles for a new media artwork [22]. In particular, the log of touch-interactions is variable, forming the basis for derivative artworks that also represent aspects of the original performance.

Borgo has drawn parallels between the swarm-like collaboration in free-improvised performances and the community collaborations that define the field:

> "One of the particular challenges of contemporary improvisation, for both players and listeners, is to remain aware of and sensitive to the many musical gestures and processes circulating between members of the group in the moment of performance and between members of the community as ideas circulate via recordings, impromptu meetings, and the overlapping personnel of various working groups." [4]

These communities of practitioners, listeners, and concert organisers are the curators of the free-improvised music world. We argue 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. In Sections 3 and 4, we describe our approach to recording improvised touch-screen performances and transforming such transcriptions into animations, gestural scores, and new artworks. These multiple representations of musical performance allow the curation and analysis of an emerging improvised practice.

3 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 [14], 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 [19], allowed a musician's performance to be automatically transcribed and converted into a paper "piano roll" that could be re-played 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 our work was the MIDI [32] (Musical Instrument Digital Interface) protocol, which was developed in the late 1970s to connect electronic music *controllers*, such electronic versions of keyboards, drums and wind instruments and new musical interfaces such as "THE HANDS" [44], with electronic synthesisers 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[20].

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 on" 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 [15] (Open Sound Control) 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 [39], although it is not commonly used in commercial electronic instruments.

Some have attempted to define protocols using OSC to standardise interaction with certain types of interface. TUIO [17] is one such protocol designed for table-top interfaces where fiducial markers and finger touches can be tracked. 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 table-top 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, acceleration, and rotation. This simplifies the requirements for software receiving TUIO which does not have to keep track of objects in between bundles of messages and does not need to worry about errors due to messages arriving out of order.

As described in Section 3.2 below, our protocol for logging touch-screen performances 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 multitouch digitiser in front of the screen is interpreted by the operating system which keeps track of individual touches and divides the incoming data into events [1]. Software developers can implement a set of callback functions to access these events individually. So-called `UIEvents` track the state of touches on the screen - they "begin", "move", "end", and may be "cancelled" if they are mis-recognised (Figure 3). 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.

```
- (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;
```

Fig. 3 Callback methods for accessing touch events in Apple iOS [1]. Our Metatone apps log each `touchesBegan`, `touchesMoved`, and `touchesEnded` event.

3.1 Metatone Apps

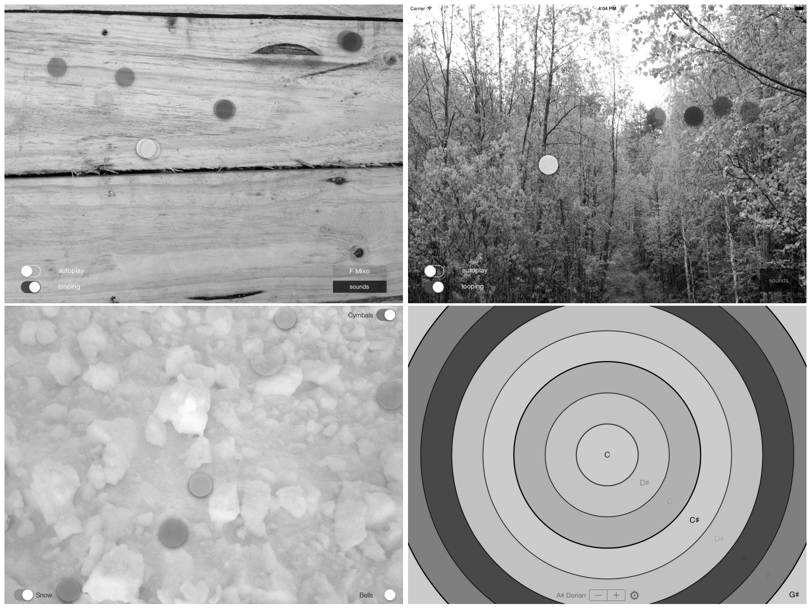

Fig. 4 Screenshots of four Metatone apps: *MetaLonsdale*, *BirdsNest*, *Snow Music*, and *Phase Rings*. *MetaLonsdale* and *BirdsNest* allow free-form touch improvisation over the whole screen with a looping function that repeats tapped notes. *Snow Music* allows performers to create snow sounds supported by algorithmically generated soundscapes. *PhaseRings* is an annular interface for performing with pitched percussion sounds selected from configurable scales.

The touch-screen device chosen for our performances was the Apple iPad. A variety of apps, including the four shown in Figure 4, have been developed for performances by Ensemble Metatone and other groups. All of the Metatone apps share the same fundamental mode of interaction: a tap triggers a single sound with a percussive envelope (i.e. a sharp attack and long decay), while swiping plays a continuous sound with volume related to the velocity of the moving touch. Apart from this commonality, the Metatone apps feature different palettes of available sounds and modes of synthesis, different arrangements of sounds on screen, a variety of special features, and different kinds of networked interactions between the iPads themselves as well as with a server application.

The earliest Metatone apps, *MetaTravels* and *MetaLonsdale* were both designed to allow percussionists to control combinations of field-recordings and pitched percussion instruments [27]. *MetaTravels* used field-recordings from around the world and all chromatic pitches were available in the interface simultaneously. *MetaLonsdale* limited performers to field-recordings from a Canberra café and pitches from a rotating sequences of scales. A related app, *BirdsNest*, allowed performers to create a soundscape reminiscent of northern Swedish forests, with samples of bird calls and field recordings from the location as well as xylophone and wood-block samples. These three apps used UI switches to control a looping feature that would repeat tapped notes and UI buttons to shuffle the sounds available to the player from a palette of sonic material. *Snow Music*, originally developed with the percussion group *Ensemble Evolution* [23], allowed percussionists to manipulate samples of snow sounds (similarly to Burtner's amplified snow) and to switch on algorithmically-produced backing soundtracks to accompany themselves. *PhaseRings* is the latest Metatone app and consists of an annular interface for performing with pitched percussion sounds. Each ring in the interface corresponds to a different pitch. Performers configure the app with a sequence of musical scales and the app displays a subset of pitch-rings taken from those scales with changes in the ring setup triggered by network interactions or UI elements[1].

In performances with our iPad apps, all touch interactions during the performance are transmitted over a Wi-Fi network. Our server software has evolved from a simple script in early rehearsals to a Python server application, Metatone Classifier[2], that can run on a local network or on a remote virtual server. Communications between the apps and server are accomplished using the OSC [15] (Open Sound Control) message format. Data is transmitted either through Unix sockets using the UDP protocol (as is typical for OSC messages) or through WebSockets [13], which is the preferred method due to increased reliability of transmission and the ability to easily open a bi-directional connection with a remote server. The Metatone apps automatically find the server and other active Metatone apps on their local network using Bonjour (zero-configuration networking). The app-to-app and app-to-server interactions have been documented elsewhere [25, 28]. While the designs of these interactions are outside of the scope of the present chapter, it suffices to say that our goal has been to introduce features that synchronise changes in the app's functionality throughout performances in response to individual and ensemble interactions. These features have been designed to emphasise and enhance the sense of group-mind that has been observed in performances of ensemble free-improvisation [4].

3.2 The Metatone Log Protocol

Over our earliest rehearsals with Ensemble Metatone and the *MetaTravels* app, we developed a protocol for capturing touch-screen information from each performer's iPad that mirrors Apple iOS's touch-event handling framework. The information is sent to a central server using the OSC data format. As we developed more features for our apps and our (Metatone Classifier) server software this protocol was

[1] The Metatone apps are available on the iTunes apps store and links can be found on `http://metatone.net`. This website also includes videos and audio recordings of performances with these apps.

[2] `http://metatone.net/metatoneclassifier/`

extended to document the performers' gestural and ensemble states and app-to-app interactions sent between iPads. A complete listing of our OSC messaging scheme is given in Table 1. When our server 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 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.

App to Server Messages:	
OSC Address	Parameters
`/metatone/online`	device
`/metatone/touch`	device, X, Y, velocity
`/metatone/touch/ended`	device
`/metatone/switch`	device, name, position
`/metatone/app`	device, name, state
Server to App Messages:	
OSC Address	Parameters
`/metatone/classifier/gesture`	device, gesture type
`/metatone/classifier/ensemble/event/new_idea`	device, measure value
`/metatone/classifier/ensemble/state`	type, value 1, value 2

Table 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 3). The "device" parameters are unique identifiers for each iPad in the ensemble.

Our 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.

Each of our iPad apps contains a small number of button and switch UI elements that are used in the 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, our iPad apps send messages to the other apps performing while connected on the same local network. These messages are also copied to the server software using the `/app` OSC address.

As will be described in later sections, Metatone Classifier has the capacity to identify the performers' touch gestures in real-time during performances. The server tracks these gestures to identify the state of the whole ensemble and, in particular, identify moments of peak gestural change where the group may have moved onto a new musical section. While this information is recorded for later analysis, it is also returned to the performers' apps, and the apps use it to update their interfaces and present new performance possibilities to the performers [28]. Our protocol for performance logging includes three OSC messages from the server to the iPad apps. Gesture classifications are returned to the iPads each second with OSC `/gesture` messages; whenever a new musical section is detected, the server sends an `/event/new_idea` message to all connected iPads. Various measures of the ensemble state are returned to the iPads each second using the OSC address, `/state`.

The scheme for logging touch-interactions (see Table 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 our protocols were created for research purposes, the CSV storage format allows us to easily transform these logs 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.

4 Transcoding Performance Protocols

Our system of iPad apps and server software records all touch events, UI interactions, and app-to-app communications that occur during improvised performances in a CSV format. These recordings are mutable new-media objects and, as suggested by Manovich, by transcoding these objects "the logic of a computer can be expected to significantly influence the traditional cultural logic of media" [22].

In this section we will describe ways in which our performance protocols can be transcoded into new representations of the improvisations, both in real-time, during a performance, and afterwards. We will explain how gestural classifications of each performer's touches leads 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.

4.1 Gesture Classification and Gestural Scores

Each second during performances, our server software analyses the previous five seconds of touch-data from each connected iPad and identifies the performers' current touch gesture. Our system calculates feature vectors of descriptive statistics for each player from these five-second windows of recorded touch interactions. The feature vectors are classified using a Random Forest Classifier [5] from the Python `scikit-learn` library [34] that is trained with examples of nine touch gestures recorded by our app designer using a formal data collection procedure [28].

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

Table 2 The nine gesture classes that our server software, Metatone Classifier, can identify in touch screen performances. This vocabulary resembles the gestural language of percussion performance, rather than typical command gestures used in HCI.

The nine gesture classes are based on those discovered through qualitative analysis of Ensemble Metatone's earliest series of rehearsals [24]. The vocabulary focusses on three fundamental touch-gesture groups: tapping, swiping, and swirling, and it includes several variations of each group. Unlike other systems for gesture recognition [45] which are designed to interpret sequences of movements with a beginning and ending as a command in a computing interface[3], our classification system aims to segment a continuous stream of free-form gestures. For this reason, our vocabulary seeks to identify "tapping", which could continue indefinitely, rather than "tap" which is completed after one touch interaction. In this way, our touch-screen gestural vocabulary resembles some of the snow gestures of *Syntax of Snow* [6] which also are open ended with respect to the number or length of interactions. 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 5, 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 5, 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

[3] This includes Apple's built in `UIGestureRecognizer` [1] class.

performer breaks out for a solo idea, their line moves up or down, or might move to the "nothing" level if they have stopped playing. Sometimes the performers split into sub-ensembles, exploring separate groups of gestures. Perhaps most interesting are the broadest structural elements where all members of the ensemble change gestures together over a short period of time. Such moments of increased change seem to segment the improvisations into large sections. In a composed piece, these might be called movements, but in an improvised setting, they appear to be related to "new idea" moments where the ensemble spontaneously transitions into a different sound or musical behaviour. While such behaviour in improvisation has been previously described by by Pressing [35], Stenström [41], Bailey [2], and others, it has not previously been observed in an automatically-generated gestural transcription.

Since these gestural scores are so helpful in understanding, at a glance, the overall flow of improvised touch-screen performances, they serve as more useful archival documents of performances than, for example, still photographs of the stage setup. Rather than simply recording the location and setting of the performers, gesture-scores store a high-level view of the performers' gestures throughout a whole performance. In fact, it would be possible to use gesture-scores as the reference material for 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 to assist the performers or even be integrated into the touch-screen interface as in the Decibel ScorePlayer [16]. 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" [40].

4.2 Identifying Special Events

Some aspects of musical structure in touch-screen improvisations are visually apparent in graphical gesture-scores, but it is also possible to automatically discriminate between different styles of touch-screen performance and identify transitions between multiple sections. To analyse such features, the performance of all performers in an ensemble must be considered at once, and our time series of gestural classifications are an ideal format for exploring these ensemble interactions. The method used in our Metatone Classifier software is that of transition matrix analysis [42], where performers' changes between gestures are summarised in matrices allowing simultaneous analysis of the whole ensemble over varying windows of time, or over a whole performance. In Metatone Classifier, transition matrices of gesture groups

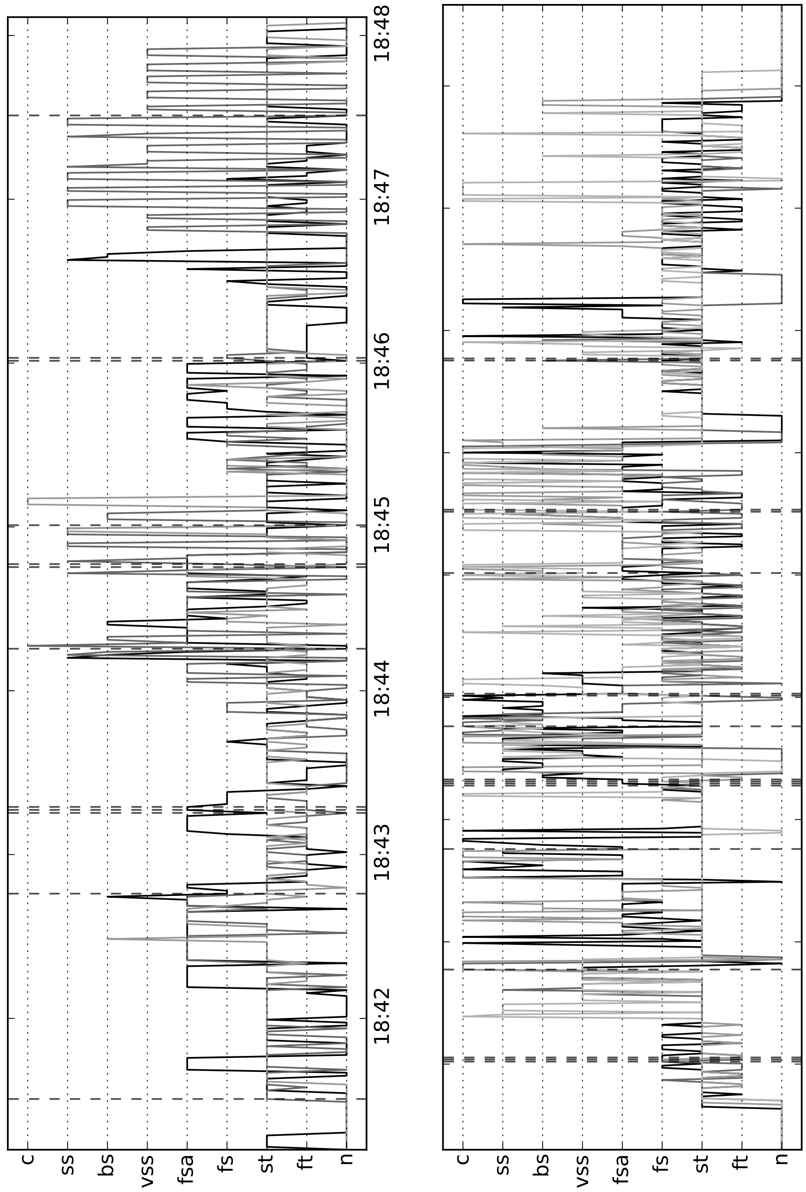

Fig. 5 Automatically generated gesture-scores for two Ensemble Metatone performances in 2014. Each line represents a single players' gestural classifications sampled once per second. The dashed vertical lines represent moments when a new-idea message was triggered by the Metatone Classifier Server. The left score was a trio improvisation by Ensemble Metatone on 2014-08-14 and the right score is of a quartet improvisation on 2015-05-07 by a group of research participants. See Table 2 for definitions of the gestures given at each level of the y-axes.

are calculated over 15-second windows, each second during a performance, a value determined through a process of trial and error. Once a transition matrix has been produced, several measures can be applied to them with little computational cost, an important factor in a real-time application.

Out of several experimental tests on transition matrices, our *flux* measure [28] has proven to be extremely promising. Flux is a measure of the rate of gestural change calculated by dividing the number of transitions from one gesture to another by the self-transitions where a gesture is followed by itself. The flux of a transition matrix has a maximum of one, when no gesture follows itself, and a minimum of zero, when each member of the ensemble stays on the same gesture for the whole window of calculation. When applied to a window of calculation that slides over a whole performance, peaks in flux tend to match moments where the ensemble shifts to a new section. In Metatone Classifier, we have implemented a strategy that tracks sudden increases in flux to identify these new ideas.

When our software identifies one of these events during a performance, an OSC message is sent to each of the connected iPad apps. We have experimented with a number of responses in the app interfaces for these messages [28]. The apps might display a series of new notes to reward the performers, or progress through a composition of soundscapes to encourage further exploration.

As with all of our app-server interactions, new-idea messages are logged to our performance protocols and have been useful in later analysis of improvised performances. These messages are shown in Figure 5 as dashed vertical lines. Since calculations over several seconds may identify the same increase in flux, new-idea messages are often grouped together closely although our apps are designed to ignore messages that arrive more frequently than once every ten seconds.

4.3 Visualisations

To understand the structure of the improvised performances we have developed a method for translating touch and gesture protocols into animated visual representations of the performance that can accompany the audio and video recordings in the documentation of each performance. Two animations are typically produced for each research-focussed performance using the Processing [36] programming language. The first is an animated version of the gesture-scores discussed in Section 4.1, with an integrated playhead that indicates the current position in the performance.

The second animation visualises the logs of performers' touches over each performance. Our 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 after about a second 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 anima-

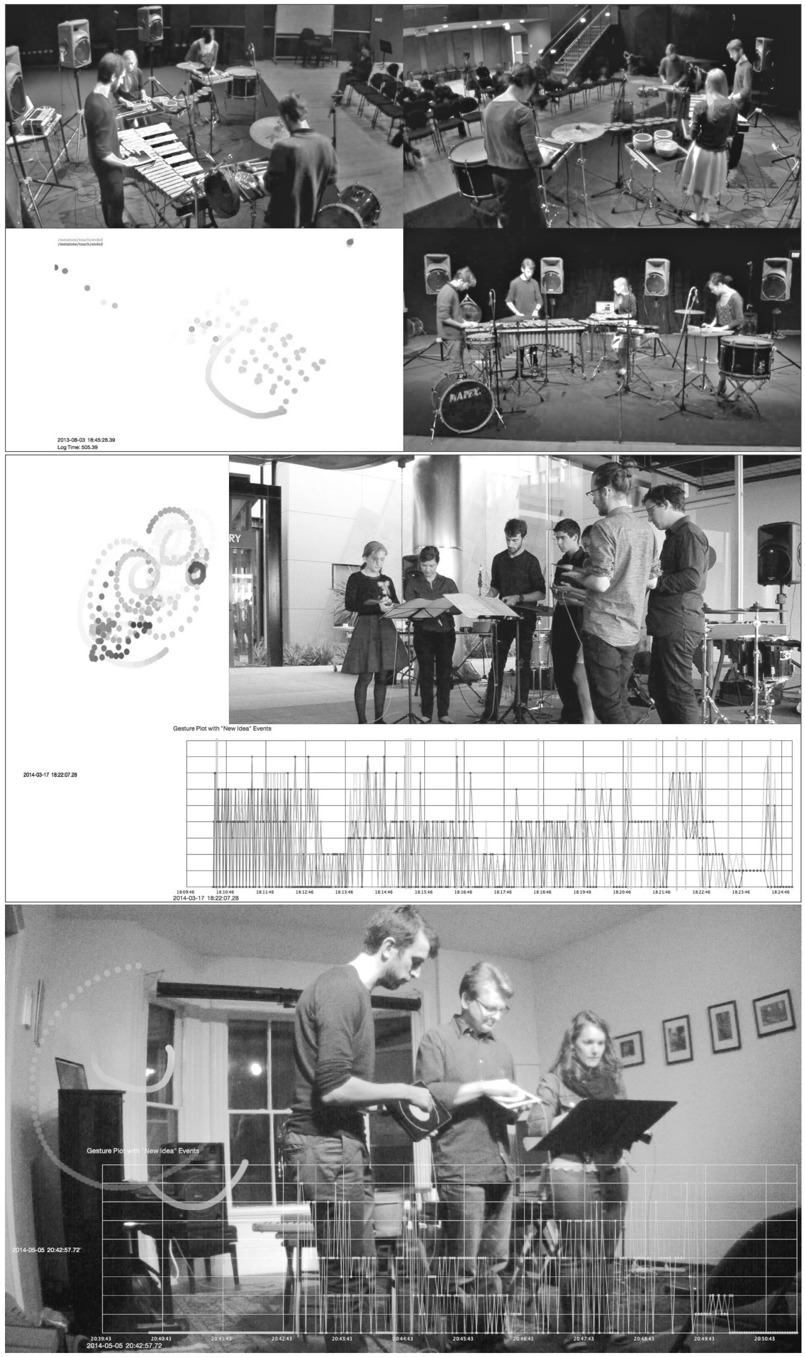

Fig. 6 Stills from hybrid visualisations of three Ensemble Metatone performances showing video, touch animations, and gesture-scores. From top to bottom, MetaLonsdale in Canberra, August 2013, Study in Bowls in Canberra, March 2014, and Touch and Tone in Boston, May 2014.

tion 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, when they form sections, or when they experiment with a new musical idea. From the researcher's point of view, the animation also gives a "performer's perspective" on touch interaction, allowing us to connect patterns of touches with musical gestures that the performers discuss after rehearsals.

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 6 shows stills from hybrid visualisations of three different performances. Each shows an alternative arrangement of visualisations from our performance protocols and multiple camera-angles of video. Such videos have been used within our research group to study the performances of Ensemble Metatone and other groups that have performed with our iPad apps. While the video component of these hybrid visualisations allows us to recall the context of the concert and to observe 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 we can see performance interactions across time, and with the touch visualisation we can see across the performers' touch spaces. In this way, the hybrid visualisation becomes a kind of "Seeing Space" [43] for improvised touch performances where multiple dimensions of the performance can be examined simultaneously. This new and useful representation of performances is only possible because of the performance-logging protocol that we have designed.

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 for the audience to view or even superimposed on the performers' iPad touch screens. A prototype touch visualisation has been developed from our software and has been used as a backdrop projection in some performances, but the implications and affordances of this addition to the performance have not yet been fully explored.

5 Conclusions

In this chapter, we have presented our protocol for logging free-improvised touch-screen musical performances which has been implemented in our Metatone Classifier software and used to record more than 100 performances by Ensemble Metatone and other groups. Our protocol records all touch interactions during each performance as well as network interactions between the performers' apps and the server software. Our protocol is a much more appropriate record of our performances than conventional systems, such as MIDI, because each touch-interaction maps directly to sound, and because network interactions are key to our ensemble improvisations. Our 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 our archive are also new-media objects satisfying Manovich's principles for new media and can be transcoded into multiple alternative representations of the performance. Such transcoding can lead to increased understanding of touch-screen improvisation and to the creation of derivative artworks. The logs themselves, as well as derivative objects, could allow improvised musical interactions to be included in spaces for curating, cataloguing, and exhibiting digital art and interaction.

In this chapter we have also presented visualisation techniques for transforming logs into gestural time series, graphical gesture-scores, and animations that afford the viewer a new performer-centric perspective on the touch improvisations. Animated visualisations have been crucial in understanding the nature of touch-screen improvisation. These have allowed the touches of all performers to be viewed simultaneously and have clearly shown the vocabulary of gestures that the ensemble explores throughout a performance. The time series of gestures have allowed us to perform statistical investigations on the performances to uncover methods for automatically segmenting performances by identifying moments of gestural change. Gesture-scores created by plotting these time series give an overview of a whole improvised performance reminiscent of graphical scores. These images allow multiple improvised performances to be directly compared at a glance and could even be used as the stimulus for future performances.

We emphasise that our protocol and our visualisation techniques have not just been used to archive touch-screen performances but also to develop an ongoing artistic practice. This has included new Metatone apps that react to gestural classifications and new-idea messages sent by the Metatone Classifier server in real-time during performances, live visualisations of touch-screen movements, and even compositions based on our vocabulary of touch-screen gestures. Such techniques could assist the broader musical community to curate and understand the long-term development of musical interactions both improvised and composed. While our touch-screen protocols allow us to archive, preserve and curate free-improvised performances, they also suggest new forms of exploratory iPad performances of the future.

References

1. Apple Inc.: Event handling guide for iOS. Apple Developer Documentation, published online (2015). URL `https://developer.apple.com/library/`
2. Bailey, D.: Improvisation: It's Nature and Practice in Music. Da Capo, Cambridge, MA (1993)
3. Blades, J.: Percussion Instruments and their History. The Bold Strummer Ltd., Westport, CT, USA (1992)
4. Borgo, D.: Sync or swarm: Musical improvisation and the complex dynamics of group creativity. In: K. Futatsugi, J.P. Jouannaud, J. Meseguer (eds.) Algebra, Meaning, and Computation, *Lecture Notes in Computer Science*, vol. 4060, pp. 1–24. Springer Berlin Heidelberg, Berlin, Heidelberg (2006). doi:10.1007/11780274_1
5. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001). doi:10.1023/A:1010933404324
6. Burtner, M.: Syntax of Snow for bells and amplified snow. Published digitally by the author (2011). URL `http://matthewburtner.com/syntax-of-snow-2/`

7. Cahn, W.L.: Creative Music Making. Routledge, New York, NY (2005)
8. Cook, G.: Teaching Percussion. Schirmer Books, New York, NY (1997)
9. Davies, S.: The ontology of musical works and the authenticity of their performances. Noûs **25**(1), 21–41 (1991). doi:10.2307/2216091
10. Davies, S.: Themes in the Philosophy of Music. Oxford University Press, Oxford, UK (2005)
11. Evens, A.: Sound Ideas: Music, Machines, and Experience, *Theory Out of Bounds*, vol. 27. University of Minnesota Press, Minneapolis, MN (2005)
12. Feldman, M.: The King of Denmark for Solo Percussionist. Edition Peters, Glendale, NY (1965)
13. Fette, I., Melnikov, A.: The WebSocket Protocol. RFC 6455 (Proposed Standard) (2011). doi:10.17487/RFC6455
14. Fowler, C.B.: The museum of music: A history of mechanical instruments. Music Educators Journal **54**(2), 45–49 (1967). doi:10.2307/3391092
15. Freed, A., Schmeder, A.: Features and future of Open Sound Control version 1.1 for NIME. In: Proceedings of the International Conference on New Interfaces for Musical Expression, pp. 116–120. Carnegie Mellon University, Pittsburgh, PA, United States (2009). URL `http://www.nime.org/proceedings/2009/nime2009_116.pdf`
16. Hope, C., Vickery, L.: The Decibel ScorePlayer - a digital tool for reading graphic notation. In: International Conference on Technologies for Music Notation and Representation, TENOR 2015. IRCAM, Paris, France (2015)
17. Kaltenbrunner, M., Bovermann, T., Bencina, R., Costanza, E.: TUIO - a protocol for table based tangible user interfaces. In: Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation. Springer-Verlag, Ile de Berder, France (2005)
18. Kania, A.: Making tracks: The ontology of rock music. The Journal of Aesthetics and Art Criticism **64**(4), 401–414 (2006). doi:10.1111/j.1540-594X.2006.00219.x
19. Kapur, A.: A history of robotic musical instruments. In: Proceedings of the International Computer Music Conference. International Computer Music Association, San Francisco, CA (2005). URL `http://hdl.handle.net/2027/spo.bbp2372.2005.162`
20. Lazzaro, J., Wawrzynek, J.: An RTP payload format for MIDI. In: Audio Engineering Society Convention 117. Audio Engineering Society, New York, NY (2004)
21. MacDonald, C.: Scoring the work: Documenting practice and performance in variable media art. Leonardo **42**(1), 59–63 (2009). doi:10.1162/leon.2009.42.1.59
22. Manovich, L.: The Language of New Media. MIT Press, Cambridge, MA, USA (2002)
23. Martin, C.: Creating mobile computer music for percussionists: Snow Music. In: M. Hitchcock, J. Taylor (eds.) Interactive: Australasian Computer Music Conference 2012 Conference Proceedings. Australasian Computer Music Association, The Basin, Australia (2012)
24. Martin, C.: Making improvised music for iPad and percussion with Ensemble Metatone. In: Proceedings of the Australasian Computer Music Conference '14. Australasian Computer Music Association, P.O. Box 284, Fitzroy, Victoria, Australia (2014). URL `http://acma.asn.au/media/2014/01/ACMC-2014r1.pdf`
25. Martin, C., Gardner, H.: That syncing feeling: Networked strategies for enabling ensemble creativity in iPad musicians. In: Proceedings of CreateWorld '15. Griffith University, Brisbane, Australia (2015)
26. Martin, C., Gardner, H., Swift, B.: Exploring percussive gesture on iPads with Ensemble Metatone. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, pp. 1025–1028. ACM, New York, NY, USA (2014). doi:10.1145/2556288.2557226
27. Martin, C., Gardner, H., Swift, B.: MetaTravels and MetaLonsdale: iPad apps for percussive improvisation. In: CHI '14 Extended Abstracts on Human Factors in Computing Systems, CHI EA '14, pp. 547–550. ACM, New York, NY, USA (2014). doi:10.1145/2559206.2574805
28. Martin, C., Gardner, H., Swift, B.: Tracking ensemble performance on touch-screens with gesture classification and transition matrices. In: E. Berdahl, J. Allison (eds.) Proceedings of the International Conference on New Interfaces for Musical Expression, pp. 359–364. Louisiana State University, Baton Rouge, Louisiana, USA (2015). URL `http://www.nime.org/proceedings/2015/nime2015_242.pdf`

29. Martin, C., Hopgood, C., Griffiths, J., Lam, Y.: Ensemble Metatone. Digital Audio Album available on Bandcamp (2014). URL `https://charlesmartin.bandcamp.com/album/ensemble-metatone/`
30. Mazzola, G., Cherlin, P.B.: Flow, Gesture, and Spaces in Free Jazz. Computational Music Science. Springer, Berlin, Germany (2009)
31. Mey, T.D.: Musique de Tables for three percussionists. Percussion Music Europe, Tienen, Belgium (1987)
32. MIDI Manufacturers Association: The complete MIDI 1.0 detailed specification. MIDI Manufacturers Association, Los Angeles, CA (1996)
33. Nunn, T.: Wisdom of the Impulse: On the Nature of Musical Free Improvisation. self-published (1998)
34. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
35. Pressing, J.: Improvisation: Methods and models. In: J. Sloboda (ed.) Generative Processes in Music. Oxford University Press, Oxford, UK (1988)
36. Reas, C., Fry, B.: Processing: programming for the media arts. AI & SOCIETY **20**(4), 526–538 (2006). doi:10.1007/s00146-006-0050-9
37. Rinehart, R.: The media art notation system: Documenting and preserving digital/media art. Leonardo **40**(2), 181–187 (2007). doi:10.1162/leon.2007.40.2.181
38. Schick, S.: The Percussionist's Art: Same Bed, Different Dreams. University of Rochester Press, Rochester, NY, USA (2006)
39. Schmeder, A., Freed, A., Wessel, D.: Best practices for Open Sound Control. In: Proceedings of the Linux Audio Conference, vol. 10 (2010). URL `http://lac.linuxaudio.org/2010/papers/37.pdf`
40. Smith, S.S., Goldstein, T.: Inner-views. Perspectives of New Music **36**(2), 187–199 (1998). doi:10.2307/833528
41. Stenström, H.: Free Ensemble Improvisation. No. 13 in ArtMonitor. Konstnärliga fakultetskansliet, University of Gothenburg, Gothenburg, Sweden (2009). URL `http://hdl.handle.net/2077/20293`
42. Swift, B., Sorensen, A., Martin, M., Gardner, H.J.: Coding livecoding. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, pp. 1021–1024. ACM, New York, NY, USA (2014). doi:10.1145/2556288.2557049
43. Victor, B.: Seeing Spaces. Available on the author's website (2014). URL `http://worrydream.com/SeeingSpaces/`
44. Waisvisz, M.: THE HANDS, a set of remote MIDI-controllers. In: Proceedings of the International Computer Music Conference, ICMC '85, pp. 313–318. International Computer Music Association, San Francisco, CA (1985). URL `http://hdl.handle.net/2027/spo.bbp2372.1985.049`
45. Wobbrock, J.O., Wilson, A.D., Li, Y.: Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes. In: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST '07, pp. 159–168. ACM, New York, NY, USA (2007). doi:10.1145/1294211.1294238
46. Xenakis, I.: Psappha for solo multi-percussionist. Universal Music Publishing – Durand Salabert Eschig, Paris, France (1975)