

A Web Application for Audience Participation in Live Music Performance: The Open Symphony Use Case

Leshao Zhang
Queen Mary University of
London
Mile End Road
London E1 4NS
leshao.zhang@qmul.ac.uk

Yongmeng Wu
Queen Mary University of
London
Mile End Road
London E1 4NS
yongmeng.wu@qmul.ac.uk

Mathieu Barthet
Queen Mary University of
London
Mile End Road
London E1 4NS
m.barthet@qmul.ac.uk

ABSTRACT

This paper presents a web-based application enabling audiences to collaboratively contribute to the creative process during live music performances. The system aims at enhancing audience engagement and creating new forms of live music experiences. Interaction between audience and performers is made possible through a client/server architecture enabling bidirectional communication of creative data. Audience members can vote for pre-determined musical attributes using a smartphone-friendly and cross-platform web application. The system gathers audience members' votes and provide feedback through visualisations that can be tailored for specific needs. In order to support multiple performers and large audiences, automatic audience-to-performer groupings are handled by the application. The framework was applied to support live interactive musical improvisations where creative roles are shared amongst audience and performers (Open Symphony). Qualitative analyses of user surveys highlighted very positive feedback related to themes such as engagement and creativity and also identified further design challenges around audience sense of control and latency.

Author Keywords

participatory art, audience performer interaction, collaborative music making, live music performance, music visualisation, vote, meta conductor, audience response

ACM Classification

H.5.5 [Information Interfaces and Presentation] Sound and Music Computing, H.5.2 [Information Interfaces and Presentation] User Interfaces

1. INTRODUCTION

In most current Western performing arts spectators are generally relegated to “receiver” status [11], having little impact on the creative process involved in performances except indirectly through the effects of appraisal reactions. Since the 1920s, attempts and experiments in theatre, dance and contemporary art, for example “The Living Stage” improvisational theatre, Adrian Piper’s collaborative performances

called “Funk Lessons”, and Dada artists’ “Mock Trial” organised by Breton [9, 17, 3], have explored new ways to encourage audience participation in performances by becoming actively involved and by interacting [7]. Often such approaches are drawn from non-Western traditions of participatory music performance where everyone present actively participates in the performance [25] as participants and potential participants “performing different roles” which often take place in rites and initiation ceremonies [1]. The form of the music in these situations is within predictable structures but open to a lot of improvisation and its aim is focussed on social bonding. Similarly, composers have encouraged performers to make their own interpretation and judgement on the form of the performed piece and make live decisions on how to play the piece. For example, Luciano Berio’s *Sequenza for solo flute* offers performers the freedom to choose how long to hold a note within a predetermined framework, Lawrence D. “Butch” Morris’ *Conduction* system proposes an exchange between composer/conductor and instrumentalists that provides the immediate possibility of altering the musical attributes of a performance [16]. Until recently, most approaches for audience participation relied on gestural and verbal communication and low technology tools. This may hinder in-situ interactions with a large number of participants, constrain the creative possibilities and impose time constraints. With advances in human computer interaction a number of digital techniques for technologically mediated audience participation emerged, exploiting various platforms and sensors, from lightsticks [8], to mobile devices [22] [27] [5], and tangible interfaces [2].

In this paper we propose a web application that enables audiences to co-create musical performances with performers in an interactive way through voting and visualisation. The system is scalable to large audiences and enables quasi real time audience-performer interactions. It makes several novel contributions to participatory live music performance systems: (i) interactions are mediated through a voting system that can be operated by audiences from a web client application on their mobile devices, (ii) audience-to-performer grouping assignments are automated based on client connection, (iii) audience members are attributed unique digital identifiers which can be tracked for personalised feedback and analysis purposes, (iv) quasi real-time visualisations are generated by a visual client following audience-driven creative data. Following a user-centred design approach an application of the system was developed to support an interactive musical system, Open Symphony, for audience-directed improvisations built on *playing modes* [10]. Surveys conducted with both performer and audience participants showed that the system transforms the traditional concert experience e.g. through changes in creative ownership as well as the performance practice, providing a challenging



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Copyright remains with the author(s).

NIME'16, July 11-15, 2016, Griffith University, Brisbane, Australia.

ground for performers to test new creative forms. Video¹ and photo² examples of Open Symphony can be found at the links provided below. Besides audience creative participation applications, the web application can also be adapted to collect audience feedback in response to the music being played during performances; this can have direct applications for music perception research, enabling experiments in real context scenarios, where ratings can be collected in parallel for a large number of listeners.

The remainder of the paper is organised as follows: Section 2 describes the background and related interactive systems for audience participation. Section 3 details the design of the proposed web application. Section 4 illustrates the Open Symphony live music performance use case and its evaluation. In Section 5 we present our conclusions and future perspectives.

2. RELATED WORKS

Recent research on audience experience [18] shows that two types of interactions may significantly improve audience satisfaction in live music performance: (i) interactions between performers and the audience, (ii) interactions between audience members. However, large audiences have only limited ways to participate in a performance on account of both logistical and cultural constraints [27]. In order to improve audience experience, many systems mediated by advanced multimedia technologies were developed for audience-performer or audience-audience interactions. Sgouros presented a framework which allows multiple users to participate in a performance either as performers by controlling some instruments, or as spectators by commenting on the event and by sending messages to other participants [21]. Maynes-Aminzade *et al.* proposed techniques adapted for music or theatre settings: e.g. tracking the audience’s movement to control an on screen game, an interactive ball which can trigger sounds and visual elements, a tracking laser point for large audience to collaboratively play on screen games [15]. Idiosyncratic digital music interfaces (DMIs) targeting audience participation have also been proposed. For example the Glimmer system [8] allows large group of audience to interact with performers by turning on and off light sticks. The system summarises the instructions from the audience using computer vision and generates sound according to these instructions. The disposable wearable motion detection device developed by Feldmeier & Paradiso enabled dancers to collaboratively produce music and control lights [6]. Smartphones have also been widely used as DMIs for audience participation probably linked to their ubiquity, the advances in their processing and sensing capabilities, and the lowered cost in design and development induced from using an already existing platform. Another advantage is that it may be easier for audiences to interact using their own familiar device. The interactive system Sound Bounce [4] generates sounds of a bouncing ball based on control gestures measured using smartphones’ motion sensors. MassMobile [27] is a client/server system for large audience participation in live performance which can be used to select lighting effects and sound sequences. The interactive system Mood Conductor [5] allows audience members to communicate emotional directions to performers to influence their musical improvisations/interpretations while they play. Mood Conductor can be divided into three parts: i) a mobile application driven by HTML5 & JavaScript which acts as interaction data collector; ii) a server side engine powered by

¹Open Symphony video: http://bit.ly/os_visvideo

²Open Symphony photos: http://bit.ly/os_photos

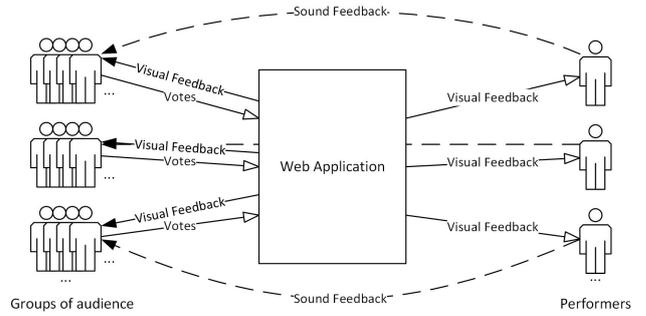


Figure 1: Multimodal Audience-Performer Interaction for Participatory Live Music Performance

CherryPy which acts as interaction data processor; iii) a visualisation client written in Python which acts as feedback generator. The architecture of the proposed web application is similar to that of Mood Conductor but extends several of its principles to allow bidirectional client/server communication, audience tracking via data identifiers, and automatic audience grouping.

3. WEB APPLICATION FRAMEWORK FOR AUDIENCE PARTICIPATION

In this section, we propose a web application for audience participation in live music performance. The application allows audiences to vote for pre-defined musical attributes and presents feedback to audiences and performers using data visualisation.

3.1 Audience Participation through Voting

In the case of a large audience, it is difficult for performers to interact with individual audience members. A general way to engage large audiences is to implement a voting mechanism as part of the performance. For example, Glimmer [8] which we introduced in Section 2 can be seen as a voting system, up to a certain extent; the light sticks enable individual audience members to vote using their hands. Mood Conductor [5] also implements a certain type of voting system as it aggregates multiple users’ mood votes and outputs a weighted-average of the votes. Our web application enables audience voting for musical attributes in live music performance. For example, by choosing different musical attribute values such as louder/quieter (intensity), higher/lower (pitch), slower/faster (tempo), or different pre-composed music sequences, audience members as a group can send their desired choices to performers thus participating in the creative aspects of the performance, in addition to receiving musical messages from the performers. With such system in place, performer should however keep the freedom to choose to fulfil or not the expectation of the audience, in trade of increased audience participation or better musical quality. We aim to make the system a communication tool for audience and performers where performers keep some degrees of freedom instead of making it a precise score-generating conduction system.

While massMobile [27] only presents feedback on the audience mobile client, our web application presents feedback both on the audience and visual clients for individual as well as collective feedback for audience and performers. Compared to Mood Conductor, audience members are divided into groups and assigned to performers in this application while Mood Conductor regards all the audience as one group. The application enables to generate different results for dif-

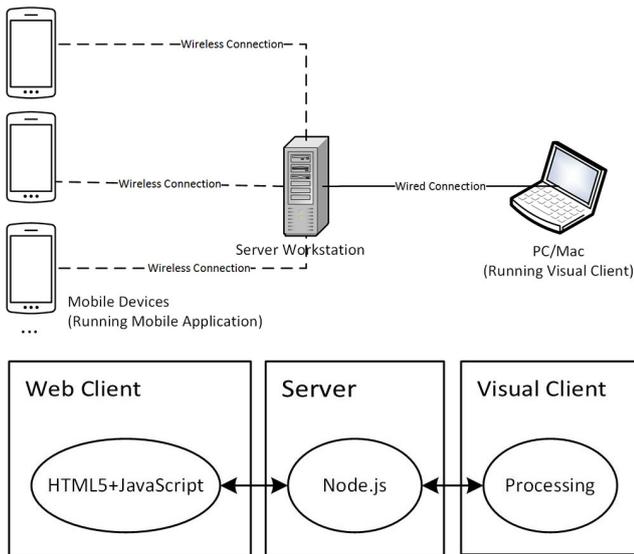


Figure 2: General Components of the Web Application

ferent performers associated to groups of audience members while Mood Conductor only presents one result for all the audience and the performers.

Audience members in each group “compete” for the selected musical attributes and groups of audiences collaborate with each other and act as a meta composer/conductor giving instructions to performers (Figure 1).

3.2 Use of Visualisation

Visualisation provides viewers an ability to comprehend huge amounts of data (“Graphics reveal data.”) [26]. Excellent graphics can communicate complex idea in a clarified, precise and efficient way. As mentioned in [24], visualisation can function as a cognitive tool and support decision making. The aim of our visual client is to simplify the musical creative data and help viewers efficiently interpret essential information. The design of the visualisation inherently depends on the specific application context (see Section 4 for an example of visualisation displaying audience-driven symbolic scores).

3.3 General Architecture

The web application consists of an audience user client which can run on mobile devices with modern web browsers, a visual client to present visual feedback for both audience and performers, and a server component to provide a bidirectional communication channel between the audience user client and the visual client and log audience votes (Figure 2). By implementing platform-independent techniques, the application was made cross-platform. The clients and server are able to communicate with each other through wired or wireless Internet connections (3G/4G or WIFI).

3.3.1 Audience User Client

The audience user client is built with HTML5 which is light weight and does not require to be installed on the device or any special configuration. Audiences can access the client using a simple URL (or URL shortener, or QR code) in the web browser. The audience user client implements three main functions for audience participation: (i) group assignment, (ii) voting, and (iii) group vote statistic. When the client is opened the first time, it is registered on the server

and gets back a unique identifier (ID). Cookies are used to store this ID and ensure the consistency of the service. Audience can use the audience user client to send votes to the server and get quasi real time feedback of the group vote statistics on their device.

3.3.2 Server

The server was developed using the cross-platform environment Node.js [23] which has an event-driven architecture capable of asynchronous I/O, optimising throughput and scalability. This is well suited for our audience participation musical application which requires real time communication of potentially large scale data (large audience, frequent votes). As described in [13], Node.js presents higher performance than e.g. CherryPy for high concurrency, low CPU consumption applications like chat and voting systems.

The server component of the web application provides five layers of services: (1) router service, (2) interactive application programming interface (API) service, (3) data storage service, (4) grouping service, and (5) log service. The router service provides the audience access to the client by URLs. The interactive API service is in charge of a set of APIs for the bidirectional communication between the audience user clients and the server, and between the visual client and the server. These APIs are introduced in the next section. The data storage service stores audiences’ digital identifiers, votes and configurations of performance such as the musical attributes to vote for, the number of audience groups, the performance duration, etc. The grouping service divides audience user clients to groups according to the configuration of audience groups number. The log service records all the events on the server including the change of configuration, the details of every votes (timing, value, voter, etc.), as well as errors in plain text files for further analysis.

3.3.3 APIs

Our web application implements six types of APIs: (1) configuration API, (2) registration API, (3) vote API, (4) heartbeat API, (5) statistic API, and (6) reset API. The configuration API allows to configure some general characteristics of the performance based on artistic director/composer’s requirements, such as performance duration, number of audience groups, musical attributes to vote for, etc. The registration API works when audience user clients are connecting to the server. As the server accepts a request from a new audience user client, the registration API generates a unique digit for the client and sends it back to the client. This unique digit is an important identifier (ID) for voting and grouping of audience user clients. The vote API provides a portal for audience votes for musical attributes through audience user clients. Audience user clients should call the heartbeat API frequently after obtaining their registered IDs in order to let the server know that they are online and to get the vote statistics of the assigned groups. The statistic API lets the visual clients get all the vote statistical analysis results to generate visual feedback. The reset API can be called before every performance to reset all the parameters from previous performances. These APIs are all accessible with uniform resource identifiers (URIs) as the web application is built according to the RESTful architecture. With such architecture, the web application uses the JSON format as a data carrier through the HTTP protocol for communication between client and server.

3.3.4 Visual Client

The visual client was developed with the Processing language [19] and can be set on any PC or Mac platforms with the Java Runtime Environment. The visual client is respon-



Figure 3: Open Symphony Music Playing Modes

sible for presenting visual feedback to audience and performers on the one hand, and setting performance configurations on the other hand. The visual feedback is generated in quasi real time according to the vote statistics fetched from the server (using the statistic API). Through key commands implemented within the visual client, changes wanted by artistic directors or composers can be operated at any time (e.g. adding another musical attribute to vote for). The changes are sent to the server which relay them and affect the audience user clients seamlessly.

4. OPEN SYMPHONY USE CASE

Open Symphony, a use case of the proposed web application, was designed and developed through an iterative participatory design approach [12] in collaboration with musician Kate Hayes (Guildhall School of Music & Drama graduate) who acted as music director for the project.

4.1 User-Centred Design

Our design approach took into account the needs of the music director, performers and audiences throughout regular team meetings and pilot experiments over the course of five months. Open Symphony live music performances target audience participants interested in new interactive musical forms and performers willing to improvise and to be guided by the audience using digital technology mediated instructions. Based on previous interactive music experiments conducted by the music director, it was deemed important to create a system where performers would not feel completely controlled by the audience with no degrees of freedom for musical expression and skilled interpretation. For audiences, it was judged important for the interaction to be easy to understand, without requiring specific musical skills. In order to provide a creative balance between performers and audience matching these requirements, a set of playing modes was established as the performance attributes the audience could vote for.

4.2 Playing Modes

Open Symphony includes the five following music playing modes: (1) ‘Drone’ which refers to single sustained notes, (2) ‘Two-note’ which refers to two notes played alternatively, (3) ‘Motif’ which refers to a small set of notes forming a melody, (4) ‘Improvisation’ which corresponds to free improvisation, and (5) ‘Silence’ which implies not to play. Figure 3 presents the graphic symbols associated to these playing modes. The symbols were designed based on three visualisation properties: pattern, color, and motion, inspired by physiological studies of how visual properties are processed [14]. As the Drone mode corresponds to sustained notes its pattern was designed as a continuous wave. The Two-note mode is drawn with two large visual parts indicating that it is composed of two elements. The Motif mode is designed with repeated sections of a wave as it is composed of several notes. The Improvisation mode is represented by two twisted waves with no obvious patterns, expressing the freedom of improvising notes. The Silence mode is represented similarly to a minim rest in traditional music notation.

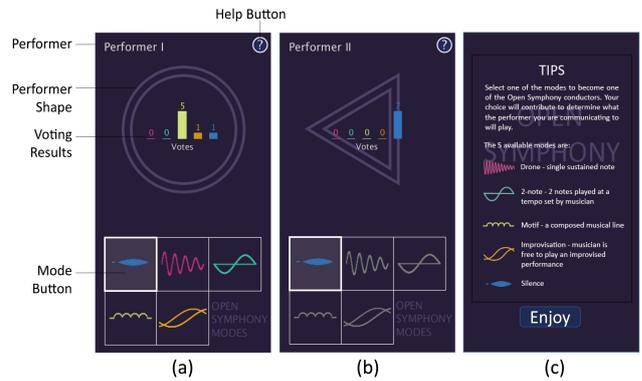


Figure 4: Graphical User Interface of the Open Symphony Web Client

4.3 Open Symphony Audience Client

The Open Symphony audience client has been designed for use on smartphones. Figure 4 presents three screen examples of the graphical user interface (GUI) of the Open Symphony audience client. In the first example (Figure 4a), the bar chart in the middle of the screen shows the current voting result for a specific audience’s group. In this instance, the user has been assigned to the first performer (Performer I) and the musical mode selected by the group is the Motif mode (highest number of votes: 5 votes). Buttons to select the five playing modes are located at the bottom of the GUI. When playing modes are not available in the composition (Figure 4b), the corresponding buttons are disabled and greyed out. Help instructions (Figure 4c) can be accessed by users at any time through a dedicated help button.

4.4 Open Symphony Visual Client

The visual client generates a graphical score for performers and audience feedback. Figure 5 presents an example of such visualisation. The designed score notations build on features of traditional music where time based events are laid out from left to right as well as on contemporary graphic notations [20]. Each performer has a timeline showing graphical symbols representing playing modes (four performers in Figure 5), and is graphically represented by a specific shape drawn at the left of the timeline (circle, triangle, squares, etc.). With this one-to-one mapping, the visual client provides direct feedback on the audience’s interaction with the aim of encouraging them to actively participate in the interaction. The visualisation provides an overview of the audience’s choices and display the elected playing modes dynamically over time.

4.5 Evaluation

We conducted an evaluation of the Open Symphony system in a real situation of interactive performance. Our method relies on the analysis of feedback from both audience and performers (the “users” of our system).

4.5.1 Participants

Four professional musicians (one clarinet player and three flute players) from Guildhall School of Music & Drama (GSMD) took part in the experiment. 13 audience participants (6 males and 7 females) were recruited through call of participation distributed online and direct invitation at Queen Mary University of London (QMUL) and GSMD and their age ranged as follows: 20-29 (5), 30-39 (6), 40-49 (1), 60-69 (1).

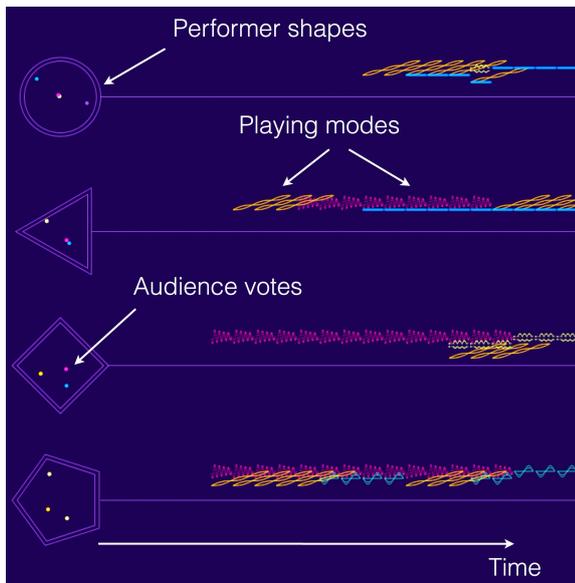


Figure 5: Graphical User Interface of the Open Symphony Visual Client

4.5.2 Procedure

The performance consisted of four pieces of four minutes with gradual increase of active modes (2, 3, 4 and 5 modes for pieces 1 to 4, respectively). This was to ensure that audience members could learn the concepts of modes and their influence progressively. Both performers and audience members were introduced to the system prior to the interactive performance.

4.5.3 Apparatus and Setting

The experiment was hosted in QMUL’s Media Arts and Technology Performance Lab. Audience and performers faced each other as in a traditional performance setting. Two screens displayed the visualisation, a large screen for the audience located behind the performers, and an HDTV screen slightly to the side for performers.

4.5.4 Survey

Different questionnaires were designed for audience and performers using a combination of opinion-scale and open-ended questions. We used online self-completion questionnaires rather than interviews to be able to collect feedback for a whole group of participants just after the performance, and to prevent the experiment from being too long for attention and availability reasons.

4.5.5 Results and Discussion

11 audience participants completed the survey (two participants were discarded being in the team of designers). Figure 6 shows the results to opinion-scale questions. A large majority of participants (9) thought that the application was easy to use being clear and intuitive; a large majority of audience participants (8) felt satisfactory about the overall experience (e.g. “it is cool and interesting to interact with performers and create different sounds”, “to connect with other people and have the performance power”; and “it draws more attention than a normal performance”). With such an active role of “conductor” audience participants felt responsible and acknowledged partial ownership in the musical creation having to make “appropriate” choices. Only two out of 11 audience participants found issues in under-

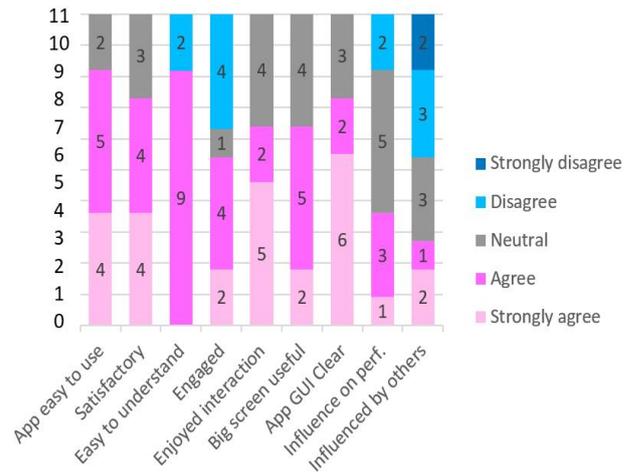


Figure 6: Open Symphony Audience Survey Results

standing or using the Open Symphony system. Results also show that most participants (6) felt highly engaged in the performance thanks to the interaction with the performers (e.g. “my choice does not always come first, but still made me feel to be engaged”, “I was paying close attention to my musician and how they were reacting to different input”, “My presence was not passive, but instead, very active”). The majority (7) very much enjoyed or enjoyed interacting with the performers during the piece, but 4 participants felt neutral about the interaction. Positive feedbacks were received both for the visual feedback and mobile app user interface: a large majority of the audience participants (7) thought that the visual feedback on the projection screen were very useful or useful and 8 participants thought that the GUI of the mobile application was very clear or clear.

A lack of control was highlighted by the results as only four participants felt that they strongly influenced or influenced the music (e.g. “it was the direct input I had towards the performance which made me feel like I had considerable influence”, “I like that you can be in contact with the performer”, “I was very happy to ‘play’ with them”, whilst seven felt they had only partial or no influence on the music (e.g. “the performer chose what was ultimately played”). With regard to within-audience interactions, three participants felt they were highly influenced by the votes of other participants, while nine had no pronounced opinion about this or felt that their choices was not influenced by that of others.

Another problem which emerged was related to latency (“There is a time delay in responding to the overall performance”, “no immediate feedback of the number of votes”, “I would aim at improving the latency.”). This is related to (i) data forwarding, due to typical broadband network delays (50-100ms), the system implementation as the visual client requires data from the server at a fixed sample rate (set up to two seconds), and (ii) control, as the selection of modes is based on the highest number of votes (plurality) which leads to individual audience member having less immediate control and perceiving a latency in the system’s response.

5. CONCLUSIONS AND FUTURE WORKS

We proposed a web application for audience participation in live music performance including a set of APIs (e.g. configuration, registration, vote, statistic) enabling flexible usage. Contrary to previous systems, our web application

allows feedback on both audience and visual clients, as well as automatic audience data tracking and assignment. The application was designed to develop novel interactive performance practices aiming at enhancing the audience experience through active engagement. The framework represents an interesting tool to investigate the relationships between audience and performers, audience and audience in a creative context, mediated by human computer interactions. Evaluation of the application in the context of an interactive performance (Open Symphony) proved the validity of the framework in terms of usability and audience engagement. Future work will aim at testing the stability of the system for larger scale performances. We will also investigate personalisation of the system for audience and performers (e.g. individual feedback), and using socket network to reduce latency and increase audience capacity. Our web application could also be adapted for measuring audience response to live music performance.

6. ACKNOWLEDGMENTS

We wish to thank creative musician Kate Hayes as well as her ensemble. This work has been partly supported by EPSRC grant EP/L019981/1, the China Scholarship Council, and Arts Council England (Sound and Music Audience Labs).

7. REFERENCES

- [1] O. Barfield. *Saving the appearances: A study in idolatry*. Wesleyan University Press, 1988.
- [2] B. Bengler and N. Bryan-Kinns. Designing collaborative musical experiences for broad audiences. In *Proceedings of the 9th ACM Conference on Creativity & Cognition*, pages 234–242. ACM, 2013.
- [3] C. Bishop. Viewers as producers. *Participations*. Ed. Claire Bishop. London: Whitechapel, pages 10–17, 2006.
- [4] L. Dahl and G. Wang. Sound Bounce: Physical Metaphors in Designing Mobile Music Performance. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 178–181, 2010.
- [5] G. Fazekas, M. Barthet, and M. B. Sandler. Novel methods in facilitating audience and performer interaction using the mood conductor framework. In *Sound, Music, and Motion*, pages 122–147. Springer, 2014.
- [6] M. Feldmeier and J. A. Paradiso. An Interactive Music Environment for Large Groups with Giveaway Wireless Motion Sensors. *Computer Music Journal*, 31(1):50–67, 2007.
- [7] J. W. Fisher. Audience participation in the eighteenth-century london theatre. *Contributions in Drama and Theatre Studies*, 101:55–70, 2003.
- [8] J. Freeman. Large audience participation, technology, and orchestral performance. In *Proceedings of the 2005 International Computer Music Conference*, pages 757–760, 2005.
- [9] S. C. Haedicke. The challenge of participation: Audiences at living stage theatre company. *Contributions in Drama and Theatre Studies*, 101:71–88, 2003.
- [10] K. Hayes, M. Barthet, Y. Wu, L. Zhang, and N. Bryan-Kinns. A participatory live music performance with the Open Symphony system. In *Proc Int. Conf. on Computer Human Interaction (Interactivity)*, 2016.
- [11] S. Kattwinkel. *Audience participation: Essays on inclusion in performance*. Number 101. Greenwood Publishing Group, 2003.
- [12] F. Kensing, J. Simonsen, and K. Bodker. Must: A method for participatory design. *Human-Computer Interaction*, 13(2):167–198, 1998.
- [13] K. Lei, Y. Ma, and Z. Tan. Performance Comparison and Evaluation of Web Development Technologies in PHP, Python, and Node.js. In *Computational Science and Engineering (CSE), 2014 IEEE 17th International Conference on*, pages 661–668, 2014.
- [14] M. Livingstone and D. Hubel. Segregation of form, color, movement, and depth: anatomy, physiology, and perception. *Science*, 240(4853):740–749, 1988.
- [15] D. Maynes-Aminzade, R. Pausch, and S. Seitz. Techniques for interactive audience participation. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*, page 15. IEEE Computer Society, 2002.
- [16] B. Morris. Conduction, <http://www.conduction.us/>, 1996.
- [17] A. Piper. Notes on funk I-IV. *Piper, Out of Order, Out of Sight*, 1, 1996.
- [18] J. Radbourne, K. Johanson, H. Glow, and T. White. The audience experience: Measuring quality in the performing arts. *International Journal of Arts Management*, pages 16–29, 2009.
- [19] C. Reas and B. Fry. *Processing: a programming handbook for visual designers and artists*, volume 6812. Mit Press, 2007.
- [20] T. Sauer. *Notations 21*. Mark Batty Pub, 2009.
- [21] N. Sgouros. Supporting audience and player interaction during interactive media performances. In *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, volume 3, pages 1367–1370, 2000.
- [22] A. Tanaka. Mobile music making. In *Proceedings of the 2004 conference on New interfaces for musical expression*, pages 154–156. Shizuoka University of Art and Culture, Hamamatsu, Japan, 2004.
- [23] S. Tilkov and S. Vinoski. Node.js: Using JavaScript to build high-performance network programs. *IEEE Internet Computing*, 14(6):80–83, 2010.
- [24] E. R. Tufte. The visual display of quantitative information, 2nd edition. *American Journal of Physics*, 53(11):400–400, 2001.
- [25] T. Turino. *Music as social life: The politics of participation*. University of Chicago Press, 2008.
- [26] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.
- [27] N. Weitzner, J. Freeman, S. Garrett, and Y.-L. Chen. massMobile – an Audience Participation Framework. In *NIME 2012 Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 92–95, 2012.