User-independent Accelerometer Gesture Recognition for Participatory Mobile Music

GERARD ROMA,
(g.roma@hud.ac.uk)

University of Huddersfield, Huddersfield, UK

ANNA XAMBÓ,
(a.xambo@qmul.ac.uk)

Queen Mary University of London, London, UK

JASON FREEMAN
(jason.freeman@gatech.edu)

Georgia Institute of Technology, Atlanta, USA

The general adoption of smartphones, and the rapid development and implementation of new web standards supporting their capabilities, have created a promising platform for participatory music. In this paper we analyze the use of accelerometer gesture recognition in this context, which brings the issue of generalizing to multiple users. We describe *Handwaving*, a system based on neural networks for real-time gesture recognition and sonification on mobile browsers. We evaluate the system using a multi-user dataset. Our results show that training with data from multiple users improves classification accuracy, supporting the use of the proposed algorithm for user-independent gesture recognition. Finally, we describe our experiences in participatory music using the system.

0 Introduction

During the last few decades, the introduction of computers in music performance has had a noticeable (and often disruptive) effect on the expectations of the audience. As noted by Keislar [1], the history of computer music has been one first of gradually abstracting the production of sound away from the body and, more recently, attempting to reconnect the body and the physical gesture to sound once again. For music where computers are involved, the audience no longer expects to see every detail of the music creation process, and it is assumed that it may as well be automated, and performers could be checking their email [2]. While the aesthetics of acousmatic music [3] have reached popular culture and some musicians prefer to perform in total darkness, some sort of interaction between performers and audience usually hints that music is being performed live.

With the ubiquity of smartphones and mobile data, the situation has become even more complex: now the audience can check their email too. Yet beyond disconnecting performers and audience, the technology offers new opportunities for creating shared experiences. In particular, the standardization of mobile technologies and the recent improvements of web standards make it easier than ever to quickly develop and deploy software for audience participation in music performances. Research on audience participation, initially an aesthetic pursuit, is now of interest to a wider community of computer-mediated music practitioners. While some research has focused on the use of smartphones for capturing and documenting music performances [4], their use for participation and interaction with the music creation process has a unique potential for engaging audiences. This includes a wide range of settings, with various degrees of participation. In the extreme case, the distinction between audience and performer can be eliminated, and a music performance can be entirely designed as an audience-driven process.

In this paper we describe *Handwaving*, a system for enhancing audience participation in music performance through mobile phone technologies. We exploit the fact that many people carry a device capable of sound synthesis and equipped with an accelerometer sensor. Our sys-

tem allows the definition of a vocabulary of gestures, which should be easy to learn by music performance audiences. Given some examples, generated by the author/s of a music piece, a machine learning model can be trained to recognize the corresponding gestures, and they can be mapped to sound synthesizers. The system is implemented using web standards, which makes it simple and quick to deploy software on audience devices in live performance settings. The training interface is also implemented as a web application, which allows a group of people to provide training examples. This collective utilization, both in the training and performance stages, can be seen as a new use case with respect to previous research on gesture recognition for mobile music. In order to explore this use case, we extend the work presented in [5] with a new dataset and additional experiments. The paper is organized as follows: Section 1 contains a brief historical account of research on mobile and participatory music. In Section 2 we review existing approaches to accelerometer gesture recognition. The proposed system is described in Section 3, and evaluated in Section 4. In Section 5 we describe some initial experiences using the system, and in Section 6 we draw some conclusions.

1 Mobile and participatory music

The idea of *mobile music* started taking shape before the introduction of smartphones, on the basis of the increasing sound capabilities of mobile phones and PDAs [6], as well as the commercialization of early tablet PCs [7]. Initial research focused on exploring the space of interaction design enabled by the different available sensors [8], as well as by social interaction enabled by ubiquity [9, 6]. While initial mobile orchestra performances also preceded smartphones [10], the standardization and ease of programming associated with them fostered the popularization of mobile orchestras following the tradition of laptop orchestras [11].
During the last few years, much research on mobile music has focused on audience participation. For example in *echobo* [12], the audience of a music performance was able to interact with a "master musician", playing along with an acoustic musician. In *massMobile* [13], the sound of the performance was centralized and users could interact with the system through a web client/server architecture. A similar centralized system with web control was implemented in *Swarmed* [14]. In *Open Symphony* [15], a group of improvising performers are directed by the audience through a voting system. The recent development of web standards, which are rapidly implemented in mobile browsers, has greatly simplified the problem of audience participation. In addition to Web Audio, many standardized capabilities and sensors are now available to web applications, such as acceleration, vibration, or location. Recent research has focused on web-based participation [16, 17]. As an example, the participatory concert of the second Web Audio Conference showcased a number of approaches for audience participation using mobiles and web standards [18, 19, 20, 21, 22].

2 Gesture recognition

Body language, and particularly hand gestures, are an important part of human and animal communication. Since the popularization of three-axis accelerometers, first in game controllers such as the Wii remote, and then in smartphones, gesture recognition has become relevant to many applications, such as mobile user interfaces. Most traditional approaches use either Dynamic Time Warping (DTW) [23] or Hidden Markov Models (HMM) [24]. It is also common to experiment with other common classifiers, such as K-Nearest Neighbors (KNN) or Support Vector Machines (SVM) [25]. These procedures usually require careful segmentation and annotation of gestures, and their evaluation is often confined to laboratory experiments. In this paper, we propose a system for gesture recognition using neural networks. Neural networks and deep learning methodologies have quickly become the mainstream method for machine learning. Like in other domains, deep learning techniques have been applied to smartphone sensors [26]. Numerous learning frameworks based on neural networks and backpropagation are now available. As an example, we use ConvnetJS[1], a Javascript library for deep learning that allows us to implement recognition in a mobile browser in real time. Unlike many systems, the data used for training the network in our system is not manually segmented, thus reducing the need of manual annotation.
In gesture recognition it is common to distinguish between *user-dependent* and *user-independent* systems [25, 27]. In the first case, the same person that will use the system is expected to train it. In the second case, the model should be able to recognize any user's gestures. In academic evaluation, user-dependent systems typically result in better performance. However, for real-world applications this adds the burden of having the user provide training data. In the field of music creation, numerous works have investigated gesture recognition [28, 29], also some works have provided generic tools for exploiting machine learning algorithms in music performances [30, 31]. Most of these systems assume a use case where the artist will train the machine learning model in an interactive setting on a personal computer. For this reason, it is rare to find reports on recognition accuracy. Hence they can be regarded as user-dependent systems. Participatory music is a different use case. In order to support collective control and understanding of the different gestures, the dictionary needs to be shared and consistent. Thus, we propose that a user-independent system is required. Also unlike previous works, our system allows mapping gestures to musical sounds directly on a mobile phone, without the need of a PC or laptop. We propose an implementation based on web standards, which makes it very easy to quickly engage casual participants in music performances as well as other settings such as installations or museums. We also provide insights on the performance of different features and different learning settings by evaluating a specific gesture dictionary. Our results show that recognition accuracy

[1] http://cs.stanford.edu/people/karpathy/convnetjs/

improves when the system is trained with gestures from multiple users.

3 Handwaving

This section describes *Handwaving*, a software protoype that supports research on mobile and participatory music through recognition and sonification of accelerometer gestures.

3.1 Repetitive Gestures

Our system is based on recognition of simple gestures, which are commonly associated to discrete events. Simple gestures are slowly making their way to mobile interaction, for example, shaking (for undoing something) is the only non-touch gesture in Apple's IOS human interface guidelines.[2] A "double twist" gesture for activating the camera was introduced in version 7.0 of Android. For music contexts, discrete events signaled by gestures can be useful, but associating specific gestures with the resulting musical events may take some time to users, especially in the context of audience participation. In addition, gestures for specific tasks must assume a "silent" (i.e. no gesture) background, while in music very diverse regimes of action vs inaction may be used. For these reasons, we opted for a general continuous recognition model, which includes a "silence" gesture class. This also means gestures are recognized in relatively short temporal windows (e.g. 2 seconds), and have no definite phases (no onset or offset). The system thus focuses on oscillatory movements, which in our experiments have been mostly limited to simple repetitive movements along each accelerometer axis. Figure 1 shows some examples.

3.2 Recognition framework

In terms of machine learning, the system is relatively straightforward. Accelerometer data consists of three coordinates, x, y, z, that represent acceleration of the phone in each dimension. All three signals are analyzed using the Short-Time Fourier Transform (STFT) and stacked to form a feature vector. The data is fed into a neural network with one hidden layer, using sigmoid activations. The hidden layer has the same number of units as the input. A final softmax layer is used to predict the gesture class. For training, we use half-window overlaps, while in the test stage, data is analyzed and input to the network for each new sample. A longer hop size can be used for saving CPU power, depending on the application.

3.3 Web application

The system is implemented using web technologies and Javascript libraries. Given the ubiquity of wireless networks and mobile broadband, this makes it very convenient to quickly prototype applications that can be executed in most recent smartphones without installing additional software. Accelerometer data is available through the DeviceMotionEvent API.[3] This API currently offers a gravity-corrected version of accelerometer data, which may be supported depending on the hardware. However the most widely supported version is "accelerationIncludingGravity", which is the raw accelerometer data. This is the call currently used in *handwaving*. Accelerometer data is thus captured in real time in a browser window. In order to train the recognition model, this data is sent to a web server and saved as a JSON file. The interface used for capturing training data consists of a simple web application that allows creating and deleting gestures, and recording examples of each class. This results in recordings of variable length which are labelled according to the gesture class. The recognition model is trained with this data using Convnetjs. This step is currently implemented as an offline task, although it could be also executed in a browser. With the amount of data used in our experiments (Section 4), a model can be trained in less than a minute with a current laptop.

The web application also allows managing sounds and mapping gestures to sounds. Sound synthesis is currently done using flockingjs[4], a library that mimics the syntax for creating synth definitions in SuperCollider, as well as a subset of its unit generators. This allows easily coding a wide variety of sounds with a low entry fee for composers and musicians familiar with SuperCollider. Synth definitions are written in the web interface in JSON format, and assigned to each gesture along with a *mapping expression*. The mapping expression defines how, in addition to the basic map that associates the gesture with the synth definition, the accelerometer axes are used directly to modify the sound in real time. A mapping expression typically consists simply of arithmetic operators. For example a mapping can associate the parameter $osc1.freq$ of the synth definition with the expression $440 * (1 + 0.01 * x)$, where x is the value for the accelerometer x-axis. Mapping expressions are evaluated as Javascript code and executed in order to update the synth in real time.

3.4 Applications

Given a model trained for a set of gestures, and the corresponding synths and mappings, a basic example web page is provided that allows using a phone as a musical instrument. This basic setup can already be used for mobile performance or audience participation by hosting the code in a web server. The audience can then just visit a web page that will download the neural network model along with the synth definitions and mappings to their mobile browser, detect gestures, and produce the corresponding sounds. More complex compositions can be coded as a sequence of web pages representing different parts of a composition (if the sound does not need to be continuous between sections), or in more involved web applications. With respect to learn-

[2] https://developer.apple.com/ios/human-interface-guidelines/interaction/gestures/

[3] https://www.w3.org/TR/orientation-event/

[4] http://flockingjs.org/

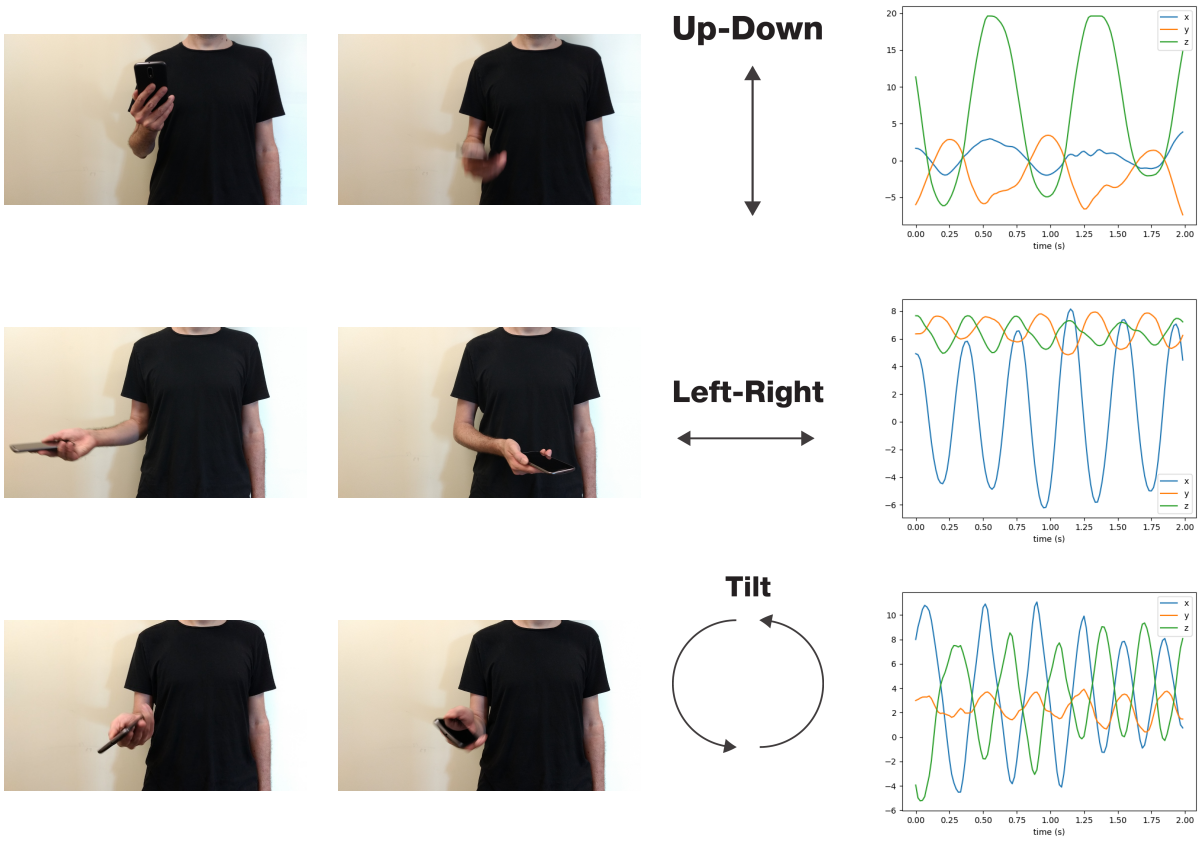

Fig. 1. Examples and accelerometer data plots for three gestures.

ing the gestures, the system allows a number of configurations depending on who trains and who uses the model. For example, a single individual could train a model for indivual use during a performance, a single individual could train the system for the audience to use, or a group of users could train it for either individual or collective usage. We expect this to have an impact on recognition performance so we analyze several configurations in section 4.3.

4 Evaluation experiments

This paper focuses on recognition of a shared dictionary of gestures for participatory music. This requires models that can be used by naive users without training the model themselves. In this section we extend the evaluation published in [5] with a new dataset that includes user identifiers, as well as further experiments to compare single and multi-user training configurations. The code and dataset for training the system can be obtained from `https://github.com/g-roma/handwaving`.

4.1 Dataset

We collected a new dataset through the web application (Section 3.3). We asked several remote participants to provide recordings of 7 classes: left / right (*lr*), up / down (*ud*), tilt (*tilt*), circles (*circ*), forward / backwards (*fb*), concave (*conc*) and convex (*conv*) (in our experiments, silence was better detected simply using a threshold on acceleration). Participants were provided an example video of each gesture class and asked to submit at least two recordings. For simple user identification, the system automatically associated each user with a device fingerprint. In order to obtain a consistent dataset, data was reduced to 9 users (including 2 of the authors) and 2 recordings per user per class. The recordings were preprocessed to remove initial silence (accelerometer values between clicking the record button and starting oscillatory movements). While the specification does not include a value for the sampling rate of accelerometer data, we tested several devices and found a consistent value of 60 Hz. The recordings were cut down to a common minimum of 500 points (8 seconds), so the same amount of data was available for each class.

4.2 Input features

We conducted a first experiment to compare the use of raw accelerometer data to the magnitude spectrum. While deep neural networks are being used to learn representations from lower level features in several domains, this often requires large amounts of training data. Initial experiments with more hidden layers showed no improvements with practical amounts of training data in our case. The experiment is the same as presented in [5], but here we used the new dataset and user-wise stratified sampling. The time series of accelerometer data was segmented using a fixed length moving window and half-window overlap. The resulting vectors were either fed directly to the neural net-

work or analyzed by an FFT module to extract the magnitude spectrum (early experiments with traditional STFT windows did not provide better results, so a rectangular window is used). The network was set to the same number of units as the input, so FFT features used half the number of units both in the visible and hidden layer. A final softmax layer was used to predict the gesture class. We compared different window sizes and the use of FFT analysis in a stratified 10-fold cross-validation setting. For each fold, the two recordings of each user were aggregated and partitioned proportionally, resulting in equal amounts of training and test data for each user in the dataset.
Figure 2 shows the result for different window sizes (8, 16, 32, 64 and 128 samples). The best accuracy was achieved when using FFT features, which are able to make better use of longer windows. This confirms the results in [5]: in this case, a more even spread of data among multiple users results in a wider difference between raw and FFT features. While in real-time usage we implemented the system to perform recognition with a hop size of one sample, it may be desirable to modify the window or hop size in order to reduce the computational cost. We found that continuous recognition performed well with most current smartphones. Older smartphones (e.g. an iPhone 4) may have both compatibility and performance issues. A capability check page is provided as part of the web application framework.

Figure 3 shows a confusion matrix for the different classes with the best set of parameters. Most confusions happen between ”up/down”, ”concave” and ”convex” gestures, all of which are dominated by the vertical axis. While we are experimenting with more complex gestures (such as alphabet letters), it is obvious that smaller number of classes will result in better recognition. At the same time, gestures which make use of different accelerometer axes will be easiest to tell apart. However, the proposed framework does not intend to explicitly model these gestures, and can in principle be trained to recognize arbitrary shapes. More complicated dictionaries may require adding more parameters to the network (e.g. more hidden layers) which would increase the computational cost and usually require more training data.

4.3 User configurations

Since our system is intended to be used by casual participants, we expected it to be more robust if trained by multiple individuals. However, it may also be interesting for single individuals to design participatory music performances. In order to provide some insight to the use of our system in both cases, we analyzed several configurations, roughly corresponding to potential use cases.

- **Multi-user**: This is the same configuration used for Experiment 4.2. Since it uses a cross-validation setting it is the most efficient use of the dataset. It represents the general case where the training does not depend on a specific user. Results were averaged across 10 folds.
- **Single-user**: In this case, we trained the system for each user using one of the recordings, and evaluated using the second recording. Results were averaged across 9 users. This represents the use case of an individual performer.
- **Cross-user**: We also tried training the system for each user (using one recording) and testing with another random user. This provides an insight (with equal amount of training and test data points) on the problems for generalizing across users.
- **One-to-Many**: In this case, we trained the system with all data from a single user and tested against the rest of data. Results were averaged across users. This represents the use case where one individual user trains the system for a participatory performance.
- **Many-to-One**: In order to test user-independent recognition, we trained the system with data from all users but one, which was used for testing. Results were averaged across users.

Table 4.3 shows the accuracy (mean and standard deviation) obtained with each of the configurations, along

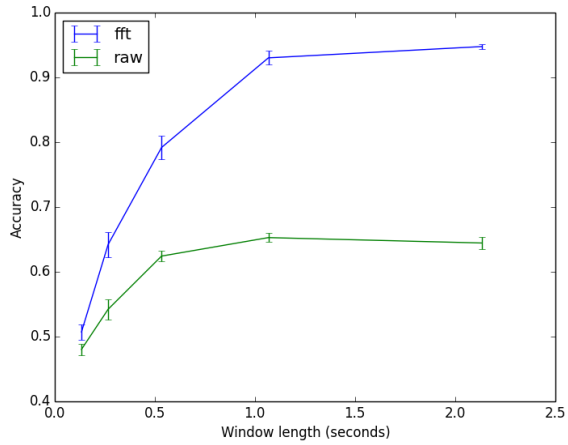

Fig. 2. Mean classification accuracy using either raw or FFT features, as a function of the window size. Error bars indicate standard deviation.

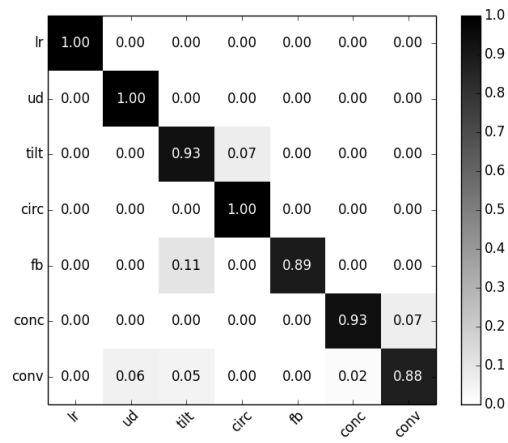

Fig. 3. Confusion matrix using 128ms window and FFT features.

with the number of training and test examples and the number of averaged models. As expected, the multi-user cross-validation setting achieves the best and most robust result, as it has been trained with more data and more models have been averaged. Using smaller amounts of training data, single-user models can also achieve good accuracy. This suggests that the system can be used both in single and multi-user settings. On the other hand, both the Cross-user and One-to-Many settings represent clearly more challenging cases. This shows that models trained by a single user will tipically not generalize so well to other users. Finally, the Many-to-One setting provides very good results, comparable to cross-validation, which shows that our system can be used for user-independent gesture recognition. In all, the results confim the intuition that robust models for participatory music should be trained by multiple users, while models for individdual performance can still be trained by the performer.

5 Experiences in participatory music

The idea of audience participation in music performance affords the possibility of a shift with respect to the traditional views of authorship, and the historical roles assigned to composer, performer and spectator of music. In the extreme case of audience participation, technology can be used to create musical experiences that are focused on the audience itself, instead of on a performer. In this sense, the development of audience participation enables a performance genre that is significantly different from both traditional setting, where the role of the audience is generally limited to signs of appreciation towards the performer/s, and the acousmatic setting, where no performer is in sight, but the audience remains passive. With respect to the expectations of current audiences, these ideas can be traced back to John Cage and to the Fluxus movement [32], however current mobile and web technologies facilitate rapid experimentation on this front, as smartphones can be trivially used both for communication and sound production. We used *Handwaving* to experiment with purely audience-led performances (in the sense that there are no performers except for the audience) in two pieces.

The first one was ”Do the Buzzer Shake” [33]. The piece was inspired by online cultural transmission through memes, while exploiting the role of imitation typically associated with gestures and gesture-based communication, music and dance. The sounds we used were based on square-wave oscillators in order to maximize the loudness of sounds produced by mobile phones.

Table 1. Results for different user configurations

Configuration	Mean Acc. (Std)	N. Train	N. Test	N. Models
Multi-user	0.94 (0.00)	882	126	10
Single-user	0.87 (0.08)	56	56	9
Cross-user	0.50 (0.13)	56	56	9
One-to-Many	0.54 (0.04)	112	896	9
Many-To-One	0.94 (0.07)	896	112	9

The piece was rehearsed several times in classroom and lab environments with groups of between 5 and 15 students, and once with a group of 100 students. It was later performed in public during the second International Conference of Live Interfaces (ICLI2016), and in the first annual Concert of Women in Music Tech held at Georgia Tech in Atlanta (GA, USA).

During the development of the piece, a structure of three parts was devised. In the first part, participants explored the use of the accelerometer and synchronization with others by trying to achieve consonance (identical phone orientations) or dissonance (different orientations). In the second part, participants explored the different gestures and their musical mappings and learnt them from each other. In the final part, synchronization was ”mandatory”: the server would count the number of participants performing each gesture, and participants performing minority gestures were ”punished” with a short vibration and a short period of silence. The duration of the silence increased progressively in order to induce a sparse ending unless a total synchronization was achieved.

During the rehearsals and public performances, it was visible that participants had no problems learning the gestures. While they were always instructed to be quiet, the disappearance of a central figure clearly created a different situation and it was very rare that participants would remain silent. However from their explorative disposition and laughters, we concluded they were engaged and enjoying the experience. Although the music was made out of drones with varying degrees of frequency stability, creating both harmonic and chaotic patterns, the atmosphere of participation had some parallels with group behavior in electronic dance music clubs, where the DJ is not necessarily the center of attention. In this sense, our research connects with previous investigations on music control by large groups [34]

A second piece, ”Hyperconnected Action Painting” was presented in the 3rd Web Audio Conference [35]. In this case, we restricted the dictionary to three gestures, associated with painting actions: ”up-down”, ”left-right” and ”splash”. Gestures performed by the audience resulted in visible traces in a collective painting that was projected. The music was based on field recordings of street jazz, with the playback speed modulated by the accelerometer. Also, in this case a ”global” music background was played through the PA in addition to the sounds from the smartphones. The painting metaphor, particularly the splash gesture, required the introduction of some constraints in the detection procedure, leaning towards discrete gestures. We hope to investigate segmentation more formally in the future. The result was also promising in terms of audience engagement, and the introduction of amplified sound improved the acoustic experience, while introducing the challenge of balancing the mixture.

6 Conclusions

With so many people carrying pocket computers with multiple sensors and sound capabilities, there is a great potential for increased audience participation in music performances. In this paper, we have proposed a framework for participatory mobile music based on mapping arbitrary accelerometer gestures to sound synthesizers on mobile phones. We have provided an multi-user dataset and shown that the system is able to learn new gestures with a few examples. We have used this system to illustrate the relevance of user-independent training for multi-user settings.
We have also described initial experiences using this system in audience-driven participatory performances. Our experiences have helped validating the system while reflecting on the potential for evolving the social organization of music performance.

7 Acknowledgements

We would like to thank all the researchers who have helped testing the system and providing training data at University of Surrey, Georgia Tech, University of Huddersfield and Queen Mary University of London. This project has received funding from the European Research Council project 725899 "FluCoMa".

8 REFERENCES

[1] D. Keislar, "A Historical View of Computer Music Technology," in *The Oxford Handbook of Computer Music*, pp. 11–43 (Oxford University Press, Oxford, UK) (2009).

[2] N. Collins, "Generative Music and Laptop Performance," *Contemporary Music Review*, vol. 22, no. 4, pp. 67–79 (2003).

[3] B. Kane, *Sound unseen: Acousmatic sound in theory and practice* (Oxford University Press, USA) (2014).

[4] L. Kennedy, M. Naaman, "Less Talk, More Rock: Automated Organization of Community-contributed Collections of Concert Videos," presented at the *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pp. 311–320 (2009).

[5] G. Roma, A. Xambó, J. Freeman, "Handwaving: Gesture Recognition for Participatory Mobile Music," presented at the *Proceedings of the 12th International Audio Mostly Conference on Augmented and Participatory Sound and Music Experiences*, AM '17, pp. 26:1–26:7 (2017).

[6] A. Tanaka, "Mobile music making," *Proceedings of the 4th International Conference on New interfaces for Musical Expression (NIME2004)*, pp. 154–156 (2004).

[7] N. Bryan-Kinns, "Daisyphone: the design and impact of a novel environment for remote group music improvisation," presented at the *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*, pp. 135–144 (2004).

[8] G. Essl, M. Rohs, "The Design Space of Sensing-Based Interaction for Mobile Music Performance," presented at the *Proceedings of the 3rd International Workshop on Pervasive Mobile Interaction* (2007).

[9] J. G. Sheridan, N. Bryan-Kinns, "Designing for performative tangible interaction," *International Journal of Arts and Technology*, vol. 1, no. 3-4, pp. 288–308 (2008).

[10] G. Wang, G. Essl, H. Penttinen, "Do mobile phones dream of electric orchestras," *Proceedings of the International Computer Music Conference (ICMC 2008)*, vol. 16, no. 10, pp. 1252–61 (2008).

[11] J. Oh, J. Herrera, N. J. Bryan, L. Dahl, G. Wang, "Evolving The Mobile Phone Orchestra," *Proceedings of the 10th International Conference on New Interfaces for Musical Expression (NIME2010)*, pp. 82–87 (2010).

[12] S. W. Lee, J. Freeman, "echobo: A mobile music instrument designed for audience to play," *Proceedings of the 13th International Conference on New Interfaces for Musical Expression (NIME2013)*, vol. 1001, pp. 42121–48109 (2013).

[13] J. Freeman, S. Xie, T. Tsuchiya, W. Shen, Y.-L. Chen, N. Weitzner, "Using massMobile, a flexible, scalable, rapid prototyping audience participation framework, in large-scale live musical performances," *Digital Creativity*, vol. 26, no. 3-4, pp. 228–244 (2015).

[14] A. Hindle, "Swarmed: Captive portals, mobile devices, and audience participation in multi-user music performance," presented at the *Proceedings of the 13th International Conference on New Interfaces for Musical Expression (NIME2013)*, pp. 174–179 (2013).

[15] Y. Wu, L. Zhang, N. Bryan-Kinns, M. Barthet, "Open symphony: Creative participation for audiences of live music performances," *IEEE MultiMedia*, vol. 24, no. 1, pp. 48–62 (2017).

[16] L. Zhang, Y. Wu, M. Barthet, "A Web Application for Audience Participation in Live Music Performance: The Open Symphony Use Case," *Proceedings of the 16th International Conference on New Interfaces for Musical Expression (NIME2016)*, vol. 16, pp. 170–175 (2016).

[17] N. Schnell, J.-p. Lambert, S. Robaszkiewicz, D. Cunin, X. Boissarie, "Collective Loops — Multimodal Interactions Through Co-located Mobile Devices and Synchronized Audiovisual Rendering Based on Web Standards," presented at the *Proceedings of the TEI '17: Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*, pp. 217–224 (2017).

[18] N. Madhavan, J. Snyder, "Constellation: A Musical Exploration of Phone-Based Audience Interaction Roles,"

Fig. 4. A moment in the ICLI 2016 performance (photo: ICLI2016 organization committee)

presented at the *Proceedings of the 2nd Web Audio Conference (WAC-2016)* (2016).

[19] S. W. Lee, A. D. de Carvalho Jr, G. Essl, "Crowd in c [loud]: Audience participation music with online dating metaphor using cloud service," presented at the *Proceedings of the 2nd Web Audio Conference (WAC-2016)* (2016).

[20] B. Houge, "Ornithological Blogpoem," presented at the *Proceedings of the 2nd Web Audio Conference (WAC-2016)* (2016).

[21] A. Bundin, "Concert for Smartphones," presented at the *Proceedings of the 2nd Web Audio Conference (WAC-2016)* (2016).

[22] W. Walker, B. Belet, "Musique Concrète Choir: An Interactive Performance Environment for Any Number of People," presented at the *Proceedings of the 2nd Web Audio Conference (WAC-2016)* (2016).

[23] J. Liu, L. Zhong, J. Wickramasuriya, V. Vasudevan, "uWave: Accelerometer-based personalized gesture recognition and its applications," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675 (2009).

[24] J. Kela, P. Korpipää, J. Mäntyjärvi, S. Kallio, G. Savino, L. Jozzo, S. Di Marca, "Accelerometer-based gesture control for a design environment," *Personal and Ubiquitous Computing*, vol. 10, no. 5, pp. 285–299 (2006).

[25] J. Wu, G. Pan, D. Zhang, G. Qi, S. Li, "Gesture recognition with a 3-d accelerometer," presented at the *Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing (UIC '09)*, pp. 25–38 (2009).

[26] C. A. Ronao, S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, vol. 59, pp. 235–244 (2016).

[27] A. Akl, S. Valaee, "Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation and compressive sensing," presented at the *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2270–2273 (2010).

[28] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, N. Rasamimanana, "Continuous realtime gesture following and recognition," presented at the *International gesture workshop*, pp. 73–84 (2009).

[29] B. Caramiaux, A. Tanaka, "Machine Learning of Musical Gestures." presented at the *Proceedings of the 13th International Conference on New Interfaces for Musical Expression (NIME2013)*, pp. 513–518 (2013).

[30] R. Fiebrink, D. Trueman, P. R. Cook, "A Meta-Instrument for Interactive, On-the-Fly Machine Learning." presented at the *Proceedings of the 9th International Conference on New Interfaces for Musical Expression (NIME2009)*, pp. 280–285 (2009).

[31] J. Bullock, A. Momeni, "ml. lib: Robust, Cross-platform, Open-source Machine Learning for Max and Pure Data," presented at the *Proceedings of the 15th international Conference on New Interfaces for Musical Expression (NIME2015)*, pp. 265–270 (2015).

[32] M. Nyman, *Experimental music: Cage and beyond*, vol. 9 (Cambridge University Press) (1999).

[33] G. Roma, A. Xambó, J. Freeman, "Do the Buzzer Shake," presented at the *Proceedings of the 3rd International Conference on Live Interfaces* (2016).

[34] M. Feldmeier, J. A. Paradiso, "An interactive music environment for large groups with giveaway wireless motion sensors," *Computer Music Journal*, vol. 31, no. 1, pp. 50–67 (2007).

[35] A. Xambó, G. Roma, "Hyperconnected Action Painting," presented at the *Proceedings of the 3rd Web Audio Conference 2017* (2017).

THE AUTHORS

Gerard Roma

Anna Xambó

Jason Freeman

Gerard Roma received his Degree in Philosophy from Universitat Aut'onoma de Barcelona (UAB) in 1997. After several years working in software development, he obtained his M.Sc (2008) and PhD (2015) in Information and Communication Technologies from Universitat Pompeu Fabra (UPF). He is currently a Research Fellow at the Centre for Research in New Music (CeReNeM), University of Huddersfield (UK)

●

Anna Xambó Ph.D. is a researcher and musician with background in CS engineering, digital humanities and digital arts, with substantial experience in both academia and industry. She is currently a postdoctoral researcher at the Centre for Digital Music, Queen Mary University of London. Her research looks into the design and evaluation of new interactive music systems for music performance in alignment with CSCW, publishing in international conferences and journals, such as NIME, ToCHI, CHI, TEI, and IwC. Her musical practice includes live coding, multichannel spatialization, tangible music, collaborative interfaces, audience participation with mobile devices, and real-time music information retrieval. Dr. Xambó works actively in the music technology and experimental electronic music scene, as a co-founder of the online music records Carpal Tunnel, co-founder of Women in Music Tech at Georgia Tech (Atlanta, GA, USA), and co-organizer of concerts.

●

Jason Freeman is a Professor of Music at Georgia Tech. His artistic practice and scholarly research focus on using technology to engage diverse audiences in collaborative, experimental, and accessible musical experiences. He also develops educational interventions in K-12, university, and MOOC environments that broaden and increase engagement in STEM disciplines through authentic integrations of music and computing. His music has been performed at Carnegie Hall, exhibited at ACM SIGGRAPH, published by Universal Edition, broadcast on public radio's Performance Today, and commissioned through support from the National Endowment for the Arts. Freeman's wide-ranging work has attracted support from sources such as the National Science Foundation, Google, and Turbulence. He has published his research in leading conferences and journals such as Computer Music Journal, Organised Sound, NIME, and ACM SIGCSE. Freeman received his B.A. in music from Yale University and his M.A. and D.M.A. in composition from Columbia University.