MAPPING MOTION TO TIMBRE: ORIENTATION, FM SYNTHESIS AND SPECTRAL FILTERING

Israel Neuman
isneuman@gmail.com

Charles Okpala
charlesokpala2000@yahoo.com

Cesar E. Bonezzi
cesar.bonezzi@gmail.com

Department of Computer Science
The University of Iowa

ABSTRACT

Motion is a time-based event situated in three-dimensional space. In the performance of most musical instruments, the musician associates physical motion with audible results. In some instruments, physical motion is used to change the timbre of the instrument. Digital techniques of analysis and re-synthesis have paved the way for the understanding of timbre as a distribution of energy among spectral frequencies in a three-dimensional space bounded by axes representing time, frequency and amplitude. In this paper, we present a musical performance system that explores the correlation between motion and sound, more specifically, timbre. A mobile device and its motion sensors function as a control element in the performance system that creates new timbres in real time.

1. INTRODUCTION

Musicians associate physical motion with audible results. Motion is a time-based event that takes place in a three-dimensional space. Terms such as displacement, direction, velocity and acceleration correspond to acoustic and psychoacoustic phenomena and are commonly used in idiomatic lexicons of musical instruments. In piano performance, displacement to the right is associated with higher frequency and to the left with lower frequency. A cellist achieves higher pitch by downwards motion of the left hand and low pitch by upwards motion. High velocity key pressing on the piano and high speed bowing on the cello will both result in a louder sound. In some instruments, physical motion is used to change the timbre of the instrument. For example, string players can change the timbre of the instrument by moving the bow towards the bridge of the instrument to the *sul ponticello* position.

The timbre of a sound is determined by the relative amplitudes of the spectral components (i.e., the partials) [2, 5]. Digital techniques of analysis and re-synthesis have paved the way for studies examining the spectral evolution of acoustic sound over time. In particular, Moorer and Grey [8] presented, in 1977, a three-dimensional understanding of timbre, in which amplitudes and frequencies of individual spectral components vary throughout the duration of a sound, predominantly in the attack and release portions. Their view was exemplified in three-dimensional graphs (Figure 1) in which the distribution of energy among the spectral components is represented in a space bounded by axes representing time, frequency and amplitude. One can easily imagine the peaks of such a graph shifting about in this three-dimensional space in response to some underlying motion, resulting in a redistribution of the energy among the spectral components and the creation of a new timbre.

In this paper, we present a musical performance system that is based on the correlation between motion and sound. The software that underlies this performance system maps readings from a mobile device's motion sensors (a gyroscope and an accelerometer) to the processing of live sound in the spectral domain. A mobile application transmits mapped values through Wi-Fi to a laptop. The latter uses the received data to control an interactive sound-processing application which is based on the fast Fourier transform (FFT). In the context of signal processing, the FFT analysis of sound enables the isolation, detection and manipulation of individual spectral components. Hence, the mobile device and its motion sensors function as a control element in a performance system that creates new timbres in real time.

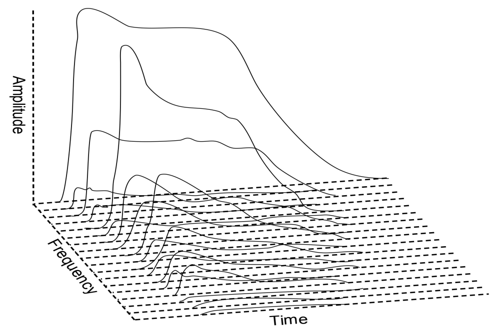

Figure 1. A representation of timbre in a three-dimensional space bounded by time, frequency and amplitude (after [8]).

We have designed a simple and effective visualization, displayed both on the mobile device and the laptop, allowing the performer to develop an intuitive control over the system and the processing of the sound of his or her own instrument. Our tests demonstrate how the system can be calibrated to fit the holding postures of different instruments and that performers have to make only minor adjustments to their

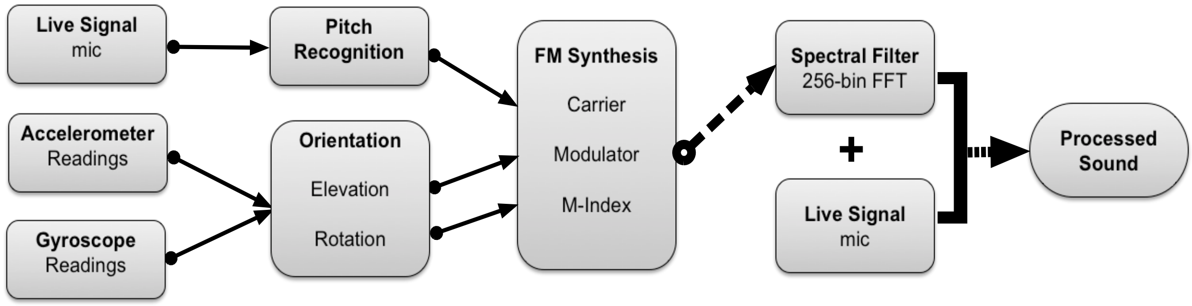

Figure 2. Multistage mapping between readings from motion and sound sensors and a dynamic spectral filter, processing live sound (see text).

performance habits in order to control the system. Our tests also demonstrate how, with a minimal amount of training and the learning of some new movements, performers are able to execute repeated musical tasks with very high fidelity between repetitions.

2. MAPPING MOTION TO TIMBRE

The foundation for the software component of our performance system is an association between motion sensors and a Pd-extended real-time sound-processing application. The unique feature of this sound processing application is that it processes the sound in the frequency domain using a dynamic spectral filter. This filter is applied to an input of a live signal from an acoustic instrument. The parameters of this filter are controlled by motion and sound sensors through the mapping presented in Figure 2. Values obtained by leveraging the six degrees of freedom available from the mobile device's motion sensors and real-time input from a musical pitch recognition system are used to derive a frequency modulation (FM) synthesis process. The latter generates a signal that constantly rewrites, in real time, 256-bin FFT vectors which in turn control the spectral filter.

As we show in Figure 2, the accelerometer and gyroscope readings are mapped to orientation values. Orientation in embedded systems is described by the three values *azimuth*, *elevation* and *rotation* (sometimes referred to as *yaw*, *pitch* and *roll*). Azimuth describes motion about the Z-axis, elevation describes motion about the X-axis, and rotation describes motion about the Y-axis (see Figure 3). In our mobile application, we use the Android methods *getRotationMatrix* and *getOrientation* to map the sensor readings to orientation. Out of the three values provided by the orientation we use only elevation and rotation in the control of the FM synthesizer as shown in the mapping in Figure 2. The azimuth value is replaced by a value that is equal to the frequency of the input signal. This value is obtained by a musical pitch recognition method (**fiddle~)** in the Pd-extended application (note that the term "pitch" here refers to identifying the musical pitch or the frequency of the sound and not the pitch in the sense of orientation, here referred to as elevation).

Frequency modulation is a sound synthesis method in which the waveform of one signal (known as the *carrier*) is varied by another signal (the *modulator*). Consequently, more elements are added to the frequency spectrum and a more complex sound is created. The characteristics of the frequency spectrum are affected by the amplitude of the modulating signal commonly referred to as the *modulation index*. An increase in the modulation index will transfer of more energy from the carrier to other spectral components. As Chowning [1] shows, varying the modulation index in time will produce a dynamic spectrum that evolves throughout the duration of the sound.

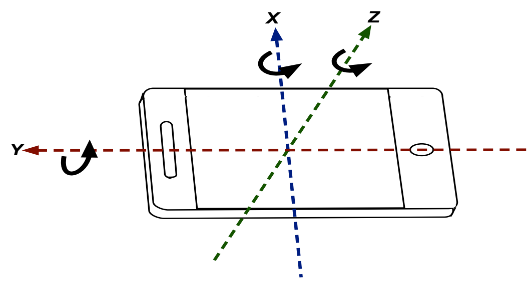

Figure 3. Orientation in embedded systems: azimuth (z) elevation (x) and rotation (y).

The FM synthesis method, hence, requires three arguments: carrier frequency, modulation frequency and modulation index. A common practice in FM synthesis is to specify a ratio by which the carrier frequency is multiplied to produce the modulation frequency. This ratio is called the *modulation ratio*. We have discovered in our system, in which the FM synthesis is used to control a dynamic spectral filter of a live signal, that the best results are achieved when the carrier frequency is matched by a pitch recognition method to the frequency of the input signal. We have empirically established that the ultimate range for the modulation ratio values as well as the modulation index values is floating-point numbers between 0 and 2. The range for the carrier frequency correlates with the ranges of frequencies of musical instruments best exemplified by the piano. Hence we have set the carrier frequency values to the range between 0Hz and 5KHz.

The basic mapping between the orientation values and the pitch recognition system to the FM synthesis arguments is specified as follows:

pitch recognition -> carrier frequency (1)
elevation -> modulation ratio
rotation -> modulation index

Since both the modulation ratio and the modulation index are floating-point numbers within the range of 0 to 2, the elevation and the rotation, which are specified in degrees, are scaled down by a division by 100. In addition the following mapping function is applied to both the elevation and the rotation:

$$y = x - (1 - \cos(x)) \tag{2}$$

We have used this function in order to filter some unintentional motion (like shaking) while maintaining responsiveness to more intentional motions. This function is represented by the graph in Figure 4. Thus the response to low degree angles is slow while the response to high degree angles is faster.

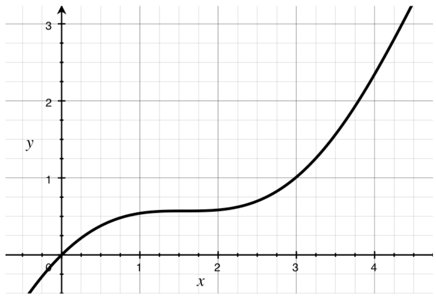

Figure 4. Graph representation of formula (2).

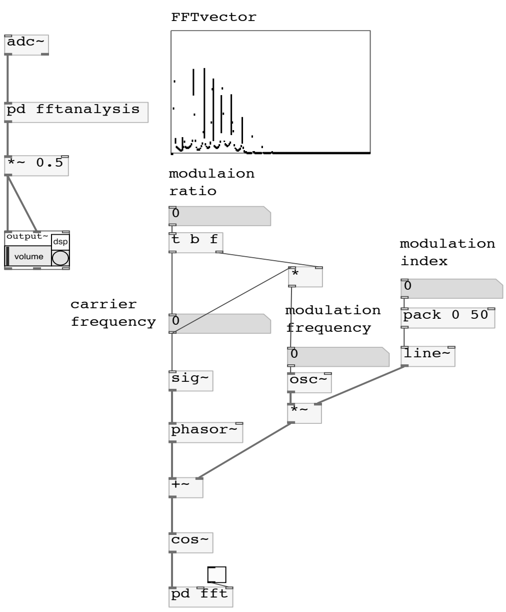

Figure 5. Pd-extended patch using FM synthesis to rewrite 256-bin FFT vectors controlling a spectral filter.

We map the FM synthesis to control a dynamic spectral filter in the Pd-extended patch shown in Figure 5. In this patch, we use FFT analysis (in the sub-patch in the upper-left part of the figure) to decompose, in the spectral domain, a live input signal into components represented in 256-bin FFT vectors. We then attenuate each component by multiplying FFT vectors from the live input signal with similar vectors from a signal generated by the FM synthesizer. Finally we use an inverse FFT to recompose the signal in the temporal domain, but with a new timbre produced by the spectral filter based on input parameters determined by the FM synthesizer. More operationally, the FM synthesis arguments, carrier frequency, modulation ratio and modulation index are entered in the patch component at the center of Figure 5. The signal generated by this FM synthesis patch component is analyzed in the spectral domain by the FFT sub-patch at the bottom of the figure. The latter rewrites the array object titled FFTvector in the upper part of the figure. This array is read by the FFT sub-patch in the upper-left part of the diagram that manipulates the input signal in the frequency domain by multiplying the values in the array by the live input signal's FFT analysis.

3. THE SYSTEM

Our performance system includes a microphone, an A/D interface, a laptop with Pd-extended, a Bluetooth foot pedal, a sound amplification system and a mobile device (although we are using only the gyroscope and accelerometer from the array of sensor available in a standard smartphone, we have chosen the latter as our mobile device because of its availability). The system components are shown in Figure 6. We have developed SoundCtrl, an Android application that incorporates the mobile device and its motion sensors in this system. In particular, the smart phone controls the Pd-extended application MotionSC, an interactive FFT-based sound processing application we have developed using the patch in Figure 5. The phone communicates with the laptop through Wi-Fi using a local network router and the TCP protocol. The Bluetooth foot pedal is used as an additional hand-free control element allowing the performer to start and stop the system as well as to freeze it in its current state. The A/D interface converts the analog input signal from the microphone to digital signal as well as the digital output signal to analog signal for amplification and playback.

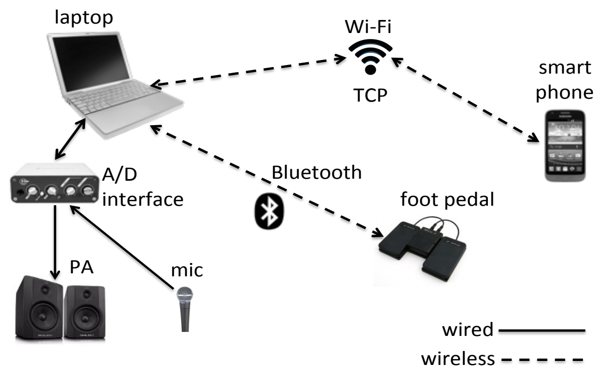

Figure 6. The performance system's component and communication diagram.

The Android application SoundCtrl maps the gyroscope and accelerometer readings to elevation and rotation and streams the data to the laptop through Wi-Fi. Note that we have replaced the magnetic field sensor with the gyroscope to allow more effective elevation and rotation values. The application SoundCtrl includes a simple color-coded bar-graph visualization of the orientation values, which we have designed using the Android GraphView library [4]. Its interface includes start and stop buttons for starting and stopping the data transmission and a field to specify the IP address of the receiving laptop (see Figure 7).

The Pd-extended sound processing application MotionSC, which is running on a laptop, receives the data from the mobile device through Wi-Fi, applies additional mapping to the data (including FM synthesis), processes the live sound based on the mapped data and outputs the processed sound. The application has a sound processing component, mapping component and Wi-Fi and Bluetooth communication component, as well as a visualization of the FM synthesis parameters carrier frequency, modulation ratio and modulation index. This visualization is a simple color-coded slider-graph that is similar to the smart phone display discussed above (see Figure 8). Additional GUI elements are included in the application's interface for controlling signal I/O, posture calibration, and monitoring Wi-Fi communication.

Figure 7. Visualization and UI in the mobile application SoundCtrl.

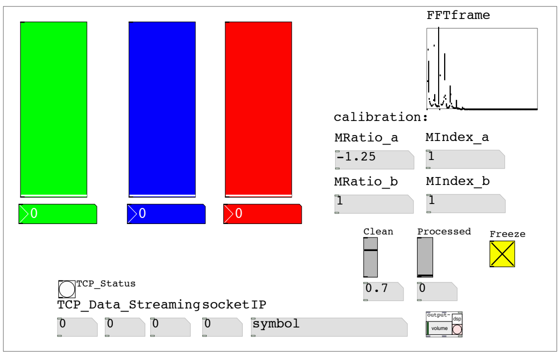

Figure 8. Visualization and GUI elements in the Pd-extended application MotionSC.

The visualizations used in the phone and laptop applications follow a unified shape and color scheme. In both cases this is an animated display with three bars or sliders that are constantly changing according to the streamed values they represent. The colors used in both applications are green, blue and red where green is the pitch and the carrier frequency; blue is the elevation and the modulation ratio; and red is the rotation and the modulation index. We believe that this simplified representation allows the performer to develop an intuitive correlation between motion and changes in the values and therefore between motion and changes in timbre. These simple animated displays and consistent color scheme also form the foundation for the graphical music notation we have developed for this performance system, which we discuss in section five.

4. PHYSICAL POSITIONING AND CALIBRATION

A motion sensor incorporated in a performance would have to be physically positioned in a location that is sensitive to body motion. Considering a variety of musical instruments, such as trumpet, saxophone, piano, and percussion, the most natural locations seems to be on the wrists or arms. With regard to other instruments, such as the guitar, mounting a mobile device on the instrument may also be a plausible solution. In our preliminary testing, we have only considered the right and left wrists as possible locations.

Depending on the instrument's holding posture and idiomatic techniques the center point of the orientation axis shifts from instrument to instrument. In particular, the difference between positioning on the wrist and mounting on an instrument (e.g. a guitar) is an inversion of the elevation axis. In the former the phone is facing the user while in the latter the phone is facing the opposite direction. Hence, in the wrist positioning the elevation values when raising the wrist are negative and when lowering the wrist the values are positive. This is obviously counterintuitive.

To accommodate for such orientation center-point shifts we have adapted formula (2) to include a and b coefficients as follows:

$$y = \left(\left(a\frac{x}{100}\right) - \left(1 - \cos\left(a\frac{x}{100}\right)\right)\right) + b \quad (3)$$

$$c_{calibration} = \begin{cases} a, & -2 \le a \le 2 \\ b, & 0 \le b \le 2 \end{cases}$$

These coefficients allow us to calibrate the center point of the orientation to fit the natural holding posture of each instrument. The goal of the calibration is to allow the performer to cover the required range of floating-point values using fairly easy motions. The a and b values obtained during calibration ensure that the control signals span the established range (0 to 2) and provide for inversion of the orientation parameter when appropriate. Formula 3 was programed as an expression in Pd-extended with *a* and *b* as the arguments of this expression. Hence, *a* and *b* where determined by entering different values as argument to this expression while the performer was asked to move the bars over their entire range. Formulae 4.1-4.6 show the calibration values (*a* and *b*) assigned to the saxophone, clarinet and flute. Figure 9 shows the resulting saxophone calibration curves.

$$y = -2\frac{x}{100} - \left(1 - \cos\left(-2\frac{x}{100}\right)\right) + 1 \quad (4.1)$$

Saxophone modulation ratio

$$y = 1.58\frac{x}{100} - \left(1 - \cos\left(1.58\frac{x}{100}\right)\right) + 0 \quad (4.2)$$

Saxophone modulation index

$$y = -2\frac{x}{100} - \left(1 - \cos\left(-2\frac{x}{100}\right)\right) + 0.7 \quad (4.3)$$

Clarinet modulation ratio

$$y = 1.5\frac{x}{100} - \left(1 - \cos\left(1.5\frac{x}{100}\right)\right) + 0 \quad (4.4)$$

Clarinet modulation index

$$y = -2\frac{x}{100} - \left(1 - \cos\left(-2\frac{x}{100}\right)\right) + 0.72 \quad (4.5)$$

Flute modulation ratio

$$y = 1.5\frac{x}{100} - \left(1 - \cos\left(1.5\frac{x}{100}\right)\right) + 1 \quad (4.6)$$

Flute modulation index

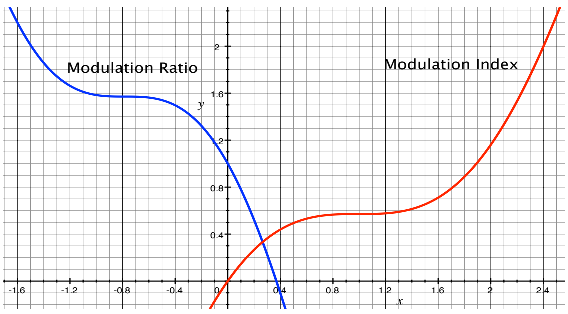

Figure 9. Graphs of system calibration used for the saxophone.

5. TESTING THE SYSTEM

A natural method of testing a motion-controlled performance system is to ask a performer to repeat specific movements and to measure the performer and the system ability to produce the same result by comparing recorded data for each repetition. Hence, we need to communicate to the performer as accurately as possible what movements he or she should repeat. In the context of music, the most accurate means of communication is musical notation. Standard musical notation allows the representation of the musical elements pitch, rhythm, dynamics and articulation, however, it does not include representations of physical motions. Therefore, we have developed additional graphical score elements that when combined with standard musical notation will allow us to communicate our requested performance tasks accurately to the performer.

Graphic notation has evolved to be a common element in contemporary music. Early examples appeared in the work of avant-garde composers in the 1950s and 1960s including scores by Ligeti, Penderecki, Stockhausen, Cage, and Feldman. More recent examples include scores by Australian composers Lindsay Vickery and Cat Hope [15]. Hope's score for *Longing* (2011), in particular, provided the inspiration for the graphical music notation we have developed for our performance system shown in Figure 10. As mentioned before, color-coding used in the visualizations in the mobile device and the laptop applications are consistent with this notation. Combined with the time representation embedded in standard notational aspects of the score, the red and blue lines shown in Figure 10 indicate to the performer in what direction and rate to move the corresponding bars or sliders in the mobile device or the laptop applications.

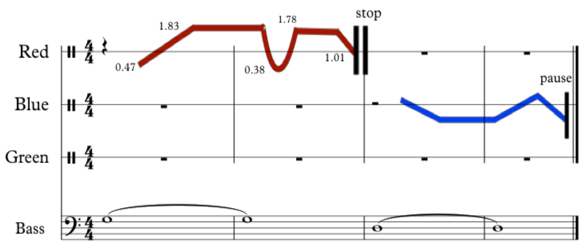

Figure 10. Graphic notion developed for the performance system presented in this paper.

Three musicians have tested our system playing a total of five different instruments (one of the participating musicians is also a co-author of this paper). The instruments tested include saxophone, flute, clarinet, cello and bass guitar. The test setting included the components described in Figure 6 and an additional large display to allow the performer to better see the visualization. Figure 11 shows the test setting for the saxophone, flute and clarinet. The test session was divided into three parts. In the first part, the system was calibrated to fit the holding posture of the instrument. In the second part, the musician was given "free play" time to get familiar with the system. And in the third part the musician played repeated notes while attempting to move the bars or sliders based on the notation given to him or her.

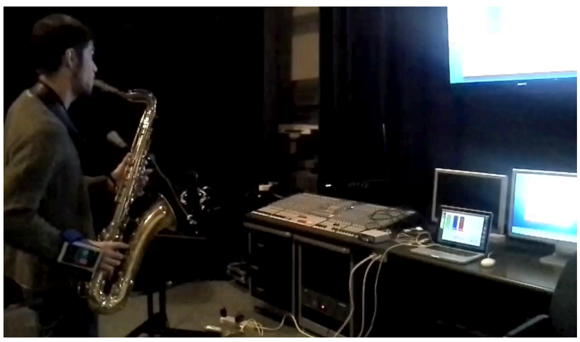

Figure 11. The test setting for the saxophone, flute and clarinet.

Figure 12 shows an excerpt from the notation given to the saxophone, in which the musician is asked to move the red bar up to the notated value, hold the bar in that position and then move it down to the next notated value. As the musician performed this task the Pd-extended application running on the laptop recorded the modulation ratio and modulation index values that correspond to the blue and red bars. Each note as the one shown in Figure 12 was repeated three to five times.

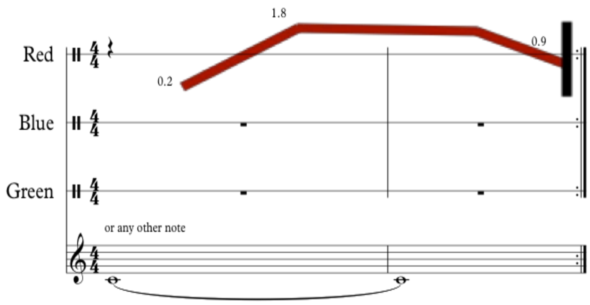

Figure 12. An excerpt from the score used for testing the system with a saxophone.

6. APPROXIMATION AND MUSICAL INTERPRETATIONS

As mentioned in the previous section, standard musical notation allows the representation of the basic musical elements pitch, rhythm, dynamics and articulation. With the adoption of standardized equal-tempered tuning in Western music the term pitch refers to an accurate measurement of frequencies. Rhythm may also be measured accurately by reference to a metronome time, although exact repetition of rhythmical values is not a realistic expectation in human performance. Dynamics and articulation are the least measurable elements. These elements vary to large extent from performer to performer as well as from performance to performance. For example, Repp [10] has shown a wide range of variations in the execution of dynamic instructions in Chopin's Etude in E major performed by highly regarded master pianists. In fact, variations of these elements are treated as defining characteristics of a personal interpretation of a composition.

We anticipated that a skillful use of our performance system would require some training time of the performer. We also anticipated that such training would not be available to the performers testing our system. Hence, in our tests, we adopted the approximation approach applied to the performance of dynamics and articulation in music. Our expectation for the tests reported here was that the performers would be able to approximate the same result when repeating the same motions with the smart phone. Figures 13a-d graph out the data collected in the tests with the saxophone, clarinet, flute, and cello. In Figures 13a-c the musician repeats notes and motion focusing first on the red bar and then on the blue bar. In Figure 13d the cellist focuses only on the red bar. While the values recorded are not exactly the same in each repetition (and in comparison with the graphic notation), the repeated patterns portrayed by these graphs clearly indicate that approximation of repeated motions is possible with our system. Furthermore, all of the graphs in Figure 13 display repeated patterns and therefore highlight the effectiveness of this system in regard to all the tested instruments. We are confident that with additional training performers will be able to execute more accurate repetitions of motions as well as more complex motions.

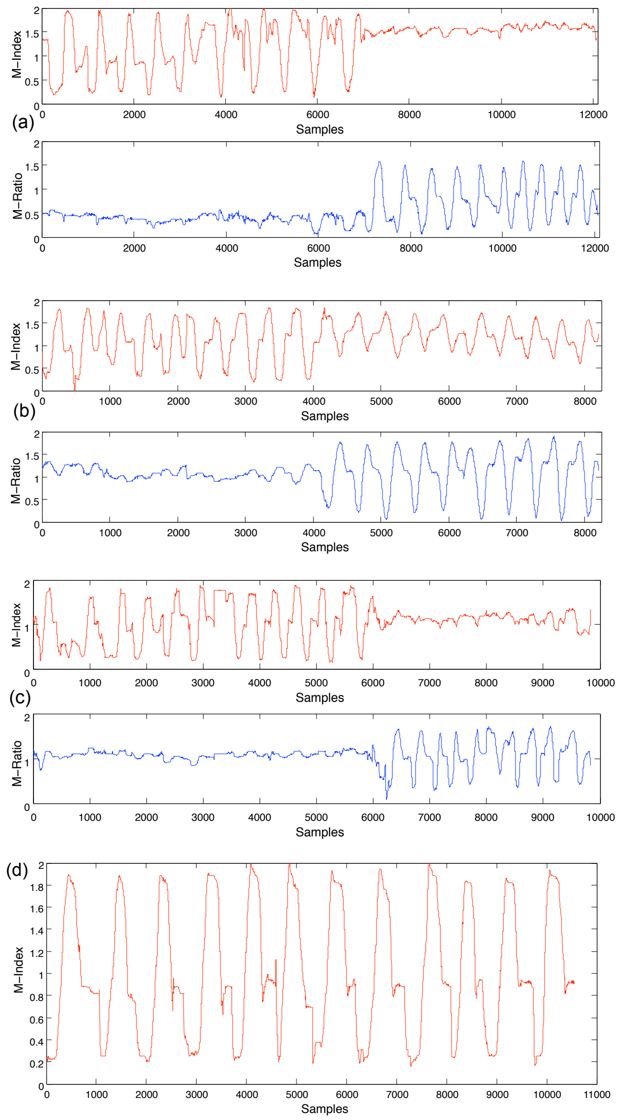

Figure 13. Plots of data collected in the testing of the system with (a) saxophone (b) clarinet (c) flute (d) cello.

7. RELATED WORK

In recent years, researchers have investigated several perspectives at the intersection of mobile technologies and musical creativity including innovations in compositional techniques, performance organizations and modes of interaction. Oh and Wang [9] explore the new genre of mobile music and audience participation performances, as well as techniques for enabling audience participation using smart phones that were incorporated in the performances of the Stanford Mobile Phone Orchestra. Settel *et al.* [11] present new modes of interaction in musical performances set in the outdoor environment using several mobile devices. Kim and Essl [6] discuss concepts and practical considerations in the design of mobile music environments as independent platforms. Wechsler *et al.* [13] present a video-based motion sensing system which allows performers to control music and images through motions and gestures. Lazzarini *et al.* [7] discuss the development of a mobile platform for sound synthesis and audio processing using the C-based audio programming language CSound. Essl [3] presents the Mobile Phone Ensemble of the University of Michigan. Winkler [14] describes the creative process of a collaborative work using the motion detection system VNS which was created by David Rokeby. Wang *et al.* [12] explore the musical implications of social interaction and collaboration using novel technologies for embedded systems.

8. CONCLUSION

In this paper, we have presented a musical performance system that is based on the correlation between motion and sound. In this system, a mobile device is used as a motion-based control device. The software, incorporated in this performance system, maps readings from the mobile device motion sensors, the gyroscope and accelerometer, to the processing of live sound in the spectral domain. As part of the development of this system we have designed a visualization and a specialized music notation. Our preliminary tests show the effectiveness of this system as a performance tool.

9. REFERENCES

[1] Chowning, John, "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation," *Computer Music Journal* 1, no. 2 (April 1977): 46-54.

[2] Dodge, Charles and Thomas A. Jerse, *Computer Music: Synthesis, Composition and Performance* (New York: Schirmer Books, 1997), 46.

[3] Essl, Georg. "The Mobile Phone Ensemble As Classroom," *Proceedings of the International Computer Music Conference* (2010).

[4] Gehring, Jonas. *Android GraphView*, http://android-graphview.org (accessed December 18, 2013).

[5] Helmholtz, Hermann L. F. *On the Sensations of Tone as a Physiological Basis for the Theory of Music,* trans. Alexander J. Ellis (New York: Dover Publications, Inc., 1954).

[6] Kim, Jong Wook and Georg Essl. "Concepts And Practical Considerations Of Platform-Independent Design Of Mobile Music Environments," *Proceedings of the International Computer Music Conference* (2011).

[7] Lazzarini, Victor, Steven Yi, Joseph Timoney, Damian Keller, and Marcelo Pimenta. "The Mobile Csound Platform," *Proceedings of the International Computer Music Conference* (2012).

[8] Moorer, James A. and John Grey, "Lexicon of Analyzed Tones. Part I: A Violin Tone," *Computer Music Journal* 1, no. 2 (April 1977): 39-45.

[9] Oh, Jieun and Ge Wang. "Audience-Participation Techniques Based On Social Mobile Computing," *Proceedings of the International Computer Music Conference* (2011)

[10] Repp, Bruno H. "A Microcosm of Musical Expression: II. Quantitative Analysis of Pianists' Dynamics in the Initial Measures of Chopin's Etude in E Major," *Journal of the Acoustic Society of America* 105 no. 3 (1999): 1972-1988

[11] Settel, Zack, Mike Wozniewski Nicolas Bouillot, and Jeremy R. Cooperstock. "Audio Graffiti: A Location Based Audio-Tagging And Remixing Environment," *Proceedings of the International Computer Music Conference* (2009)

[12] Wang, Ge, *et al.* "Smule = Sonic Media: An Intersection Of The Mobile, Musical, And Social," *Proceedings of the International Computer Music Conference* (2009).

[13] Wechsler, Robert, Frieder Weiß and Peter Dowling. "Eyecon: A Motion Sensing Tool For Creating Interactive Dance, Music And Video Projections," *Proceedings of the Society of the Study of Artificial Intelligence and the Simulation of Behavior (SSAIBB)'s convention: Motion, Emotion and Cognition* (2004).

[14] Winkler, Todd. "Motion-Sensing Music: Artistic And Technical Challenges In Two Works For Dance," *Proceedings of the International Computer Music Conference* (1998).

[15] Wyatt, Aaron, Cat Hope, lindsay Vickery and Stuart James. "Animated Music Notation on The iPad," *Proceedings of the International Computer Music* (2013).