Demo Abstract: Sensor-enabled Yo-yos as New Musical Instruments

Jung-Hyun Jun[1]
peterjun@sutd.edu.sg

Sunardi[1]
sunardi@sutd.edu.sg

Lijuan Wang[1]
wanglijuan@sutd.edu.sg

Joel W. Matthys[2]
joel@matthysmusic.com

Simon Lui[1]
simon_lui@sutd.edu.sg

Yu Gu[1]
jasongu@sutd.edu.sg

[1]Singapore University of Technology and Design, 20 Dover Drive, Singapore 138682
[2]College Conservatory of Music, University of Cincinnati, Cincinnati OH, U.S.A. 45221-0003

ABSTRACT

An interactive and reprogrammable musical yo-yo system is designed. The aim of it is to demonstrate the feasibility of converting any sensor-enabled objects into potential musical instruments. This involves three design phases. First, the physical yo-yo is designed to house Iris sensors. The software is developed to sense the movement of yo-yo and transmit its measurements to Max/MSP for corresponding music generation. Finally, aurally pleasing and real-time musical sounds are designed and generated in effect of yo-yo by the computer music composer.

Categories and Subject Descriptors

H.4 [**Computer Applications**]: Miscellaneous; D.2.8 [**Design**]: Performance—*Sensor-enabled instruments*

Keywords

Wireless sensor networks, MaxMSP, Yo-yo, Musical instrument, Cyber-physical objects

1. INTRODUCTION

Wireless Sensor Networks and Internet of Things have potentials to turn ordinary objects into programmable and interactive devices with networking capability. We envision that these technological advances will open up a possibility of transforming sensor nodes and cyber-physical objects into tunable and programmable musical instruments that are aurally pleasing, interactive, and fun to play.

In today's market, there exist some interactive musical toys and software which can convert portable electronics like smartphones into simple musical instruments. However, they often do not support the sophisticated needs of computer music composers. Smartphones, for all their versatility, provide a restricted and unintuitive interface for musical performers with limited choice of sounds and capability to interact with other musical objects. Also, they are not readily programmable and, most importantly, smartphones and tablets lack the tactile and kinesthetic feedback necessary for complex instrument design. But, with help of wireless sensor network and Max/MSP [2], a visual programming language for music and multimedia, it is now feasible to transform a non-instrumental object like a yo-yo into a complete musical performance.

IPSN'12, April 16–20, 2012, Beijing, China.
ACM 978-1-4503-1227-1/12/04.

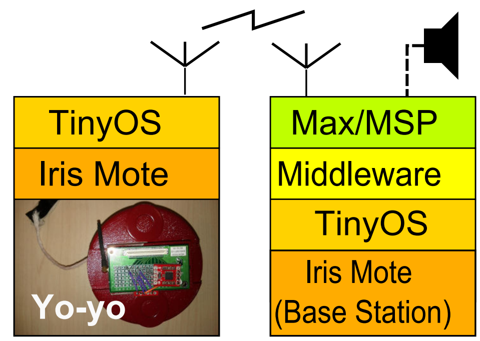

Figure 1: Musical yo-yo system architecture.

Our contributions here are twofold: (1) we have successfully developed a working musical yo-yo system and (2) we make it simple for computer music composers to use this system.

2. MUSICAL YO-YO SYSTEM

We selected the yo-yo as our platform because of its playful and familiar form. Capable of a wide range of complex motion in three dimensions, as well as rotation, the yo-yo shows great potential as a musical instrument, while also being inviting and children friendly. The motion of the yo-yo provides both tactile feedback to the performer and strong visual interest to an observer. The musical yo-yo system is of a three-layered architecture as shown in Figure 1.

a. Yoyo Hardware Design : The model of musical yo-yo is designed to house two Iris motes as shown as Figure 2. The Iris mote [1] with an external 3-axis accelerometer and 2-axis gyroscope is used to measure the force exerted from a hand tossing the yo-yo. The Iris sensors are placed close to the center of yo-yo, which helps to balance the yo-yo while it is spinning. The yo-yo is materialized by the 3D printer using polyester. Four metal rings are implanted onto the yo-yo to balance the misaligned weight caused by two AA batteries that supply power to Iris sensors. Although, the size and overall weight of this yo-yo is bigger and heavier than the regular yo-yos, it still functions like a real yo-yo.

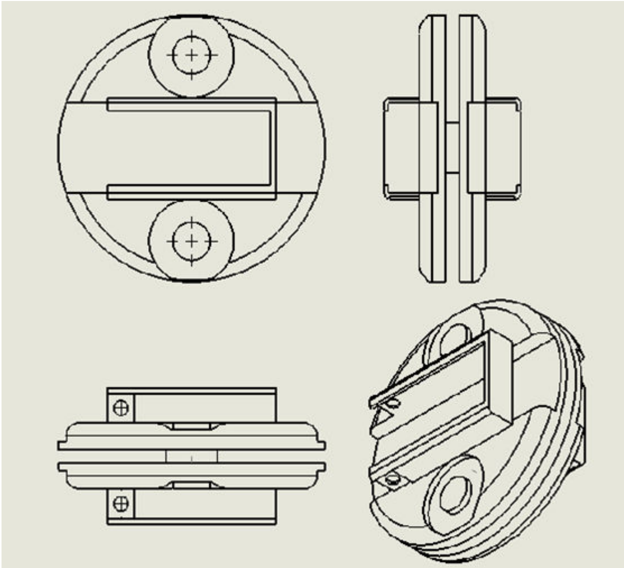

Figure 2: Yo-yo 3D layout model.

We had many failures while designing the musical yo-yo that can hold wireless sensors without disturbing the balance of its weight. However, this would not have been a challenge if wireless sensors were smaller and lighter-weighted.

b. Sensor Software Design : To the best of our knowledge, there is no formally published software/middleware which integrate the tinyOS with Max/MSP that accepts commands from the composers and outputs the data into Max/MSP. Our custom middleware is designed to translate commands and outputs between tinyOS to Max/MSP. It is an extended Java object running inside the virtual java machine (MXJ). It lists a set of available sensors to Max/MSP for the composer. Then, it enables sensors and adjusts their sampling rate and transmission rate of the musical yo-yo according to the composer's command. Sensors transmit a stream of raw sensor data to middleware periodically. Typically, the period is set to 1 (msec) for its responsiveness.

When the yo-yo system is executed the data received at the middleware is forwarded to Max/MSP and makes the data available for the composers. The data mining was done at the composers' end to optimize their usage of the sensor data.

c. Transformation from Sensory Data to Music : Four or five yo-yos are intended to be used simultaneously in a live performance as fully independent musical instruments. In designing the sound in Max/MSP, we decided to craft a kind of generative techno-music, rhythmic but not strictly repetitive, which is able to use as much of the yo-yo's motion data as possible. This gives the performer the maximal potential for musical variety and control.

Using John Chowning's frequency modulation techniques, we created a set of synthetic bells whose pitches would rise and fall with the extension of the yo-yo. Combining the X, Y, and Z values, with respect to the 3-axis acceleration from the sensor, the total velocity is normalized and mapped onto one of two alternating pitch sets: {C,D,E,G,A} and {C,Eb,F,G,Bb}, commonly known as the major and minor pentatonic scales, spanning three octaves. The tone pans from left to right based on the X values and the notes are

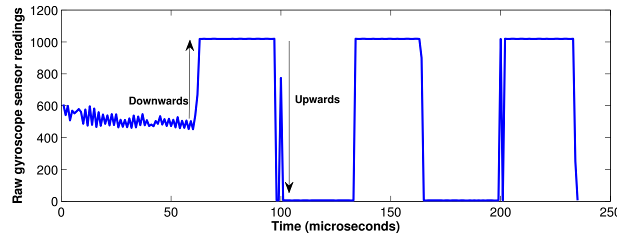

Figure 3: Gyroscope sensor readings and corresponding directions where yo-yo is spinning.

quantized rhythmically to occur twice per second, or 120 beats per minute. The Z value is used to generate a bass line, whose tone is a sine wave altered through amplitude modulation. This bass instrument is mapped on the same scale as the treble bells, but at half the tempo.

Other musical data is derived from higher-order sensor data. A sampled drum set (kick drum, snare, and hi-hat) is triggered by aspects of acceleration in the gyroscope and velocity data as shown as Figure 3. The bass drum is triggered by a trough in the acceleration data, which occurs when the yo-yo reaches either the top or the bottom of the string.

Every 8 bass drum hits triggers a change of harmony, which provides a sense of development and complexity to the music. Usually this involves alternating between the two pentatonic scales, but occasionally a different scale is randomly triggered. This scale, the traditional American blues scale {C, Eb, F, F#, G, Bb}, introduces a new harmonic element to the music.

3. CONCLUSION AND FUTURE WORK

The rehearsal of musical yo-yos was recorded and uploaded onto `www.youtube.com` [3] and it was first demonstrated for openhouse at Singapore University of Technology and Design. Due to its flexibility and user friendliness, any composers can use it for their professional performances. It is expected to be performed as a part of The Cincinnati Composers Laptop Orchestra Project (CiCLOP).

As for the future work, we will map the sensor inputs to trigger and produce more controllable music pattern. For example, using drum-loop rather than individual drum sound as atomic element, or using certain thresholds of spinning speed to trigger drum pattern instead of using signal crest and trough, or just performing a sequence of drum loop, but the spinning will trigger and switch to other loop.

4. ACKNOWLEDGMENTS

This work is supported by SUTD SRG ISTD 2010 002 and Singapore-MIT International Design Center IDG31000101.

5. REFERENCES

[1] Crossbow iris datasheet at. `http://bullseye.xbow.com:81/Products`.

[2] Max/msp - a visual programming language for music and multimedia. `http://cycling74.com/whatismax`.

[3] Musical yo-yo rehearsal video. `http://www.youtube.com/watch?v=Y_Dceb3S_tM`.