Spider King: Virtual Musical Instruments based on Microsoft Kinect

Mu-Hsen Hsu[1], Kumara W. G. C. W.[2],
Timothy K. Shih[3],
Dept. of Computer Science and Information Engineering,
National Central University,
Taoyuan Country 32001, Taiwan (R.O.C.)
[1]b_i10243@hotmail.com,
{[2]chinthakawk, [3]timothykshih}@gmail.com

Zixue Cheng,
School of Computer Science and Engineering,
University of Aizu,
Aizu-Wakamatsu City, Fukushima 965-8580, Japan
z-cheng@u-aizu.ac.jp

***Abstract*—Human Computer Interaction is becoming a major component in computer science related fields allowing humans to communicate with machines in very simple ways exploring new dimensions of research. Kinect, the 3D sensing device introduced by Microsoft mainly aiming computer games domain now is used in different scopes, one is being generation or controlling of sound signals producing aesthetic music. Here, in this paper, authors' experimental efforts on three virtual music instruments: Drum, Guitar and Spider King, based on Kinect sensor are presented. All three instruments virtually set the relevant sensing input areas, as an example, strings of the guitar or cymbals of the drum, then, the player controls the instrument through those virtual inputs through the Kinect. Sound control data is then generated and fed to the audio library based on the musically oriented human computer interaction gestures, composing a real-time musical expressive performance. A live performance using the presented virtual instruments was carried out at the end.**

Keywords—MIDI; VST instrument; OpenNI; Midi port; RtMIDI

I. INTRODUCTION

As Human Computer Interaction (HCI) evolves in many different areas of human interactions with the computers or machines, this generation and composing of music based performance with advanced sensing devices is becoming hot research topic and a promising application area for the high revenue entertainment markets based on Internet, PCs, and laptops and especially with the smart mobile devices. Here the most important hurdle is correctly sensing of the human gestures through the new age sensors. In this paper authors present their experience of musical performance with the use of three virtual instruments, Drum and Guitar and a newly proposed musical instrument SpiderKing. All three instruments capture the human gestures through the Microsoft Kinect sensor. Kinect is a sensor which is capable of capturing depth and color information of the user in front of it using an array of RGB and infrared cameras. Further it is capable of capturing the sound input though an array of microphones. OpenNI library is used to interface with the Kinect sensor in the proposed design. In detail design and connectivity information is given in section 3.

Audio format used was MIDI which stands for Musical Instrument Digital Interface. As MIDI uses very simple message structure to generate sound, it is the best suitable format when dealing with this kind of computer interfacing applications. As MIDI sends only the relevant timing and frequency information with the sound levels it consumes comparatively very small bandwidth against raw audio formats. And a device called MIDI controller is required to generate audio from that digital control information.

In traditional music it was used to have only acoustic instruments which stimulate sounds based on player's direct inputs. Now a day because of the introduction of MIDI technology a large range of computer or Internet based music tools and virtual instruments appearing with almost real or even better performance comparing to the traditional instruments. What is most interesting here is, these new devices are capable of generating new sounds that acoustic instruments are not capable of.

The rest of the paper organizes as follows. In the related works section several related recent works carried out by other researchers are discussed. Design of the musical instruments and related concepts are discussed in the section III. Section IV is devoted to present the experimental related information while section V discusses future directions. Section VI concludes the discussion.

II. RELATED WORK

Odowichuk et al. in [1] describes a study into the realization of a new method for capturing 3D sound control data. They have used a radiodrum 3D input device by incorporating a computer vision platform that have developed using the Xbox Kinect motion sensing input device. Their Kinect instrument is compatible with virtually all MIDI hardware/software platforms so that any user could develop their own custom hardware/software interface with relative ease.

Mandanici et al. presented a tool called "Disembodied voices" as an interactive environment designed for an expressive, gesture-based musical performance in [2]. They have used the motion sensor Kinect, placed in front of the performer, to provide the computer with the 3D space coordinates of the two

hands. The software, developed by the authors, interprets the gestural data and controls articulated events to be sung and expressively performed by a virtual choir. The system also provides a display of motion data, a visualization of the part of the score performed at that time, and a representation of the musical result processed by the compositional algorithm.

Qin, Ying in [3] describes some technical details about Wii, discusses its potential as musical controllers, and introduces several achievements of utilizing Wii Remote in musical contexts: virtual conducting systems, virtual instruments, imaginary dialogues, interactive mixing, and collaborative experience (Wiiband).

Trail et al. in [4] focused on the pitched percussion family and describe a non-invasive sensing approach for extending them to hyper-instruments. Their primary concern was to retain the technical integrity of the acoustic instrument and sound production methods while being able to intuitively interface the computer. This is accomplished by utilizing the Kinect sensor to track the position of the mallets without any modification to the instrument which enables easy and cheap replication of the proposed hyper-instrument extensions. In addition they described two approaches to higher-level gesture control that remove the need for additional control devices such as foot pedals and fader boxes that are frequently used in electro-acoustic performance. This gesture control integrates more organically with the natural flow of playing the instrument providing user selectable control over filter parameters, synthesis, sampling, sequencing, and improvisation using a commercially available low-cost sensing apparatus.

In Crossole, as presented by Sentürk et al. in [5] the chord progressions are visually presented as a set of virtual blocks. With the aid of the Kinect sensing technology, a performer controls music by manipulating the crossword blocks using hand movements. The performer can build chords in the high level, traverse over the blocks, and step into the low level to control the chord arpeggiations note by note, loop a chord progression or map gestures to various processing algorithms to enhance the tumbrel scenery.

Wilschrey et al. in [6] presents the development of a virtual drums prototype, based on natural interaction, through a Kinect device. Results of early tests are encouraging, but the prototype can definitively be improved. Although the delay of the Kinect device seems insignificant, it may however affect user's experience. WAV sounds played by the current prototype will be replaced by MIDI instructions.

Shamshiri, Sina in [7] aims to create a completely controller-less air guitar with the aid of Microsoft Kinect sensor, with expectations to provide a superior experience compared to the previous works on the topic. Additionally, it attempts to create a real and rich synthesized sound as well as providing digital effects in order to further enhance the user experience. The final product was positively perceived by all the users who tested the platform and it is believed that not only using it can be a fun experience but also it has very high future potential.

III. DESIGN

In this section, presented three musical instruments and their operations are discussed. Microsoft Kinect sensor[1] is used as the main interaction tool between the human player and the computer. In all three instruments, Kinect captures the RGB color and depth information in the rate of 30 fps, each with the spatial resolution of 640x480 pixels. The standard 3D sensing framework OpenNI [2] was used as the driver and API to communicate with the Kinect. Once the depth information is captured using Kinect, user skeleton can be obtained with the available functions of OpenNI with 24 joints. Then using the available coordinates of each joint, a suitable strategy can be implemented to play the instrument virtually. Fig. 1 displays the steps of the general architecture used in each instrument.

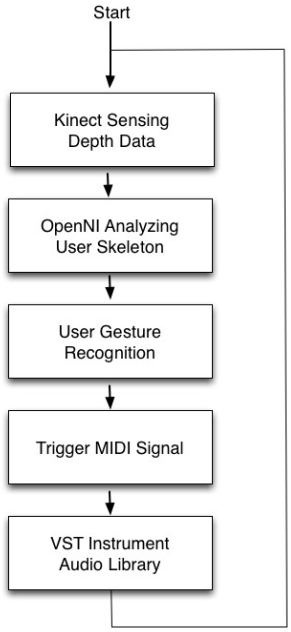


Fig. 1. Virtual instruments architecture

With the inherence qualities like ease of use, ease of communication, very low data rates etc. *Musical Instrument Digital* Interface which is known as *MIDI* is used as the audio format in the design. MIDI carries event messages that specify notation, pitch and velocity with the relevant timing information. For the MIDI signal in the presented virtual instruments, the RtMidi [3] API from Gary P. Scavone from McGill University was used to send MIDI signals to audio library for controlling the note key, duration and velocity.

Virtual Studio Technology which is abbreviated as VST instrument 4 from Steinberg Company is used as the audio library in this design which is capable of simulating the sound of real instruments. Using MIDI as the input mechanism, VST instruments output the sounds vividly as instructed.

[1] http://www.microsoft.com/en-us/kinectforwindows/
[2] http://www.openni.org/
[3] http://www.music.mcgill.ca/~gary/rtmidi/
[4] http://www.steinberg.net/en/products/vst.html

To connect the program with VST, we use virtual MIDI port software called LoopBe[5]. Since, we can choose the MIDI port we want to send data In RtMidi API, we choose the LooBE1 port and then set the input port of VST as LoopBE. Then finally we can use the program to control VST by sending respective MIDI signals.

A. Drum

First, as explained above using Kinect and OpenNI, user skeleton is acquired. Then three areas in front of the user is identified as the Kick, Snare, Hi-Hat and Cymbal. Left hand, right hand and right knee are used as the triggers against the above specified regions. When the coordinate of the triggering point is larger than a specified threshold with respect to the defined regions of the virtual drum sets, program triggers MIDI signals, and then MIDI signals trigger the sound in audio library. Respective regions and a sample of user depth data map is shown in Fig. 3 (a). Over shoulder view of the player is shown in Fig. 3 (b) with the depth map on the screen and the Kinect sensor beside.

Algorithm 1. Drum operation

1. Capture the coordinates of the Head, Left Hand, Right Hand, Left Knee, Right Knee, Torso center, Right and Left Hip positions using Kinect sensor through OpenNI depth and skeleton data (Fig. 2).
   Controlling joint set = Left Hand, Right Hand, Left Knee, Right Knee
2. Draw three horizontal lines in the display screen according to Head, center of Torso and Hips.
   Set trigger points as follows,
   A, B: in Head line for cymbal A and cymbal B
   C, D: in Torso line for snare and hi-hat
   E: in Hip line for bass drum
3. Set midiMessage as follows:
   If (d_A<threshold)
   midiMessage(144, 52, 80)
   If (d_B<threshold)
   midiMessage(144, 55, 52)
   If (d_C<threshold)
   midiMessage(144, 38, 52)
   If (d_D<threshold)
   midiMessage(144, 46, 52)
   If (d_E<threshold)
   midiMessage(144, 35, 52)
   where,
   d_X = distance of one of joint from controlling set and x, x □ (A, B, C, D, E),
   threshold = 80 in 2D coordinate, not including the depth
   midiMessage(144 for midi note on, key of unit x, 52 as the velocity)
4. Route respective MIDI signal to VST instrument through Virtual midi port
5. VST instrument plays the sound
6. Go back to 1 (next frame)

[5] http://www.nerds.de/index.html

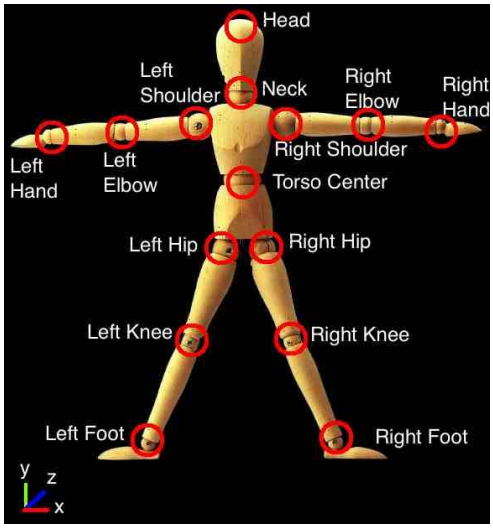
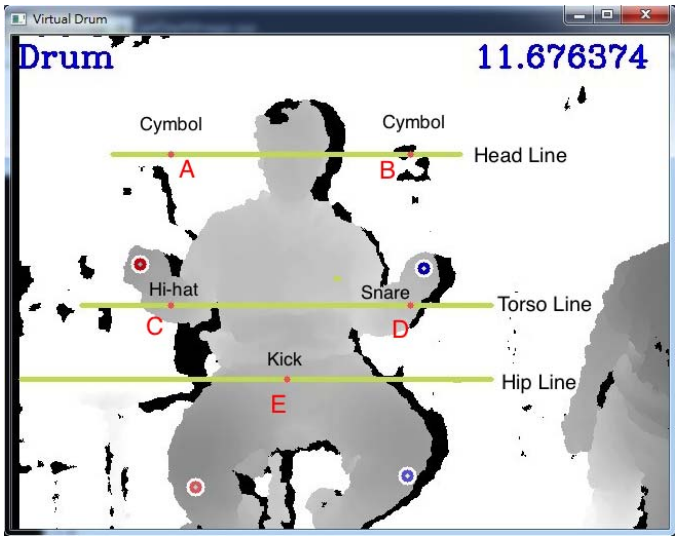
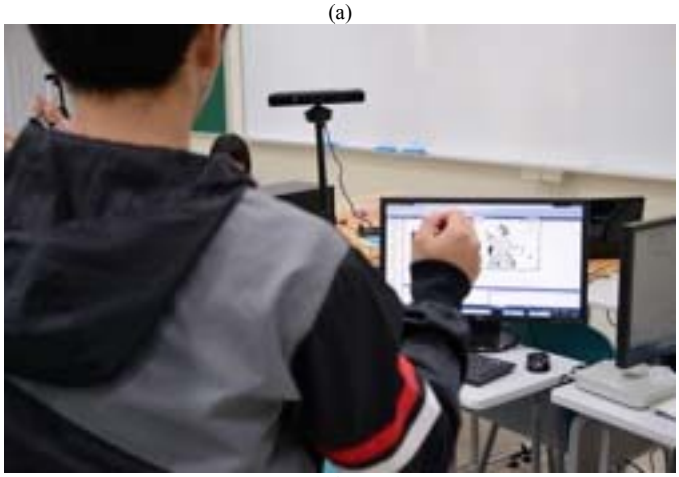


Fig. 2. Major coordinate positions of the human skeleton



(a)



(b)

Fig. 3. Design of the Drum (a) drawn horizontal lines and trigger points on the depth map (b) over shoulder view of the player

B. Guitar

Same like in Drum, using Kinect and OpenNI, user skeleton is captured first. Then according to the user skeleton coordinates, we set a Chord selection position in front of the user's left hand. In the Chord selection position, we have defined six different areas, each representing a chord. To play the virtual guitar, we also set a virtual guitar string in front of the user's right hand. When the right hand coordinate is in a special value interval, the program sends MIDI signal to the audio library and then triggers the relevant sound. Fig. 4 shows the chord and string positions of the virtual guitar setup. Fig. 5 (b) shows the detected skeleton data of the user on screen and in Fig. 5 (c) the over shoulder view of the player.

Algorithm 2. Guitar operation

1. Capture the coordinates of the Head, Left Hand, Right Hand, Torso center of the player using Kinect sensor through OpenNI depth and skeleton data (Fig. 2)
   Controlling joint set = Left Hand, Right Hand
2. Virtually set a 2 row by 3 column table in front of the left hand of the player using one depth threshold A and two horizontal thresholds B and C to dedicate 6 positions for six notes as given in Fig. 5 (a).
   Set one vertical threshold D in front of the right hand of the player for the trigger.
3. Set midiMessage as follows:
   If (LH_Z>A AND LH_X>A AND LH_X>B AND RH_Y>D)
   midiMessage(144, C chord, 80)
   If (LH_Z>A AND LH_X<A AND LH_X>B AND RH_Y>D)
   midiMessage(144, Dm chord, 80)
   If (LH_Z>A AND LH_X<A AND LH_X<B AND RH_Y>D)
   midiMessage(144, Em chord, 80)
   If (LH_Z<A AND LH_X>A AND LH_X>B AND RH_Y>D)
   midiMessage(144, F chord, 80)
   If (LH_Z<A AND LH_X<A AND LH_X>B AND RH_Y>D)
   midiMessage(144, G chord, 80)
   If (LH_Z<A AND LH_X<A AND LH_X<B AND RH_Y>D)
   midiMessage(144, Am chord, 80)

   where,
   (LH_X, LH_Y, LH_Z) are left hand coordinates of the player
   (RH_X, RH_Y, RH_Z) are right hand coordinates of the player
   midiMessage(144 for midi note on, three notes of the relevant chord, 80 as the velocity). Following are the notes in each chord respectively, Chord C contains C E G, Chord Dm contains D F A, Chord Em contains E G B, Chord F contains F A C, Chord G contains G B D and Chord Am contains A C E.
4. Route MIDI signal to VST instrument through Virtual midi port
5. VST instrument plays the sound
6. Go back to 1 for the next frame

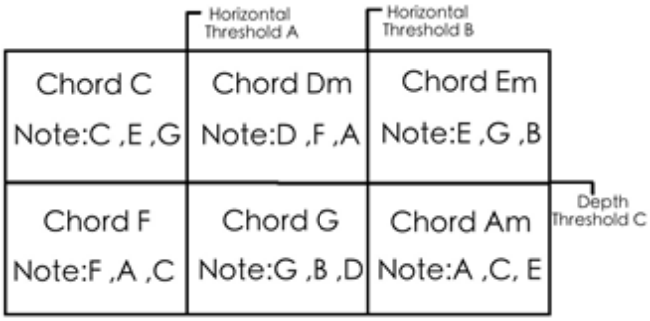
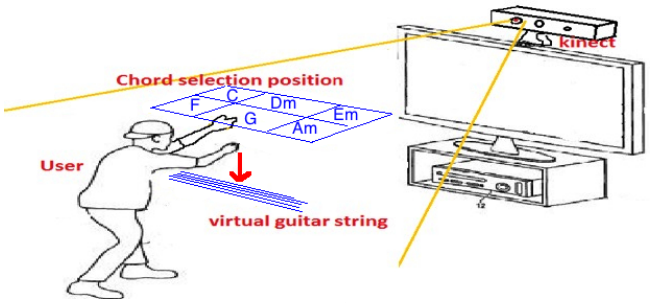
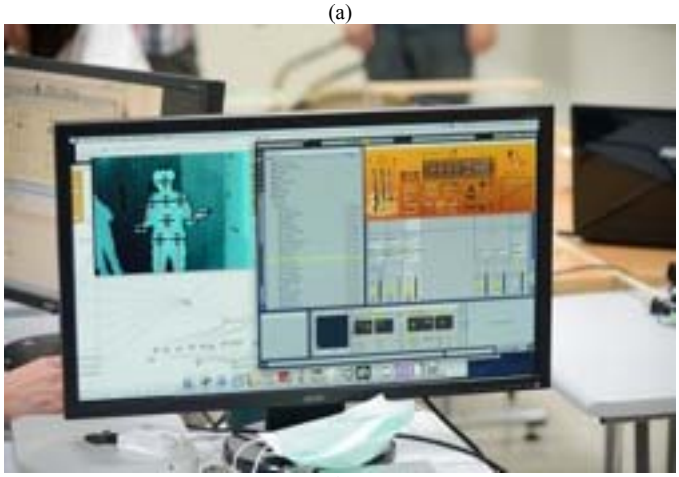


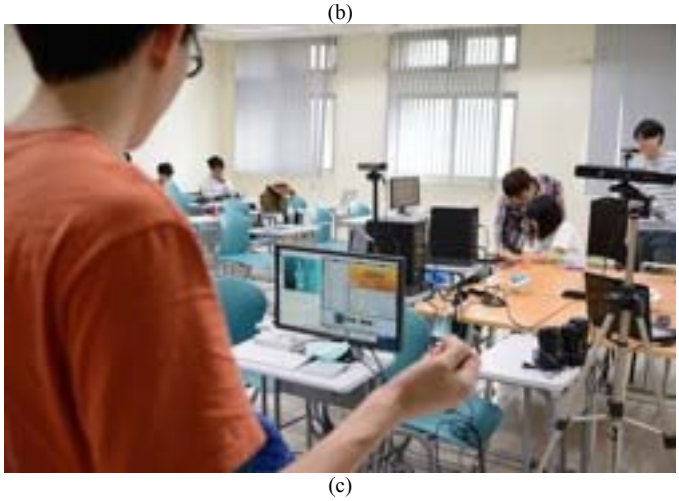
Fig. 4. Virtual guitar tones.



(a)



(b)



(c)

Fig. 5. Virtual guitar setup (a) chord selection postion and virtual guitar string (b) detected skeleton (c) over shoulder view of the player

C. *Spider King*

The program virtually draws a circle around the user. We divide the circumference into several intervals. Then users' hands are used to control the key and volume. When the hand is closer to the circumference, the volume becomes louder. The sound is also from MIDI signal as with drum and guitar, and we connect MIDI signals to different audio libraries. Fig. 6(a) shows a simple design of the Spider King with user depth data map and virtual circle while Fig. 6(b) shows the over shoulder view of the player.

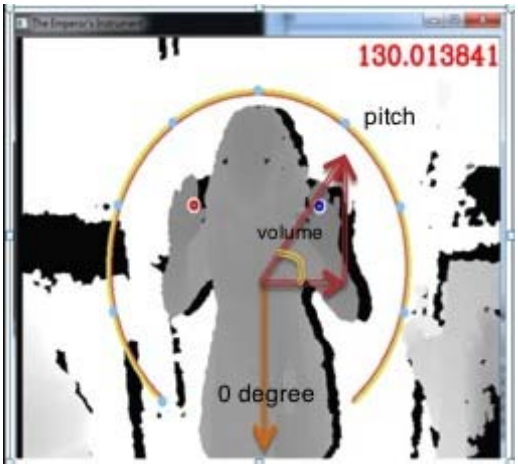
Algorithm 3. Spider King operation (basic)

1. Capture coordinates of the Head, Left Hand, Right Hand, Torso center positions from Kinect sensor through OpenNI using depth and skeleton data (Fig. 2)
   Controlling joint set = Left Hand, Right Hand;
2. Draw a circle in the display screen
   Divide the circle in to 8 sections according to the angle. Then we have 8 areas representing different notes (A, B, C, D, E, F, G and H)
3. Set the midiMessage as follows:
   If (LH in A AND d < threshold)
   midiMessage(144, 39, d)
   Repeat the same for all other points A to H for both left hand and right hand coordinates.
   where,
   LH = left hand
   d = distance of left hand and circumference
   midiMessage(144 for midi note on, relevant key note, d as the velocity)
7. Route the MIDI signal to VST instrument through Virtual midi port
8. VST instrument plays the sound
9. Go back to 1 For next frame

Algorithm 4. Spider King operation (advanced)

- The spider web is virtually located in front of the user
- Calculate the hand position and compare it with the spider web to control the spider king instrument
- Several notes are positioned on the web
- Use Depth to control volume
  - As an example, when the user's hand is nearer the Kinect camera, the sound becomes louder
- Radial distance from the center of the web controls the low, Mid and high tones
  - For example Low, Mid and High with respect to note A, represents Low A, Mid A and high A

Implementation of the basic Spider Kind instrument is completed and the users with good musical knowledge claims to perform it better. We plan to further experiment on the user experience in detail in a later stage. Algorithm 4 shows step by step details of our current ongoing work of the advanced Spider King instrument design and Fig. 7 shows the in depth design of the notes and their positions in the virtual setup.



(a)

(b)

Fig. 6. Spider King (basic) operation (a) notes positions (b) over shoulder view of the player

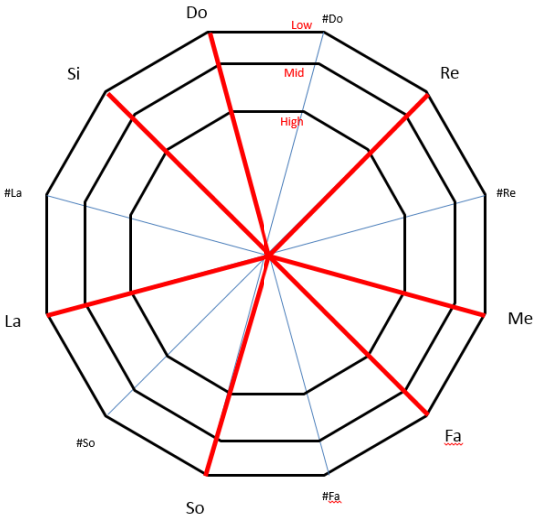
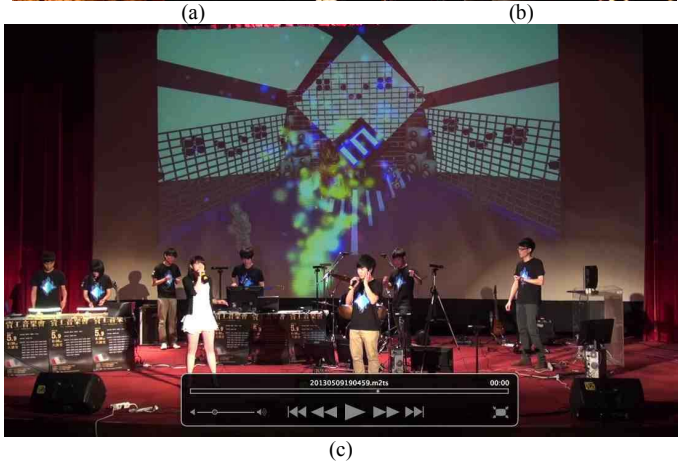


Fig. 7. Spider King (advanced) operation

IV. EXPERIMENT

All the virtual instruments presented here, Guitar, Drum and Spider King and several other virtual instruments that the current wok is going on were presented in a live performance carried out as part of the NCU CSIE Annual musical concert on 9th May 2013 as shown in Fig. 8. Full video of the performance can be accessed at http://goo.gl/ziLZND.

(a) (b)



(c)

Fig. 8. Live performance using virtual instruments (a) Spider King (b) Drum (c) All virtual instruments

V. FUTURE WORKS

To improve the research further the following works are planned to carry out in future.

- At present the virtual instrument tracks only the position of our skeleton joints limiting the playing way. If we can track more detail information such as finger position, then we can develop several playing ways in detail.
- For the latency, the program so far working at the frame rate of 11 fps only. It is not sufficient for a high quality instrument playing experience since in playing music, the accuracy is the most important. Hence, we need to improve the performance of the program.
- Integrating of robust methods to filter out wrong coordinate tracking to improve the quality of the play.
- A detailed subjective Quality of Experience (QoE) is required to compare the real experience of the users and listeners to further develop the instruments.

VI. CONCLUSION

Due to the development of the research and innovations in the HCI domain many new ways of interactions with the computers or machines are emerging. Recently introduced Microsoft Kinect motion sensor is being used all around the world by many researches to find new ways of interactions. Human gesture based music composing is one of those emerging field of research. Here in this paper a research directive based on HCI to compose music based on human gestures is presented using the Kinect motion sensor. Three virtual instruments, namely, Guitar, Drum and Spider King are presented in this paper with the technical background and implementation details. Player's gesture is captured by Kinect sensor and MIDI signals are generated accordingly. Then the generated MIDI signals are passed to VST audio library to generate sound. A live performance was carried out using the presented virtual instruments along with other in-progress instruments. Current works are going on to capture detailed human gestures, increase the fps for better performance, integrate robust methods to improve quality and specially detailed subjective Quality of Experience measurements are required to assess the users and listeners experience.

ACKNOWLEDGMENT

Authors specially thank undergraduate student team members Hsu Chia Hua, Lin Jun Yang, Chuang Chih Kai, Kuo Ping Cheng and Lin Jia Yu who extend a great help to make this research a success. All the helps received from other MINE Lab members are also much appreciated.

REFERENCES

[1] Odowichuk, Gabrielle, Shawn Trail, Peter Driessen, Wendy Nie, and Wyatt Page. "Sensor fusion: Towards a fully expressive 3d music control interface." In *Communications, Computers and Signal Processing (PacRim), 2011 IEEE Pacific Rim Conference on*, pp. 836-841 , 2011.

[2] Mandanici, Marcella, and Sylviane Sapir. "DISEMBODIED VOICES: A KINECT VIRTUAL CHOIR CONDUCTOR." *Proceedings of the 9th Sound and Music Computing Conference*, Copenhagen, Denmark, pp. 271-276 ,2012.

[3] Qin, Ying. "A Study of Wii/Kinect Controller as Musical Controllers.", *Music Technology Area of Schulich Music School, McGill University*

[4] Trail, Shawn, Michael Dean, Tiago F. Tavares, Gabrielle Odowichuk, Peter Driessen, W. Andrew Schloss, and George Tzanetakis. "Non-invasive sensing and gesture control for pitched percussion hyper-instruments using the Kinect.", *The international Conference on New Interfaces for Musical Expression*, 2012

[5] Sentürk, Sertan, Sang Won Lee, Avinash Sastry, Anosh Daruwalla, and Gil Weinberg. "Crossole: A Gestural Interface for Composition, Improvisation and Performance using Kinect." *The international Conference on New Interfaces for Musical Expression*, 2012

[6] Wilschrey, Jacob, Cristian Rusu, Ivan Mercado, Rodolfo Inostroza, and Cristhy Jiménez. "Virtual Drums Based on Natural Interaction." In *Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on*, pp. 652-655 , 2012.

[7] Shamshiri, Sina. "A Kinect Air Guitar." (2012), *Department of Computer Science, University of Sheffield*.