Evaluating Gesture-Augmented Keyboard Performance

Qi Yang, Georg Essl

Computer Music Journal, Volume 38, Number 4, Winter 2014, pp. 68-79
(Article)

Published by The MIT Press

For additional information about this article
http://muse.jhu.edu/journals/cmj/summary/v038/38.4.yang.html

Access provided by University of Michigan @ Ann Arbor (20 Jan 2015 06:39 GMT)

Qi Yang and Georg Essl
Electrical Engineering and Computer Science
University of Michigan
2260 Hayward Ave
Ann Arbor, Michigan 48109-2121, USA
{yangqi, gessl}@umich.edu

Evaluating Gesture-Augmented Keyboard Performance

Abstract: The technology of depth cameras has made designing gesture-based augmentation for existing instruments inexpensive. We explored the use of this technology to augment keyboard performance with 3-D continuous gesture controls. In a user study, we compared the control of one or two continuous parameters using gestures versus the traditional control using pitch and modulation wheels. We found that the choice of mapping depends on the choice of synthesis parameter in use, and that the gesture control under suitable mappings can outperform pitch-wheel performance when two parameters are controlled simultaneously.

In this article we discuss the evaluation of a musical-keyboard interface augmented with free-hand gestures. Keyboards are musically expressive and are well suited for performance of discrete notes. Smooth adjustments of performance parameters that are important for digital synthesizers or samplers are difficult to achieved, however. Since the 1970s, such adjustments have often been achieved using pitch and modulation wheels at the left side of the keyboard. Contemporary sensor technology now makes it increasingly easy to offer alternative means to track continuous input. We augmented the musical keyboard with a 3-D gesture space using the Microsoft Kinect, an infrared-based depth camera for sensing and top–down projection for visual feedback. This interface provides 3-D gesture controls to enable continuous adjustments to multiple acoustic parameters, such as those found on typical digital synthesizers. Using this system, we conducted a user study to establish the relative merits of free-hand gesture motion versus traditional continuous controls.

Keyboard as Interface

It is easy to understand the popularity of the piano-style musical keyboard. A keyboard enables the player to address multiple discrete pitches concurrently and directly. In contrast, wind instruments produce a single pitch at a time and require complex chorded fingering. Further, in string instruments such as violin or guitar, polyphony is limited by the number of strings and by the geometry of the hand that provides the fingering. Also, the initial activation and reactivation of notes on a keyboard does not require preparation like stopping strings or activating multiple valves on a wind instrument.

Despite the ease of keyboard playing, it does come with drawbacks. After the onset of each note, the player has limited control of the quality of the sound. This is in contrast to bowed or wind instruments, which have a range of expressive timbre controls after the onset of each note. In the case of the traditional piano, limited timbre controls are provided by pedals to control the damping of the strings and, therefore, the amount of sympathetic resonance between strings.

The pipe organ does offer means of timbre control through knobs or tabs, commonly referred to as organ stops. The player pushes or pulls on the stops to activate or mute different sets of pipes, changing the timbre of the sound produced by actuating the keys. Pipe organs have developed a wide range of timbres that are enabled by different combinations of pipes, but the physical interface has seen little change, as the the stops are not designed for timbre changes while keys are being held down (more recent pipe organs allow configurations to be saved in advance and loaded during the performance), while the crescendo and swell foot pedals provide limited continuous timbre controls. Continuous control via a pedal is an interesting possibility, but we will not be considering it here.

Digital synthesizers, sampler instruments, and MIDI controllers usually feature a keyboard for pitch selection and note activation. For parameter adjustment during live performance, they traditionally feature one or two wheels (or in some cases, joysticks) next to the keyboard to control modulation

Computer Music Journal, 38:4, pp. 68–79, Winter 2014
doi:10.1162/COMJ_a_00277
© 2014 Massachusetts Institute of Technology.

and/or pitch bend. We wanted to see if open-air hand gestures provide better means of adjustment during live performance.

It is easy to perform continuous gestures using hand motions in space, hence they make a good candidate for continuous timbre control in real time, especially in improvised music. When performing on the keyboard, the player can quickly lift their hand from the keyboard and move into and out of the gesture space. Recent advances in sensor technology make gesture sensing easy and affordable. Our prototype system uses an off-the-shelf depth camera to track a range of hand motions, positions, and gestures in real time, making it suitable for live performance and the goals of our project. The sensing of position and hand width creates a space with multiple, continuous degrees of freedom, allowing multiple parameters to be controlled simultaneously. The gesture space also allows either hand to be used for hand-gesture controls, in contrast to the fixed location of pitch and modulation wheels on the far left side of a standard MIDI keyboard.

Related Work

This article brings together two important strands in the design of new musical instruments: the augmentation of established, traditional musical instruments, and the use of gestures for continuous control of musical instruments. The prior art in both these fields is extensive, and we refer the reader to comprehensive reviews (Paradiso 1997; Miranda and Wanderley 2006).

How best to support continuous control in conjunction with the keyboard interface is a longstanding problem and has seen many proposals. When designing the first hard-wired commercial analog synthesizers, Bill Hemsath, in collaboration with Bob Moog and Don Pakkala, invented the pitch and modulation wheels (Pinch and Trocco 2004), which became the canonical forms of continuous control on electronic keyboard interfaces ever since. Early analog synthesizers had many continuous controls via rotary potentiometers and sliders, but in many canonical cases the pitch and modulation wheels were the only ones that survived the transition to digital synthesizers. Still, continuous control in keyboard performance remained an important topic. Moog, later with collaborators Tom Rhea and John Eaton, experimented for decades with prototypes to add continuous control to the surface of the keys themselves (Moog 1982; Moog and Rhea 1990; Eaton and Moog 2005). This idea has also been explored by others (Haken and Tellman 1998; McPherson and Kim 2010; Lamb and Robertson 2011; McPherson 2012).

Another idea that has been proposed is the augmentation of the action of the key itself. The classic aftertouch, where extra levels of control are available once the keys are fully depressed, is an early example of this (Paradiso 1997). Precise sensing of key position can be achieved through various means, such as optical interruption sensing (Freed and Avizienis 2000). More recently, McPherson and Kim (2011) described the augmentation of traditional piano keys through a light-emitting diode (LED) sensing mechanism that is capable of inferring performance parameters from the key action. This, in turn, can be used to augment performance.

More narrowly, our work augments the musical-keyboard interface with continuous hand gesture control in open space. Perhaps the most famous previous example of open gesture control is the Theremin, which uses capacitive sensing. Open-space gestures can be tracked using different technologies. Our prototype uses visual sensing via depth cameras. Visual tracking of hands has been explored previously (Gorodnichy and Yogeswaran 2006; Takegawa, Terada, and Tsukamoto 2011). Concurrently to our work, Aristotelis Hadjakos (2012) used the Kinect for hand, arm, and posture detection in piano performance. The key differences between his work and ours is that we consider the visual tracking for generic gesture interactions that augment piano performance, whereas Hadjakos is interested in sensing for medical and pedagogical purposes. Hence the system does not include visual feedback. Visual feedback did appear in work by Takegawa, Terada, and Tsukamoto (2011), who projected score and fingering information to guide early piano pedagogy. William Brent (2012) presented a visual tracking system based on infrared blob detection. In that work, an ordinary camera is suspended above a piano together with an array of infrared lights. The depth information is then inferred from the size of the blob. The purpose of that work was to detect central parts of the performer to allow extra control parameters to be derived from the position of the hand center relative to the lower arm. The author reports problems with independence of the control parameters thus detected. Our system avoids this problem by directly sensing position, in a 3-D volume, of the field of view using a depth camera.

In addition, literature exists on evaluation methodologies for designing digital music instruments. Notably, Wanderley and Orio (2002) suggested using musical tasks and adapting human–computer interaction (HCI) methodologies for evaluating input devices to the area of evaluating musical instruments. Sile O'Modhrain (2011) proposed a framework where the roles and goals of different stakeholders (such as the audience, performer, and the manufacturer) of the musical instruments are all considered for the evaluation of instrument designs. Sergi Jordà (2004) proposed a measure of musical instruments' efficiency based on the expressive power and diversity of the instrument and on the complexity of the input interface. Our evaluation draws ideas from Wanderley and Orio (2002) by using HCI performance metrics of input devices with a well-defined musical task.

Implementation

Our system uses a Kinect depth camera and a video projector installed above a MIDI keyboard, facing down toward the keyboard (see Figure 1). The Kinect depth camera, projector, and keyboard are connected to a single computer that processes the sensor data from the camera and the MIDI data from the keyboard, while controlling a software synthesizer to produce the sound. A white projection surface placed above the keyboard allows a clear view of the projected visual feedback.

The Kinect depth camera is used to capture three-dimensional data from the gesture space, in the form of an 11-bit monochrome, 640 × 480-pixel video stream sampled at 30 Hz, with the brightness indicating the distance from the camera. This video stream is passed through background and noise removal and fed into a blob-detection algorithm using OpenCV (Culjak et al. 2012). The chosen blob-detection algorithm was proposed by Chang, Chen, and Lu (2004). It uses a connected-pixel labeling strategy to derive contour components, including the external contour of the blobs to be detected. Using the initial keyboard setup as a background, the image is passed through blob detection (after first removing the background). We can then detect the presence and position of the player's arms as they enter the gesture space. The player's hand positions are isolated by capturing the extremity of their arms, and we use the centroid of the player's hands as the position. Using the center of their hand as reference, we also measure the distance to the camera, which in this case corresponds to the hand's height. (See Figure 2 for the stages of processing data from the depth camera.) At the same time, we can also compute the widths of the hands to see if they are open or closed. The trajectory of the hand motion, inferred from this position, is passed through an averaging filter of five frames to remove the jitter caused by noise from the depth camera.

Using the Processing framework (Reas and Fry 2006) as a bridge, the hand-position data are mapped to MIDI messages for timbre control, to be sent to a software synthesizer (see Figure 3). MIDI

Figure 1. Configuration of the augmented keyboard.

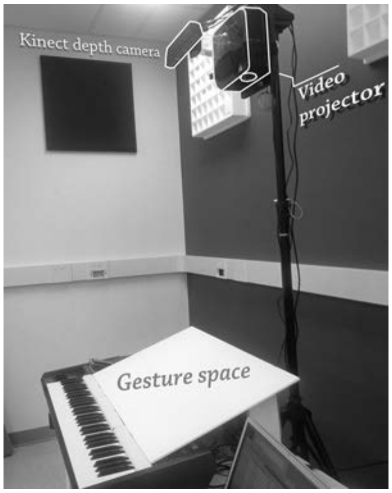


Figure 2. Kinect video stream (a), depth-camera stream (b), and image after background removed with hand position derived from blob detection (c).

Figure 3. Data flow of the augmented keyboard.

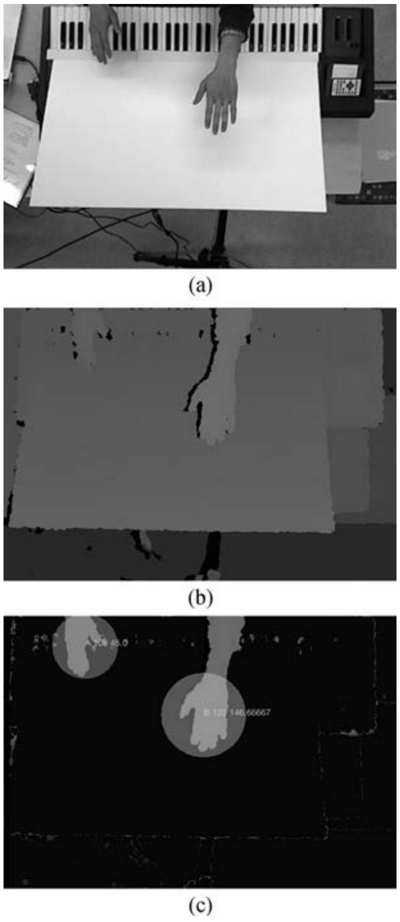


Figure 2

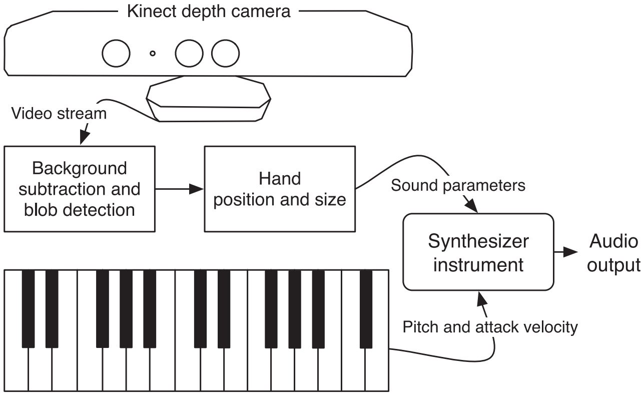


Figure 3

note-number and attack-velocity messages from the keyboard are also sent to the synthesizer. We also use Processing for visual feedback (see Figure 4), which is projected onto the surface beneath the gesture space. The detected location of the player's hands is displayed, as are (1) vertical and horizontal bars showing the gesture axes that are currently active, with their current values, and (2) circles showing both the size of the palm and the height of the player's hands.

The overall latency in the system from the Kinect sensor to visualization and MIDI control messages is estimated to be 174 msec, with a standard deviation of 23 msec, less than the 33 msec it takes for the Kinect sensor to refresh. (Note that latency measurements were conducted after an operating system update that needed to be made after the study, and these values may not fully reflect those at the time of the original user study described below.)

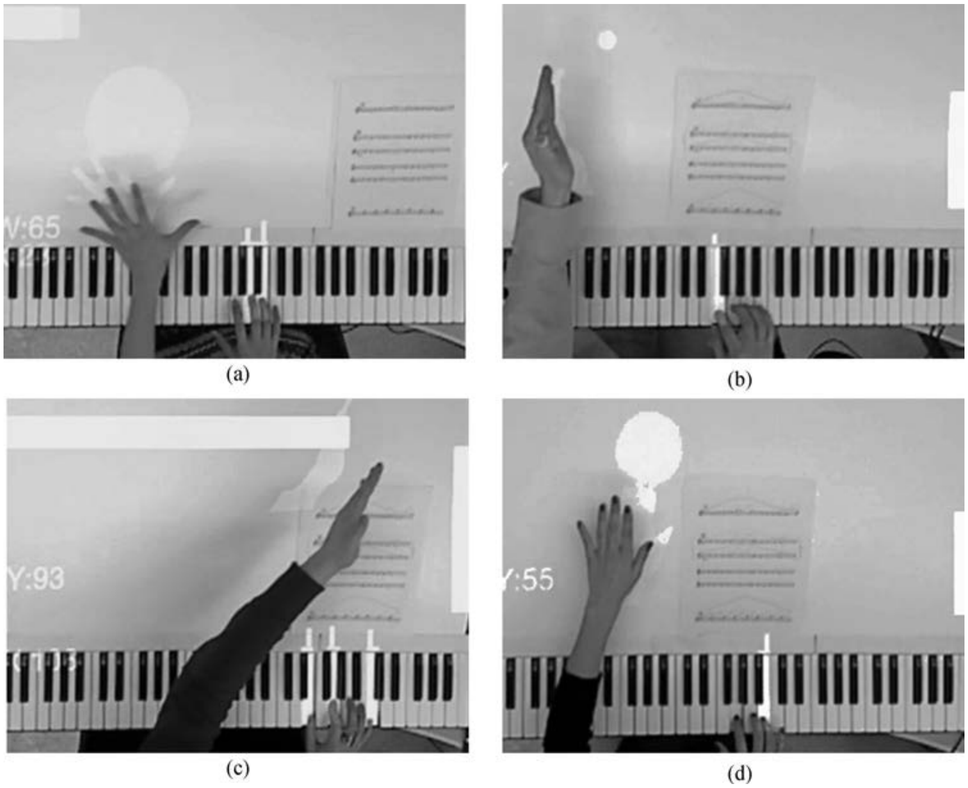
Extended Playing Technique

With our system, a keyboard player can play normally using both hands on the keyboard, just as with any traditional keyboard. For continuous gesture controls, the player can move either hand into the gesture space immediately above and behind the keyboard (see Figure 1) while using the other hand to continue playing at the same time. The gesture space can also be configured to be directly above the keys on the keyboard itself, so any wrist motion or other hand gesture during normal playing can be captured and used for continuous control.

Study with Human Subjects

We conducted a user study to evaluate how our system performs versus the physical controls featured on conventional electronic keyboards. In addition, we wanted to examine the mapping between gesture types and timbral parameters, as well as to study ergonomic issues such as fatigue, learnability, and enjoyment.

Figure 4. Visual feedback generated by the system, based on hand detection.



Experiment Design

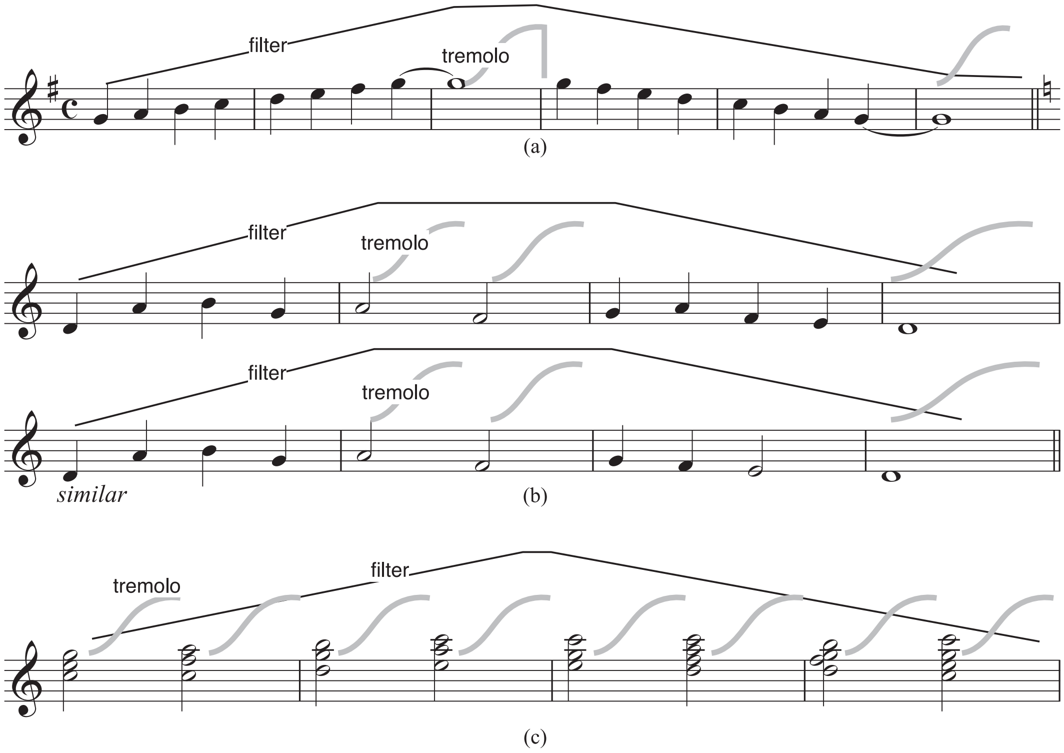
Our study consisted of two parts: a playing session on the augmented keyboard, which lasted 45–50 minutes, and an exit questionnaire.

To test continuous timbre manipulation after note onset, we asked each participant to play three simple passages of monophonic melodies and chords on the keyboard that required only a single hand to play. At the same time, the participant was to move the other hand in the gesture space to control one or two parameters of the synthesizer that affect the timbre of the sound produced.

As effects to be applied to a generic synthesizer sound, we chose a low-pass cutoff filter (henceforth "filter," for brevity) and a tremolo effect (an oscillation in amplitude but not pitch). The two effects were chosen because they have distinct timbral results, even when applied concurrently. A musical score of the passage is provided (see Figure 5), with timbral effects marked as curves above the notes, using vertical position to show the amount of the effect. The filter effect is notated as a slowly increasing or decreasing timbre change, and tremolo is notated as a gradual increase to the maximum with a sharp cutoff soon after.

For comparison, we chose three distinct gestural axes to map to the two effects, as well as two physical wheel controls on the electronic keyboard. We detected the left-to-right movement of the player's hand (X), the front-to-backback movement (Y), and the width of the hand (W, which changes when the hand is opened or closed, or alternatively when the wrist is turned). For physical control, we detected the pitch-bend wheel (wheel1) and modulation wheel (wheel2) on the keyboard. These were then mapped to one or two timbral effect parameters. As with most MIDI keyboards, on the keyboard used for the experiment the pitch-bend wheel is spring loaded and the modulation wheel is

Figure 5. Notation of timbral effects used for our study. Three passages of varying difficulty were used.



not, and zero timbral effect is always mapped to the neutral position on the spring-loaded wheel.

We tested all combinations of mapping one or two gestures to one or two effects using a full factorial design. We did the same with mapping physical wheel controls to effects, with a total of ten configurations of control scheme mapped to a single effect, and eight configurations of two controls mapped to two effects (see Table 1).

At each session, the participant was first asked to fill out the screening survey, followed by a learning period of up to five minutes, in which the participant played the passages without using any timbral effects. Then the configurations were presented. Owing to the length of each playing session, we anticipated that not all participants would be able to complete all 18 configurations. As a result, we first presented (in randomized order) only the configurations lacking an X gesture. Then, only if there was time remaining, the configurations containing X gestures were presented in randomized order. In practice, out of the 22 participants, only two were unable to complete all the configurations. For consistency, we kept the partially randomized presentation order for all participants.

For each configuration, the participants were given one to two minutes to play the passage with the notated timbral effects, and then to play one last time while their performance was recorded. This procedure was repeated for all three passages. New configurations were introduced without pause after each one was finished. Although our system makes no distinction between the left and the right hand, for consistency the participants were asked to use the right hand for playing the melody and the left hand for timbre control. After completing all the configurations, the participants were invited to improvise timbral effects on music of their choosing, or to play one of the test passages using their own timbral effects, using a control configuration of their own choice. Then they were asked to fill out the exit questionnaire. In the questionnaire we

Table 1. Configurations of Mapping Gestures and Physical Wheels to Effects

Configuration	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*
Filter	Y		W		Wh1		Wh2		X		Y	W	Wh1	Wh2	X	Y	X	W
Tremolo		Y		W		Wh1		Wh2		X	W	Y	Wh2	Wh1	Y	X	W	X

The columns indicate the different combinations of mapping gestures (X, Y, and W) and physical wheels (Wh1: pitch bend; Wh2: modulation) used to control the two effects effects (low-pass filter and tremolo). Empty cells indicate that the effect was not used.

used five-point Likert-scale questions to assess, for each configuration, ease of learning, expressiveness, fatigue, fun, and personal preference. We also used the ISO 9241-420 questionnaire (ISO 2011) to evaluate potential discomfort.

Participants

We recruited undergraduate and graduate students and faculty members at the University of Michigan. Twenty-two participants took part in the study, of whom 45 percent were and 80 percent were between the ages of 19 and 25 years. All participants had experience with keyboard instruments, with more than 80% having five or more years playing experience. One-third of the participants were currently studying music at the college level. Participants were compensated for their time.

Results

We recorded MIDI performance data from the keyboard for each configuration, as well as MIDI controller messages from the mapped gestures or physical controls. We then used this information to compute task completion time, error, and smoothness of continuous controls, which will now be discussed.

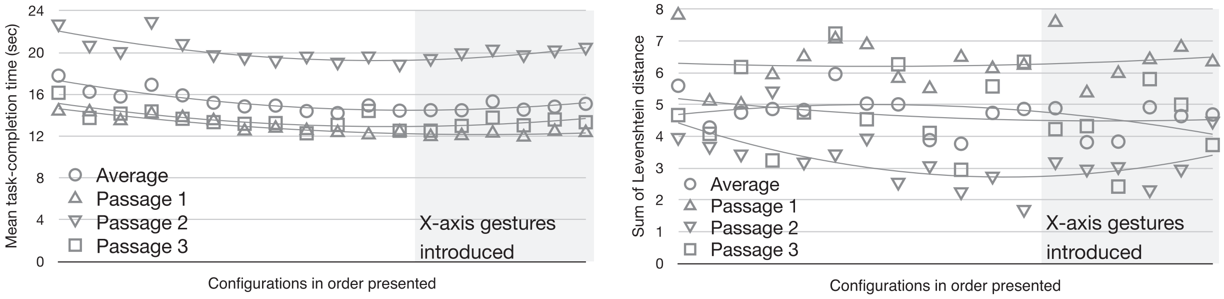
Task Completion Time

We measured the time each participant took to play each passage for the final time after one or two practices. Based on observation, participants encountering difficulties playing with hand gestures stuttered or paused more often, and were likely to take longer than the normal tempo they established during the practice phase. Task completion time can capture performance degradation due to cognitive load, motor performance difficulty, and other, related performance characteristics. Hence it serves in our view as a useful measure of performance competence.

We discarded data from five participants because of technical problems in recording data. After running a two-factor analysis of variance (ANOVA) on the task completion time of single-effect configurations (where one control is mapped to a single effect), we found that the completion time had high variance overall. Neither controls nor effect types had a statistically significant ($p > 0.05$) effect on the task completion time.

When two gestures or physical controls are mapped to two effects simultaneously, we found that passage A and B exhibited no significant difference between different control type and parameters. It is likely that this can be attributed to the fact that in neither of the two passages did two parameters need to be adjusted concurrently (see Figure 5). One parameter only needed to be held at a constant value while the other was adjusted. For passage C we found that controls have a significant effect ($F = 3.7$, $p < 0.0178$) on completion time. In particular, t tests show that the combination of X-filter and W-tremolo or Y-tremolo are better than many physical wheel configurations ($t = 4.11$, $p < 0.0008$, using $p < 0.05/N$ after Bonferroni multiplicity correction as the threshold of significance). The combination of X-filter and W-tremolo was also significantly better than X-tremolo and W-filter ($t = 4.19$, $p < 0.0007$), with no other configurations showing significant

Figure 6. Learning curves with polynomial curve fit, with some effect on task completion time (a), little effect on edit distance (b).



(a) (b)

differences. This is likely because the passage requires two parameters to be adjusted concurrently.

Although many of the configurations that use the *x*-axis are better than physical wheels, we cannot claim that the difference is statistically significant, because X-gestures were confounded by not being presented with other mappings fully randomly. The measured effect could be explained in multiple ways; one possible explanation is improvement over time.

We investigated this possibility by inspecting progression of task completion time chronologically in the order of presentation (see Figure 6a). The curve does show a slight learning effect during the first ten configurations presented. After that, before the X-gestures are introduced in the last six configurations, there is little improvement. In fact, the increase in time for passages after the first ten configurations makes it implausible for X-gestures to be confounded by learning effects, which led to a decreased performance time. This suggests that the observed advantage of X-gestures over physical controls may be a real effect. This is not conclusive, however, as the slight increase at the end can also suggest fatigue after playing for about 35 minutes.

Levenshtein Distance

We adopted Levenshtein distance (also sometimes called "edit distance"), an algorithm to compute the minimal difference between two strings in terms of basic edit operations (Levenshtein 1966), as a measure of the errors participants made during playing. Similar to task completion time, errors may correspond to difficulty in performing the continuous timbral effects. For each recorded performance, we compare the MIDI note data with a "gold standard" performance derived from the score. Each passage is considered as a sequence of notes, and the Levenshtein distance between the recording and the gold standard is computed, as the number of mistakes (missing a note, inserting an extra note, or playing the wrong note) the participant made.

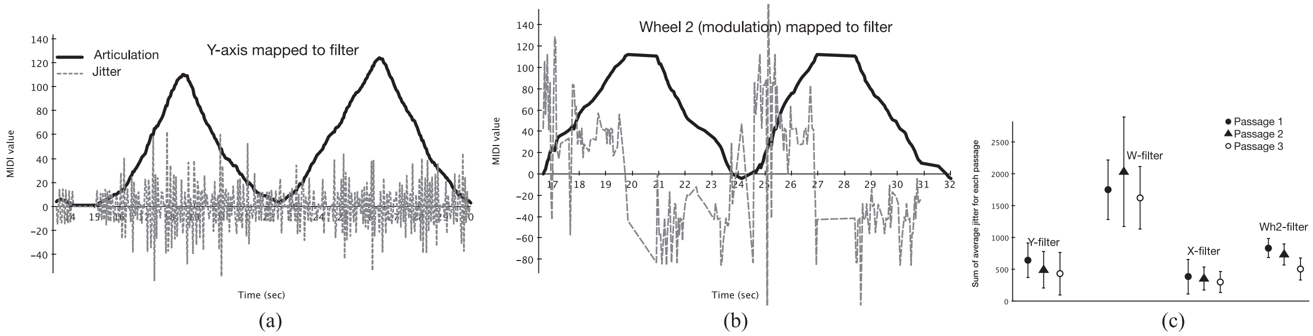
Because participants performed many passages with few errors, and some passages with no errors at all, the data are sparse. We aggregated errors from all three passages; a two-factor ANOVA shows no strong effect on either control schemes used or the effect mapped to. Similarly to task completion time, there are no significant differences for single-effect configurations. In the case of dual-effect, X-filter and Y-tremolo performed significantly better than the Y-tremolo and W-filter configuration and the configurations with only one physical wheel ($t = 3.58$, $p < 0.0028$), with no other significant differences.

Similarly to task completion time, we examined the possible effects of presentation order on Levenshtein distance. We found no clear effects of learning; only passage 2 showed some effects of presentation order (see Figure 6b). The absence of clear effects in Levenshtein distance after the first eight configurations further supports the possibility that the advantage of X-gestures may be real.

Smoothness of Continuous Control

We analyzed the MIDI controller data derived from either the hand motion or the physical wheels, to measure the smoothness of the continuous controls.

Figure 7. Jitter in typical gesture controls (a) and physical wheel controls (b). Jitter, computed as numerical second derivative, is scaled down by a factor of 100 to fit visually. Wheel control exhibits significantly more jitter (c).



Jitter in control (manifested as fluctuation in the controlled parameter) suggests possible difficulty in operating the control, or stumbling when the subject was confused by the mappings, or fatigue. Because the participants were told to make timbral effects gradually and smoothly, as notated, the presence of unintended jitter should reflect the quality of the performance.

The MIDI controller data were sampled at roughly 25 Hz and had a resolution of only seven bits (128 discrete values). To measure the jitter in the continuous controls, we used standard three-point numerical differentiation to estimate the second derivative of the effect values, thus measuring changes in acceleration. By cursory observation, the MIDI controller data derived from the Kinect sensor have a significant amount of noise, even after the necessary smoothing (see Figure 7), whereas physical wheels exhibit no noise when they are not actuated by the player.

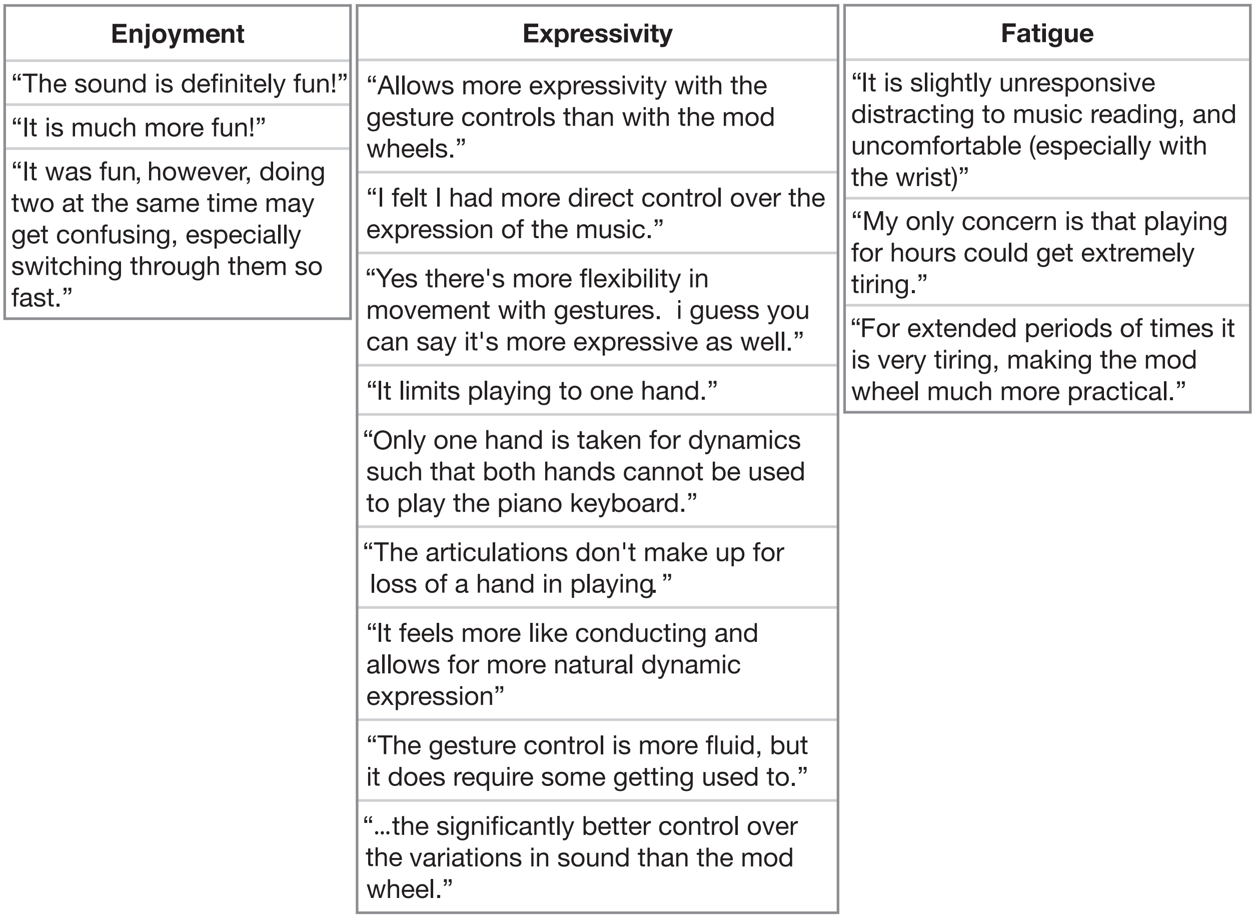
Because of technical problems, we recorded and analyzed the gestures and modulation wheel mapped to the low-pass filter for only nine subjects. Comparing only jitter in single-effect configurations, an ANOVA shows the control scheme has a significant effect ($F = 31.5$, $p < 0.000001$), W gestures have significantly more jitter than all others ($t = 3.97$, $p < 0.0063$, see Figure 7c), X gestures have less jitter than using modulation wheel ($t = 4.81$, $p < 0.0019$), and no other significant differences. Given that the Kinect sensor is generally noisier than physical wheels, the advantage of gestures producing continuous timbral effects with less jitter is significant. Our experimental setup did not have a control setup to account for noise in the wheels' potentiometers versus optical or vision sensing; we do observe, however, that the wheels have no noise when they are not being moved. Nevertheless, It should be noted that because W gestures exhibit more noise, the difference cannot be due to sensor noises in physical wheel controls.

Exit Survey

After participants completed the playing session, they were asked to fill out an exit survey consisting of five Likert-scale questions for each configuration they played, an ISO 9241-420 "Assessment of Comfort" evaluation, and a set of open-ended questions for feedback. Owing to the large number of configurations tested, we asked participants to evaluate the comfort of gesture controls in comparison to physical wheels in general.

We analyzed the five-point Likert-scale questionnaires using the pairwise Mann-Whitney U (MWU) test. The MWU only shows significance for dual-effect configurations, with gestures being easier than physical wheels ($U = 100$, $p < 0.0392$). Within gestures, W-tremolo is easier to learn than W-filter ($U = 94$, $p < 0.0245$). Most configurations are easy to learn. On expressiveness, participants responded that single-effect configurations were less expressive than dual-effect ($U = 86$, $p < 0.04257$). Within dual-effect configurations, using physical wheels were worse than some gestures ($U = 105$, $p < 0.0367$), with no other significance. When asked if the configuration was fun to play, 57% responded positively (i.e., that the configuration was fun to play), and 11% negatively. Multiple effects were always more

Figure 8. Excerpts of open-ended responses from participants.



Enjoyment

"The sound is definitely fun!"

"It is much more fun!"

"It was fun, however, doing two at the same time may get confusing, especially switching through them so fast."

Expressivity

"Allows more expressivity with the gesture controls than with the mod wheels."

"I felt I had more direct control over the expression of the music."

"Yes there's more flexibility in movement with gestures. i guess you can say it's more expressive as well."

"It limits playing to one hand."

"Only one hand is taken for dynamics such that both hands cannot be used to play the piano keyboard."

"The articulations don't make up for loss of a hand in playing. "

"It feels more like conducting and allows for more natural dynamic expression"

"The gesture control is more fluid, but it does require some getting used to."

"...the significantly better control over the variations in sound than the mod wheel."

Fatigue

"It is slightly unresponsive distracting to music reading, and uncomfortable (especially with the wrist)"

"My only concern is that playing for hours could get extremely tiring."

"For extended periods of times it is very tiring, making the mod wheel much more practical."

fun than a single effect, regardless of the control scheme ($U = 82$, $p < 0.04426$). In addition, dual-effect configurations with W-tremolo were more fun than other configurations ($U = 113$, $p < 0.0226$). When the participants were asked to rate configurations based on personal preference, the MWU shows W-tremolo to be least preferable among single-effect configurations ($U = 102$, $p < 0.02994$). For dual-effect configurations, however, W-tremolo was considered preferable to configurations where other gestures were mapped to tremolo.

For the ISO 9241-420 assessment of comfort, participants were asked about fatigue of gestures versus physical wheels in general. The gestures are considered better in terms of force required, smoothness, accuracy, and general comfort, with no significant differences in other factors. There is a clear tradeoff between finger and arm fatigue, with physical wheels causing more finger fatigue, whereas gestures cause more arm fatigue. No significant differences in fatigue are found between the individual configurations.

On the last open-ended question, participants mentioned that gestures improve expressiveness and are fun to play (see Figure 8). They also mentioned that taking one hand away for timbre control limits the complexity of the music that can be played and causes more fatigue. Participants describe gesture controls as "natural" or "fluid," but they also stated that different mappings can be confusing to learn, especially in the short time given. Although our system has an estimated latency of 174 msec, only one participant mentioned that the system could be "slightly unresponsive," probably because of the latency.

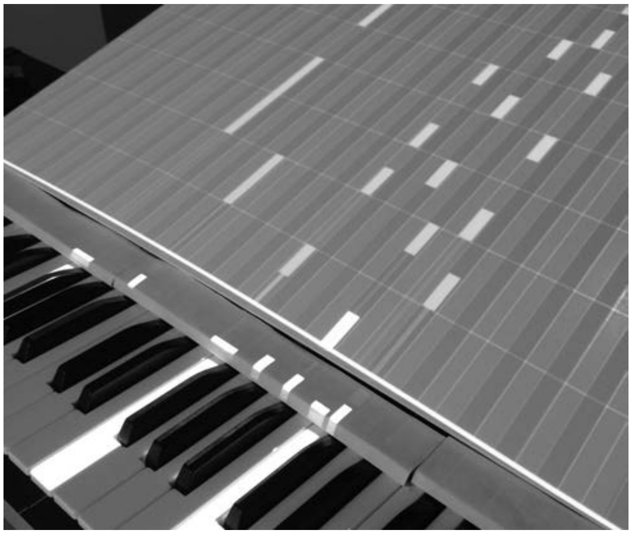
Objective metrics (task completion time, Levenshtein distance, jitter) that measure the participant's performance with the system suggest that, when multiple parameters are controlled concurrently, there are advantages in using gestures over physical wheels, as long as the gesture mappings are chosen well. The difference is insignificant, however, when only a single effect is mapped or when two parameters are not adjusted concurrently. We also found that some gesture mappings perform better than others, particularly when W is mapped to tremolo in any dual-effect configurations. This suggests that the action of opening the hand or turning the hand to affect W may be a good match to the tremolo effect. The results from subjective surveys agree with this finding. The subjective surveys also show that participants find the augmented keyboard generally fun and expressive, and that there is a tradeoff between finger and arm fatigue caused by performing continuous timbral effects, depending on whether gestures or physical wheels are used.

Conclusion

We augmented the musical keyboard with a gesture space, using a depth camera for sensing and top–down projection for visual feedback of gestures. We found that improved performance is dependent on the particular mapping between gesture and sound effect. This suggests that the choice of mapping is critical, which should be a focus for future research. As an example, using a change of hand width for a tremolo effect shows significant improvement in performance compared with traditional pitch and modulation wheels.

Our system has a wide range of potential applications. The same sensing and visual feedback setup can be adopted for other styles of playing or for applications such as pedagogy. For example, in a pedagogical scenario the hand position data can be used to display contextual information around the learner's hand on the keyboard. A guided improvisation system can show a choice of future harmonies given a history of harmonic progression, by highlighting the appropriate keys to play near the learner's hand. When not used for gesture, the large gesture space can be used to show instructional information, such as video, an adaptive musical score, or a "waterfall" notation of the music (see Figure 9).

Figure 9. A prototype pedagogical game using a "waterfall" musical notation.



We can also envision a range of performance techniques using this technology. One can imagine using the gesture space as a virtual harp by waving one's hand in midair. Furthermore, the gesture space can be used to manipulate a wide range of parameters, expanding on the rich timbre controls of a pipe organ. Additionally, a range of novel abstract gesture performances can be realized using this system.

We see several future directions for further research. Details of pedagogical benefits have yet to be studied. Also, in this work we have not investigated the interplay between visual feedback and gesture detection. Since the submission of this article for publication in *Computer Music Journal*, the authors have published a paper exploring the visualization aspect of the system (Yang and Essl 2013). Finally, the current system can be extended in various ways. For example, the projection surface can be made into a multi-touch surface, enabling more detailed contact tracking.

References

Brent, W. 2012. "The Gesturally Extended Piano." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 332–335.

Chang, F., C.-J. Chen, and C.-J. Lu. 2004. "A Linear-Time Component-Labeling Algorithm Using Contour Tracing Technique." *Computer Vision and Image Understanding* 93(2):206–220.

Culjak, I., et al. 2012. "A Brief Introduction to OpenCV." In *Proceedings of MIPRO: 35th International Convention on Information and Communication Technology, Electronics and Microelectronics*, pp. 1725–1730.

Eaton, J., and R. A. Moog. 2005. "Multiple-Touch-Sensitive Keyboard." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 258–259.

Freed, A., and R. Avizienis. 2000. "A New Music Keyboard Featuring Continuous Key-Position Sensing and High-Speed Communication Options." In *Proceedings of the International Computer Music Conference*, pp. 515–516.

Gorodnichy, D. O., and A. Yogeswaran. 2006. "Detection and Tracking of Pianist Hands and Fingers." In *Proceedings of the Third Canadian Conference on Computer and Robot Vision*, p. 63.

Hadjakos, A. 2012. "Pianist Motion Capture with the Kinect Depth Camera." In *Proceedings of the Sound and Music Computing Conference*, pp. 303–310. Available online at smcnetwork.org/node/1707. Accessed July 2013.

Haken, L., and E. Tellman. 1998. "An Indiscrete Music Keyboard." *Computer Music Journal* 22(1):30–48.

ISO (International Organization for Standardization). 2011. "Appendix D: Assessment of Comfort." In *ISO 9241-420:2011 Ergonomics of Human–System Interaction, Part 420: Selection of Physical Input Devices*. Geneva: International Organization for Standardization, pp. 28–31.

Jordà, S. 2004. "Instruments and Players: Some Thoughts on Digital Lutherie." *Journal of New Music Research* 33(3):321–341.

Lamb, R., and A. Robertson. 2011. "Seaboard: A New Piano Keyboard-Related Interface Combining Discrete and Continuous Control." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 503–506.

Levenshtein, V. I. 1966. "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals." *Soviet Physics Doklady* 10(8):707–710.

McPherson, A. 2012. "TouchKeys: Capacitive Multi-Touch Sensing on a Physical Keyboard." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 53–56.

McPherson, A., and Y. Kim. 2010. "Augmenting the Acoustic Piano with Electromagnetic String Actuation and Continuous Key Position Sensing." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 217–222.

McPherson, A., and Y. Kim. 2011. "Multidimensional Gesture Sensing at the Piano Keyboard." In *Proceedings of the Annual Conference on Human Factors in Computing Systems*, pp. 2789–2798.

Miranda, E., and M. Wanderley. 2006. *New Digital Musical Instruments: Control and Interaction Beyond the Keyboard*. Middletown, Wisconsin: A-R Editions.

Moog, R. A. 1982. "A Multiply Touch-Sensitive Clavier for Computer Music Systems." In *Proceedings of the International Computer Music Conference*, pp. 601–605.

Moog, R. A., and T. L. Rhea. 1990. "Evolution of the Keyboard Interface: The Bösendorfer 290 SE Recording Piano and the Moog Multiply-Touch-Sensitive Keyboards." *Computer Music Journal* 14(2):52–60.

O'Modhrain, S. 2011. "A Framework for the Evaluation of Digital Musical Instruments." *Computer Music Journal* 35(1):28–42.

Paradiso, J. A. 1997. "Electronic Music: New Ways to Play." *IEEE Spectrum* 34(12):18–30.

Pinch, T. J., and F. Trocco. 2004. *Analog Days: The Invention and Impact of the Moog Synthesizer*. Cambridge, Massachusetts: Harvard University Press.

Reas, C., and B. Fry. 2006. "Processing: Programming for the Media Arts." *AI and Society* 20(4):526–538.

Takegawa, Y., T. Terada, and M. Tsukamoto. 2011. "Design and Implementation of a Piano Practice Support System Using a Real-Time Fingering Recognition Technique." In *Proceeding of the International Computer Music Conference*, pp. 387–394.

Wanderley, M. M., and N. Orio. 2002. "Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI." *Computer Music Journal* 26(3):62–76.

Yang, Q., and G. Essl. 2013. "Visual Associations in Augmented Keyboard Performance." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 252–255.