Towards Touch Screen Live Instruments with Less Risk: A Gestural Approach

Edward Jangwon Lee
Graduate School of Culture Technology
KAIST
noshel@kaist.ac.kr

Woon Seung Yeo
Division of Digital Media
Ewha Womans University
woony@ewha.ac.kr

ABSTRACT

Although touch screen interfaces such as smartphones and tablet PCs have become an important part of our life and are being used in almost every situation, these interfaces are facing some difficulties in being used in live musical performances, despite the numerous benefits they can musically offer. Among those difficulties, we identify and focus on the visual dedication requirement of interaction and nevertheless high risk of making mistakes, and design a simple musical interface aiming to alleviate these problems. In order to reduce visual dedication, we employ larger on-screen controls. To reduce risk of mistakes, we choose a gestural approach and incorporate plucking gestures, which require users to pull and release a touch after initiated. The interface is qualitatively tested, focusing on playability, visual dedication, and risk of making mistakes. While playability and risk received positive feedbacks, reducing visual dedication received partial agreement and seems to require further investigation. Although the interface is yet immature and too simple to be used on stage, we believe that identifying and solving the problems that touch screens have while being used in live situations is meaningful and valuable to discuss.

1. INTRODUCTION

The introduction of touch screen interfaces such as smartphones and tablet PCs, alongside with their numerous novel, fast and accurate sensors, has changed our lives in a way that we have never imagined before. These new interfaces seem to be capable of almost anything and there are applications that are used in both casual and professional fields, leading smartphones to become an indispensable part of our life.

Many researchers and artists have seen great live music possibilities in touch interfaces, and many results can be found throughout the music computing literature. Alongside with new protocols such as OpenSound Control (OSC) [1], touch interfaces can be hooked into a network and serve as a control surface with low latency using softwares such as Control [2], enabling composing and performing in a way we have never imagined before.

However, compared to the appealing features and creative opportunities touch screen devices can offer, it seems that these devices are not gaining enough popularity on live, on-stage situations as an instrument. We believe that identifying the musical obstacles that touchscreen devices are facing and designing digital musical interfaces in a manner that can possibly overcome those obstacles will surely further promote the usability of them on stage. One of the major obstacles might be the risk of making mistakes. Touch screens highly suffer from accidental touches, which cannot be afforded to happen during live performances. This problem can be relieved by a gestural approach, since gestures that can too easily trigger interactions may be the main reason of accidental inputs. Incorporating plucking gestures [3], which requires a marginal cost of interaction while offering additional sound parameters, might be a possible remedy for this. Section 1.1 discusses the difficulties touch screens have in being a reliable on-stage instrument.

While many other issues might exist, this paper identifies and discusses a number of these obstacles, and presents a simple digital musical interface for user testing. Although this interface is yet too simple to be used in serious live situations, we hope that this piece of work provides a discussion point in finding and solving the problems that touch screen devices have in being selected in live situations.

1.1 Touch Screens and Live Performances

Despite the great possibilities that touch screen devices can offer, such as networking and versatile user interface programming, why are these devices not widely used enough in live performances? Among numerous possible reasons, we present a few of them that suit to our research. First, touch screens mostly require heavy visual dedications, unlike traditional instruments. Geiger (2006) states that throughout the history of instruments, only few instruments rely on visual feedback [4]. Moreover, in collaborative ensemble situations, the performer must interact with other players and possibly the audience – making visual dedication to interfaces even further costly. Upon this reasoning, Walther et al. (2013) devised a MIDI controller based on swipe-gestures using the whole screen as a single canvas, thereby reducing the required visual effort on finding the exact position to touch [5]. Another good example addressing visual dedication problems is CarPlay 1, which includes sev-

Copyright: ©2014 Edward Jangwon Lee et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 Unported License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

1 CarPlay by Apple (http://www.apple.com/ios/carplay) addresses visual dedication problems by employing voice inbuilt car controls to manipulate touch screen smartphones while driving.
eral ideas in enabling touch screen manipulation during driving.

Another reason is the risk of making mistakes. Due to the
nature of activation upon touch, only a slightest contact
with the screen might trigger undesired audio feedback. In
live performances, such risk might be intolerable. Pirho-
inen et al. (2002) noticed this slight tapping problem while
user testing with mobile devices while moving around
and proposed setting a specific threshold of touch movement
before accepting the tap as a valid input [6].

Finally, interactions between player and touch screens
are not revealing enough to be interesting on stage. Tra-
ditional instruments, which require physical force to gen-
erate sound, present a clear link between playing gestures
and sound. Therefore, a popular design method in digital
musical interfaces is to make both manipulations (inputs)
and effects (outputs) as visible as possible [7]. However,
touch screens do not provide such links, and the required
gestures to generate sound are usually physically small.
Additionally, their requirement of the player to visually
concentrate on the screen and risks of making mistakes
even further decrease the player’s movement.

In this paper, we propose an alleviation of these problems
by incorporating plucking gestures into touch screen based
musical instruments. An in-depth discussion of plucking
gestures addressing these problems is to be presented in
section 2.

1.2 Plucking Gestures

Figure 1. Threefold process of plucking gestures and plucking buttons.

Plucking is a touch screen input gesture that resembles
the plucking of guitar strings [3]. Plucking on strings is
a threefold process: a) hold a string with finger, b) apply
force on string by moving finger, and c) release string to
generate sound. We adopt this gesture on touch screens
as follows: a) start touch in a UI control, b) move touch,
and c) release touch to trigger interaction feedback. Touch
move distance less than a specific threshold or moving back
into the control cancels the interaction. Figure 1 illustrates
how plucking gestures work on touch screens.

Although plucking gestures are somewhat complex com-
pared to ordinary tapping, several advantages exist to com-
pensate the increased cost of interaction. First, plucking
gestures enable onscreen buttons to be touched without
triggering outcome. Many keyed or stringed instruments
allow the player to place their hands (or picks, bows, and
so on) on the keys or strings without producing any sound.

Plucking gestures can implement this feature on touch screens
by requiring users to move the touch out of the target con-
tral before activating it. This also helps in reducing visual
dedication, as placing hands on controls can offer more
comfort in remembering the positions of controls than not
touching the device.

Second, the risk of making mistakes is reduced. Plucking

gestures require a fair amount of touch movement to trig-
ger interaction and during a touch, users may head back to
the control in order to cancel the touch. Finally, additional
sound parameters can be mapped, especially using touch
move data. Additional sound parameters imply added ex-
pressiveness, which is a definite desire to all artists.

In terms of usability, a quantitative user test based on sen-
orimotor synchronization (SMS) research methods [8, 9]
reports that plucking gestures can be easily trained to ef-
ciently execute and do not show significant difference in
rhythmic accuracy compared to ordinary tapping [3].

1.3 Related Work

In addition to the works presented in Section 1.1 [4, 5],
Wang (2009) has released the well-known iPhone Ocarina
[10]. The iPhone Ocarina is a good example of resolving
the problem we are dealing with (discussed in 1.1), as
musical output is triggered by breath rather than touch (re-
ducing accidental touch), and forcing players to bring their
iPhones up to their mouth offers a very appropriate link be-
tween gesture and sound (revealing interactions). Another
work by Wang is the Magic Fiddle [11]. This instrument
also requires players to hold their iPad as a fiddle, and pro-
vides a single button pushed by the right hand for sound
generation. Pitch is controlled by the left hand, by placing
fingers on a selection of three strings.

The test interface we present in this research extends the
work of those described above, especially those of Wang,
in a sense that pitch can be controlled without triggering
sound, and sound is generated by a single control: plucking
gestures on a plucking button.

The remainder of this paper is constructed as follows.
The design and implementation of the new interface is de-
scribed in Section 2, followed by a simple evaluation in
Section 3. Conclusions and discussions are presented in
Section 4.

2. TEST INTERFACE DESIGN AND
IMPLEMENTATION

Throughout this section, we first discuss possible remedies
to the prevailing problems that prevent touch screens be-
ing used in live performances, and present a simple inter-
face which is built upon this reasoning. Although the pre-

tained interface is not yet intended to be used on stage, we
believe that it is a meaningful approach in mitigating the
weak points of touch screen interfaces.

2.1 Resolving Problems of Touch Screens in Live
Performances Through Design

In 1.1, we have described three obstacles that prevent touch
screen interfaces from appearing in live situations: a) vi-
2.2 Guitar Inspiration in Design and Mapping

The interface design is inspired by traditional guitars in two ways: the left hand determines the pitch of notes to be played by changing fingerings, and the right hand plucks on the screen as if plucking guitar strings. On the left side of the screen, eight buttons are implemented in two lines and controls pitch by four fingers on the left hand (index to pinky). This layout also has a guitar metaphor, with two strings and four frets. On the right, one gray button is placed for plucking gestures, using right hand fingers.

Characteristics that differ from traditional guitars also exist, other than not having real strings and frets. While frets closer to the sound hole should normally produce higher pitch, this interface has an opposite fingering-pitch mapping: buttons on the left are mapped to higher pitches. This decision has been made to enable players lay down the interface, rather than holding it up as a real guitar or violin: holding up the interface causes the device to move and results in higher visual dedication.

The eight buttons on the left are each mapped to a note in the C major scale (C4 to C5). This also differs from real guitars, as guitar frets always have a half-note interval. These buttons, including the plucking button on the right, are rendered as an oversized circle, to further reduce visual dedication and risk of making mistakes.

In order to further relieve accidental inputs while emphasizing guitar fingerings, higher note buttons (frets) in the same row have higher priority: pressing C4 (1-1, rightmost upper row) and F4 (1-4, leftmost upper row) simultaneously produces F4, rather than C4. This enables the interface to be more friendly to guitar players.

Sound synthesis is done by Stk::Mandolin, which is a plucked instrument simulation of the mandolin included as an example in STK. Currently, we have implemented this interface as a monophonic instrument, in order to minimize the complexity of playing. Therefore, pressing two different buttons on different lines only produce one note, rather than a chord.

For plucking gestures, sound is generated when a touch started inside the plucking button ends outside of its borders. The distance between the plucking button’s border and the touch’s end position is mapped to gain level, from 0.0 to 1.0. Pulling a touch further achieves higher gain. As pulling a touch slightly out of the button would generate a very soft sound, this type of gain mapping may also act as a feature to reduce mistakes.

The interface is implemented on Apple’s iPad, using Cocos2D/Box2D and the Momu ToolKit with STK. Figure 2 illustrates the design of our proposed interface, and Figure 3 is how the interface looks on an iPad.

3. Evaluation

As the proposed interface has strong gestural links to guitars, and the purpose of it is to assess the usability of plucking gestures in live situations, the implemented interface has been presented to and assessed by eight experienced
During free playing sessions, some participants showed interest in the multi-touch capabilities of plucking buttons, and developed additional gestures were not intended by the designers. First, plucking was done with not only one finger, but multiple fingers to execute faster attacks. This is similar to those of electric bass guitar players, who mostly play by using two fingers taking turns. Second, some players initiated several touches with multiple fingers at once and thereafter decided not to release touches until they intended, stacking up a "pile of attacks" ready to be played upon touch end of each finger. Another interesting gesture was to swipe the plucking button rapidly with two or three fingers to achieve even more speed.

Some users claimed that the left-hand mapping is not intuitive enough, having higher pitch on the left and lower on the right. However, after receiving explanations on the guitar metaphor, the experienced guitarists agreed that their fingerings were more comfortable than the opposite case. Also, some users, especially with long fingernails, found the button very difficult to pluck as they would on real instruments. This problem was mitigated by laying their fingers sideways, touching the screen with the side of their fingers.

Through a simple user test, the presented interface has been shown to a) reduce risks of making mistakes, b) add additional expressiveness compared to conventional touch screen buttons and c) slightly reduce the level of visual dedication. Additionally, the multitouchable nature of plucking buttons and plucking gestures offer a certain level of explorability, allowing new gestures to be developed and used.

4. CONCLUSIONS AND DISCUSSIONS

In this research, a problem-solving type of design approach has been used, by first setting up a problem, "What risks do touch screen devices face on stage?", and seeking possible answers. Searching answers has become the design process itself. Especially, the question "Why do players accidentally touch the screen and make mistakes?" led to solving a gestural problem, resulting in incorporating plucking gestures. In addition to plucking gestures, we believe that a third set of additional gestures can offer a chance of explorability, which further enhances expressiveness and creativity with new instrument implementations.

While three obstacles that touch screen musical interfaces should overcome in order to become a fully reliable instrument (visual dedication, risk of making mistakes, and unrevealing interaction) have been presented in this research, there must be more problems that remain unidentified by researchers: problems that we already feel while using, but not properly stated. Further research will include identifying additional problems, and applying those proposed solutions on instrument implementations. Additionally, we are

---

3 These additional gestures can be viewed at [http://aimlab.kaist.ac.kr/~noshel/plucking2/](http://aimlab.kaist.ac.kr/~noshel/plucking2/)
currently devising a quantitative user test employing sensorimotor synchronization experiment methods to statistically access the usability and accuracy of the input methods described in this research.

5. REFERENCES


