Use the Force: Incorporating Touch Force Sensors into Mobile Music Interaction

Edward Jangwon Lee[1], Sangeon Yong[1], Soonbeom Choi[1], Liwei Chan[2], Roshan Peiris[3], and Juhan Nam[1]

[1] KAIST, Daejeon, Korea
noshel, koragon2, cjb3549, juhannam@kaist.ac.kr
[2] NCTU, Hsinchu, Taiwan
liwei.name@gmail.com
[3] Keio University, Yokohama, Japan
roshan82@gmail.com

Abstract. The musical possibilities of force sensors on touchscreen devices are explored, using Apple's 3D Touch. Three functions are selected to be controlled by force: a) excitation, b) modification (aftertouch), and c) mode change. Excitation starts a note, modification alters a playing note, and mode change controls binary on/off sound parameters. Four instruments are designed using different combinations of force-sound mapping strategies. ForceKlick is a single button instrument that plays consecutive notes within one touch by altering touch force, by detecting force down-peaks. The iPhone 6s/7 Ocarina features force-sensitive fingerholes that heightens octaves upon high force. Force Trombone continuously controls gain by force. Force Synth is a trigger pad array featuring all functions in one button: start note by touch, control vibrato with force, and toggle octaves upon abrupt burst of force. A simple user test suggests that adding force features to well-known instruments are more friendly and usable.

Keywords: mobile music, force touch, 3D touch, touch gestures, relative force

1 Introduction

Recently, Apple's iPhone featured *3D Touch*[4], which captures the finger pressure of a touch on the screen. Although mobile musicians always have enjoyed designing instruments using new sensors on smart devices [1] [10], only a handful of commercial musical applications adopt the new technology after more than a year since 3D Touch's debut. It seems that the current force sensing feature is just not enough: while force mapped to aftertouch is considered to work perfectly, its relatively slow update rate[5] alongside with noisy data at the beginning

[4] 3D Touch by Apple, on models after iPhone 6s: https://developer.apple.com/ios/3d-touch/.

[5] Although Apple does not disclose 3D Touch's sample rate, our preliminary experiments indicate it to be approximately between 10 and 15ms (67-100Hz).

of a strongly struck touch acts as a hurdle for this technology being used for note velocity control [3]. However, we believe that force sensors will surely open a new possibility for more expressiveness on touchscreen music, and therefore deserve more attention by computer music researchers.

This paper aims to discuss on how force sensors on touchscreen devices can be used as a musical input gesture, and presents a compilation of simple mobile instruments each differing in musical mapping of force. Usage of force sensors on music is categorized into three categories: a) excitation (note triggering), b) sound modification (aftertouch), and c) mode change, an on/off switch similar to electric guitar effects pedals. Each instrument is implemented with a different combination of the three functions, and two among them employ relative force – using the first and second time derivatives of force data for capturing peaks in force and generate discrete events using them.

1.1 Related Work

Lack of force sensing has been considered a loss of gestural information, which in turn leads to restrictions in musical expressiveness. In this sense, several workarounds are found throughout the computer music literature. Tanaka attached an external device with force-sensing resistors (FSRs) on a PDA device [5]. Park and Oriol also addressed this issue by attaching deformable foam blocks beneath an iPad and measuring accelerometer data change during touch [2], and Apple also uses accelerometer data in iOS GarageBand for note velocity control.[6]. Recently, to overcome the restriction that accelerometer data can only measure force from a single touch, Michon et al. attached several FSRs beneath an iPad and measured multi-touch force [3].

In addition to these two approaches, FSRs and accelerometers, Essl et al. takes touch radius into consideration and presents three prototypes using different force sensing approaches [4]. Now that a force sensor has been included in smart devices, in addition to exploring *how to be able to "use the force"*, this paper discusses *how to "use the force"* and presents several examples of various force-sound mappings and force data parsing methods.

Recently, mobile music applications also began to feature 3D Touch functions, mostly mapped to aftertouch controls, where such examples include ROLI's Noise[7], Miditure[8], Aftertouch[9], and Apple's GarageBand. ROLI's Seaboard 5D supports advanced 3D Touch features such as velocity control and finger lift speed. However, the velocity value seems to be zoned into only a few levels, rather than 128, which is the standard MIDI velocity range. This restriction may have been set due to the unreliable behavior of initial touch force measurements, as discussed above. In this paper, we propose alternative mappings of force, including mappings using relative force over time, rather than the absolute measurement.

[6] Apple's GarageBand: `http://apple.com/ios/garageband/`.
[7] `http://roli.com/products/noise`.
[8] `http://facebook.com/Miditure`.
[9] `http://aftertouchapp.com`.

2 Mapping Touch Force to Sound

Borrowing real instrument metaphors that require force and applying them to force-sound mapping would be advantageous in terms of intuitiveness and usability. First, for blown instruments, pitch and timbre differ greatly depending on embouchure, lip tension, and air pressure. As mobile devices do not have the ability to recognize such input, mapping those to force sensors might be a meaningful alternate. That is, force sensors can be used for deciding pitch and timbre, which are usually a part of *excitation*. Second, in stringed instruments such as the guitar, a player may alter pitch during a note playing by pushing a vibrating string orthogonally to the neck (bending or vibrato, depending on the speed and style of push). These techniques can be categorized as *aftertouch*, and be used as a metaphor in force-sound mapping. Finally, guitar effects pedals such as distortion and chorus can be turned on and off anytime during playing. This metaphor is also taken into consideration, and is categorized as *mode change*. Each of the three mapping categories is described in detail in the following subsections.

2.1 Excitation

Although 3D Touch in the current state is inappropriate for note velocity control, force sensors can improve the excitation process, and two possibilities are presented in this paper.

First, *adaptive gain control* is for non-percussive instruments with continuous sound, such as blown and bowed instruments. Such instruments accept force in a continuous fashion, starting from zero and gradually increasing. As touch force theoretically begins at zero and gradually increases as well, mapping force measurement to output gain is an intuitive idea. However, due to the slow sample rate of 3D Touch, initial force readings tend to be non-zero when touch begins with large force, which lead to an unpleasant abrupt change in gain: therefore the gain envelope must *adapt* to the force curve over time. For this purpose, a gain control model similar to time constant models is designed:

$$g = g_{fixed}(e^{-\frac{t+\alpha}{2\alpha}}) + \frac{f}{f_{max}}(1 - (e^{-\frac{t+\alpha}{2\alpha}})) \quad (1)$$

where g: output gain (0.0 to 1.0), g_{fixed}: a fixed gain value, t: time, f/f_{max}: force value normalized from 0.0 to 1.0, and α: the time constant. 2α, rather than a, is used to produce a more natural adaptation curve.

Another model, *down peak detection*, focuses on the *timing* of excitation. This model uses relative force (time derivatives of force data) over a single touch and a note is played on every down-to-up peak. Although velocity control is not available, this technique produces notes quickly upon user intention and enables the user to play consecutive notes by controlling touch force, without lifting the touching finger [7]. This model is suitable for percussive instruments such as drums, where timing is crucial and normally do not require additional sound modification (aftertouch) after a note is played.

2.2 Sound modification

In contrast to note excitation, sound modification does not require as much as frequent force value updates, therefore implementation is straightforward. Apple's GarageBand 2.1 for iOS has aftertouch implemented into some of its instruments using 3D Touch. As described in the previous section, force is used to control not only timbre, but also gain. This concept has been borrowed from blown and bowed instruments, in which amplitude envelope follows the force of blowing.

2.3 Mode change

Mode change is a feature that can be toggled on and off. For instance, blown instruments produce different pitch depending on how they are blown, although the fingering is identical. Woodwinds feature *overblowing*, which usually causes the instrument to shift to a higher octave. On their mobile counterparts, fingerholes are implemented as buttons. Overblowing can be mapped on the buttons using force – higher octaves by applying strong force on the fingerholes.

Another method comes from guitar effects pedals, which can be turned on and off anytime during play by stepping on them. By recognizing an *abrupt, narrow and steep peak* in force, sound parameters can be toggled on and off, while steady changes of force are mapped to other continuously changing sound parameters such as amount of vibrato. In order to detect narrow peaks of force, relative force over time is used and a recognition model is devised: by analyzing first and second derivatives from 200 collected samples of narrow up-slopes of force generated by 10 different subjects, a linear regression model is obtained. Afterwards, during testing the research team added restrictions and refinements to further filter unintended mode changes.

3 Instrument Prototypes

Based on the discussion above, four instruments are implemented on iPhone 6s, using STK [8] and MoMu [9] as the sound engine. User interface is built using `UIKit`, the basic GUI library for iOS. Each instrument differs in usage of force sensors and the combinations of usages. All rectangular and circular buttons are named *force buttons*, which are designed to accept a single touch and constantly monitor the amount of force applied. As iOS reports touch data only upon change in position or force, for constant monitoring a function that calculates the change of force over time is called every 1/60 seconds.

The screenshots and force-sound mapping strategies are summarized in Figure 1.

3.1 ForceKlick

Force mapping: excitation (relative force). ForceKlick is a very simple instrument for testing force buttons, with only one force button that can play consecutive

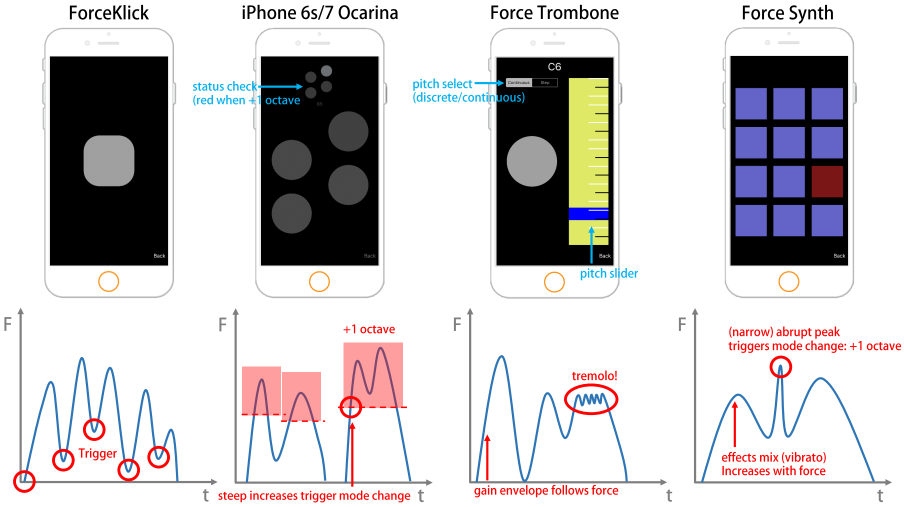

Fig. 1. Instrument screenshots and their force-sound mapping strategies.

notes with fixed pitch and gain within a single touch, using down peak detection (section 2.1). A new note is played whenever a down-to-up peak is detected. That is, note excitation occurs shortly after force begins to increase. This is efficient in a sense that the user's intention of playing a note would be first indicated to the system as an increase of force.

3.2 The iPhone 6s/7 Ocarina

Force mapping: mode change – octave. The iPhone Ocarina [1] has four buttons (holes), accepting sixteen (2^4) combinations of fingerings. Excitation is done by blowing breath into the microphone: stronger blowing increases the amplitude of sound.

The four buttons are all implemented as force sensitive buttons, and their color changes corresponding to applied force. The instrument transposes one octave higher when the first time derivative of force is higher than a fixed threshold – a rapid increase in force. The original octave is restored upon detecting either a steep decrease of force or force below an absolute threshold. This transposition feature extends the number of playable pitches to 31. [10]

3.3 Force Trombone

Force mapping: excitation, modification – gain control. Rather than blowing, the force trombone is excited by pressing the circular force button. Amplitude is

[10] Not $2^4 \times 2 = 32$, as "all holes open" does not have any fingers on screen to apply force.

increased by applying stronger force on the button. By controlling force, players can not only change gain but also execute tremolos and other amplitude-based techniques, without lifting the touching finger. The slider on the right is mapped to pitch, and it can be toggled between continuous and discrete pitch mode. For initial gain control, the model in Equation 1 is used, and the g_{fixed} and α value is manually tuned to 0.6 and 0.3, respectively for acceptable outcomes.

3.4 Force Synth

Force mapping: excitation, modification – vibrato, mode change (relative force) – octave. The Force Synth attempts to include all three proposed force functions into one button. The instrument consists of twelve force buttons, each assigned to a note following pentatonic scale. Touching a button triggers the corresponding note with a fixed amplitude (excitation), and controlling force decides the amount of vibrato (modification). Additionally, a narrow peak in force immediately toggles the button's pitch to a higher octave (mode change). The toggled octave is indicated by the button's color changing from purple to red, and vibrato amount can be noticed by darker hue of colors.

4 Evaluation

4.1 Test Design

To evaluate the experimental use of force sensors, a simple user test has been conducted. The four instruments have been evaluated by 9 participants, all in their twenties and using smartphones as if they were a part of their bodies. 8 participants had musical experience, mostly in guitar and piano.

Before handing out the instrument to participants, a demo and training session was conducted. The demos included playing performance by the research team. As force change cannot be easily noticed by watching others play, the test conductors had to hold the participants' finger and help them control force and explain how the force-sound mappings were designed. Afterwards, a free-play session was given, and the device was taken back after completing the online survey.

The questionnaire consisted of three or four questions for each instrument. The forth question is applied to ForceKlick and Force Synth only, as only these two employ relative force methods. Each question was answered in a 5-point Likert scale. The questions are as follows:

Q1. Is the force-sound mapping intuitive?
Q2. Does the instrument follow the player's intention?
Q3. Is it easy to use force sensors under this mapping?
Q4. (for ForceKlick and Force Synth) Is relative force mapping easy to use?

Questions on overall preference were also included, "Which instrument did you like (and dislike) the most, and why?". Finally, users were asked to optionally provide written feedback.

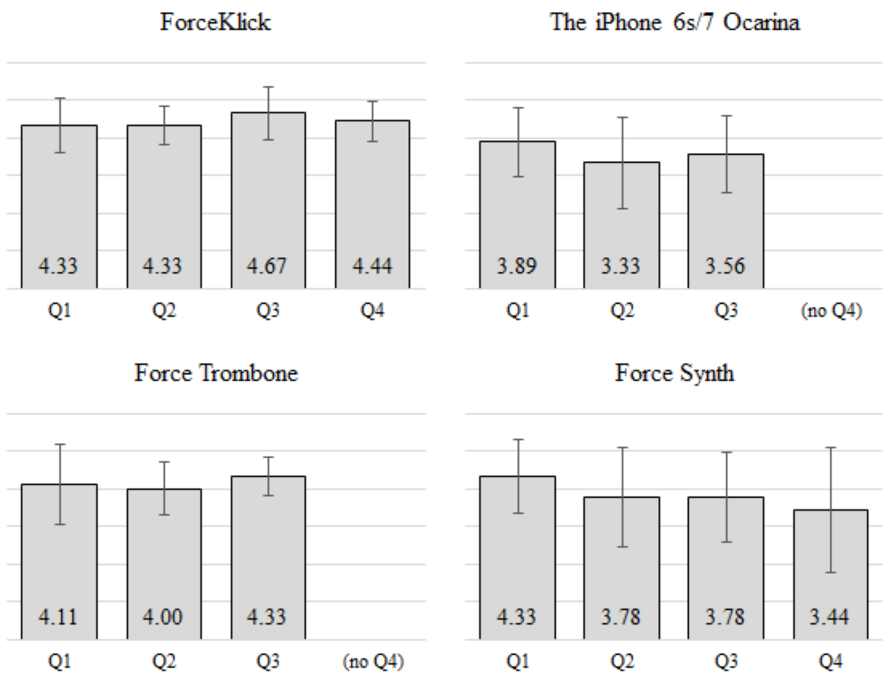

Fig. 2. Survey results for each instrument. Average and standard deviation of the corresponding question.

4.2 Results and Discussion

Figure 2 displays the test results for each instrument. ForceKlicks received high points for all items with low variance, notably 4.44 points for Q4: relative force mapping usability. Alongside with high ratings in other questions, this suggests that users felt comfortable and satisfied with note triggering via down-to-up peak detection.

The iPhone 6s/7 Ocarina received the lowest ratings in all items, with a rather high variance in Q2 and Q3 (playability and force sensor usability). The high variance in responds were explained in the comments, "Difficult to control force while holding the phone and blowing into it".

Force Trombone recorded all items higher than 4 points, suggesting that gain control by touch force is intuitive, playable, and usable to users. One participant that gave low points remarked that "The pitch slider is too sensitive, I can't stop sliding at the right pitch", which implies that low points were caused by the slider rather than the force mapping.

Force Synth received high points in Q1: intuitive force-sound mapping. This is surprising, as this instrument has the most complex mapping design: all three force features – note triggering, aftertouch, and mode change – are included in one force button. Moreover, mode changing by creating narrow peaks of force was

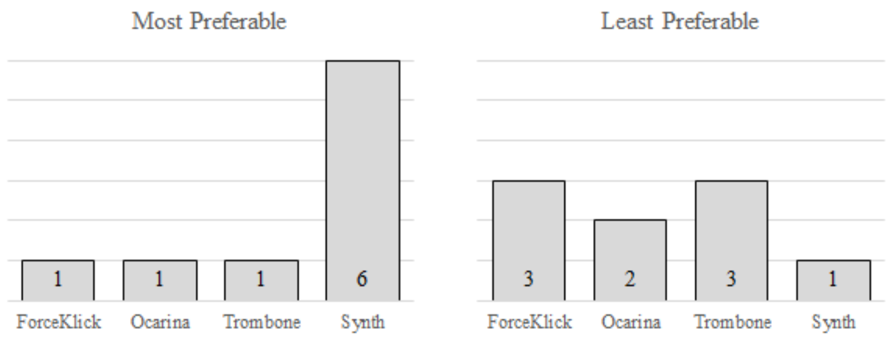

Fig. 3. Preference survey results.

Table 1. Selected comments for the most and least preferable instrument.

ForceKlick	Worst	"Too simple"
Ocarina	Worst	"Difficult to hold phone while controlling force"
	Best	"Works well and interesting"
Trombone	Best	"Good use of trombone metaphor"
Synth	Best	"Impressive force features added on usual trigger pad UI"
		"Intuitive, all features work well"
		"Easy to use and behaves as my expectation"

rather experimental and not expected to be easily accepted. The guitar effects pedal metaphor behind this mapping might have been convincing. However, although users perceived the mapping as intuitive, other questions in terms of playability, force mapping usability, and relative force usability (mode change) received low points. This discrepancy suggests that although participants agree in how the force button works, they feel rather challenged in actually generating narrow force peaks to activate mode change. One participant commented that he wanted a different action mapped to mode change.

The overall preference results are presented in Figure 3. Force Synth has received the most votes as best instrument, and votes were evenly spread out for the least preferable, from Force Synth (one vote) to ForceKlick and Trombone (three votes). Both votes required a reason, and notable comments are presented in Table 1. In contrary to the individual instrument survey where ForceKlick received the highest ratings, in preference polls users tended to focus on the instrument itself, rather than the force-sound mapping strategies: the simplicity of ForceKlick was the main complaint. Force Synth was most preferred mostly for its familiarity with prior trigger pad instruments and convincing addition of force sensors, alongside with its polyphonic capabilities.

5 Future Work and Conclusion

Although mode change feature in Force Synth works well after a certain time of training, a better method to satisfy all users is being devised. Various methods are being tested such as support vector machines (SVMs) to fit the needs of as many as possible. Personal customization of threshold parameters are also in consideration, either by providing option screens or machine learning. The user test can also be improved by setting a control case – building similar instruments without force mappings and comparing them to their force counterpart.

This paper discusses the current state and possible usage of touch force sensors on multi-touch devices. Based on the discussion, four different mobile instruments utilizing touch force, each having their own mapping strategy including force data parsing methods are presented. User tests revealed that rather than defining new mappings such as blowing the trombone with a force button, adding additional force functions to conventional button mappings (ForceKlick and Force Synth) are more acceptable to participants, although the added functions used relative force data and were not expected to be easily convincing.

6 Demo Video

A demonstration video of the four instruments introduced in this paper can be found at the following link: `https://youtu.be/quxAEBEp97Q`.

References

1. Wang, Ge: Designing Smule's iPhone Ocarina: The iPhone's Magic Flute. Proceedings of the 2009 International Conference on New Interfaces for Musical Expression. pp. 303–307. Pittsburgh, Pennsylvania (2009)
2. Park, T.H., Oriol, N.: Fortissimo: Force-Feedback for Mobile Devices. Proceedings of the 2013 International Conference on New Interfaces for Musical Expression. pp. 291–294. Daejeon, Korea (2013)
3. Michon, R., Smith, J.O., Chafe, C., Wright, M., Wang, G.: NUANCE: Adding Multi-Touch Force Detection to the iPad. Proceedings of the 2016 Sound and Music Computing Conference. Hamburg, Denmark (2016)
4. Essl, G., Rohs, M., Kratz, S.: Use the Force (or Something) – Pressure and Pressure-Like Input for Mobile Music Performance. Proceedings of the 2010 International Conference on New Interfaces for Musical Expression. pp. 182–185. Sydney, Australia (2010)
5. Tanaka, A.: Mobile Music Making. Proceedings of the 2004 International Conference on New Interfaces for Musical Expression. pp. 154-156. Singapore (2004)
6. Lee, E.J., Yeo, W.S.: Plucking Buttons: An Alternate Soft Button Input Method on Touch Screens for Musical Interaction. Proceedings of the 2013 Sound and Music Computing Conference. pp. 202–206. Stockholm, Sweden (2013)
7. Yong, S., Lee, E.J, Peiris, R., Chan, L., Nam, J.: ForceClicks: Enabling Efficient Button Interaction with Singer Finger Touch. Proceedings of the TEI '17: Eleventh International Conference on Tangible, Embedded, and Embodied Interaction. Yokohama, Japan (2017)

8. Cook, P., Scavone, G.P.: The Synthesis Toolkit (STK). Proceedings of the 1999 International Computer Music Conference. pp. 164–166. Beijing, China (1999)
9. Bryan, N.J, Herrera, J., Oh, J., Wang, G.: MoMu: A Mobile Music Toolkit. Proceedings of the 2010 International Conference on New Interfaces for Musical Expression. pp. 174–177. Sydney, Australia (2010)
10. Kim, B., Yeo, W.S.: Interactive Music Performance with Digital Compass. Proceedings of the 2012 International Conference on New Interfaces for Musical Expression. Ann Arbor, Michigan (2012)