Using Shear as a Supplemental Two-Dimensional Input Channel for Rich Touchscreen Interaction

Chris Harrison  Scott E. Hudson
Human-Computer Interaction Institute and Heinz College Center for the Future of Work
Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
chris.harrison@cs.cmu.edu, scott.hudson@cs.cmu.edu

ABSTRACT
Touch input is constrained, typically only providing finger X/Y coordinates. To access and switch between different functions, valuable screen real estate must be allocated to buttons and menus, or users must perform special actions, such as touch-and-hold, double tap, or multi-finger chords. Even still, this only adds a few bits of additional information, leaving touch interaction unwieldy for many tasks. In this work, we suggest using a largely unutilized touch input dimension: shear (force tangential to a screen’s surface). Similar to pressure, shear can be used in concert with conventional finger positional input. However, unlike pressure, shear provides a rich, analog 2D input space, which has many powerful uses. We put forward five classes of advanced interaction that considerably expands the envelope of interaction possible on touchscreens.

Author Keywords: Tangential force; handheld mobile devices; finger interaction; touchscreens; fat fingers.

ACM Classification Keywords: H.5.2 [Information interfaces and presentation]: User Interfaces – GUIs.

INTRODUCTION
Touch interaction is becoming pervasive, especially in the mobile domain, where there is limited room for static physical buttons. Conventional touch input is generally treated as binary – the user is touching or not touching. Further, touch points are typically reduced to a simple X and Y coordinate. This is not particularly high bandwidth – for comparison, a modern computer mouse provides X/Y position, a continuous 1D scroll wheel, and several buttons. Because of this, touch interaction can be unwieldy at times, requiring less-than-natural touch-and-hold [14] or multi-finger chording actions [12] to trigger secondary functions or switch modes.

In response, there has been tremendous work on sensing and processing additional dimensions of touch, for the purposes of enriching interaction on touchscreens, including pressure [15,18-23], finger orientation [27], finger posture [15], configuration of the hands [1], rolling motions of static fingers [24], and what part of the finger was used for input [6]. Handheld devices can also augment touch position with non-finger input dimensions (see [9] for overview), including tilt [28], acceleration [9], how the enclosure is being manipulated [25] and position relative to the user [13].

One dimension that has been underexplored is shear force. Shear stress (also called “tangential force” in the literature) arises from a force vector perpendicular to the surface normal of a touchscreen (i.e., parallel to the touchscreen surface). This is similar to normal stress – what is commonly called pressure – which arises from a force vector parallel to the surface normal.

Before we can discuss related work and shear applications, it is necessary to define four terms we use to describe finger movements. To describe traditional finger pointing and dragging, we use the term move or drag respectively (e.g., “the user moves their finger over an icon” or “drags an icon”). We call the application of force in the shear dimension a push (e.g., “the user pushes upward”; this is called a “pivot” in [7]). The finger does not slide during pushes due to static friction. Finally, a drag action performed with substantial pressure is called a hard drag (e.g., “a user hard drags an icon”). Shear forces are created due to sliding friction between the finger and screen. Of note, hard drags do not provide a secondary 2D input channel (i.e., you cannot drag in one direction and push in another), but rather provide an alternative drag mode.

The central contribution of this work is considering shear as a rich analog X/Y input channel. Importantly, this can operate in parallel with conventional finger X/Y tracking, potentially affording touchscreens an additional 2D input channel. For example, a user can touch the screen (providing a 2D position) and push in any direction (providing a 2D shear vector). As we will discuss, this enables a variety of compelling interaction techniques and could be used to alleviate some of the input contention experienced in today’s touchscreen devices.

RELATED WORK
Touchscreens able to capture finger shear forces date back at least as far as 1977 [8]. Exploration of the interaction space was limited, given that it predates popular use of graphical user interfaces. Minsky revisited touch+shear interaction in 1984 [16], suggesting uses for higher-fidelity digital painting and visual programming. Of note, they propose directional flicks, inferred from shear information, which could be used for sending items to different screens.
There have also been efforts to instrument the fingers with shear sensors [15], though this is particularly invasive. Most recently, Heo and Lee have taken the idea into the mobile domain, augmenting an iPod Touch [7]. The latter work describes a shear- and pressure-augmented web browser and eBook reader. These applications were primarily controlled by “force gestures” in the up/down/left/right directions (see e.g., Table 1 in [7]).

Pressure is much like shear, in that it offers a continuous input dimension that can operate in concert with conventional touch input [15,18-23]. The most significant difference is that pressure is one-dimensional. It should be noted that shear and pressure input are not entirely orthogonal. Although pressure can be provided without shear, shear force requires pressure, which alters the static and sliding friction of a finger on a touchscreen. For example, in order for users to produce a hard drag, they must press their fingers firmly to the screen. In combination with X/Y finger tracking, pressure sensors could determine a hard drag. However, for example, even with X/Y tracking, pressure cannot be used to determine the directionality of a push.

It seems likely that most interaction techniques built on top of pressure forces are immediately portable to shear input (and potentially enhanced further). For example, there has been considerable work on continuous scrolling [20,22] and zooming [18] using pressure. This is trivially implemented using shear [7,8,11], and can gain a directional component (e.g., scrolling up/down with up/down shear force).

Our prototype hardware is based on two self-centering, analog joysticks, which displace in response to shear forces roughly according to Hooke’s law. Similar are isometric joysticks [26], which are essentially 2D shear force sensors. There is a substantial body of work assessing human motor performance in this input modality [2,3,5,7,11]. Results show humans are remarkably dexterous in shear-space, though performance is slightly worse than positional input, like a mouse. Further, shear input appears to be comfortable and intuitive [7,11]. This work does not contribute new empirical results, but rather pushes the envelope in interaction design.

**UTILIZING 2D SHEAR INPUT**

As described previously, shear has been primarily utilized for simple gestures, consisting of uni-directional pushes and hard drags. This is small subset of possible uses - we now describe five advanced interactions that fully realize shear as rich, secondary, 2D input dimension. To illustrate the unique capabilities of shear, we ground our discussion with many small examples. This simultaneously serves to highlight the utility of shear input. Please also see our Video Figure, which demonstrates a collection of applications utilizing complex force gestures, continuous and 2D control.

**Shear Gestures**

In much the same way as a touchscreen can support gestures, shear too can support a rich gesture set. These can be as simple as directional pushes [7], all the way up to full alphanumeric text entry [4]. A dedicated gesture input channel on touchscreens would be invaluable, and would allow functionality typically provided in buttons and menus to be removed from the screen, freeing valuable real estate.

One simple, but useful application we built was shear-augmented copy and paste – unwieldy on touchscreens at present. With shear input, a user highlights a section of text as usual, finishing with an up push, which copies the text to the clipboard. The user then moves their finger elsewhere, and with a down push, pastes in the text at that location. A smartphone offers a more sophisticated gesture set. Writing a ‘V’ in shear-space triggers the voicemail. Clockwise and counterclockwise circling motions increase and decrease speaker volume, respectively (Figure 1A). An ‘S’ gesture puts the phone into silent mode (Figure 1B).

**In-Situ Continuous Manipulation and Control**

Two-dimensional shear input can be used in a rate-controlled manner. We implemented this for in situ, continuous scrolling of a web page. The more shear force that is applied, the faster the scroll rate. This would be useful on small screens, which often require repeated finger drags to navigate large content (e.g., maps; Figure 1C). Continuous zooming (in and out) can be handled with a single shear dimension, and with one finger (instead of +/- buttons or two finger pinches; Figure 1D).

There are many one- and two-dimensional manipulations beyond panning and zooming. For example, by moving ones finger onto a photograph, brightness and saturation can be adjusted with shear force (X and Y dimensions; Figure 1F). Note this operation is not global, but can be made to act solely on the element under the finger. This is a unique quality, in that both selection and manipulation can happen without fingers moving or lifting from the display.

---

*Figure 1. A: clockwise shear gesture increases volume. B: ‘S’ shear gesture sets phone to silent. C: In situ rate-controlled map panning, and D: zooming. E: In situ color selection using 2D shear. F: X/Y shear dimensions control brightness and saturation.*
Similarly, we built a demo application that allows users to change the color of a graphical element by placing their finger on it, and using shear forces to select a color from a popup color palette, as depicted in Figure 1E.

**Parallel High CD Gain Input**

Touchscreen interaction generally operates with a 1:1 control-device (CD) gain. While providing a tight coupling between physical and digital manipulations, fine-grained operations are challenging. In mobile contexts, this problem is further compounded by small screens and “fat fingers”. Fine-grained manipulation is important in many digital tasks. For example, in graphic design, proper sizing, placement, and alignment of objects and text is fundamental. In general, fine adjustments are common in tasks requiring continuous control, for example, when selecting colors or adjusting audio parameters.

Touchscreen inaccuracy could be mitigated by using shear as a parallel, high CD gain input means. For example, we built a demo application that lets users place their finger on an object, and adjust its position by pushing it in the desired direction; a 1:10 CD gain is used (Figure 2G). Also consider a tuning knob that could be adjusted with great precision, without lifting the finger (Figure 2H).

**Shear Menuing**

Because shear operates in two dimensions, in situ navigation of pie or marking menus is possible [14,15,17] (Figure 2 I and J, respectively). For example, imagine a user wishing to align a figure to the top-left of a document. They could touch the figure and push up, which triggers a pie menu. Users could then push in the direction (e.g., 37°) of desired menu items; e.g., alignment→top-left→margins.

**Alt-Dragging**

In the same way a right mouse button provides an alt-click function, hard drags could serve as a natural alt-drag in touchscreen interaction, and in general, enable modal dragging [22,24]. For example, we created a freehand drawing application where finger drags produce strokes. Straight line segments are drawn with hard drags, without effecting freehand strokes (Figure 2K). Another example is organizing documents. In our proof-of-concept system, conventional dragging moves the item around the desktop; a hard drag, however, “tears off” a duplicate of the file (Figure 2L).

**LIMITATIONS**

Current shear-input-capable systems [7,8,11,16], including our own, have several limitations of note. Foremost, because a single rigid touchscreen is used, only a single shear reading can be captured at any instant. This precludes shear multitouch (i.e., where each finger has a separate shear vector), but does not preclude capturing a single shear force for multitouch gestures, which is still powerful (e.g., a two finger push). Multi-user input is similarly affected.

**EXPLORATORY PLATFORM**

To help us explore shear as an additional two-dimensional input dimension for touchscreens, we created a proof-of-concept handheld device, which we briefly describe for reference (Figure 3). The device consists of a Haier 7” LCD display, on top of which we mounted a 6.7” 3M capacitive touchscreen. Operating between the display and touchscreen are two, low-profile, self-centering, analog joysticks. These are affixed to the panels in the upper-right and lower-left corners with epoxy. When sufficient shear force is applied to the touchscreen, the joysticks displace in response to the force (roughly according to Hooke’s Law), causing the touchscreen to translate relative to the display. Maximum translation in the X- and Y-axes is 2mm.

For rapid prototyping, our device is tethered to a conventional computer, which processes input and provides a graphical user interface. Single-finger touch input from the touchscreen is transmitted to the computer over USB using a 3M EXII controller. Values from our two joysticks are processed by an Arduino and also sent over USB. A series
of custom Java programs handle touch and shear input data, and provide an interactive experience (see Video Figure).

Although our sensors are capable of digitizing 2mm of shear displacement with roughly 9-bits of resolution, functional precision is much lower due to lack of human motor precision at such small scales [20]. This is similar to results found in pressure input experiments, where humans are generally only capable of accurately providing six discrete levels of pressure [14,23]. Users can also push out to the maximum displacement (physical boundary) of our sensors. When used in this fashion, our system acts more like EdgeWrite [29]. However, as our (and most) joysticks have a radial maximum, there is a continuous circular periphery instead of sides and corners.

FUTURE WORK

There are many avenues of future work that would be interesting to pursue. For example, how effectively can shear, pressure and positional input be used in concert, and what advanced interactions can leverage all three? Our system is also capable of sensing static twisting forces of a finger (i.e., screen torque) – this is an unutilized input dimension at present and could have interesting applications. Direct manipulation interfaces typically provide visual feedback about users’ touch locations. How to design visual feedback for shear forces is an outstanding question.

CONCLUSION

We have shown that shear has many compelling uses when realized as a supplemental analog 2D input channel. Drawing on a range of inspirations, we have put forward five uses of shear input that significantly advances shear input beyond uni-directional force gestures.

ACKNOWLEDGMENTS

Funding for this work was provided in part by grants from the Center for the Future of Work, Heinz College, CMU, and also a Microsoft Ph.D. Fellowship. We also thank our reviewers for their many excellent comments.

REFERENCES


