SqueezeBlock: Using Virtual Springs in Mobile Devices for Eyes-Free Interaction

Sidhant Gupta[1], Tim Campbell[1], Jeffrey R. Hightower[2], Shwetak N. Patel[1]

Computer Science & Engineering[1]
UbiComp Lab, DUB Group
University of Washington, Seattle
{sidhant, tcampbll, shwetak}@uw.edu

Intel Labs Seattle[2]
1100 NE 45th St., 6th Floor,
Seattle, WA, 98105
jeffrey.r.hightower@intel.com

ABSTRACT

Haptic feedback provides an additional interaction channel when auditory and visual feedback may not be appropriate. We present a novel haptic feedback system that changes its elasticity to convey information for eyes-free interaction. SqueezeBlock is an electro-mechanical system that can realize a virtual spring having a programmatically controlled spring constant. It also allows for additional haptic modalities by altering the Hooke's Law linear-elastic force-displacement equation, such as non-linear springs, size changes, and spring length (range of motion) variations. This ability to program arbitrarily spring constants also allows for "click" and button-like feedback. We present several potential applications along with results from a study showing how well participants can distinguish between several levels of stiffness, size, and range of motion. We conclude with implications for interaction design.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. – Haptic I/O.

General terms: Algorithms, Design, Human Factors

Keywords: Haptics, springs, eyes free interaction

INTRODUCTION

Today's mobile devices enable us to perform useful tasks such email, social networking, and listening to music. Unfortunately, their current human-computer interfaces require substantial visual attention and are not well suited for situations where computing is not someone's primary task. Extracting a phone from the pocket or purse to compulsively check the screen for new messages or respond to visual alerts has become all too frequent a behavior for many people. This recurring need for visual interaction can be frustrating and even dangerous in visually demanding situations like driving a car. We believe this interaction problem can be alleviated by designing interactions that take advantage of other human sensory channels such as auditory or, of particular interest in this paper, haptic feedback.

Auditory feedback has previously been used in mobile devices by allowing eyes-free operation through screen reading, announcing menu items, and giving user interface cues. [3,9]. In many situations audio feedback may be inappropriate, such as in social settings or loud environments.

Haptic perception relies on forces experienced when an object is touched or physically manipulated. Throughout daily life we come across a large number of different objects, each having several dimensions that can be perceived haptically: stiffness, texture, size, temperature (thermal transfer), etc. Klatzky *et al.* showed that such dimensions are the fundamental mechanical properties of an object that people use to distinguish one object from another [6]. In the context of natural user interfaces, such properties can potentially be used by a haptic feedback system to represent and distinguish between various information states that the user can then perceive in a non-visual way.

Figure 1: SqueezeBlock realizes virtual springs that can dynamically be altered to haptically convey information to a user.

In this paper, we present an eyes-free mobile haptic system that provides feedback inspired by the natural tactile dimension that humans encounter. Our system, called the SqueezeBlock (see Figure 1), conveys information by realizing a virtual spring that programmatically changes stiffness. That is, a user can interact with the device by squeezing it, much like a stress ball, and the level of "squishiness" or stiffness changes depending on the information being conveyed. For example, squishy versus stiff could convey an empty versus full battery. Intermediate battery levels map to stiffness levels between the two extremes. Another scenario where our approach can provide benefit is in non-visual interaction with a mobile phone during a call, when someone is holding it to their ear. An added benefit of our

UIST'10, October 3–6, 2010, New York, New York, USA.

approach is that both the input and output can be coupled into a single interactive gesture. Displacement of the spring can serve as input, while feedback from the changing spring characteristics provides the associated output. For example, one can imagine squeezing a mobile phone and holding it there for a brief time to put a mobile phone ringtone into silent mode or for changing its audio volume.

In addition to presenting the details of a working prototype, we report the results of a user study showing that participants perceive the virtual spring's stiffness as they would expect from a real spring and can distinguish between several levels of stiffness achievable by our prototype. We conclude with some implications for interaction design.

RELATED WORK

Providing haptic feedback for mobile devices has been explored before—particularly the use of vibrotactile feedback for providing UI cues and generating complex vibration patterns [8]. These approaches *push* information from the device to the user or register successful completion of an input. In contrast, the SqueezeBlock system uses the stiffness property, which is inherently an information *pull* as the user needs to query it by squeezing the device. Information push interfaces are generally event driven and thus are apt for notification purposes whereas our approach conveys information when and for the duration someone desires.

Similar to our approach, researchers have explored shape change (rotary deformation) and shown it to be an effective eyes-free information display [5]. Fortunately, linear deformation comes as a free feature of our prototype since, to realize a spring, one needs force and displacement and by simply setting the displacement statically we can alter the effective width of the prototype.

Virtual springs have been explored by researchers in robotics, in high-fidelity haptic feedback for tele-operations, and in virtual environments [7]. The joints of a robot can be made compliant by a virtual spring-damper system [1]. In medical haptics, similar spring-damper systems are realized for high fidelity feedback, such as surgery simulations to give the operator the sensation of operating on human skin and tissue [2]. Gurari *et al.* built a prototype that realizes a linear virtual spring behind a button the user pushes with a finger to compare Weber fraction for human perception of stiffness among various conditions [4]. Unlike this previous work, we do not try to implement a high fidelity ideal linear spring. Instead, our SqueezeBlock system is demonstrating the ability to use a compact virtual spring as an effective eyes-free output modality for mobile devices.

THE SQUEEZEBLOCK

Theory

Stiffness of an object depends on the relationship between the force applied to it and the resulting deformation and is defined as the resistance an elastic object exerts to deformation. Elasticity is the property of a material to return to its original shape after the deforming force is removed. Hooke's Law of elasticity is the relationship that defines the behavior of *linear-elastic* materials or more commonly, linear springs. To realize a virtual spring, SqueezeBlock should deform (displacement x) in proportion to the force applied (F). Changing the spring constant (K) alters the stiffness of the spring, i.e. making it larger requires proportionally more force to create the same displacement.

Prototype and Implementation Details

To assess the feasibility of using stiffness for haptic feedback and its effectiveness at conveying information, we developed a working prototype that is roughly the size of a mobile phone. Although initially prototyped using a 3D printer (FDM), we developed SqueezeBlock using high grade aluminum for durability and resistance to high forces.

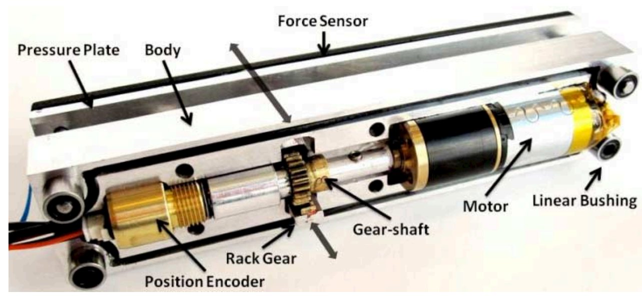

Figure 2: Cross-sectional view of the SqueezeBlock showing the mechanical components and sensors.

Figure 1 shows the completed prototype and Figure 2 shows the internal components. Mechanical components consist of a Maxon® motor with a reducing gear head, a rack and pinion gear system, and a custom metal housing with sliding pressure plates, which was designed in CAD and fabricated on a three axis milling machine. Sensing components measure the force applied to the device and the position of the motor using two FLEXIFORCE® force sensors and a US Digital® MA3 magnetic shaft encoder. Sensor data is processed using a 16MHz ATMega128 microprocessor to drive the motor in real-time via a Polulu MBD01 motor controller. A USB connection between the processor and a PC via FTDI TTL232R allows us to experimentally enter stiffness parameters and switch between the haptic modes.

A single shaft turned on a lathe connects the motor, spur gear, and encoder. A rack and pinion gear system is used to convert rotation to linear motion. The rack gear is affixed to the pressure plates, while the spur gear shares a shaft with the motor, making the pressure plates move in and out as the motor rotates. The absolute encoder reports the current position in degrees as an analog signal ranging from 0-5V, with a precision of better than a degree. This setup allows the control software to determine the exact position of the pressure plates and set them to any required position, thus implementing the displacement parameter $\boldsymbol{x}$.

The pressure plates, where someone grasps the device, consist of force sensors sandwiched between an inner solid metal plate and an outer flexible plastic plate. When pressure is applied, the plastic plates flex causing the force sensor to press against a raised area on the inner metal plate and register the total amount of force being applied, which we use as the linear-elastic force parameter **F**.

If the motor dynamically drives the pressure plates to a position proportional to the force being applied multiplied by the spring constant **K**, the result is a compelling virtual spring. However, this closed-loop control must occur at a high update rate; too much delay between the user's application of force and the displacement destroys the illusion of spring-like elasticity. Our control loop on the microcontroller provides a 1KHz update rate, which is sufficient and comparable to the update rate found in high-fidelity haptic systems. The control loop itself is implemented as a PID (Proportional-Integral-Derivative) controller.

Achievable Feedback Behaviors

Since we implement the entire linear elasticity equation in software, we can also simulate non linear springs and other arbitrary relationships between force and displacement. Figure 3 shows force versus displacement curves for several types of spring-like interactions.

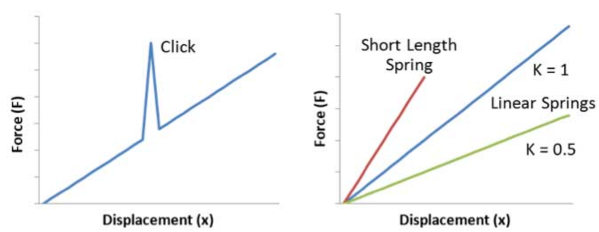

Figure 3: Simulated linear-elastic behaviors

Perception of detents or *clicks* as the device is squeezed can be simulated by one or more discontinuities between force and displacement. The total displacement range or *travel distance* can be altered by a function that holds displacement constant after a defined point; increasing force does not cause the device to yield. Setting the displacement statically, independent of the force applied, allows for changing the *size* of the device. Finally, arbitrary functions allow for complex state machines, e.g. SqueezeBlock can mimic the sticky behavior of a foot-maneuvered car parking break or a two-state stay-on push button.

EXAMPLE APPLICATIONS

The SqueezeBlock spring behaviors can be applied to convey discrete or continuous state information. For example, by setting the spring constant as a function of the number of crucial email message the device may feel stiff when the number of messages is above a certain threshold or soft otherwise. Together with the "clicks" feature, one could also count the number of such crucial messages, or more simply, the presence of a click could indicate the presence of a message from a particular important sender. The interaction is not only limited to the perception of an absolute stiffness level, but can be enriched by using relative change in stiffness of two consecutive squeezes, for example, using increasing stiffness to indicate ring volume.

The multistate function that mimics a parking brake can be used as an input gesture, where squeezing keeps the device deformed until squeezed again. Deforming the device could map to reducing the ringer volume. Another application is using the change in size to indicate upcoming calendar events such as an appointment where the size could be made to shrink as the meeting time draws near, thus allowing a user to probe this information without requiring a person to take the mobile device out of the pocket or purse.

USER STUDY

To validate our assumptions about individuals being able to perceive various changes using a virtual spring, we performed a laboratory study with our prototype. Ten people (3 female) age 22-46 (median 28) from our lab who were unfamiliar with the prototype and its haptic modality volunteered for the experiment. Prior to the experiment each participant was told what the prototype did and the tasks they would be completing. They were allowed to get familiar with SqueezeBlock and explore its behaviors using a graphical control tool running on a PC. Another PC tool allowed us to automate setting behaviors and log participant's responses to the force and displacement changes.

Participants all completed the same seven tasks. The first six tasks test detection of *relative* differences (pairs of test were generated and the participant rated the second in comparison to the first) followed by *absolute values* (label each test case on a given scale) for **stiffness** (K-change), **size**, and **travel distance**. In the seventh task, participants probed the device five times and reported when they felt a **click**. The SqueezeBlock was placed in an opaque bag during all tasks to approximate the scenario of reaching for a mobile phone in a purse or pocket and, more practically, so that visual cues did not bias participants' responses. Participants also wore noise-suppressing headphones playing pink noise during the tests so that mechanical sound variations (however slight) did not bias their responses.

To evaluate relative detection, participants were shown all pair-wise combinations. For example, in the K-change test where there are 5 levels of stiffness, participants were randomly given all 25 stiffness pairs and for each pair they responded whether the second felt softer, stiffer or the same as the first. To test absolute detection, participants were shown a single state and asked to assign an absolute mapping. Absolute states were randomly drawn with the constraint that every level was shown at least once. For example, in K-change participants would squeeze the device and choose the value from 1 to 5 which they believed matched the stiffness. For simplicity, we rendered the absolute input as a virtual fuel gauge (as on a car) with 1 corresponding to Empty and 5 corresponding to Full. Size (25 pairs) and travel distance (16 pairs) were tested similarly.

RESULTS

Figure 5 depicts accuracy in perceiving relative changes at different granularities. For example, the third bar of the second group shows that participants had a mean accuracy of 83% for detecting all 5mm differences in travel distance (i.e. this bar includes 5 versus 10, 10 versus 15, and 15 versus 20 since all these cases are 5mm variations).

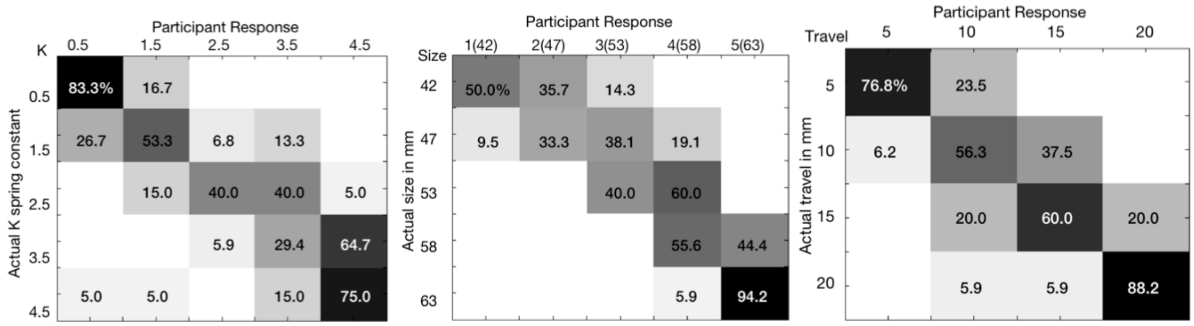

Figure 4: Confusion matrices for participants' absolute perceptions of stiffness, size, and travel distance.

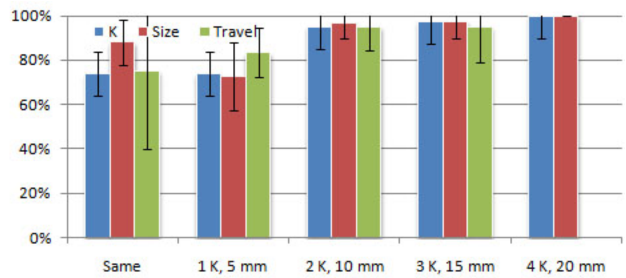

Figure 5: Accuracy in perceiving different granularities of relative change in K, size & travel distance.

We observe that when the variation is at least 2 K-levels or 10mm of size or travel distance, accuracy is above 95%. Participants were also good at noticing when size did not change, but performed poorly in correctly identifying small size variations, which suggests that varying size may be good to convey when something has changed but poor for communicating the direction or magnitude of that change.

Figure 4 shows confusion matrices for the absolute perception tests. Travel distance (right) was perceived most accurately, which correlates with participants' comments that they felt most confident about it. Four participants commented that the speed with which they could reach the end point gave an additional cue to the travel distance, which is valid since a small spring length doesn't allow the squeezing speed to build up. K-value (left) is detectable in the extremes, but was shown to be moderately difficult to distinguish as 5 absolute levels. Participants tended to strictly overestimate size (middle), which may be related to their difficulty in perceiving small relative size variations. For the clicks test, every participant correctly identified when a click was present in all cases.

CONCLUSION

Travel distance variations of 5-10mm over 4 levels performed well in both relative and absolute tests so we believe it is a good approach to convey 4 or 5 bits of absolute information, such as the battery level of a mobile phone. Size and stiffness variations perform less well in absolute and may be more suited to binary or ternary data, but they seem better at conveying dynamic changes, for example to confirm the increase of a phone's ring volume in an eyes-free way. Size appears particularly effective as an indicator that something has changed, but it performs poorly for communicating the direction or magnitude of that change.

A final note is that eight of the ten participants commented that they felt more confident on size and travel distance when they probed the gap between the force plates with their hands and fingers. These were unexpected comments and we think they suggest a ripe opportunity to explore programmatic manipulation of gaps, voids, separations, and contour variations on devices instead of (or in addition to) changes in absolute size. These more nuanced variations are potentially mechanically simpler and smaller allowing multiple haptic displays to fit into a single mobile device.

REFERENCES

1. Bernzen, W. Active Vibration Control of Flexible Robots Using Virtual Spring-damper Systems. J. Intelligent Robotics System. 24, 1 (Jan. 1999), 69-88.
2. C. Mendoza, K. Sundaraj, C. Laugier, A Fast Method to Simulate Virtual Deformable Objects with Force Feedback, IEEE ICARCV, Singapore 2002.
3. Speech enabled eyes-free Android applications. http://code.google.com/p/eyes-free/. Accessed 3/2010.
4. Gurari, N., Kuchenbecker, K. J., and Okamura, A. M. Stiffness discrimination with visual and proprioceptive cues. World Haptics (March 18 - 20, 2009).
5. Hemmert, F. *et al*. Shape-changing mobiles: tapering in one-dimensional deformational displays in mobile phones. TEI 2010 (1/24 - 27, 2010).
6. Klatzky, R.L., Lederman, S.J., & Metzger, V. (1985). Identifying objects by touch: An "expert system". Perception & Psychophysics. 37(4). 299-302.
7. Srinivasan, M.A., Beauregard G.L., Brock,D.L. The impact of visual information on the haptic perception of stiffness in virtual environments. ASME 1996.
8. Yatani, K., Truong, K.N. SemFeel: A User Interface with Semantic Tactile Feedback for Mobile Touchscreen Devices. UIST (October 4-7, Canada), 2009.
9. Zhao, S., Dragicevic, P., Chignell, M., Balakrishnan, R., and Baudisch, P. Earpod: eyes-free menu selection using touch input and reactive audio feedback. Proc. of the CHI (USA, April 28 - May 03, 2007).