Ambient Touch: Designing Tactile Interfaces for Handheld Devices

Ivan Poupyrev [1], Shigeaki Maruyama [2] and Jun Rekimoto [1]

[1] Interaction Lab, Sony CSL,
3-14-13 Higashigotanda, Shinagawa
Tokyo 141-0022, Japan

[2] Micro Device Center, Sony EMCS,
2-15-3 Konan, Minato-ku
Tokyo 108-6201, Japan

{poup, rekimoto}@csl.sony.co.jp, shigeaki.maruyama@jp.sony.com
http://www.csl.sony.co.jp/IL

ABSTRACT
This paper investigates the sense of touch as a channel for communicating with miniature handheld devices. We embedded a PDA with a TouchEngine™ – a thin, miniature lower-power tactile actuator that we have designed specifically to use in mobile interfaces (Figure 1). Unlike previous tactile actuators, the TouchEngine is a universal tactile display that can produce a wide variety of tactile feelings from simple clicks to complex vibrotactile patterns. Using the TouchEngine, we began exploring the design space of interactive tactile feedback for handheld computers. Here, we investigated only a subset of this space: using touch as the ambient, background channel of interaction. We proposed a general approach to design such tactile interfaces and described several implemented prototypes. Finally, our user studies demonstrated 22% faster task completion when we enhanced handheld tilting interfaces with tactile feedback.

Keywords: tactile feedback, mobile devices and interfaces

INTRODUCTION
Miniature handheld computing devices such as mobile phones, PDAs, digital cameras, and music players are rapidly permeating into our lives today. In the near future, we will, perhaps, spend more time interacting with these devices than with conventional desktop computers equipped with familiar graphical user interfaces (GUI). Therefore, the interface design for small handheld computing devices is an important and exciting challenge for interface researchers and designers.

There are many limitations and difficulties to overcome, well documented in interface literature [25, 29, 33, 36]. Many of these limitations, however, also reflect the assumptions and expectations inherited from traditional GUI, that fail in handheld devices, particularly heavy reliance on rich visual feedback and undivided attention from the user. In this work we attempt to extend beyond familiar visual interfaces and employ a relatively unexplored channel for interaction with handheld devices: *the sense of touch*.

There are many possibilities for tactile feedback use in mobile interfaces. Here, we are particularly interested in a small subset of this design space: using touch as the ambient, background channel of interaction. Our basic hypothesis is that in a mobile setting where the user's attention is not fixed on the computer, but on real-world tasks, such ambient tactile interfaces would become a necessary and important counterpart to the traditional visual interaction.

Tactile or coetaneous sense is defined as a combination of various sensations evoked by stimulating the skin [11]. In combination with kinesthesia, tactile feedback is often referred to as *haptics* [4] and is crucial for us to interact with our physical environment. The importance of tactile feedback has been recognized in many fields from virtual reality [8] to design of consumer electronics [1], and it is natural to extend its applications to mobile computers.

Indeed, mobile devices are naturally at a close proximity to our skin, whether it is in our hand or tucked away in a pocket. Skin is the largest human sensory organ (~1.8 m^2 [11]) and with the exception of water and heat regulation, most of it is unused. Mobile interfaces can utilize this.

Tactile feedback also provides superior temporary discrimination, e.g. when rapidly successive data needs to be resolved, the feel of touch is about five times faster than vision [19]. Hence, it allows for precise and fast motor con-

Figure 1: TouchEngine™ actuator – a new vibrotactile actuator for designing tactile interfaces for small handheld computing devices; it is only 0.5 mm thick

trol: When we roll a pencil in our fingers, we can quickly and precisely re-adjust the 3D positions and grasping forces of our fingers by relying entirely on touch [3]. Furthermore, these complex motor operations produce little cognitive load and can be performed in parallel with other activities, such as reading a newspaper. This is because large areas of the sensory cortex are devoted to processing stimuli from the skin. Moreover, a large amount of the processing occurs in the lower level of the spinal cord, where sensory and motor neuron fibers intersect [11][1]. By re-directing some of the information processing from the visual channel to touch, we can take advantage of this ability to reduce the cognitive load and make it easier to operate mobile devices.

The skin also acts as a powerful information pick-up channel. In the real world, it allows us to correctly judge object properties when vision fails: e.g. textures and surface variations can be accurately detected by touch [9]. Geldart in 1957 [18] developed a vibrotactile language called "Vibratese" and demonstrated that trained subjects were able to receive a complex message up to 38 words per minute. This and later studies [38] show that with proper encoding, messages can be transmitted through the skin. We can take advantage of this when designing mobile interfaces.

The message, however, does not necessarily need to be symbolic: touch has a strong emotional impact. Running a finger into a splinter, touching a cat's fur, or immersing into some unknown sticky substance all bring intense, though very different, emotional responses. Hence, touch is a very strong "break-in" sense: coetaneous sensations, especially if aroused in unusual patterns, are highly attention-demanding [17]. This has already been explored in some mobile interfaces, but there is still more room for research.

To conclude, these properties make touch an ideal channel of interaction with handheld devices: it is fast, needs little conscious control, allows for information encoding, and produces strong emotional responses. This paper investigates some of the implications of using tactile feedback in mobile interfaces.

We begin with a review of a related work. We then describe the design and implementation of novel tactile displays based on the TouchEngine, a miniature low-power actuator that we created specifically for small handheld devices. Unlike other previously reported actuators, the TouchEngine is a versatile device that can be used to generate a wide spectrum of tactile feelings. We continue by exploring the design space for tactile displays in mobile devices and investigate several applications using touch as the ambient, background channel for mobile communication. Finally, we report the results of the experimental studies, which demonstrated that tactile feedback resulted, on average, 22% faster task completion when used in combination with tilting interfaces in a 1D-scrolling task.

RELATED WORK

The importance of tactile feedback in human-machine interaction is well recognized [1, 7]. Much effort has been spent on simulating realistic tactile properties of our physical environment, such as roughness of surfaces. This has been achieved by either developing new, special purpose haptic devices [9, 32] or by adding tactile display to usual desktop devices, e.g. a mouse or touchpad [2, 26, 30]. In passive tactile feedback, alternatively, the user can "feel" virtual objects via manipulating physical objects with similar haptic properties, e.g. shape or surface texture. For example, in one system, a user rotates a doll's head to rotate a virtual 3D model of a human brain [24]. This approach is being also explored in the tangible and graspable interface research areas [13, 27, 35].

Tactile interfaces were intended to enhance the user experience by making interfaces more realistic, intuitive, and easy to use. Another direction of research investigated how information can be encoded and transmitted to the user through stimulation of the skin [17, 18, 38]. Results from this work were used to develop devices to assist blind or deaf people, e.g. Optacon [38]. Most of these devices rely on vibrotactile arrays to create a "tactile picture" that can be recognized through touch [e.g. 10].

Unfortunately, little work has been done to add tactile feedback to mobile handheld devices. To effectively utilize tactile feedback, we must initially overcome two major challenges. First, there are *feasibility limitations* in the current actuator technology [20]. The fundamental requirements for mobile actuators are 1) small, miniature size; 2) lightweight; 3) low voltage (~5V) and low power consumption; and 4) ease in customization to allow retrofitting to devices of various sizes and forms.

While technical characteristics of displays are important, we should not forget that they should also *feel good*. Satisfying only feasibility requirements would not produce expressive tactile displays. Although touch may not be as rich as vision, it is by no means a one-dimensional sense: we have an amazing range of tactile sensations and tactile displays should be able to evoke them. Depending on the design, the difference in feeling between tactile displays can be profound; perhaps, as profound as the difference between today's high-resolution color monitors to black and white vector displays from the 1960s.

Hence, the second obstacle is *human factors limitations*. We still have limited understanding of the sense of touch, but we can suggest that the optimal mobile tactile display should satisfy the following requirements [9, 10, 17, 35]:

1. *Fast response.* The minimal time in which humans can detect two consecutive tactile stimuli is about 5 ms [11]. Note, that this is about five times faster then vision. Hence, we can approximate the minimum required latency of a tactile actuator to be ~5ms. A large lag would significantly reduce the quality of feeling and would not allow users to effectively regulate their motor movements or perceive complex tactile patterns. For example, vibration motors in mobile phones have significant latency and, therefore, cannot be used in interactive applications.

2. *Variable intensity*. Humans can discriminate a vast range of intensities of tactile stimulus. Given appropriate conditions, the palm of a hand can feel vibratory stimulus with

[1] In a striking example of this integration, it has been shown that cat walking may be entirely controlled within its spinal cord [9].

an amplitude as low as 0.2 microns [21]. On the other hand, for a comfortable tactile feeling, a much larger displacement should also be provided. Hence, the actuator should be able to generate variable skin displacements.

3. *Wide frequency bandwidth*. The vibration frequency has a profound effect on tactile perception. For example, single tactile pulses, or taps, are perceived differently from the sinusoidal vibration, because different coetaneous receptors respond to these patterns [11]. The ideal actuator should provide variable frequencies from 1 to 1000 Hz.

4. *Multitude of different wave shapes*. Humans are able to distinguish a wide range of tactile wave shapes. Gault in 1924 [17] converted speech into vibration and found that trained subjects were able to distinguish one out of 10 short sentences with 75% accuracy. Hence, to be effective, tactile actuators should be able to produce numerous wave shapes.

We are not aware of tactile actuators that satisfy these criteria: most were designed to create a single type of tactile feeling for a narrow application. Vibration motors, for example, rotate eccentrically weighted shafts that vibrate at about 130Hz. Because its purpose is to alert the user, it is not important that there is significant latency and that only sinusoidal patterns with limited amplitudes and range of frequencies can be displayed. These limitations, however, make it impossible to use vibration motors to encode any complex tactile feelings and to use them in interactive applications where small latency is crucial.

Voice coils and speakers can also be used for tactile stimulation [20], but the displacement and force they provide is low. Recently Fukamoto [16] embedded a voice coil type actuator that provided sufficient force when driven at the resonant frequency**.** However, only a single frequency and amplitude of vibration can be produced; therefore, complex patterns of vibration cannot be encoded.

Other interesting technologies are tactile matrix arrays (e.g. [12]) and direct electrical stimulation [28]. Matrix arrays are usually large and heavy, and require a considerable amount of voltage and power, e.g. 350V in the case of [12], making them impossible to use in mobile devices. Electrocoetaneous devices can be very small and efficient, but their feel is quite different from the familiar tactile feeling.

As with any display technology, properties of tactile displays determine their interface applications. Currently few applications of mobile tactile displays are available. Vibration motors have long been used as silent alarm devices, however, because of their inherent limitations, they provide only one bit of information: an occurrence of an event, e.g. a phone call. Probably, the first interactive tactile feedback for handheld devices was implemented in the Active Click by Fukamoto [16]. Active Click was designed to provide a tactile "click" feeling when users touch the graphical buttons on a touch screen. The coil actuators that he used have very low latency and the illusion of physical button pushing was strong. However, this is only a narrow application–due to the limitations we discussed above, it might be difficult to further expand the Active Click into other applications.

Tactile displays have been proposed for wearable computers [20, 39] to be used, for example, as a navigation aid. In this work, we, however, are more interested in handheld devices and more interactive applications than display of navigating information. We were inspired by designs that attempted to deviate from the traditional GUI paradigm to create interfaces explicitly tailored for mobile devices. Examples include context-aware and ambient interfaces, embodied and gesture interfaces, augmented reality, and others [6, 15, 25, 36]. Most of these interfaces still center on visual interaction and assume focused attention by the user. By augmenting them with tactile displays, we want to expand these concepts leading to new mobile interfaces.

DESIGNING A HANDHELD TACTILE DISPLAY

Development of any haptic display involves: 1) Choosing or developing a *haptic actuator*, a transducer that converts electrical signals into mechanical motion; electrical motors are typical actuators used in haptic research. 2) Designing a *tactile display* that converts mechanical motion produced by the actuator into force communicated to the user. The same actuator can lead to various haptic displays with very different interactive properties. Indeed, motors have been used in haptic displays ranging from exoskeletons to small vibrators in mobile phones. 3) Developing *control hardware and software*. Effective operation of haptic display requires in-depth understanding of its properties.

This section describes the design of the TouchEngine–a new tactile technology that includes actuator, tactile display and control hardware and software that we have developed to overcome deficiencies of current tactile displays.

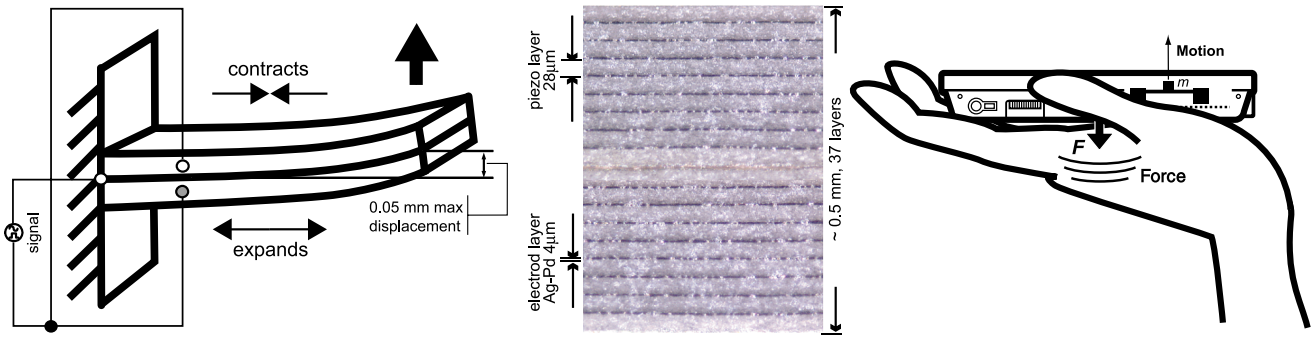

Figure 2 *Left*: The bending motor: the top layers contract and the bottom expands bending the entire actuator; *Middle*: A microscopic view: 18 layers of piezo and 19 layers of electrode; *Right*: Indirect haptic display.

TouchEngine actuator

Basic structure

The TouchEngine actuator is constructed as a sandwich of thin (~0.28μm) piezoceramic film with printed adhesive electrodes in between, forming an extremely thin (less then 0.5mm) beam (Figure 1 and Figure 2). The piezoceramic material works as a solid state "muscle" by either shrinking or expanding, depending on the polarity of the applied voltage. The material on the top has an opposite polarity to that on the bottom, so when a signal is applied the entire structure bends (Figure 2, left). This configuration is often called a "bending motor" actuator.

Bending motors that were previously used in tactile displays consisted of only two layers (biomorphs) and required extremely high voltage (e.g. 350V in [12]) making them unsuitable for mobile devices. By sandwiching multiple layers of very thin piezo film with the adhesive electrode (Figure 2, middle), we can reduce the voltage required for maximum displacement to ±8-10V. Indeed, for the voltage V and the thickness of the piezoelectric layer T, the displacement D and force F for the serially connected bending motors are as follows [40]:

$$D = 4l^2 d_{31} \frac{V}{T^2} \quad \text{and} \quad F = \frac{4w}{3g_{31}} T \cdot V \qquad (1)$$

where l and w are beam dimensions, and d_{31} and g_{31} are piezoelectric constants. We can see that by decreasing the thickness T, we can achieve the same displacement D with a lower voltage. This, however, also decreases the force F; we compensate this by layering multiple thin piezoceramic layers together (Figure 2, middle).

The resulting actuator holds unique properties. It is extremely thin, small, can be battery-operated, and manufactured in various sizes and number of layers. It is also fast allowing us to control both the amplitude and frequency of vibrations at the same time, to create a variety of tactile waveforms – this is not possible with any other actuator.

Electromechanical properties

This section discusses the TouchEngine's electromechanical properties. Note that this discussion applies to *any* piezoceramic bending motor actuators.

The relation between the signal voltage, the actuator displacement, and the force is linear (equation 1). We, however, are more concerned with the actuator dynamics, particularly how fast an actuator can reach the required displacement: The greater the acceleration of the actuator, the stronger the impulse force becomes (see equation 4) and the sharper the tactile impulses can be detected by the user. The actuator latency will depend on its electrical properties.

Electrically, the piezoceramic actuator behaves as a *capacitor*. The typical capacitances for TouchEngine are ~3μF for a 7-layer and ~4μF for an 11-layer actuators. Consequently, the voltage across the actuator–and the displacement–would change according to the capacitor charging and discharging waveforms (Figure 3). Because a capacitor reaches over 99% of it is charge in 5RC seconds, [22] we can estimate the latency of an ideal actuator as:

$$\tau = 5RC. \qquad (2)$$

where C is the actuator capacitance and R is the serial resistance [22]. Hence, the latency of an actuator is *constant* and does not depend on the voltage; an actuator would reach any target displacement in the *same* amount of time.

For a real actuator internal resistance and inductance, material dumping, physical mounting, and limitations in the current supply would cause further delay in the displacement. Figure 4 shows an actual mechanical response of an 11-layer actuator measured using a laser displacement sensor. Latency of approximately 5ms can be clearly seen, which is still near optimal latency for the tactile actuator. Increasing latency by increasing the capacitance and serial resistance can produce a softer and gentler feeling, which may be necessary in some applications.

Current requirement is another very important issue because current supply significantly affects actuator response time, while in mobile devices current supply is limited. Current flows through the piezo actuator *only* when it bends. The actuator then stays in the bended state without draining any additional current. The current equation is:

$$i = C\frac{dV}{dt}, \qquad (3)$$

where C is the capacitance and V is the signal. The current is a sinusoid for a sinusoidal signal and a narrow spike for a square wave [22]. This peak current can be fairly high: up to 450 mA for a 20V peak-to-peak signal. If the battery

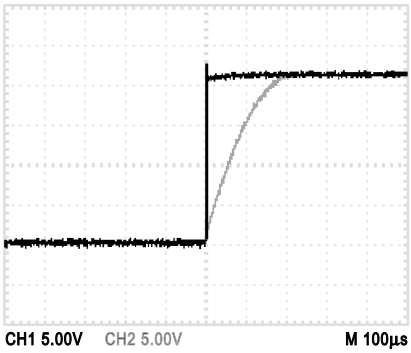

Figure 3: *Black*: input signal; *grey:* actuator voltage: a capacitive charging/discharging curve.

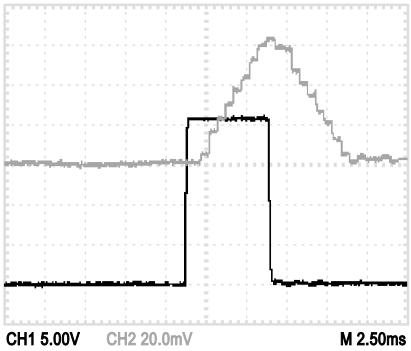

Figure 4: *Black:* input signal, *grey*: actuator displacement (its jaggy because of the limited sensor resolution).

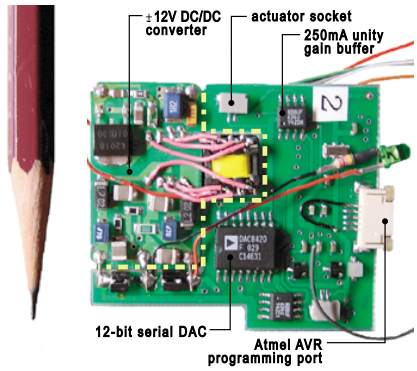

Figure 5: Control board that drives TouchEngine actuator.

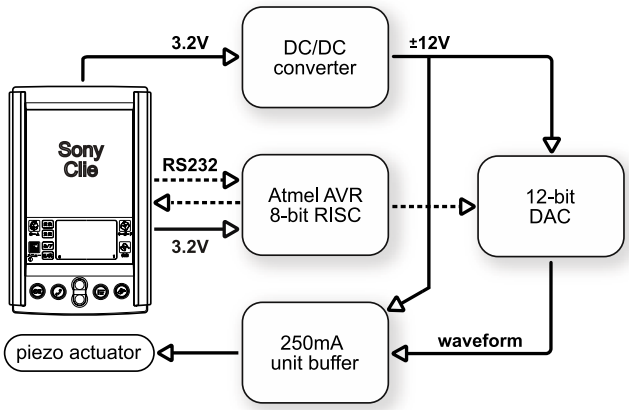

Figure 6: Basic diagram of the driving board.

cannot supply the maximum required current the actuator latency will *increase*, e.g. by integrating equation (3):

$$v(t) = C^{-1} i \cdot t$$

where t is time. Therefore, in case of limited current supply the actuator voltage (and displacement) changes not as a capacitive curve on Figure 3, but as a linear function of time: more current can be supplied, the faster the actuator bends. Thus, current amplifiers are essential for achieving low latency. We have found that 250 mA unity-gain buffers were sufficient to achieve a latency of 5ms.

While the peak current may be large, the *average* current is low and dependant on the signal frequency and maximum voltage. We have found that actuator require approximately 3 mW of power for a single 20V peak-to-peak square pulse, which can be easily achieved in handheld devices.

TouchEngine display

The actuator that we designed bends from the signal applied. The next step is to convert this mechanical motion into a detectable force. The major challenge in doing this is the very small total displacement of the actuator, less then 0.1 mm. Two strategies have been developed to utilize our actuator: direct tactile display and indirect tactile display.

In *direct tactile display* the actuator moves a part of the device, e.g. a single button or an entire PDA screen. The basic interaction metaphor is direct touch: when the user touches a part augmented with tactile feedback, various tactile patterns communicate back. Different elements of the device can be augmented with distinct tactile feedback, however, it may be difficult and expensive to retrofit all the elements of a device with its own tactile actuators.

Indirect tactile display is based on the tool metaphor: When we hit a nail with a hammer, the force is not communicated directly to our hand; we feel the effect of the impact *through* the handle. To achieve a similar effect, the actuator is placed anywhere within the device with a small weight attached to it (Figure 2, right). Force is generated using conservation of momentum; for an isolated system with no external force, its total momentum is zero. Thus, when the actuator bends, the mass moves up or down with momentum p_a and the entire device moves with *equal* momentum p_d in the opposite direction. The force F felt by the user is the time derivative of the momentum; therefore, higher acceleration of the actuator results in a stronger force:

$$\vec{F}_d = \frac{d\vec{p}_d}{dt} = -\frac{d\vec{p}_a}{dt} = -m\frac{d\vec{v}_a}{dt}. \quad (4)$$

Here, the entire handheld device acts as a tactile display. Because our actuator moves very fast, we can create very sharp, distinct force impulses. Also, the actuator can be attached anywhere inside the device making construction of a tactile display easy, even for very small devices.

Implementing a mobile tactile display

We embedded a TouchEngine tactile display into a Sony Clié PEG-N700C Palm OS based PDA. A custom board that controls the actuator is inserted into the PDA (Figure 5). We freed the space inside the PDA by removing the Memory Stick slot, Jog Dial, and pen compartment.

A basic diagram of the board is presented in Figure 6. An Atmel AVR 8-bit RISC microprocessor creates different waveforms and sends them to the actuator using a 12-bit digital-to-analog converter (DAC). The power regulator (a DC-DC converter) is then used to boost the voltage from the 3.2V of the PDA battery to ±12V, and the current is further amplified using 250 mA unity-gain buffers. Caution must be taken to prevent amplifier oscillations, which often occurs when driving large capacitive loads.

Applications running on the Clié's Palm OS communicate with the AVR chip via a serial port. We can easily reprogram the microprocessor without removing the board from the PDA using a programming socket, so various tactile feedback applications can be prototyped on the same platform. The board can be optimized further, such as using power supplies that are much smaller and more efficient. Nevertheless, we are very satisfied with this platform as it fits entirely within the PDA, invisible to the user; provides high quality tactile feedback; and is easy to use for prototyping and evaluating mobile tactile interfaces.

MOBILE TACTILE INTERFACES

This section analyzes the design space of applications for mobile tactile interfaces. We do not attempt to suggest the ultimate taxonomy of mobile tactile interfaces: there are, perhaps, more applications than we can imagine at this point; other taxonomies can also be used (e.g. [31]). Instead, this taxonomy is only meant to be a rough outline that can guide designers and developers.

Ambient displays and touch

In our analysis of tactile mobile interfaces, we were inspired by previous research on ambient and peripheral awareness interfaces [10, 27, 34, 37]. They attempt to utilize human ability to receive and process information on the periphery of attention, without shifting focus from the primary task. For example, it is relatively easy for someone to note that a meeting is about to start in the next office or that the weather is about to change [10].

The premise of ambient or peripheral awareness displays is that they would first, allow users to perceive and process incoming information with little conscious effort and without need to interrupt current activities. Second, they would provide unobtrusive notification techniques to shift the

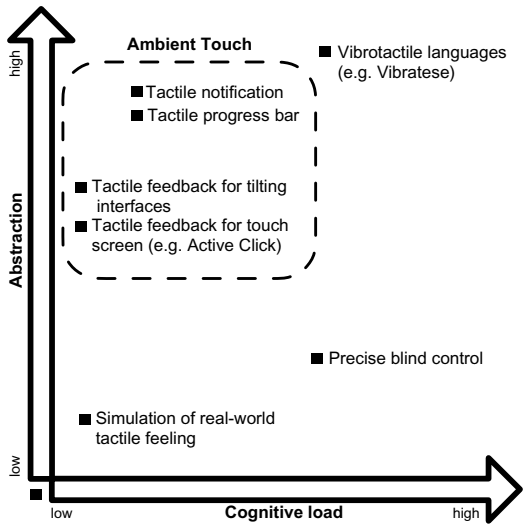

Figure 7: Design space of mobile tactile interfaces

user's attention when the interruption is needed. A number of ambient displays has been explored [27, 37], however, all of them relied on visual or aural channels to communicate information to users.

We believe that touch is the perfect ambient information channel, in many cases, more powerful then visuals or sound. We already heavily rely on the tactile sense to receive information in the background: we feel the ground with our feet and unconsciously adjust how we walk when we step from the pavement to the grassy lawn; the movement from clothes informs us of changes in the wind direction; and slight vibrations of a pen help to regulate the amount of pressure applied to the paper. Tactile feedback is an extraordinary attention management technique. We may not hear someone calling our name, but we certainly react when someone touches our shoulder. Moreover, because of the reasons discussed in the introduction, most of this is done with little focused attention, almost unconsciously.

Ambient tactile interfaces may be particularly useful in mobile applications. Indeed, as mobile users are often preoccupied with real world tasks; tactile displays that are always close to the body allow information to be received without interrupting the current activity to look at the visual display [37]. Even when a user is focused on a mobile device, constant distractions and noise do not allow constant attention on mobile devices; hence, touch may substitute vision for short distractions. Finally, due to superior temporal processing abilities, we may be able to transfer some of the information processing from the eye to the finger.

Design space of mobile tactile interfaces

We have organized the design space of mobile tactile interfaces along two dimensions (Figure 7). The first dimension is the amount of *cognitive load* the interface imposes on the user. A large cognitive load requires more focused attention on the mobile interface at the expense of other tasks.

The second dimension is the level of *abstractness* of information representation in tactile interfaces. The idea of abstract representation in ambient interfaces was first formulated by Pederson, et al. in the AROMA system [34]. They proposed that a signal should be simplified to retain only the most significant bits of information and then re-mapped into a different *abstract* representation suitable for a particular ambient display and application. This idea provides an approach to tackle an important problem in ambient interfaces: how to communicate "enough" information to make the display useful, while at the same time, produce low cognitive load and create easy to use interfaces.

The lowest amount of abstraction in tactile displays is when we directly simulate the tactile feeling of real-world objects. This was investigated in VR and telepresence applications [8]. A tactile interface in this case does not usually require much additional attention from the user. The exception is when the user has to perform precise identification or control only through touch, e.g. blind control.

A tactile signal can be abstracted to communicate only particular tactile properties, e.g. button presses can be simplified to a simple sinusoidal pulse to create the button "click" feeling. The entire simulation of the button physics may not be needed [16]. In this case the tactile signal is in a sense a metaphor of the button; other tactile metaphors can be designed to communicate various feelings, e.g. scratching, tapping, breaking, and so on. By associating these tactile signals with interface events we can communicate the state of the mobile interface through touch. We can further increase abstractness of the tactile signal by using vibrotactile languages, such as Vibrotese, that do not resemble any realistic tactile events but can carry a complex message [38]. Using them, however, drastically increases cognitive load.

Here, we are interested in the left top corner of the design space shown in Figure 7 that we refer to as Ambient Touch. We investigate some of its applications in the next section.

AMBIENT TOUCH INTERFACES

This section describes several application scenarios to evaluate mobile tactile displays. Although the TouchEngine can produce a great variety of vibrotactile patterns, we used only simple square waveforms of different frequencies and intensities (e.g. Figure 3). We still lack the in-depth understanding of how we can design vibration patterns to evoke specific user feelings; hence, for the initial investigation, we have chosen very simple patterns in which the feeling can be easily predicted and used in interface design.

We conducted a formal user study for one of the applications: tactile feedback for tilting interfaces. With the remaining applications, we informally tested them using our colleagues as subjects: the goal was not to collect data but evaluate the feasibility of our ideas.

Notification through touch

Touch is an excellent "break-in" sense and has been used to alert people since the 1960s [19]. Today, vibrotactile displays are commonly utilized for user notification in mobile phones and, recently, in handheld computers. The common problem, however, is that current tactile displays convey only a *single bit* of information: the simple occurrence of an event, e.g. a phone call. Because no related information is provided, such as who is calling or call urgency, the user is forced to interrupt their current activity, interact with the device, access the relevant information, and make the fol-

low-up decision. This can be frustrating and dangerous, e.g. when the user checks a caller while driving.

To address this problem, intelligent context-aware and ambient notification mechanisms that use sound have been proposed (e.g. [37]). Tactile notification, however, is preferable when sound cannot be used, such as during meetings and in noisy environments. Tactile feedback has not been used for intelligent notification simply because no mobile tactile display was available that could produce complex vibration patterns in a short temporal event that can be remembered and recognized.

We were encouraged by the studies on tactile communication [19, 38] reporting that humans can learn to accurately distinguish different vibrotactile patterns. To experiment with tactile notification, we created six tactile pulse sequences with a different *rhythm*, *intensity*, and *gradient* of vibration. We used the drum as a metaphor to help users remember the patterns: simple rhythms were tapped and subjects were asked to repeat them and compare them with the same patterns reproduced on a mobile device. After a brief training period, we asked subjects to recognize the vibration patterns when the device was placed in the pocket; most subjects were able to identify them easily.

We would like to further investigate tactile notification mechanisms combining them with ideas from contextual messaging, e.g. [37]. We envision interfaces, where users can feel and recognize incoming information without interrupting their activity to interact directly with the device.

Tactile monitoring

Notification techniques alert users about incoming events. A similar task is monitoring the status of a process, such as downloading large files or copying data to an external storage. The main difference from notification concepts is that here the user continuously monitors the time before process completion, not a single event. When a process is nearly complete, the user can prepare for the interruption in current activities, such as step aside from the sidewalk to access the mobile device. With desktop computers, this is easy to implement: the user can visually monitor the status of a process. For mobile users, visual monitoring would require periodical interruption of their current activity to access their device and check on the status, which can be frustrating and inefficient.

We designed a tactile progress bar, to tactually inform the user about the process status. The first variation of the tactile progress bar simply provided the user with a short tactile impulse every 0.5 second. The intensity of the click increased as the process gets closer to completion. The evaluation revealed, however, that this design had no significant difference from the notification, because it only provided binary information on whether the process has finished or not. Absolute levels of intensity were not informative as humans have poor capabilities to judge them.

We then re-designed this technique to take advantage of touch temporal discrimination abilities [19]: the current process status was mapped into the time between the two clicks. As the process progresses the time between the two clicks decreases. This modification was well received by users, as it was easy to relate the tactile feedback to the current status of the process. Certainly, more study is needed to further investigate this interaction technique.

Tactile feedback for gestural mobile interfaces

Using gestures to interact with mobile devices have been an active research area [5, 23, 25, 36]. Devices are augmented with sensors, such as tilt or pressure sensors, and users interact with them by physically manipulating the devices. For example, in tilting interfaces [14, 36] the user can scroll through data by tilting the handheld device back and forth, in which the tilting angle controls the scrolling speed. This technique is intuitive and allows for one-handed use.

Our experience with tilting interfaces, however, was not entirely satisfactory. First of all, we found that users would often *overshoot* the target destination, i.e. when the target is reached, users could not return the device to its neutral state in time to stop scrolling. Moreover, users often overshoot in the opposite direction, e.g. while trying to stop scrolling, the user would miss the neutral position and begin tilting in the opposite direction. This makes precise selection control quite difficult ([23, 25] reported similar problems). We also found that slower speeds would reduce the overshoot, but increase the time to complete tasks. Second, tilting interfaces require *constant visual attention*. Even a momentary distraction may result in a loss of control, leading to unpredictable results: e.g. an imperceptible twist of the hand while a user checks the name of a passing station might change the entire screen content without the user noticing.

We believed that tilting interfaces could be improved by combining gesture control with tactile feedback, which would allow the user to "feel" how the information moves inside the device. It would allow the user to: 1) maintain continuous awareness of the interface state during brief interruptions to perform other tasks; 2) provide rapid feedback for tilting operations; because touch is faster then vision, we believe that it allows users to more rapidly readjust their hand to reduce overshoot than only relying on vision; 3) communicate interface states, such as reaching neutral orientation or boundaries of data.

Implementation of tilting interfaces

We used an ADXL202 two-degree accelerometer connected to unused input ports on the AVR microprocessor to measure the tilt of the Clié PDA. The sensor was set to operate at the lowest bandwidth (5Hz) to minimize noise and jitters. We sampled tilt at approximately 40 Hz.

The user activates the tilt mode by pressing the *back* button on the left side of the PDA (Figure 8). Scrolling begins when the device tilts more then 5 degrees from the initial device orientation, e.g. when the button is first pressed. Two speeds follow: data scrolls once every 500 ms when the tilt angle is between 5 and 10 degrees, and once every 250 ms when the device tilts more then 10 degrees. Scrolling was carefully synchronized with tactile feedback taking into account the time to update the visual display.

Tactile applications

Two tilting applications were investigated:

2D browsing of graphical data. Figure 8 presents an example of browsing a Tokyo subway map. The map scrolls in

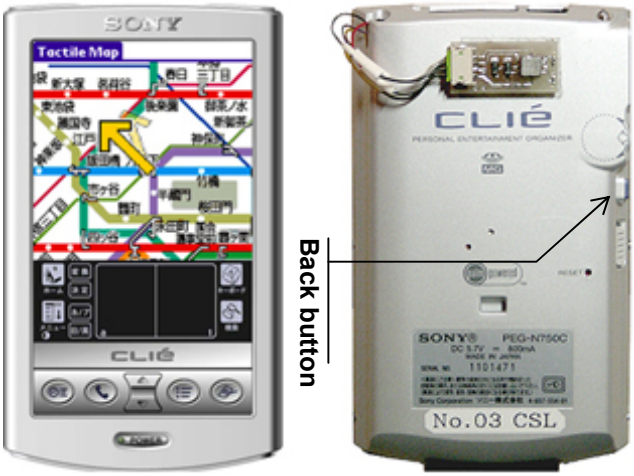

Figure 8 Tilting interfaces: 2D browsing of a Tokyo subway map. Tilt sensor board is attached on the back.

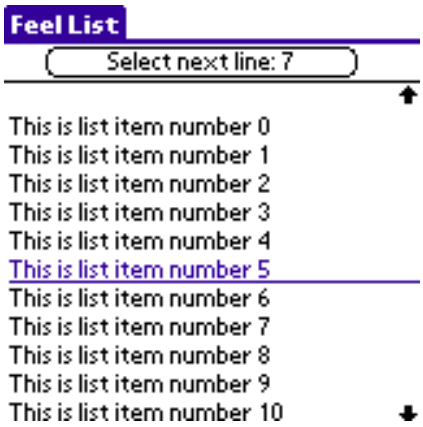

Figure 9: Screen snapshot of the task at the start of a trial: the user is required to scroll from the starting position, for example, from line 5 to the target, line 7.

the direction of the tilt where the yellow arrow indicates the direction of scrolling; The arrow increases in length as speed increases. Every time the image moves on the screen, a simple scratching tactile pattern plays, allowing the user to feel the map shifting within the device. The tactile pattern changes to a single low-intensity pulse when the image reaches its boundaries. We received positive feedback through an informal evaluation as users felt that tactile feedback made interaction easier and more enjoyable.

1D scrolling through text lists. A simple tactile "tap" occurs in the device every time one line of text is scrolled up or down. As the scrolling speed changes, the users would feel the tapping in their hand become faster or slower. The interface also produces a short buzz when the user returns the device to its initial position, reducing the overshoot when the user attempt to stop scrolling. We evaluated the text list scrolling interface in a controlled experimental study discussed in the next section.

EXPERIMENTAL STUDY

An experimental usability evaluation was conducted to investigate whether adding tactile feedback would affect user performance in tilting tasks, and if it does, how significantly. The following hypotheses were formulated prior to the experiments:

H1: Tactile feedback will result in faster task completion in one-dimensional text scrolling tasks.

H2: Tactile feedback will reduce overshot of target in one-dimensional text scrolling tasks.

Subjects and apparatus

Ten unpaid subjects, all male, all right-handed, between 19 and 35 years old, were recruited from the laboratory subjects' pool. The experiments were conducted using a Sony Clié Palm OS 4.1 PDA running in color 160x160 resolution mode equipped with a TouchEngine feedback device and tilt sensor board described above.

Experimental task

The experimental task required subjects to scroll a text list from the start to target line, where each line is numbered (Figure 9). At the start of a trial, the list is reset to the start line and the target line number is indicated at the top of the screen. The current line is underlined. To begin, the user presses the trigger button and tilts the device. Note that only the text scrolls, the position of the selection bar does not move – it remains in the middle of the screen. The task is completed when the user selects the target line and releases the button. After a delay, the next trial is presented and so on, until all trials are completed.

Experiment design and procedure

The repeated-measures within-subject experimental design was used. The independent variables are *tactile feedback* and *distance* of scrolling, defined as the number of lines from the start to the target lines with three levels: *short* (1 line), *medium* (6 lines) and *large* (12 lines). The start and target lines were never placed at the beginning or the end of the list, as these conditions would be easier to complete, thus, confounding experimental results. The dependent variables are *completion time*, i.e. the time from the moment the user presses the trigger until target selection, and *overshoot*, defined as the difference between the required and actual scrolled distances.

The experiment began with an explanation of the techniques, experimental tasks, and procedure, followed by a 5-10 minute training session. The user was then asked to operate the device with the left hand, as the left hand is usually used to operate mobile devices. The training was followed by an experiment consisting of six sessions, three with and three without tactile feedback. Each session consisted of seven trials: two trials for each of the three distances randomized and one warm-up trial at the beginning of the session, where its data was not used in the final analysis. To control for order effect, half of the subjects started with tactile feedback while the other half started without. The TouchEngine produced sound; therefore, subjects were required to listen to music with headphones during the experiment. We believe that this closely approximates to the actual conditions of mobile device use.

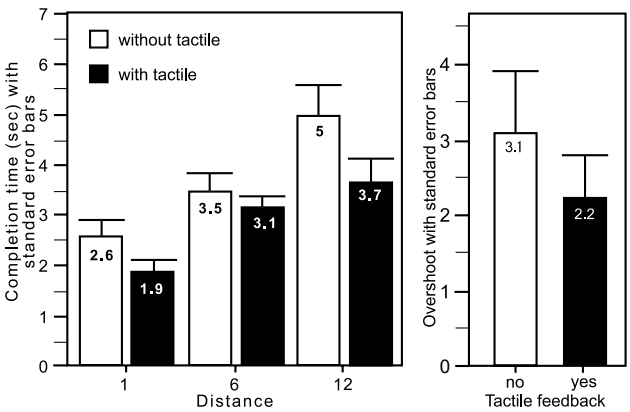

Figure 10: Experimental results

After the experiment, a questionnaire was administered, in which we asked each subjects to rate the techniques on a scale from 1 to 5 (1 = very bad, 2 = bad, 3 = OK, 4 = good, and 5 = excellent) and explain their choices.

Results

A repeated-measures two-way analysis of variance (ANOVA) was performed for each of the dependent variables with tactile feedback and distance as independent variables. Averaged across conditions, tactile feedback resulted in 22% faster task completion (2.9 sec with and 3.7 without tactile feedback), which was also statistically significant $F_{1,9} = 15.6$, $p < 0.003$. Analyzing for each distance (Figure 10) revealed that tactile feedback resulted in a 27% improvement for one-line scrolling, 26% for long distance, while for medium distance, it was only an 11% improvement[2]. The largest absolute improvement was for long-distance scrolling (1.3 second) while the smallest absolute improvement was for the medium distance (0.4 seconds, not statistically significant $F_{1,9} = 0.3$, $p < 0.6$).

Tactile feedback was weakly significant for the overshoot distance $F_{1,9} = 6.4$, $p < 0.03$: on average, it was 3.1 lines without and 2.2 lines with tactile feedback. Distance was statistically significant for both time and overshoot. User ratings were 3.9, e.g. almost good, with tactile feedback and 2.5, e.g. between bad and OK, without it.

Discussion

The experimental results supported hypothesis H1: tactile feedback resulted in 22% faster task completion for tilting interfaces. The effect of tactile feedback was strongest for short and long distances. Based on our observations and user comments we found that exact positioning, e.g. to scroll to the next line, was difficult. One reason might be that the threshold angle was too small and the initial scrolling speed was too fast, hence subjects could not stop quickly enough when the list started scrolling, resulting in an overshoot. Moreover, because no tactile feedback was provided *before* the scrolling started, users faced difficulties in planning their movements in advance. In this situation, tactile feedback was helpful as it allowed the user to react 22% faster then with only visual feedback.

[2] Conditions with no tactile feedback were taken as 100%

For long distance scrolling, subjects would switch to the fastest scrolling speed in which touch was more effective as a feedback channel then vision, as it allowed users to "feel" the scrolled amount. On the medium distance, however, the user scrolled at a comfortable medium speed, which allowed precise selection, relying only on vision; hence tactile feedback did not provide significant benefits.

Tactile feedback was also statistically significant for the overshoot, thus supporting hypothesis *H2*. However, it was not numerically significant: tactile feedback improved user performance by less then a line. One explanation might be that overshoot does not affect user performance in terms of distance, but in terms of the time that it takes the user to recover from an overshoot, re-adjust his or her hand, and correct the problem. Tactile feedback, in this case, allowed for more confident, rapid adjustment, and recovery.

The ratings and comments provided by the subjects indicate that they prefer to have tactile feedback in the device. Interestingly, half of the users preferred tactile feedback not because of functional benefits, such as better coordination or faster selection, but because of a better user *experience*. Some suggested that emotionally it feels more comfortable and familiar. On the question of what was the most difficult element of the interface, all subjects noted that selecting the next line was the most difficult and frustrating task.

CONCLUSIONS

We believe that tactile feedback can become an important feature of interfaces for future mobile devices by providing more effective, comfortable, and enjoyable interaction. Our experiments demonstrated that introducing tactile feedback resulted in a 22% improvement on user performance. This paper has only begun mapping out the design space of tactile displays for mobile devices, an area that has yet to be investigated. In our future work, we will continue our investigation of the tactile design space and search for new applications of mobile tactile feedback.

AKNOWLEDGEMENT

We are deeply indebted to Shigeru Tajima for his infinite patience while helping us with the TouchEngine circuit design, to Henry Newton-Dunn for thought-provocative and enlightening discussions, Mimi Huang for making this paper readable, James Gibson for on-screen graphics, and Sidney Fels for interesting comments. We also thank all subjects who participated in the user study and reviewers for their valuable suggestions for improving this paper.

REFERENCES

1. *Design innovations yearbook 2000*. 2000, Design-Zentrum Nordrhein Westfallen, Verlagpp.
2. Akamatsu, M., S. Sato, *A multi-modal mouse with tactile and force feedback.* International Journal of Human-Computer Studies, 1994. 40(3): p. 443-453.
3. Annaswamy, A.M., M.A. Srinivasan, *The role of compliant fingerpads in grasping and manipulation: Identification and control*, in *Essays on Mathematical Robotics: The IMA Volumes in Mathematics and its Applications.*, J. Baillieul and S. Sastry, Editors. 1998, Springer-Verlag.

4. Apelle, S., *Haptic perception of form: activity and stimulus attributes*, in *The psychology of touch*, M. Heller and W. Schiff, Editors. 1991, Lawrence Erlbaum: Hillsdale, NJ. p. 169-188.
5. Bartlett, J., *Rock'n'Scroll is here to stay.* IEEE Computer Graphics & Applications, 2000(2): p. 40-45.
6. Billinghurst, M., J. Bowskill, N. Dyer, J. Morphett. *An evaluation of wearable information spaces*. in *VRAIS'98*. 1998: IEEE p. 20-27.
7. Brooks, F.P., Jr., Ming-Ouh-Young, J.J. Batter, P.J. Killpatrick. *Project GROPE-Haptic Displays for Scientific Visualization*. in *SIGGRAPH'90*. 1990: p. 177-185.
8. Burdea, G., *Force and touch feedback for virtual reality*. 1996, John Wiley and Sons. 339 pp.
9. Buttolo, P., B. Hannaford. *Pen-based force display for precision manipulation in virtual environments*. in *VRAIS*. 1995: IEEE p. 217-224.
10. Buxton, B. *Integrating the periphery and context: a new taxonomy of telematics*. in *Graphics Interfaces*. 1995 p. 239-246.
11. Cholewiak, R., A. Collins, *Sensory and physiological bases of touch*, in *The psychology of touch*, M. Heller and W. Schiff, Ed. 1991, Lawrence Erlbaum. p. 23-60.
12. Cholewiak, R., C. Sherrick, *A computer-controlled matrix system for presentation to skin of complex spatio-temporal pattern.* Behavior Research Methods and Instrumentation, 1981. 13(5): p. 667-673.
13. Fitzmaurice, G., H. Ishii, W. Buxton. *Bricks: Laying the foundations for graspable user interfaces*. in *CHI'95*. 1995: ACM p. 442-449.
14. Fitzmaurice, G., S. Zhai, M. Chignell, *Virtual reality for palmtop computers.* ACM Transactions on Information Systems, 1993. 11(3): p. 197-218.
15. Fitzmaurice, G.W., *Situated information spaces and spatially aware palmtop computers.* Communication of the ACM, 1993. 36(7): p. 38-49.
16. Fukumoto, M., S. Toshiaki. *ActiveClick: Tactile Feedback for Touch Panels*. in *CHI'2001, Extended Abstracts*. 2001: ACM p. 121-122.
17. Gault, R., *Progress in experiments on tactual interpretation of oral speech.* Journal of Abnormal and Social Psychology, 1925(14): p. 155-159.
18. Geldard, F., *Adventures in tactile literacy.* American Psychologist, 1957. 12: p. 115-124.
19. Geldard, F., *Some neglected possibilities for communication.* Science, 1960. 131(3413): p. 1583-1588.
20. Gemperle, F., N. Ota, D. Siewiorek. *Design of the wearbale tactile display*. in *ISWC*. 2001: IEEE.
21. Gescheider, G., et al., *The effects of a surround on the vibrotactile thresholds.* Sensory Process, 1978. 2(2): p. 99-115.
22. Grob, B., *Basic electronics*. 8 ed. 1997, McGrow-Hill. 1010 pp.
23. Harrison, B., et al. *Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces*. in *CHI'98*. 1998: ACM p. 17-24.
24. Hinckley, K., R. Pausch, J. Gobble, N. Kassell. *Passive real-world interface props for neurosurgical visualization*. in *CHI'94*. 1994: ACM p. 452-458.
25. Hinckley, K., J. Pierce, M. Sinclair, E. Horvitz. *Sensing Techniques for Mobile Interaction*. in *UIST'2000*. 2000: ACM p. 91-100.
26. Hughes, R., A. Forrest. *Perceptualisation using a tactile mouse*. in *Visualization'96*. 1996: IEEE p. 181-188.
27. Ishii, H., B. Ullmer *Tangible bits towards seamless interfaces between people, bits and atoms*. in *CHI97*. 1997: ACM p. 234-241.
28. Kajimoto, H., N. Kawakami, T. Maeda, S. Tachi. *Electrocutaneous display as an interface to a virtual tactile world*. in *VR'2001*. 2001: IEEE p. 289-290.
29. Kristoffersen, S., F. Ljungberg. *"Making place" to make IT work: empirical explorations of HCI for mobile CSCW*. in *SIGGROUP'99*. 1999: ACM p. 276-285.
30. MacKenzie, S., A. Oniszczak. *A Comparison of three selection techniques for touchpads*. in *CHI'98*. 1998: ACM p. 336-343.
31. Miller, T., R. Zeleznik. *The design of 3D haptic widgets*. in *Symposium on Interactive 3D Graphics*. 1999: ACM p. 97-102.
32. Minsky, M., et al. *Feeling and seeing: issues in force display*. in *I3D*. 1990: ACM p. 235-241.
33. Pascoe, J., N. Ryan, D. Morse, *Using while moving: HCI issues in fieldwork environments.* Transactions on Computer-Human Interaction, 2000. 7(3): p. 417-437.
34. Pedersen, E., T. Sokoler. *AROMA: abstract representation of presence supporting mutual awareness*. in *CHI*. 1997: ACM p. 51-58.
35. Poupyrev, I., et al., *Developing a generic augmented-reality interface*, *IEEE Computer*. 2002. p. 44-49.
36. Rekimoto, J. *Tilting operations for small screen interfaces*. in *UIST'96*. 1996: ACM p. 167-168.
37. Sawhney, N., C. Schmandt. *Nomadic Radio: Scalable and contextual notification for wearable audio messaging*. in *CHI'99*. 1999: ACM p. 96-103.
38. Sherrick, C., *Vibrotactile pattern perception: some findings and applications*, in *The Psychology of Touch*, M. Heller and W. Schiff, Editors. 1991, Lawrence Erlbaum Associates. p. 189-217.
39. Tan, H., A. Pentland. *Tactual displays for wearable computing*. in *ISWC97*. 1997: IEEE p. 84-89.
40. Uchino, K., *Piezoelectric Actuators and Ultrasonic Motors*. 1996, Kluwer Academic Publishers. 349 pp.