
Dynamics of Tilt-based Browsing on Mobile Devices

Sung-Jung Cho

SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
sung-jung.cho@samsung.com

Roderick Murray-Smith

DCS, Glasgow University,
Glasgow G12 8QQ Scotland
& Hamilton Institute, NUIM
rod@dcs.gla.ac.uk

Changkyu Choi

SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
changkyu_choi@samsung.com

Younghoon Sung

SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
younghun.sung@samsung.com

Kwanghyeon Lee

SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
kennylee@samsung.com

Yeun-Bae Kim

SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
kimybae@samsung.com

Abstract

A tilt-controlled photo browsing method for small mobile devices is presented. The implementation uses continuous inputs from an accelerometer, and a multimodal (visual, audio and vibrotactile) display coupled with the states of this model. The model is based on a simple physical model, with its characteristics shaped to enhance usability. We show how the dynamics of the physical model can be shaped to make the handling qualities of the mobile device fit the browsing task. We implemented the proposed algorithm on Samsung MITs PDA with tri-axis accelerometer and a vibrotactile motor. The experiment used seven novice users browsing from 100 photos. We compare a tilt-based interaction method with a button-based browser and an iPod wheel. We discuss the usability performance and contrast this with subjective experience from the users. The iPod wheel has significantly poorer performance than button pushing or tilt interaction, despite its commercial popularity.

Keywords

Tilt dynamics, photo browsing, mobile interaction, motion-based interaction, accelerometer

ACM Classification Keywords

H.5.2 User Interfaces: Input devices and strategies

Copyright is held by the author/owner(s).

CHI 2007, April 28 – May 3, 2007, San Jose, USA

ACM 1-xxxxxx

Introduction

This paper develops the example of tilt-based photo browsing as a case-study in the use of pseudo-physical dynamic models as a design method for multimodal interaction. This is particularly topical given the large number of photos now stored on mobile devices with limited input capability and small screen size. Current list browsing methods are categorized by the type of input: button-based input, touch gesture based input, and tilt-based input. In the button-based one, users press soft buttons on the screen or physical buttons repeatedly to navigate. While an effective method for short paths, repeatedly pressing buttons is tedious for longer paths [1].

In touch-screen gestural input, the current view screen is moved as a user draws downward or upward gestures on the touch screen. The browsing speed and zoom level are affected by the gesture size or the screen scroll speed [2][3][4]. It is quite fast for browsing but difficult for single-handed use. An alternative is Apple's iPod click wheel [5] which enables users to scroll the list by rotating fingers over the touch area. It is fairly simple and supports fast browsing. However, it occupies a large area and requires both hands for fine control, and suffers from repeated overshooting.

In tilt-based input, the screen is scrolled proportional to the amount of tilt angle. It does not obscure the screen, or use buttons, and can support single-handed interaction. The problem of screen visibility when skewed is being gradually ameliorated by the introduction of wide viewing angle LCD [6] and OLED technology. Prior work on tilt-based screen panning method in mobile devices includes [7,8,9]. A tilt-based

document browser with a dynamic systems implementation of speed-dependent automatic zooming was also introduced in [10].

Advantages of tilt-based input are as follows. Users understand the interaction with tilt-based input easily because of the metaphor of realistically responding physical objects. Also, they have more fun in browsing photos with the embedded interaction. [11] proposed a realistic ball-in-bowl demo system which adopts a metaphor of physical model like a rolling ball in bowls. Also the simulated models can be linked to intelligence in the device, such that the properties of the dynamics of the interface, and the sounds and vibrations perceived by the user can be made a function of the content as shown in a tilt-based text-entry system [12], and in a multilingual text-browsing application [13].

Shaping the dynamics of tilt browsing

This paper investigates the dynamics of a tilt-controlled photo browser. The dynamics are shaped to simulate the behavior of sticky film strips as a metaphor, but are modified from a purely physical model to enhance usability.

The usability-focused modifications of the dynamics include the creation of 'attractors' around the photos, making it easier to settle on them. The attractors are sensitive to the speed of navigation and recent changes in input behavior, in order to be able to be agile and responsive when making fine adjustments, but easy to control when moving rapidly. They reduce *overshooting* problems, and damping down any minor oscillation around the target due to hand tremor. The use of such attractors also forces the image to settle on one photo,

rather than *partial presentation* problems, where it comes to rest between photos.

There is a *position-dependent speed control* term which reduces speed more significantly close to the centre, and damps movement beyond the center, making it easier to stop at the target. This is combined with a *position-dependent attractor velocity* term which accelerates the cursor towards the image center, from either side. A further benefit of the dynamic simulation approach is that the perceived texture of the surface can be generated as a function of image content.

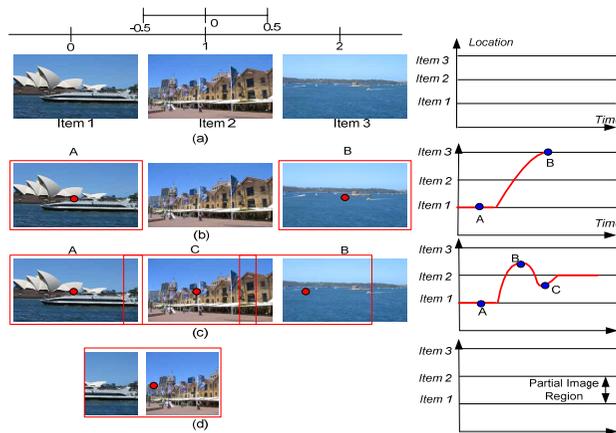


Fig. 1 Possible photo browsing behaviour (the red rectangle denotes a cursor) (a) image coordinate (b) overshooting problem (targetting centre photo, but lands on right photo) (c) oscillating problem (d) partial image problem – settling between photos.

Overview of the proposed system design

Our cursor dynamics is composed of tilt-to-velocity conversion, position-dependent speed control, position-dependent attractor velocity, dynamics parameter

update, and screen coordinate conversion, as shown in Fig. 2. We choose a nonlinear monotonous mapping from the tilt angle to the velocity for reducing the consequences of hand tremor. For the tilt θ , the velocity is given by $v(t) = \text{sign}(\theta(t)) \cdot \theta(t)^2$. The dynamic system updates the cursor parameter of velocity and position from the tilt angle and the position. Let $x(t), v(t)$ denote the position and velocity at time t . The velocity is the sum of velocities from the tilt angle and position:

$$v(t) = v_{ilt}(\theta(t)) + v_{attractor}(x(t)) \quad (1)$$

Eq. (1) is filtered by the speed v_m at $x(t)$ as follows:

$$v_f(t) = \text{sign}(v(t)) \cdot \min(v_m(x(t)), |v(t)|) \quad (2)$$

For a simulation interval Δs , $x(t+1)$ is as follows:

$$x(t+1) = x(t) + \Delta \cdot v_f(t) \quad (3)$$

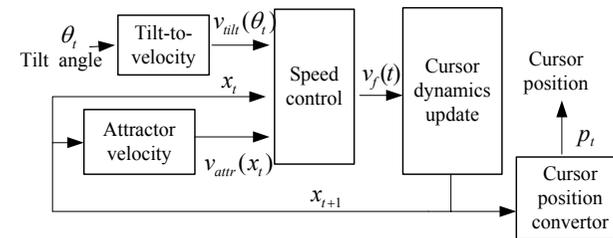


Fig. 2 Overview of the proposed tilt dynamics

Conversion of parameters to screen coordinates

The cursor position on the screen, $p(t)$, is converted from the dynamics position $x(t)$ according to the stable and transient regions for preventing the partial images as much as possible. Let $\lceil x(t) \rceil$ be the image center



Fig. 3 Prototype system with sensors and a tactile actuator

nearest to $x(t)$ and δ the predetermined distance. Then, in the stable region, the cursor is fixed at the center and in the transit region, the cursor position is updated according to $x(t)$ in real time as follows:

$$\begin{cases} p(t) = x(t) & \text{if } |x(t) - \lceil x(t) \rceil > \delta \\ p(t) = \lceil x(t) \rceil & \text{otherwise} \end{cases} \quad (4)$$

Usability experiment design

A user browses to the next or previous neighboring image by tilting to the right or left. It allows users to browse photos in a discrete or continuous manner; users can change images one-by-one, by tilting the device to the left or right and then returning it to the rest position. They can also browse photos continuously by holding the device tilted. Users sense multi-modal feedback, with sound and vibration augmenting the display. Currently, the vibration and sound are generated with each new image. The framework also permits sound and vibration to be an arbitrary function of the content of the image via adaptive dynamics. E.g. the behavior could differ for landscapes or human faces.

Hardware and software

The prototype system is implemented on a Samsung PDA (MITs 4300) by attaching a sensor board to the battery pack via a serial port (Fig. 3). The sensor pack has Kionix tri-axis accelerometer and three mono-axis gyroscopes which generate acceleration and angular velocity signal at 50 Hz. For browsing photos, we use the pitch angle (vertical axis in Fig. 3), inferred from the 3DOF acceleration sensors. It has a VBW 32 vibrotactile motor for rendering vibrotactile feedback. We used the Piccolo graphic library to render a list of

images [14], using C# and with smooth zooming in/out. The screen is updated at 13 Hz in our system.

Experimental Results

We tested the usability with 7 novice users of age 20-30 years from our company. None had experience with tilt-based input. We presented them 20 images from 100 sequentially and then asked them to find the images. We measured the task performance. For comparison with tilt methods, we used button-based browsing and the video iPod, which are among the most typical techniques. The users have time to practice all the input methods for a couple of minutes. The users' activity history (tilt angles, cursor position, button press time) are recorded to a log file (except the iPod, where its activities are recorded by a video camera and tagged manually).

Measure of usability criteria

We now compare different systems configurations for browsing photos via tilt-input. An appropriate objective metric which corresponds well to subjective perception of ease of control is required. For comparison of the performance, we employed five criteria: the number of overshooting cases (OS), the number of browsed images to the target (Dist.), the total browsing time to the target (TBT), the transition time before arriving at the target (TT) and the stabilization time after arriving at the target (ST). As an interaction method becomes more efficient, all the measures have lower values. Figure 4 shows one example of measures. The user moves from the position A and finally arrives at the position E. Then, OS is 3 (A→B, B→C, C→D). The total distance is the sum of d_1, \dots, d_4 . TT is the interval to

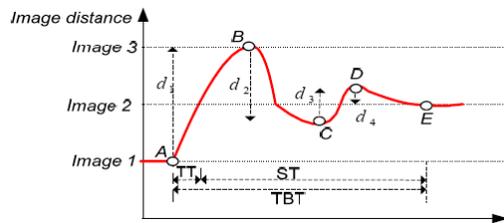
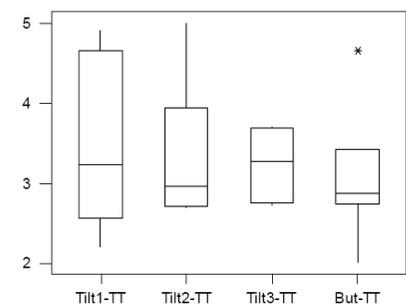
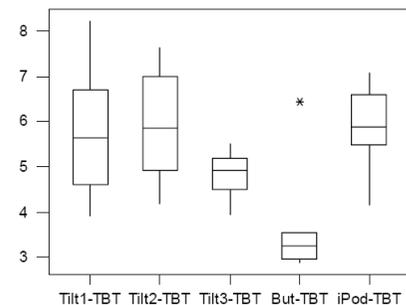
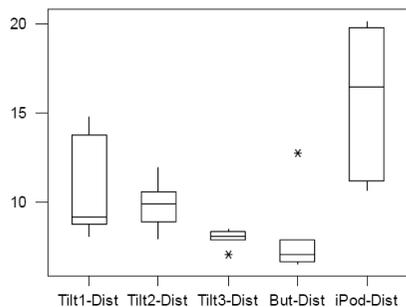


Fig. 4 Usability criteria



Boxplots of distance covered, Total Browse Time and Transient Time to first encounter with target

first arrival time at image 2. ST is the time for convergence after TT. TBT is the time from A to E, and the sum of TT and ST.

Table 1 Properties of interaction types for comparison

Input	Description
<i>Tilt 1</i>	Baseline tilt dynamics: $v_f(t) = v_{tilt}(t)$
<i>Tilt 2</i>	<i>Tilt 1</i> + Speed control capability
<i>Tilt 3 (Proposed)</i>	<i>Tilt 2</i> + Attractor velocity
<i>Button</i>	Move photos by pressing buttons
<i>iPod</i>	Move photos by rotating a click wheel

Table 2 Qualitative experimental results (mean and stdev)

Method	OS (#)	Dist. (#img)	TBT (sec)	TT (sec)	ST (sec)
<i>Tilt 1</i>	0.9(0.2)	10.6(2.6)	5.8(1.5)	3.4(1.0)	2.3(0.7)
<i>Tilt 2</i>	1.2(0.4)	9.8(1.3)	5.9(1.3)	3.3(0.9)	2.6(0.7)
<i>Tilt 3</i>	0.6(0.2)	8.0(0.5)	4.8(0.5)	3.2(0.4)	1.5(0.4)
<i>Button</i>	0.3(0.3)	7.9(2.2)	3.6(1.3)	3.1(0.8)	0.5(0.6)
<i>Ipod</i>	1.0(0.2)	15.6(4.4)	5.9(0.9)	N/A	N/A

Result analysis

Table 2 shows the usability performance of the five input methods, and in the left margin we show Boxplots comparing the different conditions for the metrics Dist (number of photos covered), TBT (total browse time) and TT (transient time to first encounter with target photo). The results are averaged over all seven users and all twenty image search tests. We have no results for the iPod for TT and ST because we could not afford to manually tag the activities by analyzing all the videos.

There is a general trend to improved performance from Tilt 1-3. The use of attractor dynamics in (*Tilt 3*) is very effective compared to *Tilt 1* and *Tilt 2*. The overshooting was reduced by about 30%, the distance by 25% (and much less variability), and the total browsing time by 17% (with much less variability). The most effective one is the button-based input. It has the minimum amount of overshooting, travel distance and total browsing time. The iPod does not give a strong result with any of the metrics, performing worse than *Button* or *Tilt 3* conditions. Especially, it leads to a large travel distance when the overshoot problem occurs. It is worth noting that all the input methods (*Tilt 1, 2, 3, Button*) have almost same transient time but different stabilization time. It suggests that stabilization is significant for usability. To analyze the qualitative results, we asked users the following questions. The results are shown in Table 3.

- How easy was it to control photos? [1-5]
- How interesting was the system? [1-5]
- How much did you like the system? [1-5]
- Please describe advantages and disadvantages.

The tilt has a low score in controllability but the highest score for interestingness. The button is effective for controlling photos but not particularly interesting to users. The iPod has low score in controllability and medium one for interestingness. Overall, the tilt-based input method is comparable to the button in preference. We expect that the controllability of tilting can be further enhanced because users only had two minutes of familiarization time. Tilt-based input is convenient for browsing a large number of photos as users can maintain a constant tilt. However, it is somewhat cumbersome to reverse the tilt to halt. Users found the 'sticky film strip' nature of the continuously moving

photos appealing. The button-based input is convenient for increments in position, but it is very tedious to push buttons serially for browsing a large number of photos. The iPod is very fast in browsing long distances. However, it has an overshooting problem for browsing short distances. We found two groups of users. One focuses on controllability and the other on novelty or appeal, where the browsing task is primarily for fun and the experience is more than just the speed of use.

Table 3 Qualitative result (mean and std. dev)

Method	Controllable	Interesting	Preference
<i>Tilt 3</i>	1.3(1.0)	4.1 (1.1)	3.4 (1.3)
<i>Button</i>	3.6 (0.8)	1.6 (0.5)	3.1 (0.7)
<i>ipod</i>	1.3 (1.3)	3.1 (0.9)	2.7 (1.5)

Conclusion

As mobile devices store more data such as photos and documents, the efficiency and appeal of list browsing is of growing importance. Tilt-based browsing is promising because it does not use space, and is often appealing to use. We shape the dynamics of tilt-based interaction and investigate the consequences for usability. The result shows that the proposed tilt dynamics reduced overshooting by about 30%, the distance by 25%, and the total browsing time by 17% compared to the baseline tilt dynamics. Its comparison with the button and iPod shows that the proposed tilting method is comparable to the controllability of buttons and more interesting than button and iPod, and performed better than the iPod, raising interesting questions about the effect of ubiquity and market image on user's perceptions of usability. Participants

commented that they were actually surprised at the poor relative performance of the touch wheel.

References

- [1] Q.Y. Wang, et. al, *Visual interface and control modality: an experiment about fast photo browsing on mobile devices*, Interact, 2005
- [2] D. Patel, G. Marsen, S. Jones, M. Jones, *An evaluation of techniques for browsing photograph collections on small displays*, Mobile HCI 2004
- [3] D. Patel, G. Marsen, M. Jones, S. Jones, *Improving Photo Searching Interfaces for Small-Screen Mobile Computers*, MobileHCI'06, Helsinki, p149-168, 2006
- [4] A. Cockburn, J. Savage, *Comparing Speed-dependent automatic zooming with traditional scroll, pan, and zoom method*, British HCI, 2003
- [5] Apple, *iPod (click wheel) manuals*, <http://docs.info.apple.com/article.html?artnum=32514>
- [6] Wikipedia, "TFT LCD," http://en.wikipedia.org/wiki/TFT_LCD, 2006
- [7] J. Rekimoto, *Tilting Operations for Small Screen Interfaces*, UIST, 1996, pp. 167-168.
- [8] B. Harrison et. al, *Squeeze Me, Hold Me, Tilt Me! An Exploration of Manipulative User Interfaces*, CHI 1998
- [9] K. Hinckley, J. Pierce, M. Sinclair and E. Horvitz, *Sensing techniques for mobile interaction*, ACM UIST 2000, p91-100
- [10] P. Eslambolchilar, R. Murray-Smith, *Tilt-based Automatic Zooming and Scaling in Mobile Devices - a state-space implementation*, MobileHCI 2004
- [11] J. Kelly, R. Murray-Smith, *Video of tilt-menu*, 2004. www.dcs.gla.ac.uk/~rod/Videos/JamesMenu2.avi
- [12] J. Williamson, R. Murray-Smith, *Dynamics and probabilistic text entry*, Hamilton Summer School on Switching and Learning in Feedback systems, Springer-Verlag, LNCS, Vol. 3355, p333-342, 2005
- [13] P. Eslambolchilar, J. Williamson, R. Murray-Smith, *Multimodal Feedback for tilt controlled Speed Dependent Automatic Zooming*, UIST 2004.
- [14] B. Bederson et. al, *Toolkit Design for Interactive Structured Graphics*, IEEE Trans. On Software Engineering, 30 (8) pp 535-546, 2004