Multi-Context Photo Browsing on Mobile Devices Based on Tilt Dynamics

Sung-Jung Cho
SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
+82-16-451-2147
sung-jung.cho@samsung.com

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

Yeun-Bae Kim
SAIT (Samsung Research)
P.O. Box 111, Suwon, Korea
+82-31-280-8227
kimybae@samsung.com

ABSTRACT
This paper presents a photo browsing system on mobile devices to browse and search photos efficiently by tilting action. It employs tilt dynamics and multi-scale photo screen layout for enhancing the browsing and the search capability respectively. 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 controllability. The multi-scale layout holds both local and global view for users to both control photos and look at the surrounding context in a single framework. We show how 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 seven users. The proposed tilt dynamics improves the usability over conventional dynamics. The iPod wheel has mixed performance comparing worse on some metrics than button pushing or tilt interaction, despite its commercial popularity.

Categories and Subject Descriptors
H.5.2 User Interfaces: Input devices and strategies

General Terms
Human Factors, Algorithms

Keywords
Tilt dynamics, photo browsing, mobile interaction, motion-based interaction, accelerometer, multi-scale view, speed-dependent automatic zooming

1. INTRODUCTION
This paper develops the example of tilt-based photo browsing as a case-study in the use of pseudo-physical dynamic models and a multi-context presentation model for enhancing the perception of list structure by users. The two kinds of models enable users to browse and search photos more fluently and efficiently by tilting actions. This is particularly topical given the large number of photos now stored on mobile devices with limited input capability and small screen size.

By the type of input, current list browsing methods are categorized into 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 and tiring for longer paths [1] given the fact that mobile keypads are not as comfortable to press as those of PCs because of size limitation.

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 [2,3] or the screen scroll speed [4-7]. It is quite fast in browsing and remedies the motion blur at fast scrolling speed by speed-dependent automatic zooming (SDAZ) capability. However it is difficult to use in a single-handed manner. An alternative is Apple's iPod click wheel [8] 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 unreadable screen when skewed has been gradually remedied these days by the introduction of wide viewing angle LCD [9] and OLED technology. A lot of researches have proposed a tilt-based screen panning method in mobile devices [10-12]. [13] used a dynamic systems implementation of speed-dependent automatic zooming in a tilt-based document browser.

Another important advantage of the tilt-based input is that it supports the metaphor of realistically responding physical objects and users have more fun by the pseudo-tentative interaction. [14] 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 [15], and in a multilingual text-browsing application in [16].

By coupling scrolling and zooming, the list browsing methods are also categorized into scrolling with fixed zoom, scrolling with variable zoom and scrolling at separate zoom levels. The first is conventionally used and items on a list are presented sequentially. It is efficient to browse items when the list size is small but not when large. For a large list, the method of automatically presenting pages or images at short time interval is proposed (RSVP: rapid serial visual processing [17]). However, it is not an easy task to determine the proper presentation rate according to intentions of users. The second browsing method usually relates zoom levels with scrolling speed like SDAZ [4-7, 13]. [18] devised an efficient and smooth zooming and panning method for transforming view positions. When the scrolling speed is high, the zoom level becomes wider and users are less bothered by visual blur from the fast moving screen. They retain an overview of the broader context in fast scrolling mode without necessity to adjust zooms manually. However, it has the problems that eyes become tired of continuously scrolling screens and users are likely to pass over target items because of the time delay to zoom in. In the last browsing method, users first select the zoom level and then scroll photos at the level. Users can figure out the global context and local context at the different zoom levels, but they should switch zoom levels manually and can watch only one of them at a time. In [19], the thumbnail view and the full page view of documents are switched by mouse click. The view modes are also changeable by the scrolling speed; at slow speed, documents are scrolled and zoomed by SDAZ, and at high speed, whole pages are shown during 50ms as RSVP [20].

In this paper, we propose a tilt-based photo browsing system with enhanced controllability from a tilt dynamics model [21] and a multi-context screen dynamics model. The dynamics model is designed for remedying major controllability errors in browsing photos. The screen dynamics is designed for presenting the local context of large size photos and the global context of neighboring thumbnails at the same time, which enhances the perception of users on the photo list structure.

The paper is organized as follows. Section 2 presents the overview of a tilt-based photo browsing system and its controllability problems. Section 3 presents a tilt-based dynamics model for solving the controllability problems. Section 4 presents a multi-context view model for enabling users to figure out the structure of photo list. Section 5 illustrates the implementation issues of the proposed system. Section 6 describes the experimental setup and the result of the proposed system. Section 7 concludes the paper.

2. Tilt-based Photo Browsing System

2.1 Conventional Tilt-based Browsing UI

A basic photo browsing method by tilting is to scroll photos vertically or horizontally as users tilt mobile devices vertically or horizontally [11]. The amount of scrolling is proportional to that of tilt angle. Fig. 1 shows its example. Photos are arranged horizontally from the left to the right. The red rectangle denotes the position of the device view screen (referred as a screen cursor or a cursor in this paper). When a user tilts a device leftward, the cursor moves leftward or rightward as if it is a heavy object or a light bubble under water respectively.

Fig. 1. Example of a tilt-based browsing UI (The red rectangle denotes the current view screen, referred as a cursor)

In tilt-based photo browsing, the conversion of tilt angle to a cursor position, a cursor dynamics, is very important for users to control photos. The tilting action is inherently an indirect pointing method so that users can not point or select photos directly like fingers on touch screens but can only move cursors relatively to the current position. Therefore, the cursor movement should be controlled sophisticatedly to reflect the intention of users naturally and effectively. The control of zoom contexts is also very important for users to learn the photo structure and predict the location of target photos during browsing and searching tasks.

2.2 Controllability Problems of Tilting

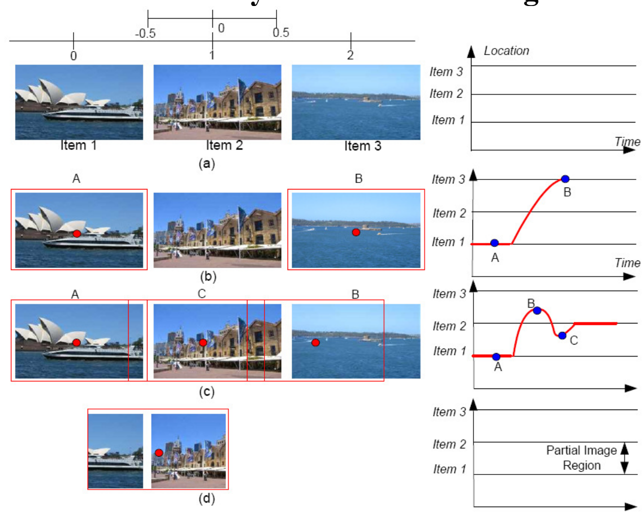

Fig. 2 Analysis of a photo browsing behavior (a) image coordinate (b) overshooting problem (targeting the center photo, but landing on the right one) (c) oscillating problem (d) partial photo problem – settling between photos.

Even though the tilt action is convenient for continuously browsing photos by holding the device, and enables realistic sensation of controlling photos as if they are physical objects, it has following three problems to locate a cursor on a target photo on the list (Fig. 2).

First, it suffers from an overshooting problem; users do not stop at the target photo but move beyond it. The problem occurs when they tilt the device too much and for too long. It requires them to tilt back to return to the target.

Second, it suffers from an oscillation problem of a cursor around the target photo without converging to it directly. It occurs when a user suffers from hand tremor or he can not give the exact tilt angle to converge to the target.

Finally, it has a partial photo presentation problem. Users usually want to watch full-sized photos on the screen. There are few cases where users want to watch different portions of photos on the screen at the same time. The partial photos are a kind of side effect during moving a cursor.

2.3 Controllability Problem of View Screen

The control of screen zoom levels is important to search and browse photos (Fig. 3). The small-size zoom (a global view) is useful for searching target photos on a list. Users can watch as many photos as possible on a screen at a time and predict where the target is located and when it can be reached by the tilting action. On the other hand, the large-size zoom (a local view) makes users to see only small number of next coming photos so that they may pass over fast approaching target photos without acting within proper time. However, it is better suited for a browsing task. Users want to watch large-size photos to recall experiences and people in photos more vividly and pleasantly.

To support the searching and the browsing task in a single framework, the local and the global view need to be supported seamlessly. One well-known approach is SDAZ which supports variable zooms depending on the scrolling speed. It has one zoom level at a time so that the risk of overshooting increases at the fast scrolling speed because the dwell time per photo decreases rapidly. The continuously refreshing screen also requires continuous visual attention from users.

Fig. 3 Controllability problem by zoom levels (a) Local view: delay in figuring out next photos (b) Global view: increase in overshooting because of the short dwell time per photo

2.4 Overview of the Proposed System

This paper investigates the tilt and the screen dynamics of a tilt-controlled photo browser. The tilt 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 include the use 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 damp down any minor oscillation around the target due to hand tremor. The use of such attractors also forces the photo to settle on the center of a screen rather than off its center, which reduces the case that the screen shows portions of two successive photos.

The screen dynamics are shaped to present the local and the global context at the same time. In browsing photos, users want to view photos as large as possible with enough time to watch them in detail. For the purpose, the local context view presents full-size photos on the screen. On the other hand, the global context view presents small size photos like thumbnails to make users scan as many of them as possible in searching targets.

The two context views are mediated by the tilt angle. A small tilt angle suggests that users watch photos cautiously with much time. Therefore, a local view (LV) dominates on the screen. Full-sized photos are scrolled for generating an agile and responsive screen. A large tilt angle suggests that users skip photos to find target photos quickly. Therefore a global view (GV) dominates on the screen and thumbnail photos are presented. To help users to feel less disorientation by the fast scrolling screen, thumbnail photos in GV are fixed and only a cursor moves on the screen. It is similar to the approach of presenting small thumbnail document images for helping users to figure out the overall document structure [19].

Fig. 4 illustrates a control sequence of LV and GV. At a small tilt, LV takes the whole screen and its photos are scrolled sequentially (Fig. 4 (a)). At a large one, LV is zoomed out and its area becomes reduced (Fig. 4 (b)). GV then occupies the area freed by LV and presents thumbnails (Fig. 4 (c)). There is a cursor in GV which represents the current photo position on LV. On the contrary of LV, the photos in GV stay stationary and only the GV cursor moves. If it goes out of the thumbnail list, the next photos are fed to GV.

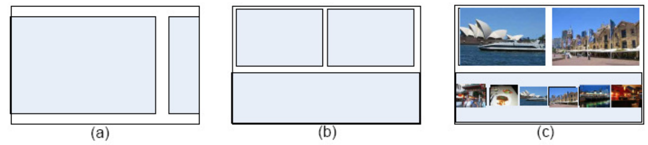

Fig. 4 Control of local and global views (a) a local view (b) the area zoomed-out by LV (c) filling of photos in GV

To support the two views seamlessly with tilting activities, the photo browsing system is composed of six components as shown in Fig. 5. The tri-axis accelerometer detects the amount of acceleration in X, Y, and Z axis. The detected acceleration is converted into the tilt angle (roll and pitch). The cursor dynamics calculates the position and speed of the LV cursor based on the tilt angle and the previous cursor position. From the tilt angle, the zoom level of LV is determined. LV area is reduced to fit photos by the zoom level. The screen area freed by LV is taken up by GV and thumbnails are shown on the GV screen. The multimodal output control part generates sound and vibrotactile feedback according to the tilt cursor position and the category of photos for enhancing the interest and presence of the interaction to users.

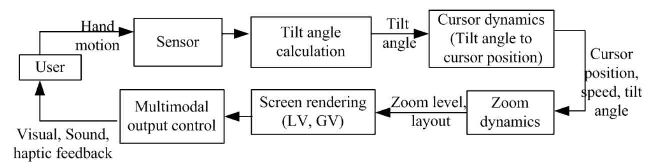

Fig. 5 Components of the proposed photo browsing system

3. Tilt-based Dynamics Model

The cursor dynamics can be designed effectively by utilizing the domain knowledge of photo browsing tasks. The cursor is likely to be located on the center of photos rather than in arbitrary positions. Also, it is likely to move sequentially on the photos.

By reflecting the domain knowledge, we remedy the three controllability problems of Section 2.2 with a cursor dynamics which gives more control time to users near the photo center but hold the average browsing time not increased. Also, the proposed dynamics enables users to perceive the texture of the surface of photos by generating multimodal outputs from a function of the photo content and the dynamics.

- Reduction of overshooting

We limit the cursor speed based on the photo position. It is reduced more significantly near the center, and movement beyond the center is damped. It is similar to pressing foot-brake for reducing the speed of a running car.

- Reduction of oscillation

We introduce an attractor velocity term which drives a cursor to converge to the photo center. It accelerates the cursor when it moves towards the photo center and decelerates when it moves out of the photo center.

- Reduction of partial photo presentation

We divide a photo area into a stable region near its center position and a transition one near its boundary. In the stable region, the photo is fixed on the screen. In the transition one, partial photos are presented in real-time.

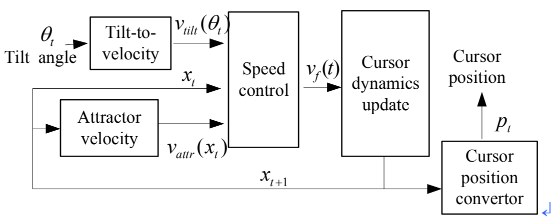

Fig. 6. Tilt dynamics for converting a tilt angle to a cursor position

Fig. 6 shows the cursor dynamics which remedies the three problems. The current cursor position depends on the previous cursor position and the current tilt angle. It has tilt-to-velocity conversion, photo position-based speed control, cursor dynamics update and cursor position conversion parts. The features and algorithms of components are as follows:

- Tilt-to-velocity conversion

To control the cursor movement by tilt, the tilt angle should be converted into dynamics parameters. The cursor speed should increase as the tilt angle becomes larger. For two tilt angles $\theta_A \le \theta_B$, the cursor speed should satisfy $v(\theta_A) \le v(\theta_B)$ (monotonous property). Also, jitter induced by muscle tremor in the hand should be minimized. Therefore, we propose following nonlinear conversion function:

$$v_{tilt}(\theta(t)) = sign(\theta(t)) \cdot \theta(t)^2 \quad (1)$$

- Photo position-based attractor velocity

To drive the cursor converged to the photo center, an attractor velocity (AV) toward the photo center is added to the cursor speed. The absolute value of AV becomes increased when the cursor moves out of the center and vice versa. Its direction is same as that of the cursor when the cursor moves toward the center and opposite when out of the center. Fig. 7 shows its one example. X-axis corresponds to photo coordinates and its integer values denote centers positions. At the center, AV has zero value. Its sign is opposite to the direction toward the center in other areas.

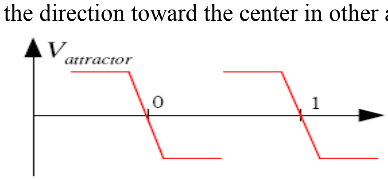

Fig. 7 Position-based attractor velocity

- Photo position-based speed control

To give users enough time to control the system, the movement speed is decreased as the cursor goes beyond the center. There is less speed restriction when it approaches to the center, for reducing the approaching time as shown in Fig. 8.

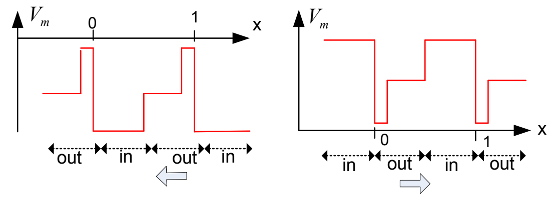

Fig. 8. Position-based speed control (left: a speed control graph of a cursor moving leftward, right: moving rightward)

- Cursor dynamics update

The position and speed $x(t), v(t)$ of a cursor at each time are updated from the previous cursor position $x(t-1)$ and the tilt-incurred velocity $v_{tilt}(\theta(t))$ at the tilt angle $\theta(t)$ as follows:

$$v(t) = v_{tilt}(\theta(t)) + v_{attractor}(x(t)) \quad (2)$$

The velocity $v(t)$ is further limited by the position-dependent speed control filter. The final speed v_f with the maximum allowable speed v_m at the position $x(t)$ is given as follows:

$$v_f(t) = sign(v(t)) \cdot \min(v_m(x(t)), |v(t)|) \quad (3)$$

For a simulation time interval Δ, the next time position $x(t+1)$ is updated as follows:.

$$x(t+1) = x(t) + \Delta \cdot v_f(t) \quad (4)$$

- Cursor position converter

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 partial photos as much as possible. Let $\lceil x(t) \rceil$ be the photo center 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 one, its position is updated according to $x(t)$ in real time as follows:

$$\begin{cases} p(t) = x(t) & \text{if } |x(t) - \lceil x(t) \rceil| > \delta \\ p(t) = \lceil x(t) \rceil & \text{otherwise} \end{cases} \quad (5)$$

4. Multi-Context View Model

To help users browse photos in slow and fast scrolling speeds in a single framework, we compose a display screen with a local view (LV) and a global view (GV). The former presents photos in large scale so that users can watch photos in detail. The latter presents thumbnail photos so that users can search targets among many photos, reducing the need to scroll them frequently.

LV does not have an explicit cursor mark on the screen but the screen itself becomes a cursor (Fig. 9 (a)). Photos in whole screen sizes are scrolled leftward or rightward according to tilting actions of users, which emphasizes the sensation of realistic manipulation of photos. The amount of scrolling distance becomes larger when users tilt at large angle.

To minimize the screen blur by fast moving photos in LV, the zoom level is determined by the tilting speed for making pixels scrolled in an almost constant amount (similar to SDAZ [4-7]). For the tilt angle $\theta(t)$, the zoom level $Z(t)$ is determined as follows:

$$Z(t) = f(\theta(t)) \tag{6}$$

The function $f(.)$ of Eq. (6) has the constraint that it is inversely proportional to $\theta(t)$. We use the tilt angle rather than the cursor speed in controlling zoom levels because the cursor speed is nonlinearly proportional to a tilt angle so that users have difficulty in controlling a zoom level. For $Z(t) < 1$, the width and height of photos in LV are shrank respectively by $Z(t)$. The area unoccupied horizontally is taken up by neighbor photos sequentially, and the vertical area is taken by GV.

Fig. 9. Movement of a cursor (a) Local view: a cursor, the screen itself, moves leftward or rightward (b) Global view: a cursor mark (the red rectangle) moves over photos. The photos are scrolled only when the cursor moves out of them.

In GV, thumbnails are presented on the screen with a cursor mark (Fig. 9 (b)). The cursor denotes the center position of LV on the photo list and synchronizes its position with that of the LV cursor. It is different from the LV cursor in that only the cursor mark rather than photos is scrolled. The set of photos scrolls only when the cursor moves out of them. Thumbnails are grouped and updated per pages in GV, which aims at helping users to memorize the photo list structure more easily by presenting consistent groups of photos.

The zoom level does not affect sizes of photos but their numbers in GV. The GV thumbnails are already small so that they are not easy to identify when further zoomed out. If GV area is expanded with a height H at a zoom value and its thumbnails are of $1/T$ from their original size, GV has rows of $T \cdot H$.

The areas of LV and GV are mediated by the tilting angle as shown in Fig. 10. Both of them have same width and their photos are scrolled horizontally. However, their heights compete each other. When the tilting angle is small (Fig. 10(a)), only LV dominates the screen. Users can watch large photos on the whole screen. When medium (Fig. 10(b)), the LV area is shrunk as the zoom level becomes smaller. The horizontal space reduced by the zoom-out is filled up with neighbor photos sequentially. The vertical space freed from LV is taken up by the growing GV area. Thumbnail photos are presented on GV. When it is high (Fig. 10 (c)), the GV area grows further and presents more thumbnails.

Fig. 10. Speed-dependent context view configuration (a) LV dominates when the tilting angle is small (b)GV grows when medium (c) GV dominates when large

Fig. 11 illustrates the overall screen dynamics of LV and GV. The zoom level is inversely proportional to the tilt angle (Fig. 11 (a)). The SDAZ feature makes the zoom level reduced at the large tilt angle, which in turn reduces the LV size. The height of LV then becomes linearly proportional to the zoom level (Fig. 11 (b)). The height of GV is negatively proportional to the LV size (Fig. 11 (c)). By the screen dynamics, users can watch more global context at high speed and more local context at low speed.

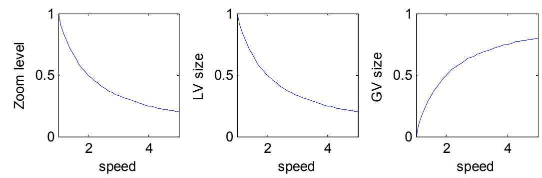

Fig. 11. Speed-dependent (a) zoom level (b) LV size (c) GV size (speed = tilt angle)

The advantages of the proposed screen dynamics are as follows. First, users can control photos by tilting in LV and watch surrounding neighbor photos in GV at the same time in a single framework. Second, there is no need to explicitly change tasks of browsing and searching photos. They are automatically determined by the tilt cursor speed. Third, the time to settle on target photos is reduced because users can predict when they will reach them among the incoming photos in GV without being annoyed by the fast scrolling screen.

5. System implementation

5.1 Prototype System

The prototype system is implemented in a Samsung PDA (MITs 4300, 11.5cm * 5.8cm *2.4cm) by attaching a sensor board to the battery pack via a serial port (Fig. 12). The sensor pack has Kionix tri-axis accelerometer and three mono-axis gyroscopes which generate acceleration and angular velocity signal at 50 Hz.

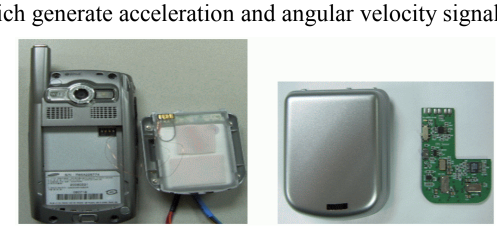

Fig. 12 HW system with sensors and a tactile actuator

Among 3-DOF (X, Y, Z) in tilting the device, we use a pitch angle (vertical direction in Fig. 12) to control photos. A user holds the device horizontally to make photos shown in landscape. Users browse photos one by one by tilting rightward or leftward and then returning it to the rest position. They also browse them continuously by holding the device tilted at the angle larger than a predetermined threshold. For tri-axis acceleration A_x, A_y, A_z, the pitch angle is calculated as follows:

$$Pitch = \tan^{-1}\left(\frac{A_z}{\sqrt{A_x^2 + A_y^2}}\right) \tag{7}$$

It has a VBW 32 vibrotactile motor for rendering vibrotactile feedback. The screen is updated at 13 Hz in our system.

5.2 Multi-Modal Feedback

The proposed framework makes it easy to adjust the dynamics of movement, and present vibrotactile feedback to the user according to a function of photo content. For instance, the vibration can be generated according to the photo type like scenery, human faces, indoors/outdoors etc. Parameters of sound and tactile patterns such as frequency, volume and duration can be changed according to the photo contents, augmenting the visual display, and supporting intermittent interaction, where the user does not need to devote his entire visual attention to the screen during the interaction. Currently, the vibration and sound are generated as discrete feedback with each new photo, but functions of content have not been implemented yet. However, richer content-dependent continuous multimodal feedback is an interesting research challenge.

6. Experiment and Result

6.1 Experimental Setup

6.1.1 Evaluation of the tilt-based dynamics model

The usability of tilt-based dynamics model is tested with seven novice users of age 20-30 years from our company. We expect that the proposed system will be used mainly by the young generation people because they are very acceptable to new devices and technologies. None had experience with tilt-based input. We presented them 20 photos from 100 sequentially and then asked them to find the photos. Among them, 9 photos have short movement distance (less than 3) from a starting photo, and the other have long movement distance (10-12). The proposed tilt method is compared with the conventional and the modified tilting methods. For comparison with tilt methods, we used button-based browsing and the video iPod, which are among the most typical techniques. Table 1 summarizes the input methods for comparison in photo browsing tasks. To make the comparison of input methods fair and independent from other effects, we fixed the zoom level of the screen as one. We did not use the multi-context screen dynamics in this comparison experiment task.

Table 1 Input methods for comparison in photo browsing tasks

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

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 at 50 Hz (except the iPod, where its activities are recorded by a video camera and tagged manually because we do not afford to change the iPod SW). On average it took about five minutes for a user to complete the browsing task by each input method and about 30 minutes to finish all the five. After the usability experiment, we performed qualitative analysis by asking them to evaluate each method subjectively. Fig.13 shows pictures of testing each input method.

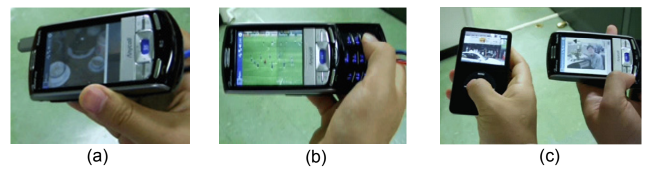

Fig. 13. Example of using input methods (a) *Tilt 1, 2, 3* (b) *button* (c) *iPod* wheel (The device to the right shows the target photo.)

6.1.2 Evaluation of the multi-context view model

The usability of multi-context view model is tested with four users of ages 20's and 30's who are different from the previous group. We choose *Tilt 3* as a baseline system and apply the multi-context view model to it (referred as *Tilt 3+MV*). The experimental condition is same as that of the tilt-based dynamics model.

6.2 Usability criteria

We now compare different configurations of systems 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. Fig. 14 shows one example of measures. The user moves from the position A and finally to the position E. In this case, OS is 3 (A→B, B→C, C→D). The total distance is the sum of $d_1, \cdots, d_4$. TT is the interval to the 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.

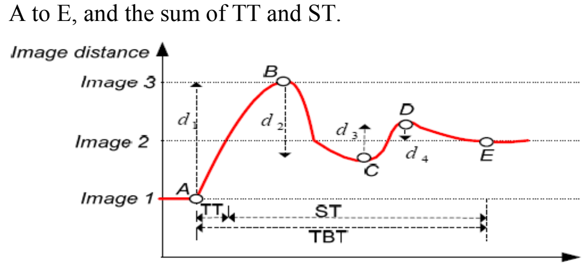

Fig. 14. Usability criteria

6.3 Experimental Results

Table 2 shows the usability performance of the five input methods and Fig. 15 shows *Boxplots* comparing the different conditions for the metrics Dist (number of photos covered), TBT (total browse time), TT (transient time to first encounter with target photo) and ST (stabilization time after arriving at the target). The results are averaged over all the seven users and all the 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.

Table 2 Qualitative experimental results (mean and stdev)

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

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* other conditions. Especially, it gives 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 the usability depends on the stabilization very strongly.

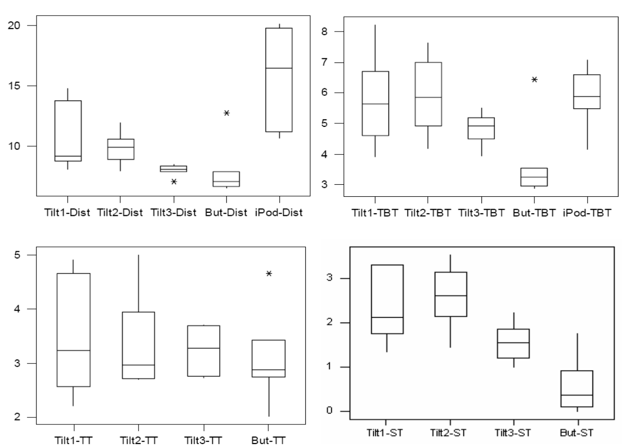

Fig. 15. *Boxplots* of experimental results according to input methods (Dist, TBT, ST and TT in clockwise order).

To analyze the qualitative results, we asked users the following questions. The results are shown in Table 3. A large number denotes the degree of positive responses (five Likert scale).

- 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.

Even though the number of subjects (seven) is rather small to draw statistically significant conclusions, the pilot results indicate following tendency; the tilt has a low score in controllability but the highest score in the interestingness. The button is convenient in controlling photos but not 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 has more room to be enhanced because users had only a couple of minutes of familiarization time.

Table 3 Qualitative result (mean and std. dev)

Method	Controllable	Interesting	Preference
Tilt 3	1.3(1.0)	**4.1 (1.1)**	3.4 (1.3)
Button	**3.6 (0.8)**	1.6 (0.5)	**3.1 (0.7)**
Ipod	1.3 (1.3)	3.1 (0.9)	2.7 (1.5)

Fig. 16 shows the effect of the multi-context view model to the proposed tilt-based dynamics (*Tilt 3*). We measure how it reduces the usability criteria relatively from *Tilt 3* (the measure of the multi-context view model over that of Tilt 3).It shows that the overshooting is greatly reduced by about 40%. It is because the multi-context view helps users to search target photos and estimate the moment to stop tilting. On the other, the other measures are reduced only slightly. We suspect that changing zoom levels and view area sizes disturb users' attention to some extent and incur delay in their control activity. Another reason might be the same tilt dynamics on both configurations. We may use much higher scrolling speed on GV screen for increasing browsing speed.

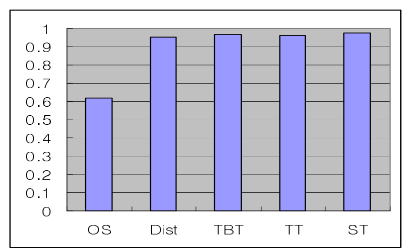

Fig. 16 Relative performance of *Tilt 3 with multi-View* over *Tilt 3* (OS, Dist, TBT, TT, ST)

Comments from users and our observations are summarized as follows. 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 tilt backward to halt. Users found the 'sticky film strip' nature of the continuously moving photos appealing. The button input is convenient for increments in position, but it is very tedious to push buttons serially for browsing many 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.

Regarding the multi-context view, users responded positively saying that the global view is very helpful in figuring out target photos in one visual scan and planning tilt activities to reach the targets. They also mentioned that the scrolling speed in the large GV area is somewhat slow and needs to be increased for skipping irrelevant photos rapidly. However, they mentioned the difficulty in giving attention to both LV and GV at the same time. They switched their attentions between both areas usually when starting scrolling and stopping tilting.

Regarding the multimodal feedback, users' comment was that multimodal feedback, especially tactile feedback, is very interesting and gives more realistic sensation to photos. Many of them felt that photos with tactile vibration feel heavier than those without it and liked to browse photos with the vibration feature. It is the research challenge to devise more objective experimental setup and criteria for measuring the contribution of multimodal feedback to the satisfaction and preference of users.

6.4 Discussion of Experiments

To make the experimental results more general and statistically significant, our future researches are to extend the experiment as follows. First, the subjects should be extended in numbers and samples. At least 30 subjects will be required for statistical significance. The subjects should be extended outside the company for fairness. People within the company might give positive answers to the survey questions in favor of the proposed system. Second, the proposed usability criteria (Section 6.2) should be revised to include qualitative features such as interestingness. Some people comment that they focus on novelty rather than speed. Buttons have the best performance in the proposed measure but not in the preference of the qualitative analysis. The criteria should resolve the discrepancy.

7. Conclusion and Future Work

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 emotionally appealing to use. We shape the dynamics of tilt-based interaction and design multi-scale photo screen layout. We then investigate their 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. Also, the proposed multi-context view reduced the overshooting by 40 % further.

The 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 even though the number of subjects seven is rather small for statistically significant results. It raises interesting questions about the effect of ubiquity and market image on user's perceptions of usability.

To draw more statistically significant result and analyze the photo browsing task in more detail, our future research will focus on the extension of subjects in volumes and samples. We need to extend the measure of usability to include the evaluation of subjective feelings like interestingness. The contribution of multimodal feedback to the usability should be analyzed further quantitatively with proper metrics.

8. References

[1] Q. Wang, S. Harada, T. Hsieh, A. Paepcke, M. Costabile, F. Paterno, "Visual interface and control modality: an experiment about fast photo browsing on mobile devices," *IFIP Interact*, p240-252 2005

[2] D. Patel, G. Marsden, S. Jones, M. Jones, "An evaluation of techniques for browsing photograph collections on small displays," *Mobile HCI*, Glasgow, UK, p132-143, 2004

[3] D. Patel, G. Marsden, M. Jones, S. Jones, "Improving Photo Searching Interfaces for Small-Screen Mobile Computers," *Mobile HCI*, Helsinki, Finland, p149-168, 2006

[4] A. Cockburn, J. Savage, "Comparing Speed-dependent automatic zooming with traditional scroll, pan, and zoom method," *British HCI*, Bath, England, p87-102, 2003

[5] A. Cockburn., J. Looser, and J. Savage, "Around the World in Seconds with Speed-Dependent Automatic Zooming" Demo in *ACM UIST, Vancouver, Canada*, p35-36, 2003

[6] A. Cockburn, J. Savage, A. Wallace, "Tuning and testing scrolling interfaces that automatically zoom," *ACM CHI* , Portland, USA, p71-80, 2005

[7] S. Jones, M. Jones, G. Marsden, D. Patel.,A. Cockburn, "An evaluation of integrated zooming and scrolling on small screens," *I. J. Human Computer Studies,* 271-303, 2005

[8] Apple, iPod (click wheel) manuals, http:// docs.info.apple.com/article.html?artnum=32514

[9] Wikipedia, "TFT LCD," http://en.wikipedia.org/wiki/TFT_LCD, 2006

[10] J. Rekimoto, "Tilting Operations for Small Screen Interfaces," *ACM UIST,* Seattle, USA, p167-168, 1996

[11] B. Harrison, Harrison, K. P. Fishkin, A. Gujar, C. Mochon, R. Want, "Squeeze Me, Hold Me, Tilt Me! An Exploration of Manipulative User Interfaces," *ACM SIG CHI*, LA, USA, p17-24, 1998

[12] K. Hinckley, J. Pierce, M. Sinclair , E. Horvitz, "Sensing techniques for mobile interaction," *ACM UIST*, San Diego, USA, p91-100, 2000

[13] P. Eslambolchilar, R. Murray-Smith, "Tilt-based Automatic Zooming and Scaling in Mobile Devices- a state-space implementation," *Mobile HCI*, Glasgow, p120-131, 2004

[14] J. Kelly, R. Murray-Smith, Video of tilt-menu, 2004. www.dcs.gla.ac.uk/~rod/Videos/JamesMenu2.avi

[15] 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

[16] P. Eslambolchilar, *Making Sense of Interaction Using a Model-Based Approach*, Ph.D. thesis, Hamilton Institute, National University of Ireland, Maynooth, 2006.

[17] O. Bruijn, R. Spence, "Rapid serial visual presentation: a space-time trade-off in information presentation," *Advanced visual interfaces*, Palermo, Italy, p189-192, 2000

[18] J. Wijk, W. Nuij, "Smooth and efficient zooming and panning," *IEEE Symp. on Information Visualization,* Seattle, Washington, USA, p15-23, 2003

[19] A. Cockburn, C. Gutwin, J. Alexander, "Faster Document Navigation with Space-Filling Thumbnails," *ACM CHI*, Quebeck, Canada, p1-10, 2006

[20] L. Sun, F. Guimbretiere, "Flipper: a new method of digital document navigation," Ext. Abs. of *ACM CHI*, Oregon, USA, p2001-2004, 2005

[21] S. J. Cho, R. Murray-Smith, C. Choi, Y. Sung, K. Lee, Y-B. Kim, "Dynamics of Tilt-based Browsing on Mobile Devices," Ext. Abs. of *ACM CHI*, San Jose, 2007