Toss-It: Intuitive Information Transfer Techniques for Mobile Devices

Koji Yatani Koiti Tamura Keiichi Hiroki Masanori Sugimoto
University of Tokyo
5-1-5, Kashihwanoha, Kashiwa,
277-8561, Chiba, Japan
{yatani, tamura, hiroki, sugi}@itl.t.u-tokyo.ac.jp
+81-4-7136-3891

Hiromichi Hashizume
National Institute of Informatics
2-1-2, Hitotsubashi, Chiyoda-ku,
103-8430, Tokyo, Japan
has@nii.ac.jp
+81-3-4212-2511

ABSTRACT
In recent years, mobile devices have rapidly penetrated into our daily lives. However, several drawbacks of mobile devices have been mentioned so far. The proposed system called Toss-It provides intuitive information transfer techniques for mobile devices, by fully utilizing their mobility. A user of Toss-It can send information from the user's PDA to other electronic devices with a toss or swing action, as the user would toss a ball or deal cards to others. This paper describes the current implementation of Toss-It and its user studies.

Author Keywords
Mobile devices; information transfer; gesture recognition; location recognition.

ACM Classification Keywords
H.5.2. User Interfaces: Input devices and strategies; Interaction styles.

INTRODUCTION
In recent years, mobile devices have rapidly penetrated into our daily lives. Several drawbacks of mobile devices have been mentioned so far, such as their limited computational capability, or small screen real estate, and so on, as compared with notebook or desktop computers. However, by fully utilizing the most notable feature of mobile devices, that is, mobility, we can explore possibilities for a new user interface of the devices.

Suppose you want to copy a file in your mobile device to that of your friend around you. Although a memory card or an infrared communication is available as information transfer methods, these methods require several steps to complete the task. When you want to pass a file to several colleagues, you have to conduct the same procedures repeatedly. In another case, if you want to print out a photo in your mobile device through a printer in front of you, you have to conduct frustrating operations on its graphical user interface, such as selecting menu items with a stylus pen several times, in order to specify the printer.

CHI 2005, April 2–7, 2004, Portland, Oregon, USA.
ACM 1-59593-002-7/05/0004.

On the other hand, when you pass something to another person around you in the real world, all you have to do is just tossing it toward the person. If you can send information from your mobile device to other devices as you would pass physical objects to others, you will be liberated from bothersome and awkward operations on your device. Therefore, we propose a system called Toss-It that enables users to transfer information in their mobile devices (PDAs in this work) in an intuitive manner by utilizing their mobility, as shown in Figure 1.

Several research projects have proposed intuitive techniques for information transfer [2, 3]. However, these systems do not allow a user to send information to multiple devices in an intuitive manner as shown in Figure 1(c). Moreover, Toss-It allows a user to send information to a receiver beyond people in-between like Figure 1(a), because it sends information by a “toss” action, not by a “pointing” action.

In order to realize information transfer techniques with toss or swing actions, Toss-It must satisfy the following requirements:

- **Req1**: Toss-It can recognize user's toss and swing actions conducted with his PDA.
- **Req2**: Toss-It can automatically identify the positions and orientations of multiple users' PDAs and electronic devices.
- **Req3**: Based on a user's action, Toss-It can transfer digital information from his PDA to other users' PDAs or to the corresponding electronic devices.

It is desirable to satisfy the requirements without any external equipment embedded or installed in an environment where Toss-It is used, in order to make it available anywhere. This is the final goal of our project. Our initial goal, however, is to investigate if the proposed idea (information transfer by toss or swing actions) is possible and practical. This paper, therefore, discusses issues mainly related to Req1. In order to recognize users' actions, we attach inertial sensors to a PDA. This approach has several merits as compared to approaches such as using external equipment. For example, in order to capture a user's action in any location, installing and calibrating multiple cameras are necessary.

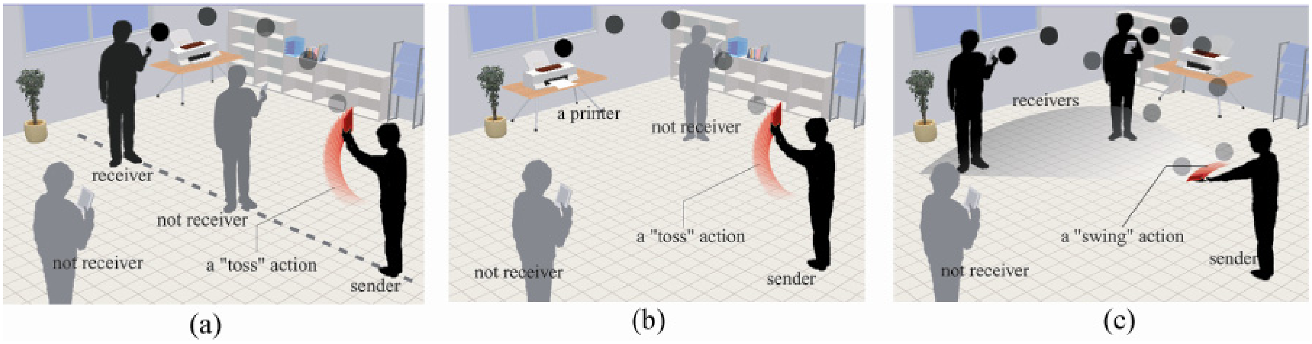

Figure 1. Intuitive information transfer techniques with Toss-It. (a) from a PDA to another PDA (b) from a PDA to a printer (c) from a PDA to multiple PDAs

As for Req2, there are several existing location recognition technologies applicable to Toss-It [1]. We are now developing a novel location and orientation recognition technology that can acquire relative positions and orientations of multiple devices without any external equipment, such as beacons in the ceiling. In the current implementation, however, we use a camera-based technology described later in this paper.

As for Req3, it may be possible to use P2P or ad-hoc network technologies. In this paper, however, we use a wireless LAN and a server computer. The server also manages data on positions and orientations of multiple devices. When a user conducts a toss or swing action with his PDA to transfer information, the software of Toss-It calculates the trajectory of the PDA, identifies target devices, and sends the information to the devices via wireless LAN.

RECOGNITION OF TOSS AND SWING ACTIONS

Hardware

In order to recognize a toss or swing action, we have developed a circuit board that mounts accelerometers (Analog Devices ADXL210), gyroscopes (Murata ENC-03J, ENC-03M) and a microprocessor (Hitachi H8 microcomputer) as shown in Figure 2. The circuit board is designed to be attached to a PDA. To capture users' quick actions as accurate as possible, four 2-axis accelerometers and three 1-axis gyroscopes are embedded in the board. They are connected to a PDA through a serial communication via a microprocessor.

Recognition Algorithm

In order to identify target devices that receive information by a user's toss or swing action, Toss-It is required to recognize not only the action, but also the strength of the toss action in a unicast situation (as shown in Figure 1(a)), and the trajectory of the swing action in a multicast situation (as shown in Figure 1(c)).

Eliciting Toss or Swing Actions

Ideally, Toss-It can recognize a toss or swing action through the output data of the inertial sensors. Several informal experiments, however, have indicated that non-negligible fluctuation occurs in the output data just after the action has been completed. Figure 3 shows a typical example of the output data of an accelerometer when a user conducted a toss action with Toss-It. In order to eliminate this fluctuation, we have devised a new recognition algorithm. To apply this algorithm, an assumption is made that a toss or swing action is initiated and finished in a state of rest. This assumption justifies the idea that the area of the positive part (P in Figure 3) is equal to that of the negative part (N in Figure 3).

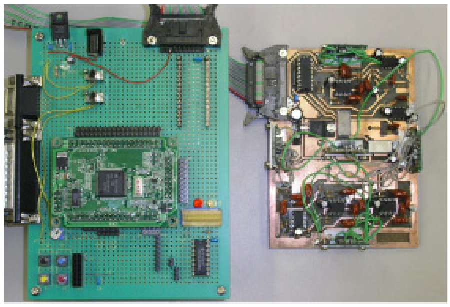

Figure 2. A circuit board with inertial sensors

The recognition process is summarized as follows: First, Toss-It searches an intersecting point of the output data curve and the zero acceleration line as shown in Figure 3. When Toss-It has found a new intersecting point, it calculates the integral of the acceleration values between the intersecting point and the previous intersecting point, named the “starting point”. If the value is greater than a specified threshold, Toss-It regards the current intersecting point as the “inversion point” and begins to calculate the integral of the acceleration values from the inversion point. While Toss-It makes the calculation, it evaluates the summation of the two integral values (the integral between the starting and inversion points, and the integral from the inversion point). When the value of the summation becomes approximately zero, Toss-It stops the calculation and regards the current point as the “end point”. Finally, Toss-It recognizes that a user's action happened between the starting point and the end point (the highlighted region in Figure 3).

In order to calculate the strength of a toss or the trajectory of a swing, a transformation matrix between the absolute coordinate system and the PDA coordinate system must be determined. During a user's action, Toss-It updates an Euler matrix by using angular velocities gained through the gyroscopes, and calculates the transformation matrix.

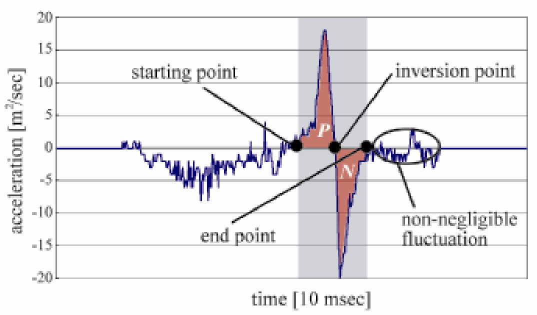

Figure 3. A typical example of the output data of an accelerometer when a user conducted a toss action with Toss-It

Estimating the Strength of a Toss Action
Toss-It estimates the strength of a toss, in order to determine how far "tossed" information travels and which devices receive the information. After several informal experiments of a toss action, we have made an assumption for reasonably accurate estimations and less complex calculations: When we toss something, we release it at the maximum speed. A toss action is started at the vertically downward position to the floor and finished without a follow through. Toss-It also assumes the launch angle is always 45[deg]. After calculating the maximum speed in a toss action, Toss-It estimates the flying distance with the equation of motion.

Estimating the Trajectory of a Swing Action
Toss-It calculates the trajectory of a user's swing action through the second integral of accelerations. To determine how many degrees a user has swung his PDA around him, we assume that a trajectory of a swing is an arc. Toss-It calculates an angle of the arc, and recognizes the devices inside the arc angle as receivers of information as shown in Figure 1(c). To simplify the calculation of the angle, we use only horizontal moves and neglect virtual moves of a swing action. The radius of the arc is decided by each user's arm length. An arc angle of a user's swing action can be calculated with the chord length and the radius of an arc.

LOCATION RECOGNITION

A marker with infrared (IR) LEDs and a stereo camera are used for the position and orientation recognition of a user's PDA. The IR LEDs on a circuit board are arranged to form an isosceles triangle (*A*, *B*, and *C* in Figure 4). The position and orientation of a user's PDA are determined by the shape of the triangle captured through the stereo camera fixed to a ceiling. Different blinking patterns of LEDs are assigned to individual PDAs for their identification.

Experimental results to evaluate this method have proved that the position and orientation recognition errors are less than 6[cm] and 10[deg], respectively. In the current implementation, it takes less than two seconds to identify individual devices, when the stereo camera can continue to correctly capture blinking patterns of LEDs. This means that a

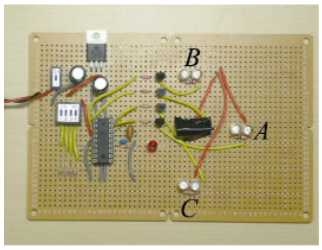

Figure 4. A marker with infrared LEDs

user is required to hold his PDA steadily for two seconds before conducting toss or swing actions, so that Toss-It can recognize the PDA successfully.

EXPERIMENTS AND EVALUATIONS

Recognition of Toss and Swing Actions

Preliminary tests proved that Toss-It could completely distinguish between a toss (vertical move) and a swing (horizontal move). We, therefore, evaluated Toss-It on how accurately it could recognize receivers with toss and swing actions. Six subjects participated in the following two experiments. In the first experiment, the subjects were asked to conduct a toss action and send information to devices placed at three different locations (1[m], 2[m], and 3[m] away from a subject). In the second experiment, the subjects were asked to conduct a three different horizontal swing with their PDAs (45[deg], 90[deg], and 135[deg]). Each subject repeated toss and swing actions 25 times for each of the three locations and three angles, respectively.

The average and standard deviation (SD) for each target distance and those for each target angle are described in Table 1 and in Table 2, respectively. Differences of toss actions between two target distances (1[m] and 2[m], or 2[m] and 3[m]) and those of swing actions between two target angles (45[deg] and 90[deg], or 90[deg] and 135[deg]) proved to be statistically significant by a Welch's t-test (two-tailed, $p < .01$), respectively.

User Studies

Figure 5 shows the experimental setting for evaluating Toss-It. Five subjects (male, right-handed) participated in the user studies, and four of them were asked to conduct the following tasks: E1 and E2 to transfer information by toss actions to (user1) and (user2), respectively, and E3 and E4 to transfer information by swing actions to (user1, user2, user4) and (user1, user2, user3, user4), respectively. The subjects were asked to conduct each of the four tasks ten times.

The results are summarized as Table 3. In the studies, each trial of the tasks by the subjects was judged as ``success'', if all and only the target users received the information (e.g., in E3, user1, user2, and user4). In E1 and E2, Toss-It transferred information to the nearest person within a two-meter radius from a landing point estimated through a sender's toss action (If no person existed in the area, no information

Target Distance[m]	Average[m]	SD
1	1.06	0.418
2	1.90	0.513
3	2.88	0.8056

Table 1. Average and standard deviation for each target distance

Target Angle[deg]	Average[deg]	SD
45	54.2	14.8
90	89.2	14.3
135	129	8.80

Table 2. Average and standard deviation for each target angle

transfer was done.). The analyses of the user studies clarified the following issues:

- Distance estimation errors: In 10.0[%] of all the trials in E1, Toss-It sent information to user2, because estimated distances of toss actions were between 2.25[m] and 5[m]. In 2.5[%] of all the trials in E2, estimated distances by Toss-It were between 5[m] and 8[m], and about 1[m] in 20.0[%] of the trials.
- Orientation recognition errors: In 10.0[%] of all the trials in E1, Toss-It sent information to user4. This means that the recognition error of a sender's orientation was more than 20 [deg] (about a half of the angle formed by the lines from the sender to user1 and user4), which is much larger than the maximum orientation recognition error (less than 10 [deg]). Video analyses of the user studies clarified that although the sender conducted toss actions by exactly facing toward the target receiver, his PDA did not always direct exactly toward the receiver. This type of failure also happened in the trials in E2, E3, and E4.
- Features in swing actions: The success transfer rate of E3 was higher than that of the other tasks, because the senders often swung their PDAs around themselves a larger angle than necessary. The senders in E4 also swung their PDAs a larger angle than necessary, which deteriorated its transfer success rate.

Discussions

The experiments suggested that more precise estimation of users' toss and swing actions are necessary. We are re-examining the current design of Toss-It from hardware and software aspects. It may be effective to improve the transfer success rate by capturing all users' gestures or utilizing contextual information. Suppose that receivers express their intention to receive information by tilting their own PDA vertically. Toss-It first identifies candidates of the receivers conducting the tilting gesture, and then sends information to some of them determined by sender's toss or swing actions. It may also be possible to utilize orientations of users' PDAs. For example, when their PDAs do not face toward a

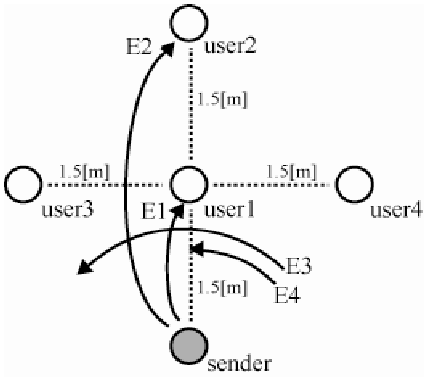

Figure 5. Users' positions and four experiments in the user study

Experiments	E1	E2	E3	E4
Transfer Success Rate[%]	75.0	70.0	85.0	60.0

Table 3. Success rate of information transfer in the experiments

sender, Toss-It judges that they have no intention to receive information from the sender. Capturing users' gestures or orientations is also effective for blocking the users to receive unnecessary information.

By utilizing situational and contextual information, the transfer success rate will be improved. For example, it is reasonable that Toss-It regards user2 as the correct receiver in E2, when the estimated location is much beyond the user2's location.

We believe that through the improvements, the proposed information transfer techniques will become more practical and usable.

CONCLUSIONS AND FUTURE WORKS

In this paper, we described the current implementation of Toss-It and its user studies. Several issues remain to be investigated: the improvement of the precise estimation of a toss and swing action is one of the most important issues. What kind of feedback (e.g. auditory or tactile) should be given to senders and receivers must be investigated. We also have a plan to conduct intensive usability studies in order to evaluate Toss-It as a user interface for mobile devices.

REFERENCES

1. Hightower, J. Location Systems for Ubiquitous Computing, *IEEE Computer*, 34, 8 (2001), 57-66.
2. Rekimoto, J. Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments, In *Proc. of UIST'97*, ACM Press (1997), 31-39.
3. Swindells, C., Inkpen, K. M., Dill, J, C. and Tory, M. That one there! Pointing to establish device identity, In *Proc. of UIST2002*, ACM Press (2002), 151-160.