Input Techniques to the Surface around a Smartphone using a Magnet Attached on a Stylus

Tetsuya Abe

University of Tsukuba 1-1-1 Tennodai, Tsukuba City, Ibaraki Pref., 305-8573, Japan abe@iplab.cs.tsukuba.ac.jp

Buntarou Shizuki

University of Tsukuba 1-1-1 Tennodai, Tsukuba City, Ibaraki Pref., 305-8573, Japan shizuki@cs.tsukuba.ac.jp

Jiro Tanaka

University of Tsukuba 1-1-1 Tennodai, Tsukuba City, Ibaraki Pref., 305-8573, Japan jiro@cs.tsukuba.ac.jp

CHI'16 Extended Abstracts, May 07-12, 2016, San Jose, CA, USA ACM 978-1-4503-4082-3/16/05. http://dx.doi.org/10.1145/2851581.2892376

Abstract

We present input techniques for smartphones using a stylus based on magnetism. Our techniques use a magnet attached on a stylus and a three-axis internal magnetic sensor in a smartphone. Using our techniques, users can input into a smartphone using a stylus on the surface on which the smartphone is placed. This enables users to employ a wider surface than the display of the smartphone for input. In addition, our techniques require no additional sensors on the smartphone. In this paper, we present two input techniques: nonelectric and electric ones. In addition, we describe our stylus for use with each technique and a coordinates-estimation algorithm we implemented. We also describe two applications: a digit recognizer and a paint application.

Author Keywords

Pen interaction; pen-shaped device; around device interaction; mobile device; magnetism; magnetic sensor; hall sensor; magnetometer.

ACM Classification Keywords

H.5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces - Input devices and strategies (e.g., mouse, touchscreen)

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author. Copyright is held by the owner/author(s).

smartphone







Figure 1: An example of a user's input.



Figure 2: Styli (a: nonelectric stylus, b: electric stylus).

Introduction

This paper presents input techniques to smartphones. Our techniques extend the input surface of the smartphone to one around the smartphone. We employ a stylus where a magnet is attached at the tip. The stylus enables users to input into a smartphone using everyday surfaces in a similar manner when inputting into a computer using a pen tablet. Our techniques can be used in combination with touch input or external input devices such as a Bluetooth keyboard. Figure 1 shows an example of a user's input.

Our techniques use only an internal magnetic sensor for sensing to estimate coordinates of the stylus. Our techniques estimate coordinates based on the assumption that the stylus is placed on the surface perpendicularly. Users are thus not required to attach any sensors to the smartphone. Moreover, the aforementioned assumption greatly simplifies the algorithm for the estimation, allowing the smartphone to estimate coordinates in real-time.

This paper presents two techniques: non-electric and electric ones. This paper also presents the algorithm to estimate the coordinates of the stylus, and two applications that use our techniques.

Related Work

Our techniques extend the input surface of a smartphone to the wider one around a smartphone. Our techniques employ a stylus where a magnet is attached at the tip and use an internal magnetic sensor. In this section, we review previous work related to our techniques.

Around Device Interaction using Magnets

GaussSence [10] is a technique to detect the place and direction of a magnet embedded in a stylus using a hall sensor array attached to the back of a smartphone. TMotion [13] is a technique to realize 3D input into a smart-

phone using a pen-shaped device which contains a 9-DOF inertial measurement unit and a magnet. By contrast, our techniques require no additional sensors on the smartphone in to estimate the coordinates of the stylus because the movable range of the stylus is limited to the surface on which the smartphone is placed.

MagPen [8] is a gesture-input technique over 3D space near a smartphone using a pen to which a magnet is attached. MagGetz [7] is a user-customizable passive control widgets that realize tangible interaction on and around a smartphone using magnets and an internal magnetic sensor. MagiSign [9] is a user identification/authorization technique based on 3D magnetic signatures around a mobile device that use an internal magnetic sensor. These techniques utilize the 3D space around a smartphone, whereas our techniques use the surface around the smartphone as an input surface.

Input Techniques using Existing Surfaces

Scratch Input [4] is an acoustic-based technique for gesture input. It uses a stethoscope attached to the audio interface of a computer to capture unique sound produced when a fingernail is dragged over a surface. Anywhere Surface Touch [12] allows any flat or curved surface to be used as an input area by using a camera and microphone. Omni-Touch [3] and Skinput [6] are techniques that use human skin as an input surface. By contrast, our techniques use the surface around a smartphone for input.

Estimating the position of a stylus is also explored to detect inputs on existing surfaces. MEMO-PEN [11] is a ballpoint pen equipped with a CCD camera on its tip. It reproduces user's handwriting on a computer by capturing them using the camera. Abracadabra [5] allows users to input gestures to a smartphone using 3D space around the mobile-devices using magnets. By contrast, our techniques use only an



Figure 3: Joint mechanism.



Figure 4: Coordinates system in our algorithm.

internal magnetic sensor to estimate the coordinates of the stylus on the surface around the smartphone.

Input Techniques to Computers using Magnets uTrack[1] is a technique that converts the movement of the thumb and ring finger into 3D input using a permanent magnet on the thumb and two magnetic sensors on the ring finger. Han et al. [2] presented a pair of wearable devices with two 2-axis magnet sensors and a magnet for inputting characters into a computer. By contrast, our techniques estimate the coordinates of a magnet on the surface around the smartphone using only an internal magnetic sensor.

Our Input Techniques

We present two input techniques: nonelectric and electric ones. The difference between our two techniques is the attached magnet. A permanent magnet is used in our nonelectric technique, whereas an electromagnet is used in our electric technique. We implement both a nonelectric stylus (Figure 2a) and an electric stylus (Figure 2b) for our nonelectric and electric techniques, respectively.

We describe how to input into smartphones using our techniques. First, a user prepares a stylus. A stylus is a stick where a magnet is attached at the tip. The user can easily make a stylus by simply attaching a magnet to an ordinary pen which contains no magnetic substances. Next, the user places his or her smartphone on a surface such as a desk and inputs using the stylus to the surface around the smartphone, as shown in Figure 1. When the user inputs using one of our techniques, the stylus must be held on the input surface perpendicularly. However, we remove this limitation by employing a joint mechanism at the tip of the stylus as shown in Figure 3. This joint mechanism between the magnet and the ordinary pen causes the magnet automatically to align perpendicularly to the surface regardless of the direction of the stylus. Thus users can hold the stylus in a natural manner.

Implementation

In this section, we describe the hardware and then the software that runs on smartphones. We use an ASUS Zenfone 5 (OS: Android 5.0.2; size: 72.8 mm \times 148.2 mm \times 10.34 mm; CPU: Qualcomm Snapdragon 400 1.2GHz; RAM: 2GB) for our implementation. The internal magnetic sensor in the smartphone is AK09911 3-axis magnetic sensor (Asahi Kasei Microdevices).

Hardware

We implemented two styli: nonelectric and electric ones. The nonelectric stylus (Figure 2a) is implemented by connecting a wooden stick (10 mm in diameter \times 100 mm in length) and the S pole of a neodymium magnet (10 mm in diameter \times 15 mm in length) in series. We connected two magnets to increase the magnetism of the stylus. The electric stylus (Figure 2b) is implemented by wrapping a polyurethane copper wire (0.4 mm in diameter) around a ferrite bar (10 mm in diameter \times 50 mm in length) approximately 480 times. We attached a microswitch to the N pole of the electromagnet so that magnetism turns on only when the user is writing on the surface. We applied 5 V to the electric stylus with a current of approximately 0.5 A.

The nonelectric stylus can be used without electricity except on smartphones. However, the software must estimate whether the user is in the process of writing because the magnetic sensor observes the magnetic field of the magnet even when the user is not in the process of writing (i.e., the stylus is away from the surface). For this estimation, we adopted a user-based approach: users must hold the stylus in the place for a while before and after a stroke is produced. In our implementation, our system determines

Equations.

$$\begin{pmatrix} b_r \\ b_z \end{pmatrix} = \frac{m}{4\pi\mu_0} \begin{pmatrix} r(1/R^3 - 1/r^3) \\ \ell/R^3 \end{pmatrix}$$
(1)

$$\frac{b_r}{b_z} = \frac{r}{\ell} \left(1 - \left(1 + \left(\frac{\ell}{r}\right)^2 \right)^{3/2} \right)$$
(2)

$$\left(\frac{\ell}{r}\right)^5 + 3 \left(\frac{\ell}{r}\right)^3 + \left(3 - \left(\frac{b_r}{b_z}\right)^2 \right) \frac{\ell}{r} + 2\frac{b_r}{b_z} = 0$$
(3)

$$x_A = r \cos \left(\arctan \left(\frac{b_y}{b_x}\right) \right)$$
(4)

**m*: magnetic moment; μ_0 : magnetic permeability; $b_r = -\sqrt{b_x^2 + b_y^2};$ $R = \sqrt{x_A^2 + y_A^2 + \ell^2};$ $r = \sqrt{x_A^2 + y_A^2}.$ the start and end of a stroke by determining whether the estimated coordinates are changing. That is, the distance between the former and current coordinates is longer than a threshold empirically determined as 30 pixels. On the other hand, using electric stylus, the software does not need to estimate whether the user is in the process of writing because the magnetic field is turned off when the stylus is away from the surface, while the stylus requires electricity.

Software

Our software obtains the readings of the magnetic sensor at a fixed interval (20 ms in our current implementation), estimates the coordinates of the stylus based on these readings, and sends the coordinates to the application. Our software finishes this process within 20 ms. Moreover, when a user inputs using the nonelectric stylus, the software also determines whether the user is in the process of writing by determining whether the estimated point is moving.

The algorithm for estimating the coordinates of the stylus requires a preliminary process. Because the software must cancel the effect of terrestrial magnetism, the software stores the readings of the magnetic sensor as a terrestrial magnetism component when the software starts. The algorithm uses the difference between the current reading and terrestrial magnetism component for estimation.

Moreover, if the stylus is too far from the sensor, the difference becomes too small to estimate coordinates. Thus, our software estimates the coordinates only when the measured reading is greater than the highest terrestrial magnetism (i.e., the constant MAGNETIC_FIELD_EARTH_MAX in SensorManager class for our current implementation). Otherwise, our software stops estimating.

For the algorithm used to estimate the coordinates of the stylus, we use the orthogonal coordinate system (x, y, z)

as shown in Figure 4. In this coordinate system, the xy plane corresponds to the input surface and the origin O corresponds to the position of the magnetic sensor. Let the magnet of length ℓ is placed on the point $A(x_A, y_A, 0)$ of the xy plane facing the N pole. Let $B(b_x, b_y, b_z)$ be the difference between the current reading and the terrestrial-magnetism component at O. Then, (b_x, b_y) can be composed into b_r , which is in the direction of \overrightarrow{OA} . Coulomb's law leads (b_r, b_z) by (1). By calculating b_r/b_z as shown in (2), (3) is delivered. Here, r can be calculated by (3) using the bisection method because ℓ is known. Then, x_A and y_A are calculated by (4).

Evaluation

We conducted two experiments to investigate the effectiveness and the limitations of our system. First, we investigated the area in which our system can estimate the coordinates of the stylus (Experiment 1) and then the accuracy of the estimated coordinates (Experiment 2). We used the same environment as described in the *Implementation* section. Both experiments were conducted on a plastic desk.

Experiment 1: Area where our System can Estimate Coordinates

We conducted Experiment 1 to investigate the area where our system can estimate the coordinates of the stylus. First, we placed the smartphone in the center of an A3-sized sheet of paper taped to a desk. We then moved each stylus throughout on the paper and determined whether our system estimates the coordinates of the stylus.

Figure 5 shows the results of Experiment 1. For the nonelectric stylus, the results reveal that the area is an annulus whose inner and outer radii are approximately 4 cm and 15 cm centered on the magnetic sensor. The area is approximately 580 cm^2 , which is 8.3 times as large as that of



Figure 5: Result of Experiment 1 (2 cm per unit).



Figure 6: Result of Experiment 2 (2 cm per unit).

	Distance (mm)	Error (mm)	Ratio (%)
	40 - 70	3.55	6.47
	70 – 90	5.26	6.40
	90 – 110	7.49	7.64
	110 – 130	8.26	6.98
	130 – 150	11.3	8.24
1	40 - 150	8.03	7.37

Table 1: Average error andaverage error ratio by distancefrom Experiment 2.

the smartphone display. For the electric stylus, the results similarly reveal that the area is an annulus whose inner and outer radii are approximately 2.5 cm and 12 cm centered on the magnetic sensor. The area is approximately 350 cm² which is 5 times as large as that of the smartphone display. An inner area where our system cannot estimate coordinates of the stylus exists because the magnetic sensor is saturated when the magnet is too close to it.

Experiment 2: Accuracy of the Estimated Coordinates In Experiment 2, we measured the difference between the real and estimated coordinates in the area revealed in Experiment 1. We printed a facsimile of that area on an A3sized sheet of paper overlaid with a grid spacing of 2 cm. We placed the nonelectric stylus at each intersection point and recorded the estimated coordinates.

Figure 6 shows the results of Experiment 2. Here, the blue points indicate the estimated coordinates. Table 1 shows the average error of the position of the stylus and the average error ratio for each range of distance. As Table 1 shows, the longer the distance, the greater is the average error. However, the average error ratio is nearly the same for all the distances. This result suggests that we may be able to obtain a smaller average error with calibration by tracing a known line using the stylus.

Applications

We implemented two applications that employ our techniques: a digit recognizer and a paint application.

Digit Recognizer

We implemented a digit recognizer, which recognizes digits based on the coordinates of the stylus. When a user inputs a digit using the stylus on the surface around the smartphone, the application recognizes the digit based on the series of estimated coordinates. We used machine learning (Support Vector Machine) for recognition instead of image recognition because we want the application to recognize digits in real-time. We calculated the feature vector as follows: (1) divide the coordinates of the stroke into twelve groups; (2) calculate the coordinates of the center point $C_n = (x_n, y_n)$ ($n = 0, \dots, 11$) for each group; and (3) calculate $\cos \theta_i$ ($i = 1, \dots, 10$) of each θ_i formed by $\overrightarrow{C_{i-1}C_i}$ and $\overrightarrow{C_iC_{i+1}}$. We use these $\cos \theta_i$ as a feature vector.

Figure 7 is a screenshot of the digit recognizer. It shows a recognized digit at the top of the screen and the estimated coordinates at the center. The color of the background is cyan when the stylus is within the area shown in Figure 5 and white when it is outside that area. The digit recognizer changes the background color to inform users whether the stylus is within the area.

Paint Application

We implemented a paint application in which users can draw some figures. When a user inputs using the stylus on the around surface of the smartphone, the system adds a figure in the canvas of the application based on the estimated coordinates. We implemented four kinds of figures for drawing: rectangle, triangle, oval, and free-hand line.

Figure 8 is a screenshot of the paint application. Users can select the desired figure to draw using the buttons on the top of the screen. When the stylus is in the area shown in Figure 5, a cyan frame is displayed.

Discussion

Implementation

Even if a user moves the stylus quickly, our system can estimate an adequate series of coordinates for the applications. Specifically, our algorithm can estimate the coordinates of a



Figure 7: A screenshot of the digit recognizer.



Figure 8: A screenshot of the paint application.

stylus every 20 ms. If an application requires more coordinates per unit time, the interval can be shortened depending on the specification of the smartphone.

The type of application determines the stylus to use. If an application is unable to feed electricity to the stylus, the nonelectric stylus should be used. Otherwise, the electric stylus is better because this stylus can determine whether a user is in process of writing. In our implementation, users used the nonelectric stylus for the digit recognizer and the electric stylus for the paint application. In the digit recognizer, users must hold the stylus in the place for a while before and after a stroke so that the recognizer can divide coordinates into strokes. We intend to improve the function of the software to estimate whether a user is in the process of writing while using the nonelectric stylus. This will be accomplished by assessing the vibration or noise generated when the user writes using an internal accelerometer or microphone.

One limitation of our techniques is that if the input surface contains a magnetic substance, our system cannot estimate the coordinates correctly. This is because a magnetic substance diffuses magnetic field lines. Therefore, users must use a surface that contains few or no magnetic substances or use an area remote from a magnetic substance.

Evaluation

Although the size of the area that our algorithm can estimate coordinates was sufficient to use our applications, users sometimes require a wider surface. Generally, when users strengthen the magnet (e.g., connecting another magnet or feeding more current), the area in which our system can estimate coordinates becomes wider. However, additional magnetic substance may then exist in the expanded input area. The inner area in which our system cannot estimate the coordinates becomes larger. Thus, users should choose a suitable magnet for their environment.

The accuracy of the estimated coordinates of the stylus can be improved by means of calibration. However, the advantage of our algorithm is its ability to estimate coordinates without calibration. Thus, the developers of an application should determine whether they force users to calibrate the coordinates for their application.

Application

Although the digit recognizer is not so accurate in its current implementation (accuracy was approximately 70% in our preliminary evaluation), it can be improved by changing the number of stroke division or the number of training sets. This represents our immediate future work. In addition, the input for the digit recognizer is a single stroke. However, inputting characters from various languages with only a single stroke is a challenge. Therefore, we plan to develop a character recognizer that recognizes multiple input strokes (e.g., image recognition).

Conclusion and Future Work

In this paper, we presented input techniques based on magnetism using the surface around a smartphone. We showed nonelectric and electric techniques and implemented a stylus for use with each. We also developed a coordinatesestimation algorithm and examined two applications that use our techniques.

For future work, we will conduct a user study to evaluate our system, improve the electric stylus to consume less electric power and more portable, implement other applications (e.g., character recognizer, personal identification, pointing). We are also hope to apply our techniques to smartwatches, because the touch input area of a smartwatch is severely limited. One or several of our techniques may be a powerful solution to this problem.

References

- Ke-Yu Chen, Kent Lyons, Sean White, and Shwetak Patel. 2013. uTrack: 3D Input Using Two Magnetic Sensors. In Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13). ACM, New York, NY, USA, 237–244. D0I: http://dx.doi.org/10.1145/2501988.2502035
- [2] Xinying Han, Hiroaki Seki, Kamiya Yochitsugu, and Masatoshi Hikizu. 2007. Wearable Handwriting Input Device using Magnetic Field. In SICE, the Society of Instrument and Control Engneers, 2007 Annual Conference. 365–368. DOI: http://dx.doi.org/10.1109/SICE.2007.4421009
- [3] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011a. OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, New York, NY, USA, 441–450. D0I:http://dx.doi.org/10.1145/2047196.2047255
- [4] Chris Harrison and Scott E. Hudson. 2008. Scratch Input: Creating Large, Inexpensive, Unpowered and Mobile Finger Input Surfaces. In Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08). ACM, New York, NY, USA, 205–208. D0I: http://dx.doi.org/10.1145/1449715.1449747
- [5] Chris Harrison and Scott E. Hudson. 2009. Abracadabra: Wireless, High-precision, and Unpowered Finger Input for Very Small Mobile Devices. In Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09). ACM, New York, NY, USA, 121–124. DDI: http://dx.doi.org/10.1145/1622176.1622199
- [6] Chris Harrison, Desney Tan, and Dan Morris.

2011b. Skinput: Appropriating the Skin As an Interactive Canvas. *Communications of the ACM* 54, 8 (Aug. 2011), 111–118. DOI: http://dx.doi.org/10.1145/1978542.1978564

- [7] Sungjae Hwang, Myungwook Ahn, and Kwang-yun Wohn. 2013a. MagGetz: Customizable Passive Tangible Controllers on and Around Conventional Mobile Devices. In Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13). ACM, New York, NY, USA, 411–416. DOI: http://dx.doi.org/10.1145/2501988.2501991
- [8] Sungjae Hwang, Andrea Bianchi, Myungwook Ahn, and Kwangyun Wohn. 2013b. MagPen: Magnetically Driven Pen Interactions on and Around Conventional Smartphones. In Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services (Mobile-HCI '13). ACM, New York, NY, USA, 412–415. DOI: http://dx.doi.org/10.1145/2493190.2493194
- [9] Hamed Ketabdar, Kamer Ali Yukse, Amirhossein Jahnbekam, Mehran Roshandel, and Daria Skirpo. 2010. MagiSign: User Identification/Authentication Based on 3D Around Device Magnetic Signatures. In *The Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBI-COMM 2010).* IAREA, 31–34.
- [10] Rong-Hao Liang, Kai-Yin Cheng, Chao-Huai Su, Chien-Ting Weng, Bing-Yu Chen, and De-Nian Yang. 2012. GaussSense: Attachable Stylus Sensing Using Magnetic Sensor Grid. In Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12). ACM, New York, NY, USA, 319–326. DOI: http://dx.doi.org/10.1145/2380116.2380157

- [11] Shinji Nabeshima, Shinichirou Yamamoto, Kiyoshi Agusa, and Toshio Taguchi. 1995. MEMO-PEN: A New Input Device. In Conference Companion on Human Factors in Computing Systems (CHI '95). ACM, New York, NY, USA, 256–257. DOI: http://dx.doi.org/10.1145/223355.223662
- [12] Takehiro Niikura, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2014. Anywhere Surface Touch: Utilizing Any Surface as an Input Area. In *Proceedings of the 5th Augmented Human International Conference (AH*

'14). ACM, New York, NY, USA, Article 39, 8 pages. D0I:http://dx.doi.org/10.1145/2582051.2582090

[13] Sang Ho Yoon, Ke Huo, and Karthik Ramani. 2015. TMotion: Embedded 3D Mobile Input Using Magnetic Sensing Technique. In Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15 Adjunct). ACM, New York, NY, USA, 71–72. DOI: http://dx.doi.org/10.1145/2815585.2815723