A Pointing Method Using Two Accelerometers for Wearable Computing

Yohei Tokoro, Tsutomu Terada, Masahiko Tsukamoto
Grad. School of Engineering,
Kobe University
1-1, Rokkodaicho, Nadaku,
Kobe 657-8501, Japan
{tokoro@stu., tsutomu@eedept., tuka@} kobe-u.ac.jp

ABSTRACT

A variety of real-world situations are beneficial for wearable computing since it provides information services while users are doing other jobs. Therefore, a simple and hands-free input interfaces have suitable for these computer operations. However, such interface has not been achieved with conventional input devices such as mice or track balls. Although gesture or eye-gaze input techniques have also been developed for wearable computing, they also suffer from problems i.e., a slow pointing speed, difficulty in carrying devices, and complexity in parallel use when doing tasks. We propose a new method of pointing using input of simple gestures with two accelerometers. By dividing the specifications of two coordinates into a combination of two independent motions, we accomplish accurate and intuitive pointing. A user attaches two small accelerometers to both his/her hands or both elbows. The pointing is done by using the intersection of two straight lines, and the movement of the lines is synchronized with that of the accelerometers. In addition, we also propose a method of changing the position of objects being pointed at that is new approach. The results we obtained from our evaluation experiments confirmed that our method was effective.

1. INTRODUCTION

Wearable computing has recently attracted a great deal of attention due to computer miniaturization [13]. Users in wearable computing environments acquire a variety of information providing by the computers attached to their bodies, which are typically PDAs or note PCs. Different to conventional tasks in desktop computing, their main intended purpose is to help people to work in the real world. Wearable computers have capabilities for capturing user intentions in their day-to-day lives to enhance their capabilities in achieving daily tasks.

Graphical User Interface (GUIs) are popular in conventional desktop-computing environments and are commonly and easily used to control various applications such as e-mail, Web pages, and Internet phones. In using these, the pointing mechanism is essential because icons and buttons on GUIs must be operated with a pointing device such as a mouse or a pen. The pointing mechanism is also important in wearable computing for compatibility, i.e., for users who want to use applications in both desktop and wearable computing environments. However, since wearable computing imposes several restrictions in using devices, it is difficult to use conventional ones for input such as mice. In other words, we need a method of hands-free pointing that can be used for work, walking, and doing other tasks.

Various conventional pointing devices such as mice, track balls, and joysticks have been developed [17]. However, as it is necessary to hold them when using them, they are obviously not hands-free. Although there has been research on hands-free pointing including glance and motion input, operations are limited or difficult and there are no methods that are compatible with conventional approaches.

We propose a new method of pointing using gestures with two small accelerometers, which is almost compatible with conventional pointing devices. With this approach, a user operates the right and left axes with the two accelerometers. Simple and intuitive operations with our method contribute to achieving quick and accurate pointing.

This paper is organized as follows. Section 2 describes related work on pointing methods for wearable computing. Section 3 describes the design of our pointing method. Section 4 describes the evaluation experiment and presents a discussion on our method. The paper is summarized in Section 5.

2. RELATED WORK

2.1 Input methods

Portable input devices such as track balls, joysticks and wireless mice with gyro sensors can be used for wearable computing [2, 5]. However, they must be held in hand, and users must operate them with their hands. In the types installed on fingers [12], a device with a built-in optical image sensor is attached to the tips of index fingers and pointing is accomplished by moving the index fingers. In Finger-Mount [7], an ultrasonic sensor and two buttons are attached to the tips of index fingers. However, devices attached to fingers do not satisfy the demands of hands-free pointing.

There has been a great deal of research on hand gestures.

SAC'09 March 8-12, 2009, Honolulu, Hawaii, U.S.A.

In Gesture Pendant [14], a user utilizes a camera and an infrared-ray floodlight attached to the pendant and inputs gestures by sight. In WearTrack [4], pointing is achieved using ultrasonic waves by detecting the position of the finger tip with three microphones installed on the head. In the hand mouse [8], a user's hand is used as a pointing device by detecting the location of his/her hands from the image obtained with a wearable camera.

There was been some research on pointing devices that use the head. In MAGI-MOUSE [9], the pointer is moved by inclining the head slowly and clicking is done by nodding quickly. In TrackIR [10] and HeadMouse Extreme [11], pointing is done by detecting a marker attached to the head with an infrared sensor.

There has also been much research on pointing with eye-gazes, such as [3]. Eye-Tracking [15] is a device that controls the pointer through eye-gazes. However, eye-gaze pointing is not sufficiently for pointing at small targets such as desktop icons. Moreover, in order to achieve clicking by eye movement, the system must select one of two ways: using a camera to detect the specific motion of eyes or using other sensors. The former burdens the user due to unusual eye motions, and the latter needs additional devices such as microphones that make the system complex and too heavy to wear.

2.2 Pointing with accelerometer

Accelerometers are commonly used in this field. In Ubi-Finger [16], a glove-shaped device is used in which bend sensors are installed on parts of the forefinger, and two-axis accelerometers are attached to the wrist. Ubi-Finger assumes simple operations for information appliances such as turning lights on or off, and it cannot be applied to pointing on a GUI.

Past research on pointing with accelerometers has introduced numerous pointing systems with one unit including two- or three-axis accelerometer sensor and occasionally gyro-sensors, such as [6]. They are typically divided into two categories: inertial navigation and gravitation sensing types. The former uses the accumulation of sensor values as the velocity of the pointer. As a result, second-order integration indicates the location of the pointer. It is widely known that this method is inaccurate because of inertial drift and error in accumulation. Although the Kalman filter is commonly used for reducing errors in the field of global navigation systems, it is not sufficiently accurate for pointing. In gravitational sensing, the directions of gravitation, i.e., roll and pitch, are used as the velocity of the pointer. It is generally difficult to use the gravitational sensing because the association between motion and the movement of the pointer is not intuitive. Another problem in using accelerometers is trembling. To employ these as pointing devices, they must be sufficiently sensitive to follow quick motions by users, which cause overly the too-sensitive detection of their movement. The buttons and menus used in present window systems are too small compared with the range of the screen. Although they are suitable on very stable desktop environments, they are not very good for wearable pointing. Moreover, these two methods requires users to pay careful attention to the screen because the locating systems fully depend on visual feedback from the user. Without watching the pointer on the screen, the user easily loses the position of the pointer.

3. POINTING METHOD

3.1 Pointing for wearable computing

In designing a method of pointing for wearable computing, we needs to consider seven requirements;

- Speed: Pointing in some wearable computing environments must be done much more quickly than in conventional desktop environments. For example, when a hotel receptionist uses a wearable computer to obtain information, he has to quickly respond to the requests by guests. There are other situations involving emergency medical treatment and rescues in a disaster areas.
- Wide accuracy range: Pointing generally requires a wide range of movements from very subtle to abrupt changes. Because pointing is often difficult in wearable computing environments, it more effective to control the object being pointed at.
- Comfort: Since input devices are used for long periods, they must be small enough and sufficiently comfortable to transport on the body.
- Intuition: The association between operation of the device and movement of the pointer should be sufficiently intuitive. If there are too difficult to understand, users must concentrate on operation and their attention on real-world tasks or other phenomena will decrease, which may cause accidents.
- Low cost: Pointing must be done with as little computational power as possible. High processing costs such as in image processing degrades the performance of primary applications. Since low power consumption is especially required in wearable computing environments due to the restricted power resources, the effect of processing power increases.
- Hands-free operation: Since computers must be operated in parallel with other real-world tasks using hands, hands-free operations are required.
- Eye-free operation: In some cases such as driving and surgery, computers must be operated without watching the screen [1]. Therefore, the pointing method should not require a steady gaze for pointing.

3.2 Design of pointing method

Considering the requirements described in the previous section, we propose a new pointing method of that fulfills these by employing two accelerometers to make operation accurate, focusing on the assumption that people can precisely and quickly control only one axis by moving a body part. By using two sensors, two parts of a user's body, e.g., both elbows, both hands, or both kees, can be used to control the values of the two coordinates. As described in Section 2, inertial navigation is not sufficiently accurate for actual use, and the type using gravitational sensing that converts the angle of gravitation into the velocity of the pointer is not intuitive for most users. The new method converts the angle of gravitation into the angle of straight lines on a screen, which is intuitive because the angle of the elbows or hands is almost equal to the angle of the lines. These are

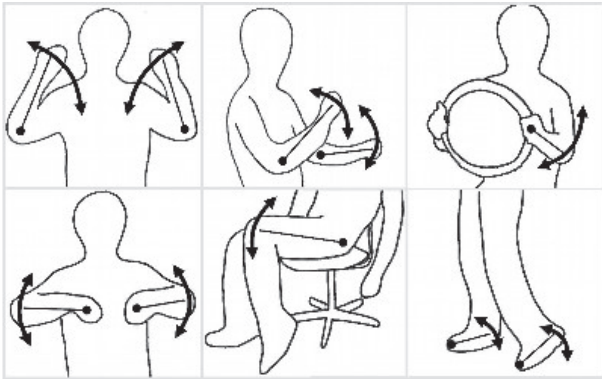

Figure 1: Positions of accelerometers and limb movements

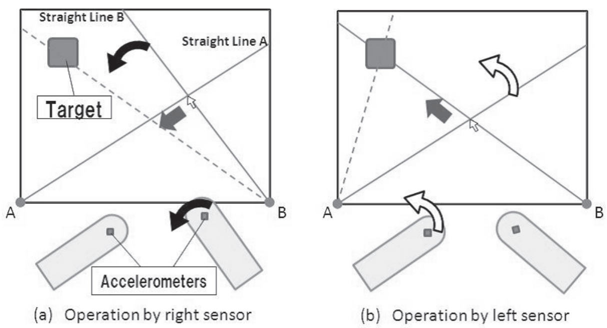

Figure 2: Pointing operations with our method

an examples of accelerometer positions and limb movements for our method in Figure 1.

A straight line (called line A) rotates with the fixed point at the lower left of the screen (point A), and another straight line (called line B) rotates with the fixed point at the lower right of the screen (point B). The pointer appears on the intersection of these two lines. These straight lines A and B correspond to the values from the two accelerometers, and move by changing the angles from the bottom of the screen according to the changes in the values of the accelerometers. As a result, pointing is achieved by moving the body parts.

The four steps explain pointing operations where sensors are attached to both hands of a user as shown in Figure 2:

1. By moving the accelerometer attached to the right hand, straight line B moves, and the pointer moves along the straight line A.
2. Straight line B arrives at the target.
3. Straight line A moves along with the movement of the left hand, and the pointer moves along straight line B.
4. When straight line A arrives at the target, the pointer reaches the target.

3.3 Implementation

Based on the method proposed in the previous section, we used a wireless sensor module that our research group has advocated, whose dimensions are 20 × 20 × 3.9 mm. Figure 3 is a photograph of the module. It employs a Nordic nRF24E1, which is integrated with a 2.4-GHz RF transceiver (nRF2401) and an 8051 compatible MCU. The data rate is 250 Kbps, the maximum output power is 0 dBms, and it has 125 Tx/Rx channels. This module has a three-axis digital 12-bit output linear acceleration sensor, the LIS3LV02DQ made by STMicroelectronics.

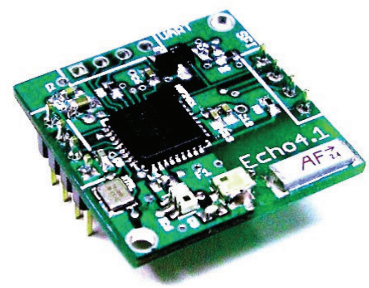

Figure 3: Wireless-Sensor accelerometer

We implemented the software that managed the hardware and cursor by using the Microsoft Visual C ♯ of Visual Studio 2005 on Microsoft Windows XP Professional.

3.4 Object allocation

In conventional GUIs, input devices employ XY coordinates and displayed objects are arranged along with the X and the Y axes. However, when using the new method of pointing we propose, control was achieved by the angles of two lines. Therefore, the to enable objects being pointed at should be arranged taking into consideration the unique characteristics of our method to enable easy and intuitive operation.

3.4.1 Approach

Fast pointing can be achieved with the new approach by using it as a hands-free interface, but there is a mismatch between the pointing method and the arrangement of objects being pointed at. For example, the menu structures on most GUIs use orthogonal coordinates, and desktop icons are also arranged along with the xy-coordinates. However, these arrangements are not suitable for the method of pointing we propose. Because our method is based on the slanted lines specified by the angle, intuitive operation cannot be attained with these conventional GUIs. Therefore, we propose a new way of arranging objects being pointed at to improve pointing.

GUIs generally have various objects as pointing targets, such as icons on desktops and in folders, on toolbars and in pull-down menus in windows, and specific operation buttons in the software. These objects to be pointed at should be arranged along with the axis in our method.

3.4.2 Arrangement of icons on desktop

The first step in constructing the arrangement of objects with our method was to find a way of arranging icons on the desktop. In our new approach, pointing is done by controlling the angle of two lines from the fulcrums. Therefore, the icons should also be allocated radially from the fulcrums.

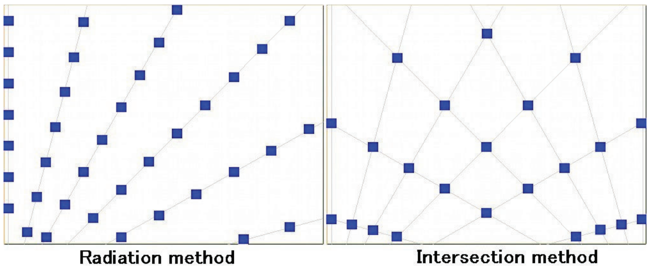

Figure 4: Radiation and intersection methods for arranging icons

We propose two patterns of arranging objects based on this concept:

(1) Radiation method

Let us assume several virtual lines have been prepared at certain intervals of angles. The icons are arranged along with the lines at the intervals shown in Figure 4 (left). Rough selection can be done by using the straight line from the fulcrum, and detailed selection is achieved with the other line. In other words, the role of the two lines are fixed to wide-ranging and detailed selection.

(2) Intersection method

Virtual lines at certain intervals of angles are prepared from both fulcrums, and the icons are arranged at each intersection. The role of the two lines can be freely switched. Therefore, flexible pointing is achieved according to the user's preferences. Figure 4 (right) shows an example of intersection arrangement.

4. EVALUATION

We first conducted an experiment to evaluate our new method of pointing and then evaluated object allocation.

4.1 Evaluation of pointing

To evaluate how effective our proposed method was, we set up experiments using a prototype. In the evaluation, eight test subjects each pointed at a target on the screen. The target was 64 × 64 pixels and the resolution of the PC screen we used for the evaluation was 1024 × 768 pixels. The target jumped to random positions on the screen when the pointer arrived at the target. Every test subject used ten variables in the evaluation, i.e., four different settings for the proposed method and six existing pointing devices. All eight test subjects participated in the experiment for five minutes for all ten variables. An evaluation program calculated the average elapsed time for one target to be selected in each pair of coordinates.

As a result of preliminary experiments, we found the following. When we arranged the fulcrums under the lower right corner and the lower left corner of the screen, the pointing speed at the bottom was slower than the speed at the top. One reason for this is that the effect of minute changes in the angle of the line became much greater at the bottom of the screen. Therefore, we repositioned the fulcrums by 200 pixels (about 25% of the width of the screen) under the original position.

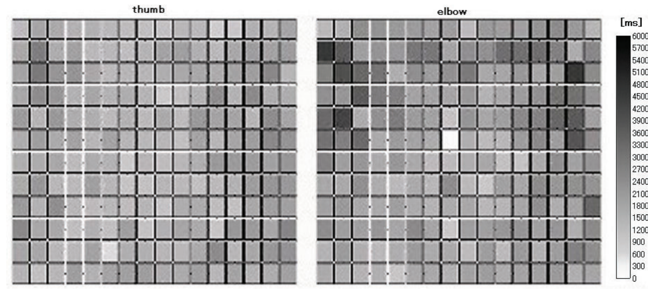

Figure 5: Elapsed time in pointing

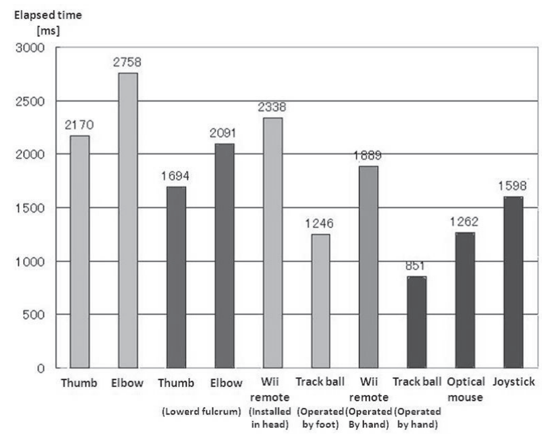

Figure 6: Comparison of pointing speeds

Comparison of accelerometer placement on thumbs and elbows

Figure 5 shows the evaluation results for the pointing time for each display position where the two accelerometers were attached to both thumbs or both elbows. When they were attached it to the thumbs (the left figure), faster input was achieved because this is intuitive. However, when attached to the elbows, the input speed was slower. The reason for this is the gesture of 'raising elbows' is not that familiar in daily life.

Comparison with existing devices

We conducted the same experiment with existing devices, to compare them with the accelerometers we used in our new method. The devices were an optical mouse manufactured by Hitachi Wearable Information Appliance (WIA), a handy track ball (FDM-G51) made by Beige, a joystick made by Victor, and a Wii remote controller made by Nintendo. Moreover, we carried out an experiment with the Wii remote controller strapped to the head, and a large track ball operated by foot as examples of hands-free devices. Figure 6 is a bar chart showing the results of the experiment. According to the detailed comprison, there is little difference in the pointing speed from the position on the screen between these existing devices. Therefore, we have shown the average pointing speed for each device.

As a result, the pointing speed with the track ball, the optical mouse, and the joystick was higher than that with

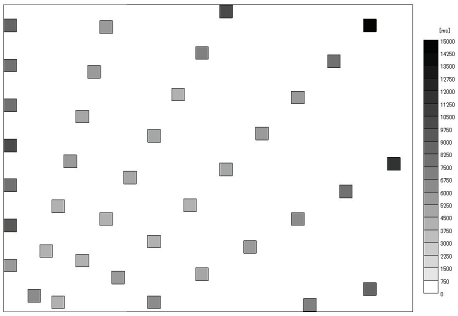

Figure 7: proposed method (1)

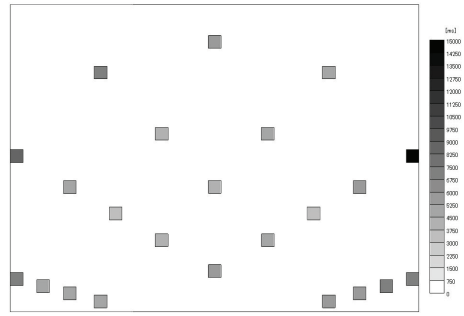

Figure 9: proposed method (2)

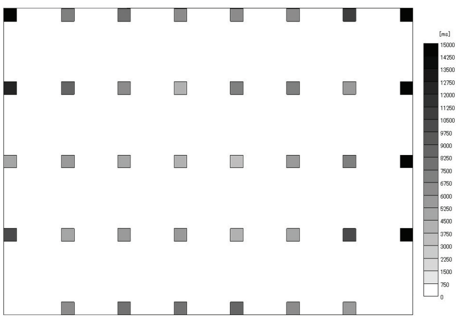

Figure 8: XY coordinates (1)

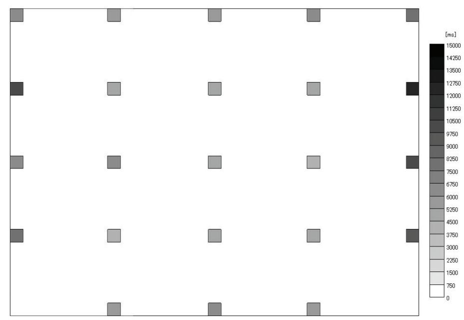

Figure 10: XY coordinates (2)

the proposed method. However, they required subjects to hold the device making hands-free operation impossible. Although the method of operating the track ball by foot was sufficiently compared with the other methods, there was a problem with the track ball being fixed to the ground making it impossible for subjects to move locations while using it. The proposed method, where the thumbs and lowered fulcrum were, used achieved almost the same speed as that of the joystick. Since the new approach achieved hands-free operation and there where no environmental restrictions, it had sufficient input speed although it was still slower than the other devices. Moreover, compared with the Wii remote controller, the hand-operated controller had sufficient input speed. However, it encountered the same problem as the other methods in that the device had to be held. However, hands-free controller achieved a worse speed than that with our method. This means that the pointing speed with our approach is faster than that with the other hands-free methods.

4.2 Evaluation of object allocation

To find how effective our method of object allocation was, we conducted the same kind of evaluation as that described in Section 4.1 using the new pointing method with the methods of arranging icons. The target was 32×32 pixels, which is the size of normal icons, and the screen resolution of the PC was 1024×768 pixels. Note that the target was smaller than that in the previous experiment because we used the actual icon size on a conventional desktop. The target in the experiment appeared at random positions chosen from positions candidate used for assigning icons in each method.

We carried out the experiment with four allocation algorithms, two of these were the radiation (1) and intersection (2) methods described in Section 3.4 and the other two were conventional arrangements along the X and Y axes with the same number of targets as that in methods (1) and (2). Each of the ten subjects clicked the target 30 times for both allocation methods. There were ten subjects.

Figure 7–10 show the results for pointing time for all positions of the target, and Figure 11 indicates the average pointing speed for both methods. Although the pointing speed for the edge of the screen is slower, this influence is reduced more with the proposed method than that with the XY coordinate methods. The average pointing speed with both methods (1) and (2) is faster than that with the XY coordinate methods.

The major reason the speed improved with our methods was that they achieved intuitive operation. The subjects achieved intuitive operation trough the associations between the lines with the virtual line of the target alignments. In addition, when subjects pointed at target on the same line on which the last pointed target had appeared, easy and accurate operation was accomplished because they only needed to move one line to select the target. This is one reason the input speed with method (2) was faster than that with method (1). Moreover, since the degree of improvement with method (2) was larger than that with method (2), we found that the new method we proppose is effective when there are numerous targets. From these results, we confirmed that the new method could effectively control the arrangements of objects being pointed at.

5. CONCLUSION

We proposed method of new where pointing with two ac-

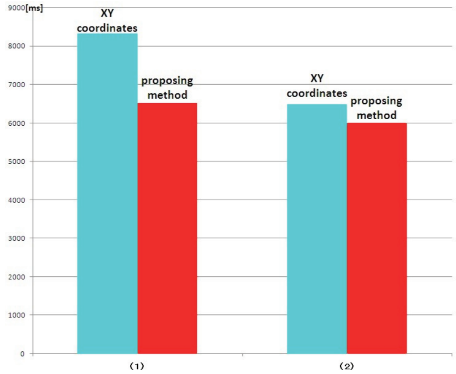

Figure 11: average time for pointing

celerometers for wearable computing where the pointer is located at the intersection of two straight lines that move in synchronization with the user's movements. In addition, we wrote a program that automatically arranges the icons on the desktop, organizing them in a way that is appropriate for the new methods of pointing. The methods are intuitive and comprehensive because they can be operated by easy input of gestures. They are also easy to operate while doing other daily work because the devices are small, can be installed on the body, and targets can be easily captured. As a result, they are suitable for use in wearable computing.

Pointing in the evaluation experiment, was fastest when accelerometers were installed on thumbs. The pointing speed slowed due to the position on the screen because it used the angle. However, by keeping away a straight line fulcrum from the pointing screen, the problem was solved, and the pointing speed was further improved. The new method records the pointing speed that equals a part of an existing device that has it in the hand though it is hands-free, effectiveness was able to be shown for the method with two accelerometers. As a result of the experiment on object allocation, more intuitive operation was possible compared with the past XY coordinates, and faster pointing was achieved.

In future research, it is necessary to improve the pointing accuracy further by introducing a new means of acquiring data with the accelerometers and drawing straight lines. Moreover, not only should the icons on the desktop be moved, but it is necessary to add improvements where the icons in folders are dynamically moved and menu structures other than icon arrangements are changed, which would be appropriate for the proposed method. In addition, it is enumerated to make it to the system that corresponds to the operation of a digital music player and special software of a web browser etc.

Acknowledgements

This research was supported in part by Grant-in-Aid for Scientific Research on Priority Areas "Methods for Handling Information using Wearable I/O Devices for Information Explosion" numbered 19024056, from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

6. REFERENCES

[1] G. Blasko and S. Feiner. Evaluation of an eyes-free cursorless numeric entry system for wearable computers. *Proceeding of The Tenth International Symposium on Wearable Computers (ISWC2006)*, October 2006.

[2] BuffaloTechnology. Bomu-w24a. http://buffalo.jp/products/catalog/item/b/bomu-w24a/.

[3] P. de la Hamette and G. Troster. Fingermouse - architecture of an asic-based mobile stereovision smart camera. *Proceeding of The Tenth International Symposium on Wearable Computers (ISWC2006)*, October 2006.

[4] E. Foxin and M. Harrington. Weartrack:a self-referenced head and tracker for wearable computers and portable vr. *Proceeding of The Forth International Symposium on Wearable Computers (ISWC2000)*, pages 155–162, 2000.

[5] Gyration. Go 2.4 optical air mouse. http://www.gyration.com/.

[6] T. Huynh and B. Schiele. Towards less supervision in activity recognition from wearable sensors. *Proceeding of The Tenth International Symposium on Wearable Computers (ISWC2006)*, October 2006.

[7] K. Kameyama and M. Yoshida. Wearable computer system using a finger-mounted device. *Information Processing Society of Japan (IPSJ)*, 99(9):13–15, 1999. (in Japanese).

[8] T. Kurata, M. Kourogi, T. Kato, T. Okuma, and K. Sakaue. The handmouse and its applications: Color- and contour-based hand detection and tracking. *ITE Technical Report, VIS2001-103*, 25(85):47–52, 2001. (in Japanese).

[9] Magitek.com. Magi-mouse. http://www.magitek.com/.

[10] NaturalPoint. Trackir. http://www.naturalpoint.com/trackir/.

[11] OriginInstrumentsCorporation. Headmouse. http://origin.com/access/headmouse/index.htm.

[12] H. Sakaguchi, H. Nonaka, and M. Kurihara. Fingertip pointing device optimized for mobile use. *The Hokkaido Chapters of the Institutes of Electrical and Information Engineers, Japan*, October 2004. (in Japanese).

[13] P. Siewiorek. New frontiers of application design. *it Communications of The ACM*, 45(12):79–82, 2002.

[14] T. Starner, J. Auxier, D. Ashbrook, and M. Gandy. A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. *Proceeding of The Fourth International Symposium on Wearable Computers (ISWC2000)*, pages 87–94, 2000.

[15] TobiiTechnology. Eye-tracking. http://www.tobii.com/.

[16] K. Tsukada. Ubi-finger : Gesture input device for mobile use. *Proceeding of 5th Asia Pacific Conference on Computer Human Interaction (APCHI '02)*, pages 388–400, 2002.

[17] J. E. Zucco, B. H. Thomas, and K. Grimmer. Evaluation of four wearable computer pointing devices for drag and drop tasks when stationary and walking. *Proceeding of The Tenth International Symposium on Wearable Computers (ISWC2006)*, October 2006.