uTrack: 3D Input Using Two Magnetic Sensors

Ke-Yu Chen[1], Kent Lyons[2], Sean White[2], Shwetak Patel[1]

University of Washington[1]
Seattle, WA (USA)
{kychen, shwetak}@uw.edu

Nokia[2]
Sunnyvale, CA (USA)
{kent.lyons, sean.white}@nokia.com

ABSTRACT

While much progress has been made in wearable computing in recent years, input techniques remain a key challenge. In this paper, we introduce *uTrack*, a technique to convert the thumb and fingers into a 3D input system using magnetic field (MF) sensing. A user wears a pair of magnetometers on the back of their fingers and a permanent magnet affixed to the back of the thumb. By moving the thumb across the fingers, we obtain a continuous input stream that can be used for 3D pointing. Specifically, our novel algorithm calculates the magnet's 3D position and tilt angle directly from the sensor readings. We evaluated uTrack as an input device, showing an average tracking accuracy of 4.84 mm in 3D space – sufficient for subtle interaction. We also demonstrate a real-time prototype and example applications allowing users to interact with the computer using 3D finger input.

ACM Classification: H.5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

Keywords: Finger; interaction; magnetic-field; MF; sensing; 3D; pointing

INTRODUCTION

Wearable devices such as Google Glass[1] extend user experiences and increase interaction richness. While these types of devices offer a unique visual experience, input is a challenge. The user sees a virtual image in 3D space so there is nothing physical to touch or manipulate for input. Traditional input modalities are not appropriate in this case because they are mainly designed for 2D interactions[2,3]. Therefore, there is a strong need for a new, mobile, 3D input device. Such pointing devices need to be low-cost, low power, and easily accessible [3].

Commercial 3D pointing devices have been broadly used in recreational or medical systems[4,5]. However, these tools are designed for desktop computers and cannot be instrumented on the human body easily for wearable applications. To overcome this problem, researchers have leveraged acoustic characteristics to use human skin for mobile, always-existing input interfaces [5,9]. Unfortunately, these acoustic-based methods only detect discrete 2D fingertip events and may become brittle in noisy environments.

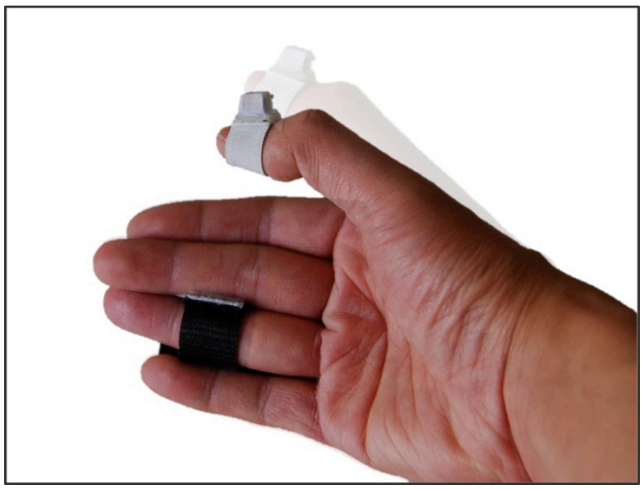

Figure 1. uTrack enables real-time 3D input from the thumb and fingers using one permanent magnet (thumb) and two magnetometers (ring finger).

In order to achieve truly 3D continuous cursor control, others explored computer vision techniques for finger-level gesture detection [10,11], including commercial products[6]. Vision-based techniques suffer from occlusion problems and are generally sensitive to lighting conditions, limiting their use in mobile contexts. To resolve these issues, some traditional approaches have used inertial sensors for cheap, occlusion-free arm gesture recognitions [14] or in-air writing [1]. While impressive, this previous work was either limited to 2D pointing or requires frequent recalibrations to resolve drifting errors.

To this end, we present *uTrack*, a pointing technique that enables 3D interactions for wearable devices. uTrack turns the user's thumb and finger into an input device using magnetic field (MF) sensing (see Figure 1). In particular, our algorithm calculates the magnet's x/y/z position plus an axis of orientation from the magnetic field vectors. Our system leaves the pads of the user's fingers free, and with only two

UIST'13, October 8–11, 2013, St. Andrews, United Kingdom.

http://dx.doi.org/10.1145/2501988.2502035.

[1]Google Glass: http://www.google.com/glass
[2]HDE Finger Mouse: http://www.shophde.com
[3]Mycestro: http://www.inn-dev.com
[4]Novint: http://www.novint.com
[5]Phantom Omni: http://www.sensable.com
[6]Leap motion: https://www.leapmotion.com

attachment points needed on the back or sides of the fingers, there is flexibility in designing form factors based on this technology.

Specifically, the contributions of this paper are:

1. A novel, low-cost technique for turning the thumb into a 3D input device for wearable systems.
2. An algorithm that derives the magnet's 3D position and tilt angle from sensor readings with minimal instrumentation to the user.
3. An analysis demonstrating tracking accuracy across 3D space within a volume of 70 mm (W) x 30 mm (D) x 60 mm (H) and a study showing the use of a thumb for pointing with this technology.

RELATED WORK

There are various devices that can be used for pointing with wearable computers. Zucco et al.'s studies compare a variety of prop-based pointing techniques including a trackball, cursor key physical buttons, gyroscopic mouse, and touchpad [18]. These each require either two-handed use or a device held in the hand for operation. In contrast, uTrack instruments the hand and uses novel sensing to enable the interaction.

Next, there is a set of techniques that utilize on-body sensing to obtain input. Researchers have shown that the distinct sounds generated by the movements of the body can be used for gesture-based interactions [5]. Harrison et al. used skin as an input surface and identified the locations of finger taps by analyzing the bio-acoustic sounds propagating through the human body [9]. Other techniques have leveraged electromyography (EMG) for sensing hand movements [16]. A common issue with all of these techniques is that they rely upon machine learning techniques to recognize and classify the underlying data sensed from the user. As such they fundamentally offer discrete input, which is often mapped to symbolic actions as opposed to a continuous signal that can be used for direct manipulation.

Others have explored optical or computer vision techniques for hand input. For example, infrared (IR) proximity sensors have been used in various ways to obtain input [4]. An IR camera was mounted in a pendant form to image the hands with the Gesture Pendant [17]. Kim et al. put an IR camera on the wrist to see the hands [11] and Harrison et al. used a depth camera mounted on the shoulder [10]. While these methods enable accurate, continuous 2D or 3D positioning, they all require a line-of-sight view of the hands, which limits the applicability of these techniques. For instance, one could not use these techniques with a hand in a pocket.

Magnetic field (MF) sensing also has a long history in tracking human body dynamics and computer input. Polhemus uses MF sensing to track objects with 6 degrees of freedom (DOF) [15]. In this work, Raab et al. derived a series of transformations that convert the non-linear magnetic field equations into a set of linear equations that can be directly solved. However, this method requires *active* magnetic sources (*i.e.*, three powered, orthogonal electromagnetic fields). The need for active sources limits the mobility of this technology since users have to stay in the range of the signal sources, which are physically large to generate fields that can be sensed at a distance.

Some systems use arrays of magnetic elements to obtain the 2D or 3D position of a tracked magnet. For example, Wacom tablets use an array of coils under the surface and Liang et al. use an array of 192 Hall effect sensors [13]. Unfortunately, these techniques require a 2D sensor array with a size that corresponds to the sensed area and thus have limited applicability to wearable uses.

Some research has used permanent magnets worn on the body, creating a magnetic field that can be sensed for input. Abracadabra uses a magnet attached to a finger and tracks the radial position of the finger with respect to the watch face [8]. Nenya uses a magnetic ring and tracks rotation of the ring about the finger to obtain a 1D input to a watch [2].

In a sequence of work, Han et al. showed that it is possible to calculate a magnet's 2D position using a pair of magnetometers if one imposes some constraints on movement [6]. In particular, they decompose the sensed magnetic field into two different eigenspaces and derive a direct relationship between the sensed data and the magnet's x and y positions. A key limitation of this work is that the magnet and the sensors must remain axis aligned in order for the assumptions to hold [7].

uTrack is conceptually similar to Han et al.'s work [6, 7]; however, we derive a completely new algorithm to remove the fixed orientation limitation in [6] and extend the sensing capability. By using one permanent magnet and only 2 magnetometers, uTrack is able to track the magnet's x/y/z position plus an axis of rotation. We focus on the configuration shown in Figure 1 where the magnet is worn on the back of the thumb and the magnetometers are on the back of the ring finger; however, other geometries are also possible with this system.

MAGNETIC FIELDS AND SPACE TRANSFORMATION

In this section, we first introduce background knowledge on magnetic field sensing in 2D space and then detail how we convert our 3D sensor readings into this 2D domain.

We can completely describe the magnetic field emanating from a fixed magnet in 3D space using only 2 degrees of freedom. To do this, we define a plane called the *magnetic field space*, in which the sensor's center is at the origin, and its x-axis is oriented in the direction of the magnet's north pole. Any point in magnetic field space can be represented in (r, θ) polar form, where r is the distance between the magnet and the sensor, and θ is the angle between the sensor and the magnet's north pole. This is a two dimensional quantity as the magnetic field is symmetric about the magnetic pole. The magnetic field vector $\mathbf{H}$ can therefore be decomposed into its tangential and radial components, $\mathbf{H_r}$ and $\mathbf{H_\theta}$, as shown in Figure 2 and defined in below:

$$\boldsymbol{H_r} = M \cos\theta / 2\pi r^3 \quad (1)$$

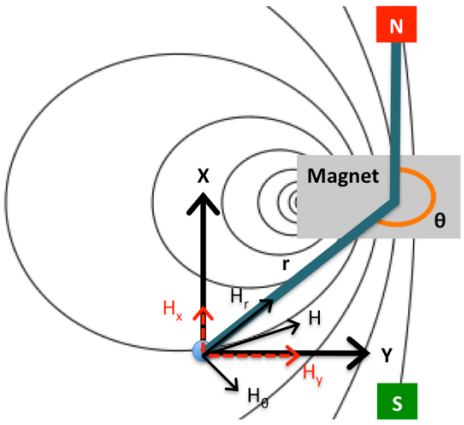

Figure 2. The magnetic field and the polar coordinate system. By decomposing the sensor vector **H** into two different eigenspaces, x and y positions of the magnet can be derived from H_x and H_y.

$$\boldsymbol{H_\theta} = M \sin\boldsymbol{\theta} / 4\pi \boldsymbol{r^3} \quad (2)$$

where M denotes the magnetic moment [12]. These two equations describe the magnetic field anywhere on the plane defined by the magnetic field space.

The two orthogonal vectors $\mathbf{H_r}$ and $\mathbf{H_\theta}$ form the basis of this space. **H** can also be decomposed into another basis, $\mathbf{H_x}$ and $\mathbf{H_y}$ (Figure 2) which produces a Cartesian coordinate system with the sensor at the origin and the directions of $\mathbf{H_x}$ and $\mathbf{H_y}$ parallel to the x and y-axis of the sensor. By using similar approaches to [6], the sensor-magnet relationship can be derived as:

$$H_x = H_r \cos\theta - H_\theta \sin\theta = \frac{K[2(x_m)^2 - (y_m)^2]}{r^5} \quad (3)$$

$$H_y = H_r \sin\theta + H_\theta \cos\theta = \frac{K(3x_m y_m)}{r^5} \quad (4)$$

$$K = \tfrac{M}{4\pi},\ r = \sqrt{(x_m)^2 + (y_m)^2},\ \sin\theta = \tfrac{-y_m}{r},\ \cos\theta = \tfrac{-x_m}{r}$$

where x_m and y_m denote the magnet's Cartesian position in the magnetic field space. For a given vector pair $\mathbf{H_x}$ and $\mathbf{H_y}$, the magnet's positions x_m and y_m can be calculated from the nonlinear equations (3) and (4).

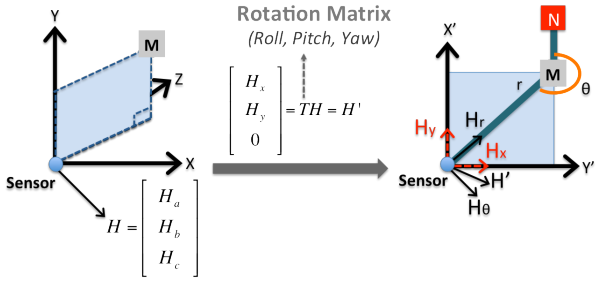

Figure 3. The incident plane of MF lines (blue rectangle) and space transformation. The rotation matrix **T** transforms the magnetic vector **H** from sensor space (left) into magnetic field space (right).

The relationship between the magnet and sensors can be derived through equations (1) to (4) *only if* the incident plane of magnetic field lines is known. The incident plane is the 2D space in which $\mathbf{H_r}$ and $\mathbf{H_\theta}$ are defined (*i.e.*, the magnetic field space). So if the magnet and sensor axes are aligned, a 3D magnetometer's sensor readings correspond directly to $[H_x, H_y, 0]^T$. This need for alignment is the fundamental constraint in Han's work, which requires the depth and orientation of magnet be held constant, and hence enables only 2D tracking [7].

Arbitrary Rotations

uTrack overcomes these limitations and allows the magnet to move and to rotate through any position in the 3D space. The key insight to make our system work is a space transformation between the 3D *sensor space* and the corresponding magnetic field space.

Figure 3 shows the incident plane and space transformation from the sensor space to magnetic field space. The transformation is achieved by the multiplication of sensor vector **H** with an unknown rotation matrix **T**. In 3D space, **T** can be defined with three Euler angles *R* (Roll), *P* (Pitch) and *Y* (Yaw), which are the rotation angles along sensor's x-, y- and z-axis respectively. Mathematically, this is denoted as below:

$$\begin{bmatrix} H_x \\ H_y \\ 0 \end{bmatrix} = TH = T_R T_P T_Y H \quad (5)$$

Once H is converted to magnetic field space, the system can then solve x_m and y_m by applying H_x and H_y to equations (3) and (4). Finally, we need to reverse the transformation back to the original 3D sensor space:

$$\begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} = T^{-1} \begin{bmatrix} x_m \\ y_m \\ 0 \end{bmatrix} = T_Y^{-1} T_P^{-1} T_R^{-1} \begin{bmatrix} x_m \\ y_m \\ 0 \end{bmatrix} \quad (6)$$

where x_s, y_s and z_s denote the magnet's x, y and *z* position in the sensor space. To find a solution, we must therefore find the unknown transformation matrix **T**. We will detail our searching algorithm in the next section.

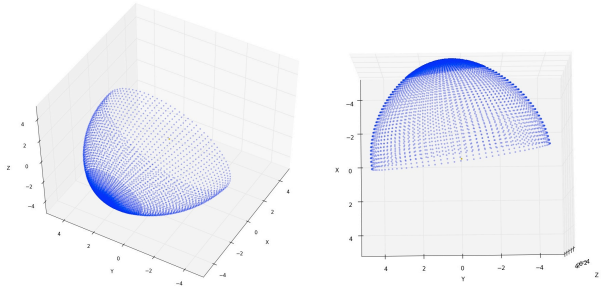

Figure 4. The search space for a given sensor vector **H** from the exhaustive search analysis. Each blue spot represents one combination of P, R and Y and its corresponding solution in the sensor space.

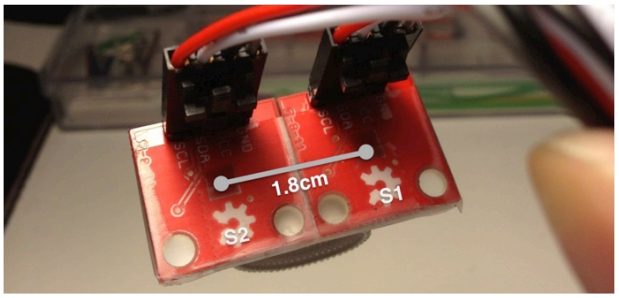

Figure 5. Using the second sensor. The aligned x axes of two sensors and the constant distance between them (1.8cm) are used in eliminating the ambiguity problem.

Understanding the Solution Space

To better understand the solution space, we performed an exhaustive search for a given sensor vector **H**. This analysis visits a subsampling of all possible combinations of R, P, and Y which satisfy the constraint of converting **H** to the vector $[H_x, H_y, 0]^T$.

Figure 4 shows the result of this exhaustive search for a single sensor reading. Each blue point represents a valid combination of R, P and Y and its corresponding solution in the sensor space. The banding is a result from our subsampling and the actual solution space would be a solid surface. This half-egg shape indicates the *ambiguity* in locating the magnet's 3D positions using only one sensor. For a given **H**, there are an infinite number of valid solutions to our equations. Thus, a single sensor is not sufficient to uniquely locate the magnet's 3D position.

Using a Second Sensor

The ambiguity can be resolved by adding a second sensor to the system with a known fixed position relative to the first sensor. In particular, we align both sensors, *S1* and *S2* so that their axes remain parallel, but there is an arbitrary shift along the x-axis of 1.8 cm (Figure 5). The axis alignment and constant distance between sensors is the key factor to resolve the ambiguity and locate the unique solution from two sensor readings.

Using this information, we are now looking for the set of rotation matrices that convert both sensor vectors to their correct magnetic field space. However, we note that the rotation matrixes, **T** and **T**', for S1 and S2 are different, but related. The offset along the x-axis means S2's incident plane needs to be rotated with one additional angle α about the S2's z-axis (Figure 6), that is, $\mathbf{T}' = \mathbf{T}\mathbf{T}_\alpha$.

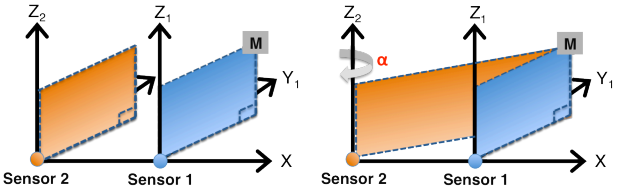

Figure 6. Magnetic field spaces for both sensors. If the same **T** is applied to both sensors, their spaces become two parallel planes, which is incorrect (left). Sensor 2 requires an additional rotation angle α to rotate it into the correct magnetic field space (right).

ALGORITHM

Figure 7 shows the system flow of uTrack. Given the non-linear nature of the system (*i.e.*, equation 3 and 4), we decided to use an exhaustive approach. The algorithm iteratively searches the three unknown angles R, P and Y for the rotation matrix **T** that eliminates the z component of S1 and converts **H** into the magnetic field space. Additionally, the angle α is used to rotate from the S1 magnetic field space to the corresponding field space of the second sensor, S2.

Now we label sensor readings from S1 and S2 as $\mathbf{H_1}$ and $\mathbf{H_2}$. The algorithm proceeds as below:

1. The system iteratively searches the three angles of **T** that converts $\mathbf{H_1}$ to a 2D vector $\mathbf{H_{m1}}$ (Eq. 5).
2. Once a candidate **T** is found, the system searches across α to find valid **T'** candidates which converts the S2 reading $\mathbf{H_2}$ into $\mathbf{H_{m2}}$.

   At this point, we have a set of rotations that satisfy our magnetic field readings. However, these candidates do not yet take into account the known physical displacement between the two sensors.
3. The two transformed magnetic field values ($\mathbf{H_{m1}}$ and $\mathbf{H_{m2}}$) are turned into spatial positions in the magnetic field space using a non-linear function solver (Eq. 3 and 4).
4. The solutions, $[x_{m1}, y_{m1}]^T$ and $[x_{m2}, y_{m2}]^T$, are then transformed back into the sensor space (Eq. 6). We denote these transformed solutions $\mathbf{V_1}$ and $\mathbf{V_2}$ as $[x_{s1}, y_{s1}, z_{s1}]^T$ and $[x_{s2}, y_{s2}, z_{s2}]^T$.
5. $\mathbf{V_1}$ and $\mathbf{V_2}$ represent the magnet's 3D position in their *respective* sensor spaces (see Figure 8, left). Since the sensors' axes are aligned and the distance between sensors is known (1.8cm), a valid solution pair $\mathbf{V_1}$ and $\mathbf{V_2}$ *must* satisfy,

$$V_2 - V_1 = \begin{bmatrix} x_{s2} - x_{s1} \\ y_{s2} - y_{s1} \\ z_{s2} - z_{s1} \end{bmatrix} = \begin{bmatrix} 1.8 \\ 0 \\ 0 \end{bmatrix} \quad (7)$$

as shown in Figure 8 (right). This spatial constraint is the

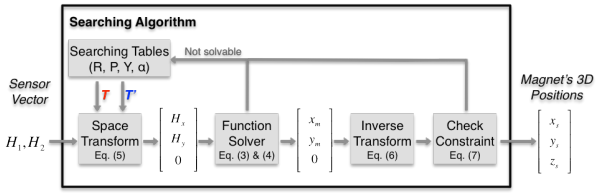

Figure 7. The searching algorithm finds three unknown angles R, P, Y and α, and converts **H** to the magnetic field space where the magnetics can apply. The magnet's 3D position is then calculated by the inverse transform.

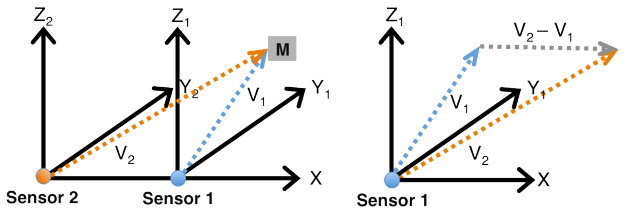

Figure 8. A valid solution pair (V_1, V_2) must satisfy the equation $V_2 - V_1 = [1.8, 0, 0]^T$. This key condition enables uTrack to identify a unique solution using only two sensor vectors.

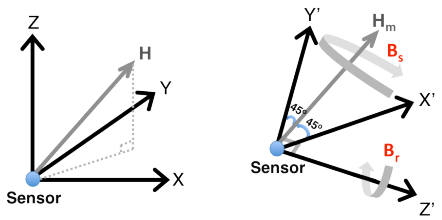

Figure 9. The orientation of a sensor before (left) and after (right) applying T_u to H. Our enhanced algorithm uses two angles B_s (spin) and B_r (rotate), which significantly reduces search complexity.

key element that enables uTrack to uniquely identify the magnet's 3D positions using *only* two sensors.

6. If at any point (1, 2, or 5) the system fails to pass the corresponding constraint, the system goes back to step 1 and continues until it finds a valid solution.

Optimization of the Searching Algorithm

In practice, the search through all angular permutations is not feasible, so we incorporated a small optimization based on the nature of the magnetic fields.

We first reduce the complexity of our algorithm by decreasing the number of angles to search for **T** from three to *two*. Intuitively, we apply a transformation to directly convert **H** to a 2D plane (i.e., making z component 0). At this point the system needs to search only two angles in this new 2D domain. To do this, we first convert **H** to its unit vector $\mathbf{H_u}$ and directly calculate the rotation matrix $\mathbf{T_u}$ that transforms $\mathbf{H_u}$ to a unit vector in the magnetic field space:

$$\begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ 0 \end{bmatrix} = T_u H_u \qquad (8)$$

Next we multiply **H** by $\mathbf{T_u}$, which converts **H** to a 2D vector (called $\mathbf{H_m}$) with equal x and y values. It is noted that H_m is on the X'-Y' plane (*i.e.,* $\mathbf{H_m}$ is perpendicular to Z'-axis) and has no z component. We therefore can search only two angles B_s (spin) and B_r (rotate), which are the angles that rotate the sensors along $\mathbf{H_m}$ and Z'-axis (see Figure 9).

To further improve the efficiency, we track the delta movement instead of searching the whole space for each new sensor reading. That is, we keep B_s, B_r, and α from the previous solutions and use it as the starting search point for the current sensor reading. To this end, we divide the searching range of B_s, B_r and α (*i.e.*, -π to π) into 90 steps. Moving outwards from 0, we add the candidate delta rotation angle to the last B_s and B_r. This corresponds to the magnet's delta movement from the previous point. Once a valid combination from S1 is found (*i.e.*, satisfying Eq. 5), the system searches for a valid α using the same delta approach. The system stops searching when a valid answer pair ($\mathbf{V_1}$, $\mathbf{V_2}$) is found or exceeds an arbitrary searching upper bound (*i.e.*, ±8 degrees).

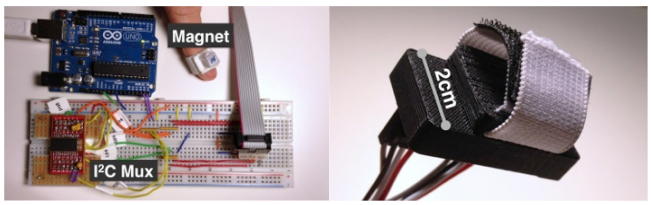

Figure 10. uTrack prototype (left) and the case with the two magnetometers (right). The data from two sensors are multiplexed and streams to Arduino.

EVALUATION OF TRACKING ACCURACY

Our first experiment is designed to understand the 3D pointing accuracy of the system with respect to ground truth. In the experiment, our uTrack prototype includes two 3-axis MEMs magnetometers (Honeywell's HMC5883), an I^2C multiplexer (NXP's PCA9527), and an Arduino Uno (Figure 10, left). The two magnetometers sample at 50Hz and are physically aligned along their x-axis as shown in Figure 5. Here we used a breakout module from Sparkfun for the magnetometers. A smaller custom board could be made to house both sensors. To ensure temporal consistency, the data from the two sensors is multiplexed through the I^2C mux and is streamed through the Arduino to the same computer. The magnetometers sit in a custom designed case to prevent unwanted movement or torqueing of the sensors (Figure 10, right). The computer for calculating the locations has a quad-core 2.8GHz CPU with 8GB of RAM running Mac OS X. The core algorithm was written in Python using the NumPy and SciPy libraries.

Experiment I: Pointing Accuracy

In the first experiment, we evaluate the accuracy of uTrack for obtaining the 3D position of the magnet. The ground truth for the magnet's 3D position was obtained using a PhaseSpace motion tracking system[7]. The PhaseSpace uses active LED markers to capture movements in real time and has 1 mm resolution. In this study, we utilized 6 cameras to capture movements of markers sampled at 480 Hz. The markers are attached on sensor 1 and the magnet.

In this experiment, the magnet was worn by the first author on the index finger; and, the sensors were affixed to the table. The researcher moved randomly in a 3D volume of ±80 mm (x-axis), ±60 mm (y-axis) and 0–80 mm (z-axis).

[7]PhaseSpace: http://www.phasespace.com

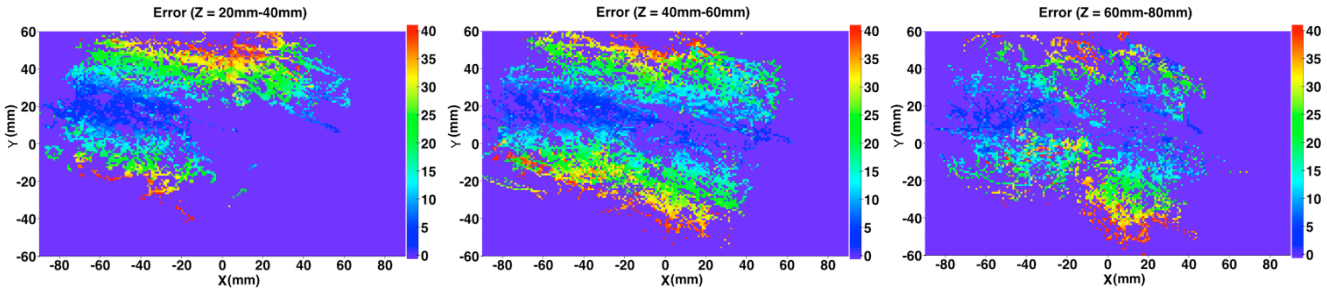

Figure 11. Tracking accuracy (unit: mm) of uTrack. The interactive space is divided into three 20 mm slices along the z-axis and shown in these three plots. The color indicates the amount of error in absolute position. The system is most accurate in the center of the volume, with a mean error of 4.82 mm.

This volume is much larger than what would be needed for our target mobile applications; however, we chose this relatively large volume to explore the limits of the system. The uTrack sensor data and 3D positions of the optical markers are saved on the same computer (*i.e.*, using the same clock and aligned in time) for off-line analysis. There were 39384 data points collected in total.

Results

Figure 11 shows Euclidean distance between the results of uTrack and the optical tracking results which we assume to be ground truth. The interaction space is divided into three 20 mm slices along z-axis to improve the visualization. In each slice, the mean of error for points with same x and y position is calculated.

For the entire volume of 38,400 mm^3, the mean Euclidean error is 24.54 mm (σ=9.17). This high overall error is caused by the environmental magnetic noise. In this study, we use a relatively strong magnet to overcome environmental interference such as the earth's magnetic field and surrounding electromagnetic fields. However, when the magnet moves away from the sensors, the data becomes noisier and therefore degrades the accuracy of the system. This can be seen in Figure 11 with increased error moving out from the center of the graphs. It is useful to note that the error in x-, y- and z-axis is 1.84 mm (σ=7.69), 2.72 mm (σ=5.51) and -13.80 mm (σ=19.04), respectively. This result indicates that the performance drop is mainly in the z-axis and the tracking on x-y plane remains quite accurate. Hence, despite the accuracy fluctuations in z-axis, uTrack is still capable of fine-grained controls on x-y plane and relatively coarse-grained controls on the z-axis.

We next explore the region where the points are within 10 mm of ground truth. In Figure 11, this region is depicted with the blue points and corresponds to a box of approximately 70 mm (W) x 30 mm (D) x 60 mm (H) (*i.e.*, the x-, y- and z-axes range from -70 mm to 0 mm, 5 mm to 35 mm and 20 mm to 80 mm, respectively). When we examine only this region, the error significantly reduces to a mean of 4.82 mm (σ=1.01). If the sensors are instrumented on the ring finger (Figure 12, left), this region approximately corresponds to the space that thumb can move around the fingers.

Besides the analysis of accuracy, we also evaluate the resolution of uTrack within the same volume of 10 mm accuracy. For this purpose, we calculate the Euclidean distance between two consecutive data points (*i.e.*, the relative movement) measured by the optical markers and the results of our system, and use their difference as the drifting error. The Euclidean mean error is 3.78 mm (σ=1.7), indicating $(25.4 \text{ mm}/3.78 \text{ mm})^3 = 303$ dpci (dots per cube inch).

APPLICATION TASK EVALUATION

To further verify the feasibility of using uTrack in real applications, we conducted a second study. Here we had participants conduct three tasks.

Experiment II: User Evaluation

For the first task, we evaluated the performance of using our system for absolute cursor control on a monitor. Here, we had participants move the magnet in the x-y plane (i.e., parallel to the table), and in x-z plane (i.e., the vertical plane approximately aligned with the monitor). The participants are asked to move the cursor along a 120-pixel square on a 24-inch, 1920x1080 LCD monitor. This square corresponds to a 20 mm square of physical movement (*i.e.*, 1 mm magnet movement corresponds to 6 pixels on the display). When the participant moves the magnet in either of the two target dimensions, they can see the cursor moving on the screen and adjust their path in real time. Movement in the third dimension was allowed, but discarded and not represented onscreen with visual feedback.

Tasks 2 and 3 were more open ended. We explored the ability of uTrack for slow, fine-grained manipulation (task 2) and fast, coarse-grained (task 3) movements. For these tasks

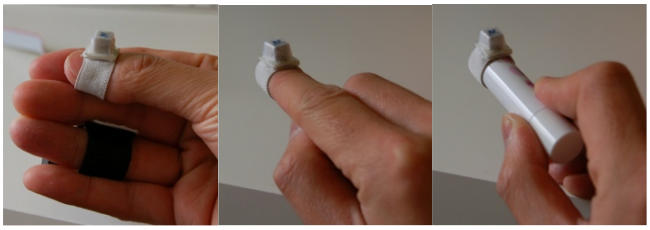

Figure 12. Three form factors using the thumb (left), the index finger (center), and a stylus (right) for 3D pointing.

	X-Y	X-Z	Avg.
Thumb	14.9	14.0	**14.5**
Finger	15.1	13.5	**14.3**
Stylus	14.6	14.2	**14.4**
Avg.	**14.9**	**13.9**	**14.4**

Table 1. Error in the cursor control task (in pixels).

participants manipulated a 3D model and played the Fruit Ninja game respectively. For these tasks, we only collected subjective feedback.

In experiment II, we explored three different physical configurations for uTrack (see Figure 12) in various scenarios. The first used the sensors on the ring finger and a magnet attached on the back of the *thumb* as the pointer. This is our original targeted one-handed pointing interaction. The second represents having two sensors in a device such as wristwatch and using the *index finger* of the opposite hand as a 3D pointer. Our third configuration would embed the two sensors in a mobile phone or tablet and use a *stylus* for 3D interactions. For scenarios two and three, instead of embedding the sensors into a device, we simulated the configuration by having participants use their non-pointing hand to hold the sensors in an appropriate configuration.

We recruited ten participants (6 males, 4 females) ranging in age from 22 to 33 years from a university. Eight participants have more than 10 years of experience with computers, while the other two participants reported having only an intermediate level of experiences with computers. Two participants were left-handed. One of the left-handed participants used their non-dominant hand for pointing and all other participants used their dominant hand. At the end of each session, the participants were asked to fill a questionnaire regarding their adaptation to these three input methods.

For all conditions, we smoothed the tracking path of the 3D points produced by our underlying algorithm by applying cosine interpolation and moving average (window size 10) on the tracking results.

Results

The results from the user evaluation for Task 1 are shown in Table 1. Because the system maps 6 pixels to 1 mm, the mean error of 14.4 pixels (σ=1.26) represents approximately 2.4 mm of movement error of the magnet (we excluded data from one participant as an outlier as the user had difficulty with the thumb condition). Since the participants move the finger or thumb in air without the benefit of a surface for support, this small error is likely due as much to the difficulty of the task as the underlying uTrack system. We also want to note that there was minimal training with this system and all participants started testing after only 30 to 60 seconds of practice.

All participants were able to successfully perform our other two open-ended tasks. We also asked participants for qualitative preferences for our different configurations. For Task 1, among three input methods, participants preferred the finger and stylus for cursor control. Participants indicated no preference between the three configurations for manipulating the 3D model and preferred the finger configuration for playing Fruit Ninja.

DISCUSSION AND FUTURE WORK

Our evaluation showed that uTrack enables 3D pointing with sufficient accuracy for real-time applications, and is perceived well by study participants. In this section, we discuss some opportunities for improving uTrack, the calculation of the magnet's tilt angle, and some future work.

There are several improvements we would like to explore with uTrack. First, our algorithm is relatively naïve in searching for the 4 unknown angles. For our prototype, this was sufficient to allow real-time performance using the computing power of a PC; however, a mobile solution would require a different implementation. It might be possible to derive some alternative linear formulations that allow for more efficient computation similar to [15].

Also, in order to reduce the search space, our current implementation utilizes deltas for tracking movements of the magnet. It requires a somewhat known starting point to seed the search. To resolve this, we constrain the magnet to stay in the first quadrant of the magnetic field space when the system is initially launched.

In addition to 3D positions, uTrack also tracks the tilt angle of magnet (4 degrees of freedom total). Here we define the tilt angle as the angle between the magnet's north pole and sensor's x-y plane. As described earlier, our algorithm searches for a rotation matrix **T** that converts **H** to the magnetic field space where x-axis is parallel to magnet's north pole. The direction vector of the magnet's north pole $\mathbf{V_N}$ can therefore be derived by:

$$V_N = T\begin{bmatrix}1\\0\\0\end{bmatrix} = \begin{bmatrix}V_x\\V_y\\V_z\end{bmatrix} \tag{9}$$

The tilt angle of magnet $\boldsymbol{\theta}_t$ can then be calculated by:

$$\theta_t = \arcsin\left(\frac{V_z}{\|V_N\|}\right) \tag{10}$$

Figure 13 illustrates some examples of angles that uTrack is able to detect. Evaluating and utilizing this additional degree of freedom would be an interesting area of future work. For

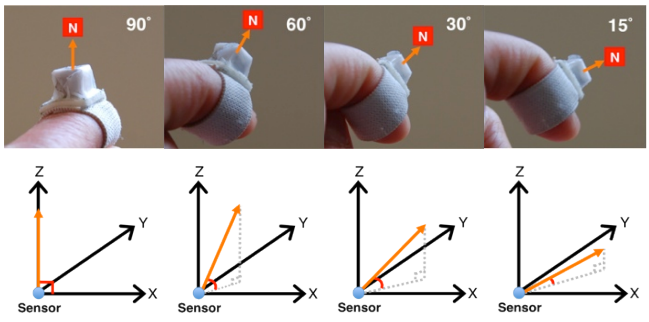

Figure 13. In addition to tracking 3D position, uTrack continuously tracks the direction of magnet's north pole to derive the magnet's tilt angle.

example, the tilt angle can possibly be used to enable "clicking", a proxy for pressure, or to switch the cursor from hover to click-and-drag mode.

Although the experiments in this paper were conducted while seated, we believe that uTrack will work equally well even when the user is in motion. To prove this, we informally conducted experiments in which a person utilized the system for 3D cursor control while walking around a building. Although the signal fluctuates more relative to the signal seen in a static location, the system appeared to be functional. We plan to further explore the ability to use uTrack in a noisy environment. An additional sensor may also be useful in these situations to help account for environmental magnetic field variations similar to Han et al. [6].

CONCLUSION

In this paper, we present a novel interaction technique using magnetic field sensing that turns the fingers and thumb into a 3D input device. Our evaluations show that for an active volume of 70 mm (W) x 30 mm (D) x 60 mm (H), we obtain an absolute positioning accuracy of 4.82 mm and 303 dpci. A user study showed participants were able to effectively use uTrack for several different pointing tasks.

This system requires only a single active module consisting of a *pair* of MEMs magnetometers that could be worn on a finger like a ring. We described how we derive the position of a permanent magnet attached to the back of the thumb. This configuration allows the user to move the thumb around fingers to provide a continuous, 3D input. As various forms of wearable computing become more commonplace, the need for a good mechanism for continuous pointing will become critical; uTrack provides such a solution.

ACKNOWLEDGEMENTS

We would like to thank Shigeyuki Seko for his help with our hardware and Daniel Ashbrook for his early feedback on magnetic field sensing. We would also acknowledge Vikash Kumar for assistance with the evaluation.

REFERENCES

1. C. Amma and T. Schultz. 2012. Airwriting: demonstrating mobile text input by 3D-space handwriting. In *Proc. of IUI* '12, pp.319-320.
2. D. Ashbrook, P. Baudisch, and S. White. 2011. Nenya: subtle and eyes-free mobile input with a magnetically-tracked finger ring. In *Proc. of CHI* '11, pp.2043-2046.
3. D. Ashbrook, J. Clawson, K. Lyons, N. Patel, and T. Starner. Quickdraw: The impact of mobility and on-body placement on device access time. In *Proc. of CHI*, 2008, pp.219-222.
4. A. Butler, S. Izadi, and S. Hodges. 2008. SideSight: multi-"touch" interaction around small devices. In *Proc. of UIST* '08, pp.201-204.
5. T. Deyle, S. Palinko, E.S. Poole and T. Starner, "Hambone: A Bio-Acoustic Gesture Interface," *Wearable Computers*, 2007, pp.3-10.
6. X. Han, H. Seki, Y. Kamiya, and M. Hikizu, Wearable handwriting input device using magnetic field. In *Proc. SICE'07*, pp.365-368.
7. X. Han, H. Seki, Y. Kamiya, and M. Hikizu, Wearable handwriting input device using magnetic field: 2nd report: Influence of misalignment of magnet and writing plane, Precision Engineering, July 2010, pp.37-43
8. C. Harrison and S. E. Hudson. 2009. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proc. of UIST* '09, pp.121-124.
9. C. Harrison, and D.M. Tan, Skinput: Appropriating the Body as an Input Surface. In *Proc. of CHI* 2010, pp.453-462
10. C. Harrison, H. Benko, A.D. Wilson, OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proc. of UIST* '11, pp.441-450
11. D. Kim, O. Hilliges, S. Izadi, A. D. Butler, J. Chen, I. Oikonomidis, and P. Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In *Proc. of UIST* '12, pp.167-176.
12. M.R. Kraichman, Handbook of Electromagnetic Propagation in Conducting Media, 2nd ed. Washington, D.C.; Head-quarters Naval Material Command, 1976.
13. R. Liang, K. Cheng, C. Su, C. Weng, B. Chen, and D. Yang, 2012. GaussSense: attachable stylus sensing using magnetic sensor grid, In *Proc. of UIST '12*, pp.319-326.
14. J Lementec and P. Bajcsy. Recognition of arm gestures using multiple orientation sensors: gesture classification. *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, 2004, pp.965-970
15. F.H. Raab, E.B. Blood, T.O. Steiner, and H.R. Jones, Magnetic Position and Orientation Tracking System, Aerospace and Electronic Systems, IEEE Transactions on, Sept. 1979, pp.709-718.
16. T. S. Saponas, D. S. Tan, D. Morris, R. Balakrishnan, Jim Turner, and J. A. Landay. 2009. Enabling always-available input with muscle-computer interfaces. In *Proc. of UIST* '09. ACM, New York, USA, pp.167-176.
17. T. Starner, J. Auxier, D. Ashbrook and M. Gandy (2000), The Gesture Pendant: A Self-illuminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring, In *Proc. of ISWC*'00, pp.87-94
18. J. E. Zucco, B. H. Thomas, K. Grimmer-Somers and A. Cockburn. 'A Comparison of Menu Configurations and Pointing Devices for use with Wearable Computers while Mobile and Stationary'. In Thirteenth International Symposium on Wearable Computers, Linz, Austria, September 2009. IEEE, pp.63-70