

Magnetic Input for Mobile Virtual Reality

Boris Smus

Google / smus@google.com

Christopher Riederer

Columbia University / mani@cs.columbia.edu

ABSTRACT

Modern smartphones can create compelling virtual reality (VR) experiences through the use of VR enclosures, devices that encase the phone and project stereoscopic renderings through lenses into the user's eyes. Since the touch screen in such designs is typically hidden inside an enclosure, the main interaction mechanism of the device is not accessible. We present a new magnetic input mechanism for mobile VR devices which is wireless, unpowered, inexpensive, provides physical feedback, requires no calibration, and works reliably on the majority of modern smartphones. This is the main input mechanism for Google Cardboard, of which there are over one million units. We show robust gesture recognition, at an accuracy of greater than 95% across smartphones and assess the capabilities, accuracy and limitations of our technique through a user study.

INTRODUCTION

When a smartphone is placed inside a VR enclosure (eg. Durovis Dive, FOV2GO and Google Cardboard), interaction with the touchscreen and hardware buttons becomes impossible. Alternative interaction modes such as voice recognition, and camera input suffers from high latency, high battery consumption, and low robustness, and requires favorable ambient conditions. External electronic devices are expensive and either require a pairing step or a wire.

Related work. The research community has developed several interfaces based on magnetic input, broadly falling into the class of Around-Device Interaction (ADI) [2]. One category of magnetic input devices, such as Abracadabra [3], Nanya [1] and MagiTact [5] and MagGetz [4] facilitate interactions near the device, through a worn or a held magnetic object. This research has not been shown to work well with a variety of smartphones, and often requires a calibration step. For haptic feedback, extra mechanical components like springs are required. Our approach doesn't require calibration, works reliably on the majority of smartphones, and doesn't require complex mechanical components.

Our key contributions are the following: (1) The design of a button with tactile feedback implemented using only a smartphone's magnetometer and affordable external magnets; (2) A reliable, calibration-free, efficient button-press detection algorithm agnostic to differences in phone model, magnetometer hardware, and operating system; (3) An evaluation of the input on real-world users and a broad deployment of over one million units.

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). *ISWC '15*, September 7–11, 2015, Osaka, Japan.
Copyright 2015 © ACM 978-1-4503-3578-2/15/09...\$15.00.
<http://dx.doi.org/10.1145/2802083.2808395>

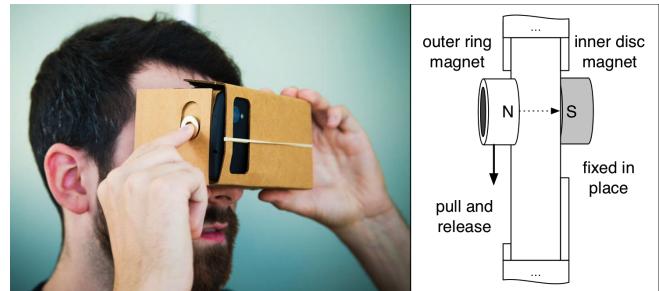


Figure 1: Using the Google Cardboard magnet input to interact with VR (left). A side view illustration of the input mechanism (right).

HARDWARE MECHANISM

Permanent magnets are inexpensive, robust, require no power to operate, and do not degrade over time. Due to their importance in navigation applications, high quality magnetometers are built into almost all modern smartphones.

Magnetometer sensor characteristics. Magnetometers sense the direction and magnitude of nearby magnetic fields. The sensor subsystem provides a stream of 3D vectors (in μT) in the phone's coordinate system. To convert a stream of magnetometer readings into magnetic north, the phone calibrates for the presence of nearby magnetic fields, seen as rapid changes in the stream of readings (see Figure 2).

Our input relies on a magnet on the outside of an enclosure, whose sudden movements are interpreted by the phone's software as a "button press". Care should be taken to disambiguate certain events from button presses: (1) rotation of the enclosure, which causes changes in the magnetic field due to the Earth, (2) rapid insertions and removals of the phone from the enclosure, and (3) calibration events.

Physical design. We created our input mode out of two magnets, one fixed and one constrained in its movements. Our enclosure's interaction area consists of three cardboard layers glued together as shown in Figure 1. A 3/4" diameter ferrous disc magnet with 1.5 lb pull is glued onto the middle layer and embedded into the inner layer (close to the phone) in a nook the size of the magnet. An outer magnet, a 3/4" outer diameter neodymium ring with 5 lb pull, is free to move along the middle layer of cardboard, constrained by a rail-like indentation cut into the outer layer of cardboard. Magnetic strengths were selected to produce sensor readings far stronger than those due to the Earth. The hole in the ring magnet invites the user's finger and the rail-like indentation suggests that the only meaningful action is to pull the magnet downward.

SOFTWARE AND INTERACTION

Unidirectional motion of the magnet appears similar to the non-interactions depicted in Figure 2. To reduce complexity, we chose a down-up motion of the magnet as the gesture.

Data collection. To test a wide variety of algorithms, we recorded magnetometer readings during typical VR use (684 true positives) and while performing the gesture with six pop-

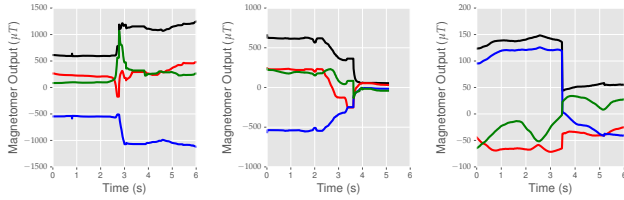


Figure 2: Sample magnetometer output for insertion into enclosure, removal from enclosure, and a calibration event. Red is X, green is Y, blue is Z, and black is the magnitude.

ular smartphone models. Plotting the sensor data recorded during this gesture depicts a standard pattern consisting of a spike in the signal followed by a return close to the original value. This is distinguishable from spikes in the signal due to non-interactions. Analyzing our collected data, we found that 95% of the gestures took place within 400ms.

Implementation and tradeoffs. We implement and evaluate two classes of sliding window gesture detection algorithms: pattern-based and state-machine-based. The *pattern-based* algorithm uses a gesture template averaged from previously recorded true positives. It calculates a similarity score between each window of values and the template, registering a press if the similarity is above a specified threshold. We also created a *state-machine-based* algorithm with three states: inactive, active, and firing. The state changes depending on whether the magnitude of the sensor vector exceeded or was below certain thresholds.

Algorithm evaluation. Calibration events, which we observed often due to increased movement [7], typical in mobile VR, often caused false negatives and positives in the pattern-based approach. Multi-scale correlation helped mitigate this problem, but required more processing, hindering graphics and head tracking performance and potentially leading to motion sickness [6]. Our state-machine-based algorithm was both more efficient and robust, generalizing well to many devices and properly handling calibration events:

1. The last vector in the window is subtracted from all previous values, mitigating variations between magnetometers.
2. We calculate the minimum over the first half of the window (min_1) and the maximum over the second half (max_2).
3. If $min_1 < T_{active}$, move to the *active* state.
4. If $max_2 > T_{firing}$, and you're in *active*, move to the *firing* state and we register a button press, else move back to the *inactive* state.
5. From the *firing* state, we move to the *refractory* state to prevent multiple detections for a single gesture.

EVALUATION

We conducted a ten-subject user study to evaluate the input's reliability and smartphone compatibility, and get their feedback. Each subject was given a Moto X 2014 phone and a Google Cardboard enclosure running a sample application, which prompted the subject to find a digital object placed in their VR surroundings, look directly at it, and activate the input. After the user understood the input gesture, the experimenter recorded when gestures were made using a Bluetooth keyboard synced to the smartphone. Both the magnetic detections and the keyboard events, along with the corresponding timestamps, were recorded on the test phone for later analy-

sis. Finally, subjects completed a qualitative survey and tested whether or not the input worked with their own phone.

Results. We compared the times of the smartphone-detected gestures and our manual recordings of when gestures took place. 208 of a total of 219 magnet pulls were correctly detected, with only one false positive throughout the entire session. This corresponds to a 0.9498 recall, 0.9954 precision, and 0.9720 F-Score.

Our subjective survey consisted of several questions answered on a 1-to-5 likert scale. Users reported that the magnet input worked consistently ($\mu=4.1$, $\sigma=0.7$), that the sample application responded quickly to magnet interactions ($\mu=4.3$, $\sigma=0.6$), and that the input was easy to use ($\mu=4.5$, $\sigma=0.5$). However, we found that some users did not understand how the input worked ($\mu=3.4$, $\sigma=0.7$), and that it wasn't necessarily clear how to use the input the first time they saw it ($\mu=4.1$, $\sigma=1.0$); indeed 2 of 10 subjects were unable to independently discover the pull-and-release action of the input. 8 of 10 subjects owned an Android phone, and the input worked with 5 of 6 of their phones. The non-working device, an HTC One M7, was determined to have a magnetometer located at the bottom of the device.

This input mechanism ships on over a million Google Cardboards. The app rating of 4.2 stars and consistently positive media reviews indicate that there is no widespread interaction problem. From user reviews on the Play Store, we conclude that a few device models are unable to detect the magnetic interaction. This problem can often be solved by flipping the phone around, bringing the magnetometer closer to the magnet.

CONCLUSION AND FUTURE WORK

We presented an input mechanism for VR enclosures that is wireless, unpowered, inexpensive, requires no calibration, and works reliably on most smartphones. We have shown that the input method works well and is pleasing to users. To help other developers create similar magnetic input methods, we have open-sourced the suite of tools that we built to create our input method, available at <https://github.com/dodger487/MIST>.

REFERENCES

1. Ashbrook, D., Baudisch, P., and White, S. Nanya: subtle and eyes-free mobile input with a magnetically-tracked finger ring. In *SIGCHI*, ACM (2011).
2. Butler, A., Izadi, S., and Hodges, S. Sidesight: multi-touch interaction around small devices. In *UIST*, ACM (2008).
3. Harrison, C., and Hudson, S. E. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *UIST*, ACM (2009).
4. Hwang, S., Ahn, M., and Wahn, K.-y. Maggetz: customizable passive tangible controllers on and around conventional mobile devices. In *UIST*, ACM (2013).
5. Ketabdar, H., Yüksel, K. A., and Roshandel, M. Magitact: interaction with mobile devices based on compass (magnetic) sensor. In *Intelligent User Interfaces*, ACM (2010).
6. Moss, J. D., and Muth, E. R. Characteristics of head-mounted displays and their effects on simulator sickness. *Human Factors: The Journal of the Human Factors and Ergonomics Society* (2011).
7. Ozyagcilar, T. Calibrating an ecompass in the presence of hard and soft-iron interference. *Freescale Semiconductor Ltd* (2012).