Demo: Implementation and Benchmark of Magnetic Tracking on Mobile Platforms

Zhenyu Chen
Shanghai Jiao Tong University
Shanghai, China
zhenyu.chen@sjtu.edu.cn

Jike Wang
Shanghai Jiao Tong University
Shanghai, China
jikewang@sjtu.edu.cn

Dongyao Chen*
Shanghai Jiao Tong University
Shanghai, China
chendy@sjtu.edu.cn

ABSTRACT

Magnetic sensing is a promising approach for enabling high-accuracy and fine-grained tracking. Recent research has highlighted the superior performance of magnetic tracking in various subtle motion sensing tasks, such as hand and joint tracking. These tasks typically rely on computational resources available on laptops. However, the feasibility of migrating magnetic tracking techniques to mobile platforms, such as smartphones with limited computing power and energy capabilities, remains uncertain. This demo demonstrates the successful deployment of magnetic tracking techniques on commodity smartphones. Our implementation is benchmarked on a Commercial Off-The-Shelf (COTS) Android smartphone. Empirical studies indicate that our implementation on mobile platforms is both energy-efficient and highly effective.

CCS CONCEPTS

• Hardware → Sensor devices and platforms; **• Human-centered computing → Empirical studies in ubiquitous and mobile computing**.

KEYWORDS

Magnetic tracking, Mobile computing, Ubiquitous computing

ACM Reference Format:
Zhenyu Chen, Jike Wang, and Dongyao Chen*. 2024. Demo: Implementation and Benchmark of Magnetic Tracking on Mobile Platforms. In *AdaAIoTSys 2024 - Workshop on Adaptive AIoT Systems (AdaAIoTSys '24), June 3–7, 2024, Minato-ku, Tokyo, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3662007.3663885

1 INTRODUCTION

Magnetic sensing has made significant strides in a wide range of application scenarios, including hand tracking [4], joint tracking [3], and magnetic road markings [6]. In comparison to existing mainstream sensing techniques such as cameras and IMUs, magnetic sensing offers unique advantages.

- **Occlusion-free:** Unlike camera-based methods, magnetic sensing is immune to occlusion issues.
- **Drift-free:** IMUs suffer from accumulative errors due to drift, which limits their effectiveness. In contrast, magnetic sensing is drift-free, enabling fast motion capture.
- **Low cost:** Magnetic sensing only necessitates COTS magnetometers and miniature magnets, resulting in significantly reduced expenses.

One can achieve magnetic tracking with three key modules: the magnetic front end, sensing array, and computing model. Specifically, a low-cost magnetic object, such as a magnetic ring, can serve as the magnetic front end. The sensing array typically comprises several magnetometers that meet the application requirements, along with a communication module (e.g., BLE) to transmit sensor data to the computing unit. The computing model can be utilized to solve the posture of the magnetic front end.

With the rapid advancement of ubiquitous computing, there is a growing demand for deploying complex computational models on edge devices with limited computing and energy resources. However, existing magnetic tracking systems are usually deployed on laptops which limits their usability. Although [4] achieved a lightweight magnetic tracking pipeline on Raspberry Pi, it is not clear how the system would perform on commodity smartphones. Hence, we pose the question: "*Can we integrate magnetic tracking techniques into COTS smartphones?*"

We leveraged the existing hardware and software architecture from MagX [4] to implement our system. In contrast to MagX, we integrated the magnetic tracking pipeline onto a COTS smartphone, specifically the Xiaomi 11 Pro. We also conducted a benchmark test to evaluate the energy consumption of our tracking pipeline on the smartphone. We open sourced the code base here: https://github.com/zychen-cs/MagAndroid

In summary, we make the following contributions:

- We migrated the implementation of the existing magnetic tracking pipeline to a mobile platform.
- We benchmarked the performance of magnetic tracking on COTS Android devices.

2 SYSTEM DESIGN

As shown in Fig. 1, the magnetic tracking pipelines on mobile platforms contain four steps: *Configuration*, *Scan the targeted BLE*, *Read sensor/Calibration*, and *Tracking the magnet's pose*.

Configuration. Users provide the configuration parameters for the tracking system. Specifically, the number of magnetometers, the position of each magnetometer, and the magnetic moment and initial position of the magnets. Note that the configuration step is a one-time effort tailored to a specific sensing platform and magnet.

*Dongyao Chen is the corresponding author.

AdaAIoTSys '24, June 3–7, 2024, Minato-ku, Tokyo, Japan

ACM ISBN 979-8-4007-0664-6/24/06
https://doi.org/10.1145/3662007.3663885

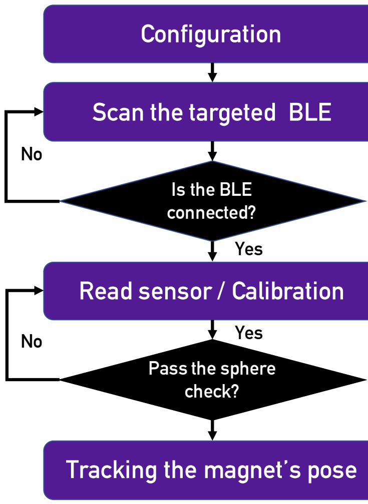

Figure 1: The tracking pipeline on the mobile platform.

Scan the targeted BLE. We have utilized BLE to transmit sensor data from the sensing platform to the mobile platform. Consequently, the app must establish a connection with the BLE module to continuously stream data from the sensing array. The challenges of scanning the targeted BLE are as follows:

- Based on the communication services provided by Bluetooth Low Energy (BLE), we have configured the "UART_TX_UUID" parameter to facilitate real-time data transmission. The difficulty lies in enabling the CCCD (Client Characteristic Configuration Descriptor) functionality of the nRF chip on the mobile phone side. This functionality is necessary to allow the mobile app to receive active notifications and indications when sensor data is updated.
- By default, the nRF chip transmits data in 20-byte packets when using Bluetooth in mobile phones. To support real-time data transmission of 96 bytes, which includes three-axis data from eight sensors, we tuned the "GATT_MAX_MTU_SIZE" to the maximum allowed by the mobile device.

Read sensor/calibration. The calibration step is essential for enhancing tracking accuracy. Users can employ the full-sphere calibration approach [5] to calibrate the sensing array. After completing the calibration, the profile will be saved on the mobile platform. Users can then initiate the tracking procedure. To maintain high tracking accuracy, the users may need to recalibrate their sensing array when the environmental magnetic field changes significantly. To address this, one can implement a calibration notification system that alerts users upon detecting abnormal changes in magnetic field data. We will incorporate this modification in our future work.

Tracking the magnet's posture. Based on the magnetic tracking algorithm of MagX [4], we utilized the dipole model to represent the magnet. Subsequently, we formulated three equations along the x, y, and z-axes of each sensor. We employed the Levenberg–Marquardt (LM) algorithm to solve these equations to compute the magnet's posture. To enhance computational efficiency, we used the Ceres Solver [1] to optimize the LM algorithm. However, integrating Ceres Solver into an Android project presented new challenges, including environment configuration and method call implementation.

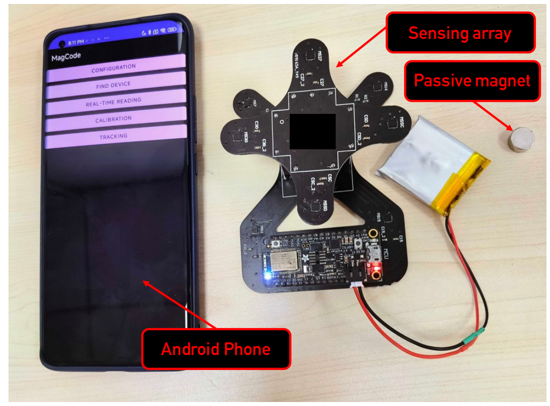

Figure 2: Experiment setup.

Specifically, environment configuration involved downloading the appropriate Android package. In our implementation, we bridged Kotlin code (i.e., the implementation language for Android platforms) with the Ceres Solver through Android JNI.

The final implementation of the magnetic tracking pipeline on a mobile platform is shown in Fig. 2. We will open-source our Android implementation on GitHub.

3 BENCHMARK

Now we analyze the performance of our implementation. We focus on its energy efficiency. We conducted an analysis of our implementation on a mobile phone using the Battery Historian tool [2]. This tool allows for inspecting battery-related usage on Android devices. Specifically, we installed the app on a Xiaomi 11 Pro running Android 12, which is equipped with a 5,000 mAh battery capacity. After running the app for ten minutes, including scanning for the BLE device and real-time tracking of a cylindrical passive magnet, the total battery consumption was only 0.32% (i.e., 16mAh). Based on the experimental results, the app can be used on a phone with a 5000, mAh battery for over 52 hours. We also tested the battery consumption of using a web browser for ten minutes, which was 0.55% (i.e., 27.5mAh). The result further demonstrates the energy-efficient feature of our implementation.

REFERENCES

[1] 2021. Ceres Solver. http://ceres-solver.org/.
[2] 2023. Analyze power use with Battery Historian. https://developer.android.com/topic/performance/power/battery-historian.
[3] Dongyao Chen, Qing Luo, Xiaomeng Chen, Xinbing Wang, and Chenghu Zhou. 2024. MagDot: Drift-free, Wearable Joint Angle Tracking at Low Cost. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 4, Article 150 (jan 2024), 25 pages. https://doi.org/10.1145/3631423
[4] Dongyao Chen, Mingke Wang, Chenxi He, Qing Luo, Yasha Iravantchi, Alanson Sample, Kang G. Shin, and Xinbing Wang. 2021. MagX: Wearable, Untethered Hands Tracking with Passive Magnets. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New Orleans, Louisiana) *(MobiCom '21)*. Association for Computing Machinery, New York, NY, USA, 269–282. https://doi.org/10.1145/3447993.3483260
[5] Timo Pylvänäinen. 2008. Automatic and adaptive calibration of 3D field sensors. *Applied Mathematical Modelling* 32, 4 (2008), 575–587.
[6] Jike Wang, Shanmu Wang, Yasha Iravantch, Mingke Wang, Alanson Sample, Kang Shin, Xinbing Wang, Chenghu Zhou, and Dongyao Chen. 2024. METRO: Magnetic Road Markings for All-weather, Smart Roads. In *Proceedings of the 21th ACM Conference on Embedded Networked Sensor Systems (SenSys '23)*. Association for Computing Machinery, Istanbul, Turkiye. https://doi.org/10.1145/3625687.3625809