

A Comprehensive Toolkit for WiFi Sensing: Decoding urban spaces

Juhyeon Park

2023-06-23

Table of contents

Preface

This book is a dedicated resource for anyone interested in leveraging affordable, commercially available sensors to measure non-motorized traffic in urban environments.

Quantifying non-motorized traffic—such as pedestrians and cyclists—plays a crucial role in urban studies. Understanding the flow and patterns of non-motorized traffic can inform urban planning strategies, enhance public safety, and contribute to the development of sustainable cities. Moreover, sensing technologies provide a robust and non-invasive method for capturing this vital information in real time, offering insights that traditional surveys or manual counts might miss.

The advent of the Internet-of-Things (IoT) has spurred a wave of urban sensing projects worldwide. Examples include the [Array of Things \(AoT\) in Chicago, USA](#) and [S-DoT in Seoul, Korea](#), which utilize a network of sensors to gather a wide range of data.

With the increasing accessibility of DIY technologies, individuals now have the opportunity to engage with their urban environment in new and innovative ways. These tools democratize the field of urban sensing, previously the domain of expert scientists, by equipping anyone with the interest to build their own sensors.

This book is designed for those interested in understanding and monitoring non-motorized traffic. We provide comprehensive guidance on building your own urban DIY sensors for this purpose. With hands-on advice, practical examples, and detailed breakthroughs, our aim is to empower you with the skills and knowledge necessary to contribute to the rapidly evolving field of urban sensing.

Scope of this document

This document demonstrates 1) how to build a smart sensor that detects pedestrians outdoors through WiFi sensing, and 2) how to analyze the resulting data to produce meaningful insights. This includes:

- Getting
- WiFi data preprocessing
- WiFi data analysis

Why WiFi sensing?

WiFi sensing technologies are among these tools, providing a non-invasive method for monitoring pedestrians outdoors via sensors that detect WiFi packets sent regularly by access points (APs) and WiFi-enabled devices. Most pedestrians today carry smart devices equipped with WiFi network interfaces, and each WiFi packet includes unique 48-bit addresses, known as Media Access Control (MAC) addresses, enabling a device to be tracked by multiple WiFi sensors. Many recent studies have utilized these sensing technologies to identify pedestrian movements and behaviors¹²³.

¹Duives, D. C., van Oijen, T., & Hoogendoorn, S. P. (2020). Enhancing Crowd Monitoring System Functionality through Data Fusion: Estimating Flow Rate from Wi-Fi Traces and Automated Counting System Data. *Sensors (Basel)*, 20(21). <https://doi.org/10.3390/s20216032>

²Soundararaj, B., Cheshire, J., & Longley, P. (2019). Estimating real-time high-street footfall from Wi-Fi probe requests. *International Journal of Geographical Information Science*, 34(2), 325-343,. <https://doi.org/10.1080/13658816.2019.1587616>

³Zhou, Y., Lau, B. P. L., Koh, Z., Yuen, C., & Ng, B. K. K. (2020). Understanding Crowd Behaviors in a Social Event by Passive WiFi Sensing and Data Mining. *IEEE internet of things journal*, 1-1,. <https://doi.org/10.1109/jiot.2020.2972062>

1

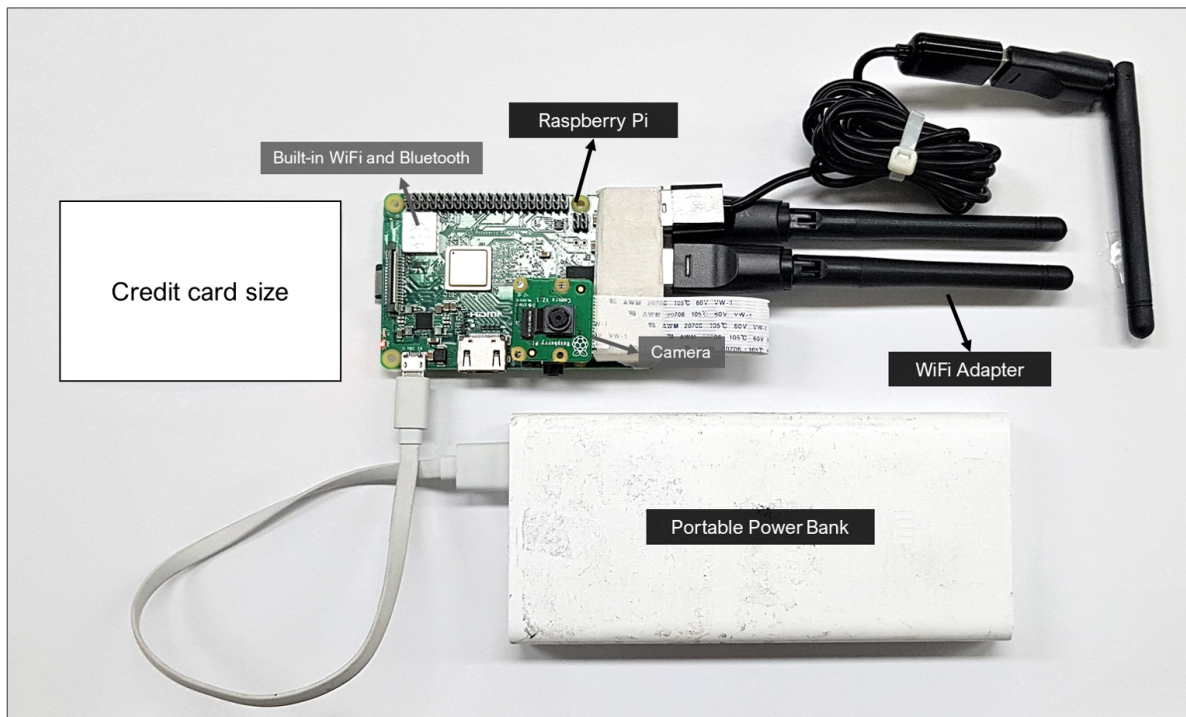
Part I

2. Setting up WiFi Sensors

2.1. Requirements

2.1.1. Hardware Requirements

This is the hardware setup illustrating the necessary components required for Wi-Fi sensing:



To build up this WiFi sensor, you'll need the following hardware:

Item	Function	Requirement
Laptop and LAN cable	Accessing and controlling the sensor	
Raspberry Pi board	Building the sensor	Pi 3B/3B+ or a higher model Pi
WiFi adapter	Capturing WiFi packets	Check chipset compatibility for 'monitoring mode' (here)
Micro SD card and adapter	Building and storing data	At least 16 GB size

Item	Function	Requirement
Ethernet cable	Connecting the Pi to your laptop	
Portable power bank	Powering the sensor in outdoor environments	Battery capacity: +20,000 mAh (lasts one day in our setting)
Pi camera	Recording the scene in front of the sensor	

2.1.2. Software

This is a list of the main programs needed to build a Wi-Fi sensor and handle the sensor data. I will also provide the download link for each step later.

Item	Function	Link
PuTTY	To access the Raspberry Pi remotely	here
SQLite	Database program to store sensor data	here
Raspberry Pi Imager	Tool to write Raspberry Pi OS images onto SD cards	here

2.1.3. Skill required

You should have a basic understanding of programming, specifically in R and Python. You should be able to write, edit, and debug code. If you need to brush up on these skills or are just starting out, I recommend the following courses: Data Science: Foundations using R Specialization:

- [Data Science: Foundations using R Specialization](#): This course will help you build a strong foundation in data science using R.
- [Python for Everybody Specialization](#): This course is a great introduction to programming in Python and covers the basics you'll need for this project.

Placement and Coverage Considerations

Item	Function	Requirement
Laptop and LAN cable	Accessing and controlling the sensor	
Raspberry Pi board	Building the sensor	Pi 3B/3B+ or a higher model Pi
WiFi adapter	Capturing WiFi packets	Check chipset compatibility for 'monitoring mode' (here)
Micro SD card and adapter	Building and storing data	At least 16 GB size
Ethernet cable	Connecting the Pi to your laptop	
Portable power bank	Powering the sensor in outdoor environments	Battery capacity: +20,000 mAh (lasts one day in our setting)
Pi camera	Recording the scene in front of the sensor	

Software

Item	Function	Link
PuTTY	To access the Pi by your laptop	here
Raspberry Pi Imager	To build Raspberry Pi OS	here
Raspberry Pi Imager	To build Raspberry Pi OS	here

Skill

Learning R and Python will be necessary for sensor building and data analysis. I recommend these classes: [Data Science: Foundations using R Specialization](#) and [Python for Everybody Specialization](#)