INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA INFORMÁTICA




SubwayNav - Navegação de pedestres no metro usando o smartphone

AFONSO GONÇALO AMORIM VOGENSEN
Setembro de 2019

POLITÉCNICO
DO PORTO

SubwayNav

Subway pedestrian navigation using the smartphone

Afonso Gonçalo Amorim Vogensen

Thesis for the Degree of Master in Computer Engineering, Area of Specialization in Graphic and Multimedia Systems

Advisor: Filipe Pacheco

External Supervisor: Marília Barandas

Júri:

Presidente:

[Nome do Presidente, Categoria, Escola]

Vogais:

[Nome do Vogal1, Categoria, Escola]

[Nome do Vogal2, Categoria, Escola] (até 4 vogais)

Porto, October 2019

Acknowledgements

I sincerely thank everyone who supported and motivated me throughout the course of the project, particularly:

My ISEP advisor, Engineer Filipe Pacheco for his curiosity, idea suggestions, and willingness to help, offer advice and comments in order to draft the best version of this document.

To my Fraunhofer advisors, Pedro Alves and Marília Barandas, for helping me to hone research skills, produce a quality thesis and a solution that follows Fraunhofer's standards and conventions.

And, finally, to my family, friends and partner for the strength, support, and advice that made this experience so much more memorable.

Thanks, from the bottom of my heart.

Resumo

Pessoas de todo o mundo confrontam-se com pobre informação nas redes de metro em turismo, o que causa atrasos ou enganos no percurso. Uma pesquisa extensa revela que as soluções de navegação subterrânea são escassas e maioritariamente para fins académicos. A procura insuficiente, juntamente com as limitações da tecnologia de navegação em ambientes subterrâneos existente, como GNSS, são as principais razões para essa escassez a nível global. No entanto, representa uma oportunidade para fornecer informações adicionais para turistas e cidadãos nativos, melhorando a experiência do utilizador nas redes de metro.

Para alcançar esta solução, SubwayNav aborda a navegação nesses ambientes usando sensores digitais disponíveis na maioria dos dispositivos móveis Android, como o acelerómetro, magnetómetro e giroscópio. Colaboradora no campo da mobilidade urbana, Ubirider, propôs as principais funcionalidades que incluem um contador de paragens, detector de direção e um simples planejador de rotas, contando apenas com sensores de smartphones e possuindo adaptabilidade, precisão e eficiência.

A contagem de paragens implica a distinção entre pelo menos dois estados: num metro em movimento e parado em qualquer outra situação. Preliminarmente, considerando isto como um problema de classificação, dados de sensores são repetidamente adquiridos em viagens de metro, extensivamente analisados e processados para identificar as features que melhor distinguem os estados uns dos outros. Recorrendo a algoritmos de aprendizagem supervisionados, treinados com as melhores features para melhor robustez, o dispositivo móvel introduz janelas curtas de dados dos sensores em tempo real e obtém assim o seu estado. Para detectar a direção, o tempo entre as paragens é comparado com a duração entre as margens fixas antes e depois da duração média até a estação correta. Com essa informação, os utilizadores conseguem chegar ao destino a tempo e em segurança com mais facilidade. Finalmente, uma prova de conceito visualmente básica é desenvolvida e testada em campo para avaliar o desempenho, o valor e a fiabilidade do algoritmo desenvolvido.

Palavras-chave (Tema): *Mobilidade, Navigação, Redes Metro, Localização no Interior, Sensores Digitais, Processamento Sinal*

Palavras-chave (Tecnologias): *Android, Python, Supervised Machine Learning*

Abstract

People across the world are no strangers to poor information in unknown subway networks, which cause delays or mistakes in their trajectory. A thorough research reveals that underground navigation solutions are scarce and mainly for academic purposes. Insufficient demand coupled with the limitations of existing navigational technology in the environment, such as GNSS, are the main reasons for this scarcity on a global level. However, this poses as an opportunity to provide additional information to tourists and natives, improving the user experience in subway networks.

To achieve this solution, SubwayNav tackles navigation in these environments using digital sensors which are present on most Android mobile devices, such as the accelerometer, magnetometer and gyroscope. Collaborator in the field of urban mobility, Ubirider, proposed the main functionalities which include a stop counter, direction detector and a simple route planner, relying solely on smartphone sensors and emphasizing adaptability, precision and efficiency.

Counting stops requires distinguishing between at least two states: moving within a metro, and stopped in any other situation. Preliminarily, considering this a classification problem, sensor data is repeatedly acquired in metro trips, extensively analyzed and processed to identify the features that best distinguish the states from each other. Resorting to supervised machine learning algorithms, trained with the best performing features for better robustness, the mobile device may feed short windows of sensor data in real time, thus keeping track of its state. To detect direction, time between stops is compared to a time window ranging between fixed margins before and after the average duration to the correct station. With this information, users are likely to reach their destination safely and on time. Finally, a visually basic proof of concept was developed and tested in the field to assess the performance, value and reliability of the developed algorithm.

***Keywords (Topic)**: Mobility, Navigation, Subway Networks, Indoor Location, Digital Sensors, Signal Processing*

***Keywords (Technologies)**: Android, Python, Supervised Machine Learning*

Contents

Table of Figures

Table Index

Acronyms

NOTATION	***MEANING***
AHP	*Analytic Hierarchy Process*
CI	*Consistency Index*
CR	*Consistency Ratio*
FFE	*Fuzzy Front End*
GNSS	*Global Navigation Satellite System*
GPS	*Global Positioning System*
MVP	*Minimum Viable Product*
NPD	*New Product Development*
QEF	*Qualitative Evaluation Framework*
SFS	*Sequential Feature Selector*

1 Introduction

In this section the SubwayNav thesis is presented, which includes a brief description of all contributions, from a business, personal and professional perspectives.

1.1 Problem and Context

Considering that Portugal has recently reached a big milestone, hosting record-breaking numbers of tourists in recent years [1], and set further achievable goals [2], people are turning towards tourism for business opportunities. Particularly, public transportation can be explored further to break the next milestone all while improving the user experience for tourists and regular users alike.

One the best modes of public transportation is the metro which, despite being used less than road transports, is the best to avoid traffic and difficulties switching to other modes, as well as the most environment-friendly option [3]. But due to poor pricing options, customer service and lack of workers to attend to faults, the quality of experience is negatively affected, even though plans are in motion to expand the networks [4] despite the unresolved issues.

Based on this information, namely the poor customer service, and metro reviews[1], it is noted that insufficient information is at least one of the major issues. And seeing as metro usage has increased due to tourism and market opportunities [2] [5], improving the quality of information regarding services and navigation would reduce delays and boost the flow of pedestrian traffic. One way to provide said information is through mobile apps.

Regarding navigational information within subway networks, Global Navigation Satellite Systems (GNSS) [6], such as the Global Positioning System (GPS) [7], are not reliable enough and most alternative technologies require specific infrastructures which are expensive to install or manage [8][9]. Popular apps such as Google Maps [10][11] and Moovit [12] rely on GNSS and so have not provided a solution for indoor location. Therefore, indoor location

---

1 Porto & Lisbon Metro - 2019 All You Need to Know BEFORE You Go, TripAdvisor.

2 L. Villalobos, “Metro de Lisboa bate recorde de passageiros e de receitas”, O Público. [Online] https://www.publico.pt/2019/01/31/economia/noticia/metro-lisboa-bate-recorde-passageiros-receitas-1860142 [Accessed: 06-Jul-2019].

solutions are particularly relevant and desired not only for underground public transport networks, but also for large buildings, such as shopping centres and hospitals.

Furthermore, the integration of various sensors, specially inertial sensors, in mobile devices [17] presents itself as a massive opportunity to improve the quality of underground navigation systems. Resorting to multi-sensor data fusion in the detection of a user's state on-board a train can be a solution capable of guiding and locating an individual with enough accuracy.

So, the challenge is the elaboration of a reliable and versatile algorithm to locate and guide subway users, as well as the mobile proof of concept in which it will be integrated. The solution should be capable of circumventing the limitations of the popularly used tools and facilitating the use of its components by users and technicians alike.

1.2 Objectives

The main goal is to create a mobile system to locate and navigate a user on underground public transport networks. Specifically, subway station map information, magnetic/radio fingerprint maps, smartphone built-in sensors and transportation schedules are many of the existing options for data fusion that can precisely navigate users. Once located the starting point, we can then guide users to take the correct vehicle and exit at the correct stop, while alerting on possible mistakes made along the way.

The performance of the algorithm should be extensively evaluated and tested, and a proof of concept will be developed to demonstrate the capabilities of the system.

In order to achieve these primary objectives, a research phase must be carried out, particularly on existing navigation systems in underground environments and on the use of sensors for indoor mapping and navigation. Later, a proof of concept will be designed and the use cases that the algorithm will depend on will be defined, with special attention to modularization, scalability and universality, if applicable.

In the development phase of the solution, it is necessary to adopt an agile methodology that will facilitate the distribution of tasks and the monitoring of the project's progress. In addition, it is necessary to adopt conventions and good practices, as well as use development patterns, when the situation so requires.

1.3 Concerns

In order to stand out within the vast sector and maximize the benefits of a solution, regarding cost, efficiency, maintainability, innovation, etc, several additional non-functional and critical concerns are to be addressed and abided:

- Battery consumption:
  - Due to the usage of multiple sensors over the course of a trip in the subway, smartphone batteries are expected to be drained faster. Preferably, its consumption should be minimalized.
- Simple Architecture:
  - The developed prototype is not to be integrated in a complex system but only simply demonstrate the capability of smartphone sensors in subway navigation.
- Confidentiality:
  - It must be clear that sensitive and private data is not collected and used for the purpose of the project unless it is anonymised or authorized.

1.4 Contributions

This subchapter is dedicated to the main contributions from a personal and involving companies' perspective, including impacts on the business sector.

1.4.1 Company Perspective

Proposed by Fraunhofer's partnering company Ubirider [14], the purpose of SubwayNav is to have its algorithm integrated in a journey assistant app, PickApp, to provide, in the long term, universal navigation throughout a trip, indoors and outdoors.

This challenge and its milestones are an important opportunity to explore tools and technologies seldomly considered to design an innovative solution. By gathering information and experimenting, experience and knowledge is expanded significantly which translates into better ideas and practices in the future, regardless of the success of the solution.

The combination of all efforts and resources bring to fruition a desirable solution for Ubirider while simultaneously predicting a positive impact to both companies' reputation, considering the influence the project will have in the public transportation market.

1.4.2 Impact in the Public Transportation Sector

Due to reviews[1] revealing a lack of clear, simple and straightforward information in metro stations, it is certain that its inclusion would shift some transportation usage from personal to public.

Despite going against tech giants such as Google and Moovit, providing an adaptable system, while simultaneously serving information on the go, the solution should stand out, endure and bring value to every stakeholder.

On a larger scope, the integration of these solutions with quick and reliable responses in underground environments may boost the usage of public transportation and, in turn, substantially improve the impact on the environment.

1.4.3 Personal Perspective

The project serves as an opportunity to explore smartphone sensors on a deeper level, as well as develop a solution born from ambition to improve the current system in subway stations around the world, which in turn could be environment friendly.

It was planned from the start that newer programming languages and practices would be involved, which also broadens my current bubble of knowledge and experience in the field.

Finally, because the project involves extensive testing and experimentation, even in the field, the introduction to new research practices to familiarize with is highly desired.

1.5 Expected Results

In order to verify whether smartphone sensors can reliably guide people in subway networks, there are two main expected results: the proof of concept which acts as the Minimum Viable Product (MVP) and the analysis of all data acquired with smartphone sensors, of which a degree of satisfaction will be attributed based on its value.

1.5.1 Proof of Concept

The developed prototype, defined with Fraunhofer's guidance, is expected to solely include a single component corresponding to a standalone mobile application, capable of gathering data from its integrated sensors and accurately assess the current situation the user is in. It should cover the following states:

I. Inside a metro in motion and knowing its direction of travel;

II. Inside a metro, arriving at a station;

III. Unexpected stop while inside a metro;

IV. Standing by a station's exit/entrance;

V. Standing by a station's point of interest (e.g. kiosk, shop, platform, etc).

In any unexpected scenario or following a mistake on the user's part, appropriate questions regarding the situation are asked in order to properly assess is and provide better directions. As a result, the user is able to correct their path without wasting time and, in extreme cases, money spent on tickets, for example.

Figure 1 below represents an example of the envisioned system, in which the Sensors framework will serve the prototype application with data collected by several sensors in real time.

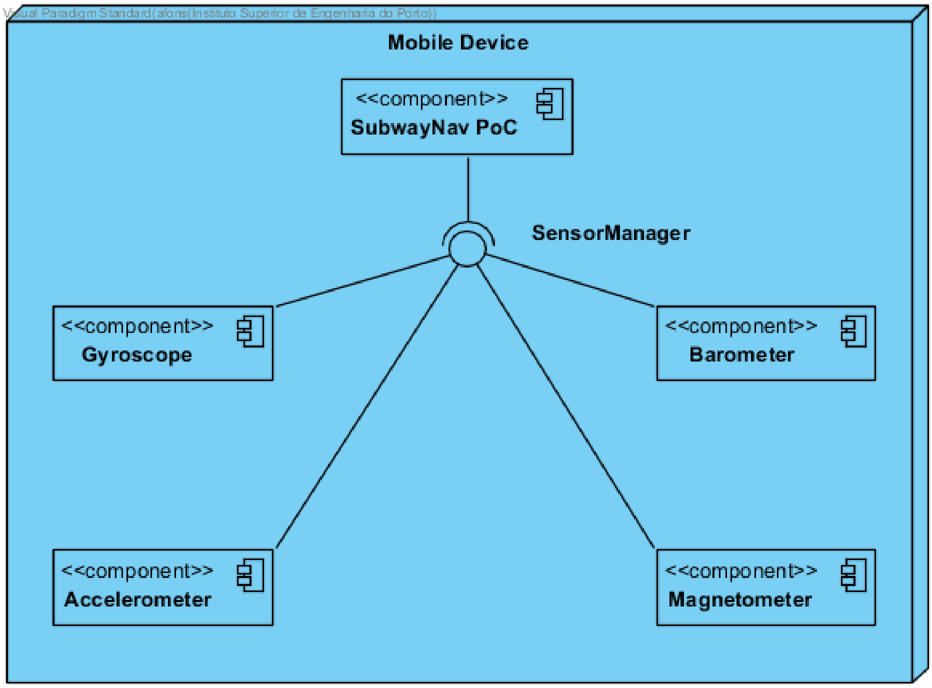


Figure 1 – Proof of concept Deployment Diagram example

To establish communication with external components, mobile data or certain adapters like the Wi-Fi and Bluetooth adapters would need to be switched on, implying more battery and resource consumption.

Through a REST Application Programming Interface (API), the mobile device could obtain, for instance, train schedules, fingerprint map information and other relevant information to better locate users in stations.

1.5.2 Data Analysis

The value of the utilized sensors highly depends on the data collected by them after a significant number of tries and whether the data can accurately distinguish a user's state and position or not.

Among the various inbuilt sensors in mobile devices, the following captured our interest:

- Gyroscope;
- Magnetometer;
- Accelerometer;
- Barometer;
- Microphone;
- WiFi Adapter.

However, there is a chance that all the mentioned devices are not needed even if they check all the desired capabilities, the reason being the battery consumption and availability. The analysis will allow for the essential sensors to be selected while minimizing resource usage.

After collecting a significant number of data samples in stations and trains, with various mobile devices, a proper analysis using adequate tools to plot graphs is carried out to check for consistencies between sensors and between devices. Furthermore, existing papers and experiments will serve as an influence to prioritize sensors based on their reliability and performance. This approach facilitates the selection of the sensors best suited for the test environments.

1.6 Document Structure

This document is divided in the following main chapters:

- Introduction: context and presentation of the project, the problem which led to its genesis, the objectives, stakeholder concerns, and the resulting contributions.
- State of Art: description of the current state of art of the respective business sector and existing approaches. Also included in this chapter is the value analysis of the project and a thorough comparison between alternative solutions.
- Value Analysis: includes an internal and external analysis of the business environment, a value network detailing the exchanged valuables between stakeholders, benefits and sacrifices of conceiving or supporting the solution, the innovation process behind the idea for the project, and a business model canvas.
- Solution Evaluation: presents the tools and methodologies to assess the resulting solution, as well as the hypothesis to be verified.
- Development: addresses the project plan, preliminary phases such as data acquisition and algorithm development, in addition to the implementation of the final prototype.
- Conclusions: dedicated to the conclusions drawn from the entire process of research and development, and the final remarks.
- Annex: contains data acquisition and testing protocols, as well as graphs for several recorded test sessions.

2 State of Art

This chapter sheds light on the current state of art of the respective business sector of the project and existing solutions. The value analysis of the project and a thorough comparison between alternative solutions is also included.

2.1 Business Sector

The project is inserted in the mobile navigation sector, having influence in other sectors such as public transportation and tourism.

Along with the exponential evolution of mobile devices came the integration of many sensors and features to allow for diverse types of measurements, many of which served as cheaper alternatives for researchers, students and engineers alike.

Despite the vast reliability and usefulness of the GNSS, the next frontier has yet to be universally crossed, which is why demand for indoor navigation systems has increased.

There are two types of solutions, industrial and academic.

2.1.1 Industrial Solutions

In the industry, these systems usually integrate devices such as beacons [15], and other signal emitters, repeaters and amplifiers, which allow for better data transmission in subway networks. As opposed to academic approaches, these better inform users on their location and path with near total accuracy.

The biggest disadvantage is the high cost of installation and maintenance, which is why these systems are still uncommon.

2.1.2 Academic Solutions

For academic purposes, usually inertial sensors [16] are used to locate individuals indoors, opening the door to many additional features that may depend on location (e.g. activity detection). Generally, the sole purpose of the solutions is guiding people.

Due to financial limitations, diverse existing academic approaches [17] have also obtained data from magnetic fields, Wi-Fi access points, air pressure, among other sources that vastly reduce the cost of installing a complex system that would serve the same purpose. High-level information such as station maps and timetables can also be incorporated to ensure accuracy and utility.

2.2 Approaches

The following subchapters will address several existing solutions, both academic and commercial, and identify the crucial aspects that influenced the envisioned prototype.

2.2.1 Accelerometer-based Solutions

An extremely common approach is using a smartphone's accelerometer to distinguish a stationary state from a moving state, considering most smartphones nowadays have one installed.

Existing solutions [18]–[20] often use an empirically defined threshold and noise filtering to detect shifts in states while minimizing false positives and negatives.

The core mechanic consists of calculating the probability of being stationary or in motion, based on accelerometer sample readings. However, due to sudden jolts and shifts in acceleration, initial experiments yielded false positives. In other words, the probability of being stationary was relatively high even if the vehicle was in motion, as observed in cumulative distribution graphs plotted by the respective researchers (see Figure 2).

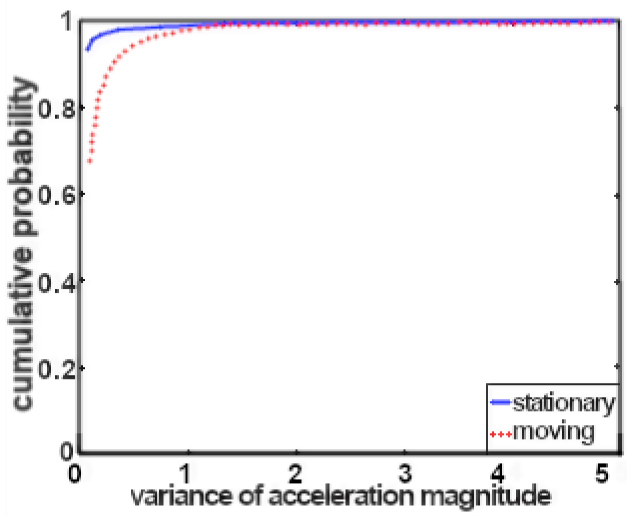


Figure 2 – Cumulative distribution of acceleration variance [18].

To overcome this obstacle, researchers resorted to filtering accelerometer data and fusing their current algorithms with algorithms based on other sensors, particularly magnetometers and gyroscopes. By further filtering the data collected in real time, the distributions of probability for both states had become vastly different (see Figure *3*), resulting in high stop detection precision and recall values in the following test phases.

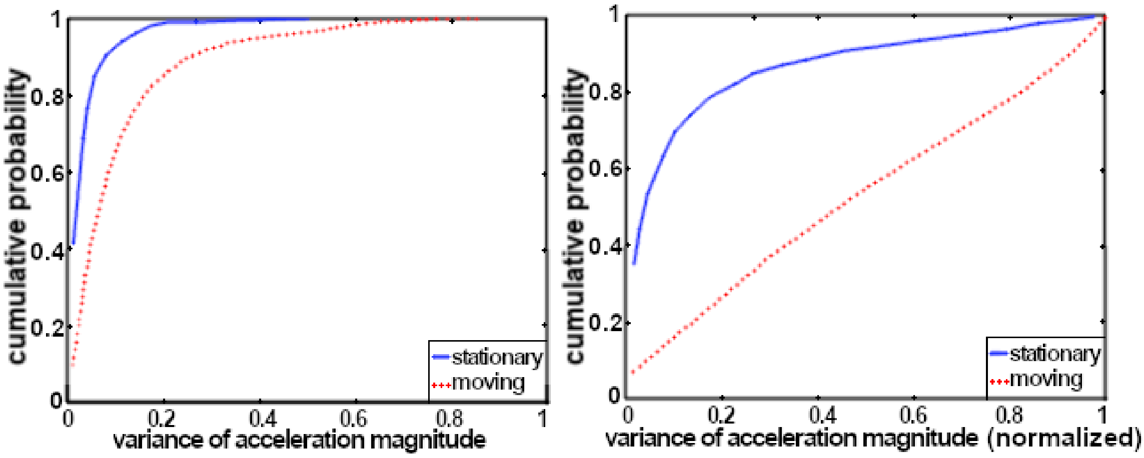


Figure 3 – Cumulative distributions of smoothed acceleration variance [18].

2.2.2 Barometer-based Solutions

Since the recent integration of barometers in smartphones, many researchers [17] [18] have seen its potential in underground positioning and stop detection, specially due to a phenomenon observed when trains accelerate in the tunnels, called the Venturi Effect [23].

Essentially, as observed, tested and reported by Snips [21], pressure levels would shift drastically whenever the metro accelerated and stabilize when it slowed and stopped. Despite other variables such as vent shafts and open sections of the lines, the data revealed to be generally consistent.

By establishing a threshold from which a variation could be detected, stop detection accuracy has reached 90%. Furthermore, the algorithm could be fused with additional ones, such as direction detection based on intervals between stops, time schedules, machine learning and crowdsourcing.

So, these types of solutions have demonstrated flexibility and versatility by facilitating stop and motion detection, even without resorting to any system components other than a smartphone and its inertial sensors.

2.2.3 Non-standalone Solutions

Even though standalone solutions imply fewer costs, it is worth considering the integration of other devices and components, especially for implementing navigation within subway stations.

A diverse range of innovative approaches have surfaced, tackling technology such as the following:

a) External Sensors

- **BLE Beacons** [15] – One of the most popular devices indoor mapping which emits a short range Bluetooth signal and possesses a unique ID to detect movement within particular areas. Compatible with smartphones but slightly imprecise, also requiring a vast network to be installed and large amounts of data in order to offer informative guides to users.
- **Ultra Wideband (UWB)** [24] – Considered a better alternative to beacons in terms of responsiveness and precision, due to data transmission over WiFi networks. While easier to maintain, it is more costly and exhausting to install if pinpoint position is required. Furthermore, it needs users to connect a dongle to their device, which may be inconvenient and unsafe to distribute them to a location's visitors as they could be stolen or damaged.

b) Crowdsourcing w/Machine Learning [25] – Requiring a Backend and management of large volumes of data, this approach relies on users to repeatedly collect data and correct false results. It would, however, translate into more accurate results in the long term as context is better analysed and trained algorithms become more robust.

c) Fingerprint-Based Solutions [26]– These solutions rely on signal fingerprint maps, constructed from numerous data collections, to identify a user's location in distinguished areas of a space. Such data can be gathered from the strength of magnetic fields and WiFi signals, which they heavily depend on.

Once gathered the data, following strict protocols to cover the entirety of a space, the same space is divided into smaller and identifiable areas, whose characteristics are stored in a server. Then a sample of any smartphone's gathered data is used to match with a specific area in order to display it to users, with a certain radius of tolerance.

2.3 Comparison of Alternatives

The enumerated solutions will be compared with the purpose of extracting the best aspects and fulfilling the stakeholders' needs.

Despite the reported results of other researchers, none of the solutions check all the objectives for the envisioned prototype. However, it must first be determined which approaches should the project be based on.

For that, the alternatives will be analysed using Thomas L. Saaty's Analytic Hierarchy Process (AHP)[27] on the following main criteria:

- Accuracy;
- System Complexity;
- Cost;
- Flexibility;
- Resource consumption.

The first step is to pair the criteria using a 1 through 9 scale, 1 meaning equal importance and 9 meaning absolute importance of one over another. The following Table 1 represents the pairings and the sum of each column, which is used to normalize the matrix (Table 2).

The weights were attributed based on multiple meetings with Ubirider*[14]*, a partner with hopes to integrate the developed algorithm in their system, who prioritize flexibility and simplicity above all, do not demand pinpoint accuracy and show little concern for battery consumption.

Criteria	Flexibility	Cost	Accuracy	Simplicity	Battery Consumption
Flexibility	1	5	2	1	6
Cost	1/5	1	1/2	1/3	2
Accuracy	1/2	2	1	1/3	2
Simplicity	1	3	3	1	6
Battery Consumption	1/6	1/2	1/2	1/6	1
Total	2 7/8	11 1/2	7	2 5/6	17

Table 1 – AHP Criteria Pairing Matrix

15/43	10/23	2/7	6/17	6/17
3/43	2/23	1/14	2/17	2/17
15/86	4/23	1/7	2/17	2/17
15/43	6/23	3/7	6/17	6/17

Table 2 – Normalized Pairing Matrix

The next step, as presented in Table 3, is calculating the average of each row, which represents each criteria's score, the consistency, which corresponds to the product of a row's values with the respective criteria score, and the λmax value, used to test the pairing values for consistency.

Criteria	Score	Preference	Consistency	Consistency/Score	
Flexibility	0.355	1	1.788	5.036	
Cost	0.086	4	0.465	5.387	
Accuracy	0.152	3	0.734	4.827	
Simplicity	0.347	2	1.767	5.082	
Battery Consumption	0.058	5	0.285	5	
				5.080	λmax

Table 3 – Resulting Score, Consistency and λmax

As observed, flexibility and simplicity are the most significant criteria, however, the score is only valid if the assigned values are consistent. So, the Consistency Index (CI) and Consistency Ratio (CR) need to be calculated, which are given by the following formulas:

$$CI = \frac{\lambda max - n}{n - 1}$$

$$CR = \frac{CI}{RCI}$$

Where n corresponds to the number of criteria and *RCI* corresponds to Thomas Saaty's Random Consistency Index, which is defined as 0.90, meaning that, if the resulting CR is under 0.1 then all the results are satisfactory.

Calculating CI and CR:

CI = (5.080 – 5)/4 = 0.020
CR = 0.020/0.90 = 0.022

Because CR is lower than 0.1, it is safe to assume that both the assigned weights and resulting scores are consistent.

In the next sub-chapters, the alternative approaches will be analysed and assessed on each criterion, using the same process to calculate a score and assign a preference.

2.3.1 Cost

The assigned weights in the initial pairing matrix (Table 4) are based on estimated cost of installation of the systems relative to one another. Because solely sensor-based solutions are low-cost as opposed to larger systems, they are regarded as preferable.

Note that, although accelerometers and barometers are considered smartphone sensors, barometers are far more uncommon, adding to its cost.

Table 5 reveals accelerometer-based solutions to be the most cost-efficient, because a negligible number of smartphones nowadays do not have one installed.

Cost	Accelerometer-Based	Fingerprint Map	Barometer-Based	External Sensors
Accelerometer-Based	1	3	2	4
Fingerprint Map	1/3	1	1	2
Barometer-Based	1/2	1	1	3
External Sensors	1/4	1/2	1/3	1

Table 4 – Price Weight Pairing

Alternatives	Score	Preference
Accelerometer-Based	0.4717	1
Barometer-Based	0.2381	2
Fingerprint Map	0.1931	3
External Sensors	0.0970	4

Table 5 – AHP Price Score Results

2.3.2 Flexibility

In terms of flexibility, the higher the weight, the more modular and versatile the solution is. In other words, it eases management and addition of new features.

Since none of the approaches on their own meet all the project's objectives, the weights are consistent, as represented in tables 6 and 7. Despite that, external sensor approaches revealed to be far more flexible due to all additional data that can be gathered at the cost of more equipment [15][24]. Meanwhile, barometer-based solutions came last, considering it has only been used to detect shifts in motion.

Flexibility	Accelerometer-Based	Fingerprint Map	Barometer-Based	External Sensors
Accelerometer-Based	1	1	2	1/5
Fingerprint Map	1	1	2	1/3
Barometer-Based	1/2	1/2	1	1/6
External Sensors	5	3	6	1

Table 6 – Flexibility Weight Pairing

Alternatives	Score	Preference
External Sensors	0.5865	1
Fingerprint Map	0.1733	2
Accelerometer-Based	0.1537	3
Barometer-Based	0.0866	4

Table 7 – AHP Flexibility Score Results

2.3.3 Simplicity

Simplicity refers to the complexity of the system as a whole, which becomes harder to invest, build and maintain the more components it has. Out of all alternatives, the inertial sensor approaches are the simplest as no communication is established between smartphones and external components, while external sensor approaches imply an undefined number of components, depending on the complexity of a subway's network.

Simplicity	Accelerometer-Based	Fingerprint Map	Barometer-Based	External Sensors
Accelerometer-Based	1	3	1	9
Fingerprint Map	1/3	1	1/3	4
Barometer-Based	1	3	1	9
External Sensors	1/9	1/4	1/9	1

Table 8 – Simplicity Weight Pairing

Alternatives	Score	Preference
Accelerometer-Based	0.4058	1
Barometer-Based	0.4058	1
Fingerprint Map	0.1461	2
External Sensors	0.0422	3

Table 9 – Simplicity Score Results

2.3.4 Accuracy

While moderately important, there is no potential in a solution without a certain degree of precision. Therefore, the attributed weights are based on reported results (referenced in chapter 2.2), translating into bigger scores for external sensor and inertial barometer solutions.

Even though results appear one-sided, it is important to note that every solution achieved satisfactory levels of accuracy for their intended purpose, however, it does not apply to the broader scope of the project.

Accuracy	Accelerometer-Based	Fingerprint Map	Barometer-Based	External Sensors
Accelerometer-Based	1	3	1/3	1/6
Fingerprint Map	1/3	1	1/5	1/7
Barometer-Based	3	5	1	1/5
External Sensors	6	7	5	1

Table 10 – Accuracy Weight Pairings

Alternatives	Score	Preference
External Sensors	0.6115	1
Barometer-Based	0.2221	2
Accelerometer-Based	0.1114	3
Fingerprint Map	0.0550	4

Table 11 – Accuracy Score Results

2.3.5 Resource Consumption

Since the long-term objective of indoor positioning research and solutions is continuous use by visitors and tourists, resource consumption must be taken into consideration, otherwise users would find themselves without directions and a smartphone before the day is over.

For all solutions, weights are mainly based on [28], a study on smartphone sensors' battery usage. For fingerprint map and external sensor approaches, it is assumed that data is collected using magnetometers, Bluetooth and WiFi.

The referenced article and resulting scores in Table *13* reveal that magnetometers, WiFi, Bluetooth and gyroscopes generally consume battery the most, placing the barometer as the preferred solution to save battery.

Battery Consumption	Accelerometer-Based	Fingerprint Map	Barometer-Based	External Sensors
Accelerometer-Based	1	6	1/2	5
Fingerprint Map	1/7	1	1/5	2
Barometer-Based	2	7	1	6
External Sensors	1/5	1/2	1/6	1

Table 12 – Resource Consumption Weight Pairing

Alternatives	Score	Preference
Barometer-Based	0.5113	1
Accelerometer-Based	0.3345	2
Fingerprint Map	0.0904	3
External Sensors	0.0638	4

Table 13 – Resource Consumption Score Results

2.3.6 Final Decision

Finally, a final matrix with all the alternatives' results is squared with each criterion's weight to calculate the rating for each approach. As observed in Table 14, the best overall alternative under these criteria is the accelerometer-based approach.

	Cost	Flexibility	Accuracy	Simplicity	Battery	Rating	Preference
Accelerometer-Based	0.4717	0.1537	0.1114	0.4058	0.3345	0.358327	1
Barometer-Based	0.2381	0.0866	0.2221	0.4058	0.5113	0.296721	2
External Sensors	0.0970	0.5865	0.6115	0.0422	0.0638	0.196586	3
Fingerprint Map	0.1931	0.1733	0.0550	0.1461	0.0904	0.148012	4

Table 14 – AHP Final Results

Considering how the sensor is present in most modern mobile devices and its final score, the proof of concept will be based on it but also use data fusion in case other data is available such as barometer readings and fingerprint maps. Like so, the algorithm operates on a wide range of devices.

3 Value Analysis

3.1.1 Inovation Process

Innovation is crucial for the development of a product with advantage over competition, but it always carries risks because a poorly planned concept may lead to business failure.

In order to mitigate risks, innovation process techniques are used, such as Peter Koen's model [29].

As observed in Figure *4*, Peter Koen divides the innovation process into three distinct phases: Fuzzy Front End (FFE), New Product Development (NPD) and Commercialization.

The FFE is characterized by informality and its unpredictable activities and uncertain dates, whereas the second phase has a greater degree of formalization, being goal oriented and more rigorous in relation to deadlines.

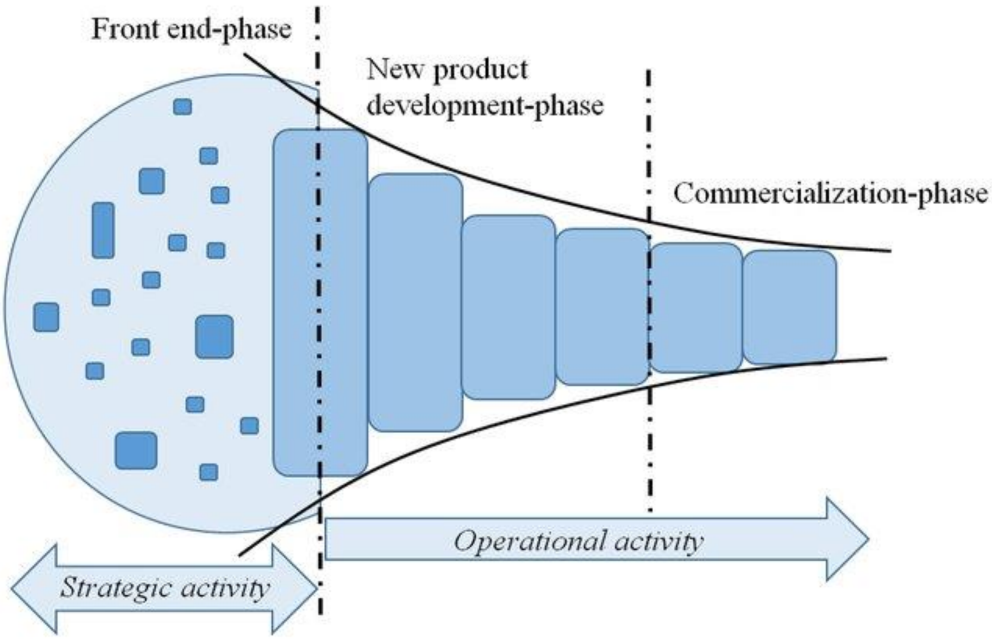


Figure 4 – Front End Innovation Process *[30]*

In 2001 the New Concept Development (NCD) was developed to simplify the FFE process, which is composed of three essential parts (see Figure 5):

• Engine: which represents the organization's culture and business strategy;

• Influencing factors: external factors that influence the engine and the key elements;

• Key elements: includes the identification and analysis of the opportunities, genesis and analysis of the idea, and concept development. As seen in Figure *5*, the process is not entirely linear, so it is possible to repeat any phase.

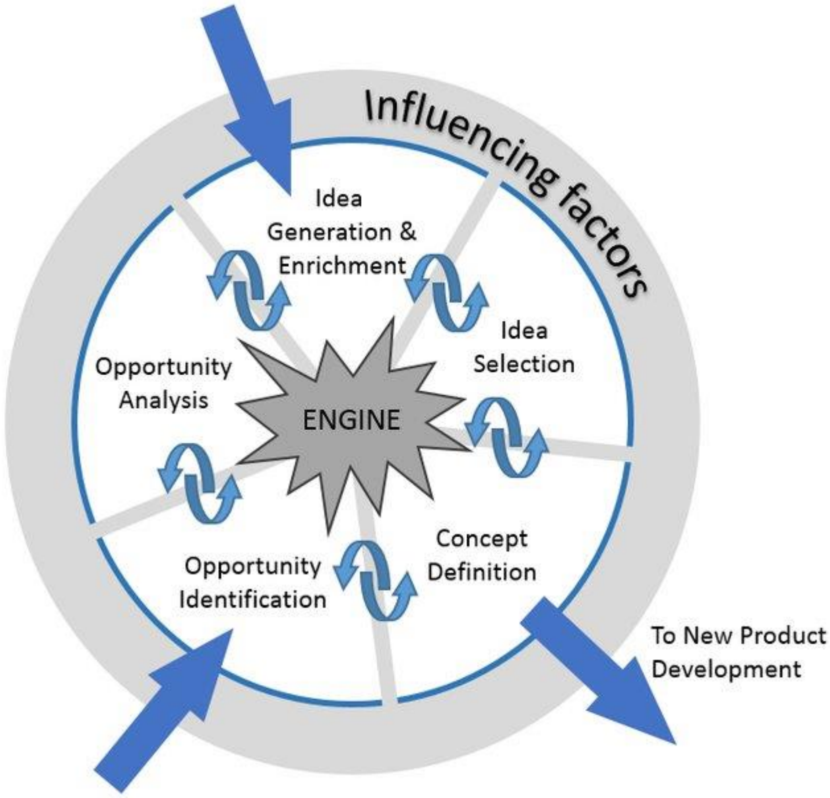


Figure 5 – New Concept Development Model *[30]*

3.1.1.1 Opportunity Identification

In this phase, the market and technology trends are analysed, communities are studied, problems are searched for, and ideas are brainstormed in order to identify various business opportunities.

For this project's case, it was observed that local subway station services did not provide enough information to guide users, specially foreigners[1], within them and that the increasing integration of sensors in smartphones are useful in indoor mapping [17].

3.1.1.2 Opportunity Analysis

All identified opportunities should then be analysed and evaluated to determine their viability to create a product or service.

Having identified the problem and opportunities, it was concluded that information on navigation within the station, metro arrivals and departures, and other services, like buying tickets, would improve user experience. In addition, it would reduce waiting times and distractions, possibly increasing the number of passengers.

3.1.1.3 Idea Generation

This phase consists of formulating concrete ideas, taking previous opportunity analysis into account, until a concrete, executable and detailed resulting solution is achieved.

So, with the base solution and respective consequences in mind, a smartphone application to guide users and provide detailed information on vehicles and other processes became the chosen solution to develop.

3.1.1.4 Idea Selection

Normally, during the innovation process, many ideas are generated and, while encouraged to do so, it is important to select those that bring the most value to the business.

The many ideas that have been generated have been filtered, the various approaches were also assessed and studied in order to obtain an envisioned product that meets the requirements of the stakeholders.

3.1.1.5 Concept Definition

The final step is usually the development of the business using the selected ideas, even though the nonlinearity of the model means this step could be taken earlier.

The chosen course of action was to research about inertial smartphone sensors for subway navigation and develop a proof of concept before proceeding.

The proof of concept, however, should be able to prove whether inertial sensors are enough to detect train moving direction, departures and arrivals, and so on, to alert a user of their current state.

3.1.2 Value Network

To clarify the exchanged valuables between all stakeholders and clients, a Verna Allee value network [31] was designed, which can also reveal possible risks in certain dependencies.

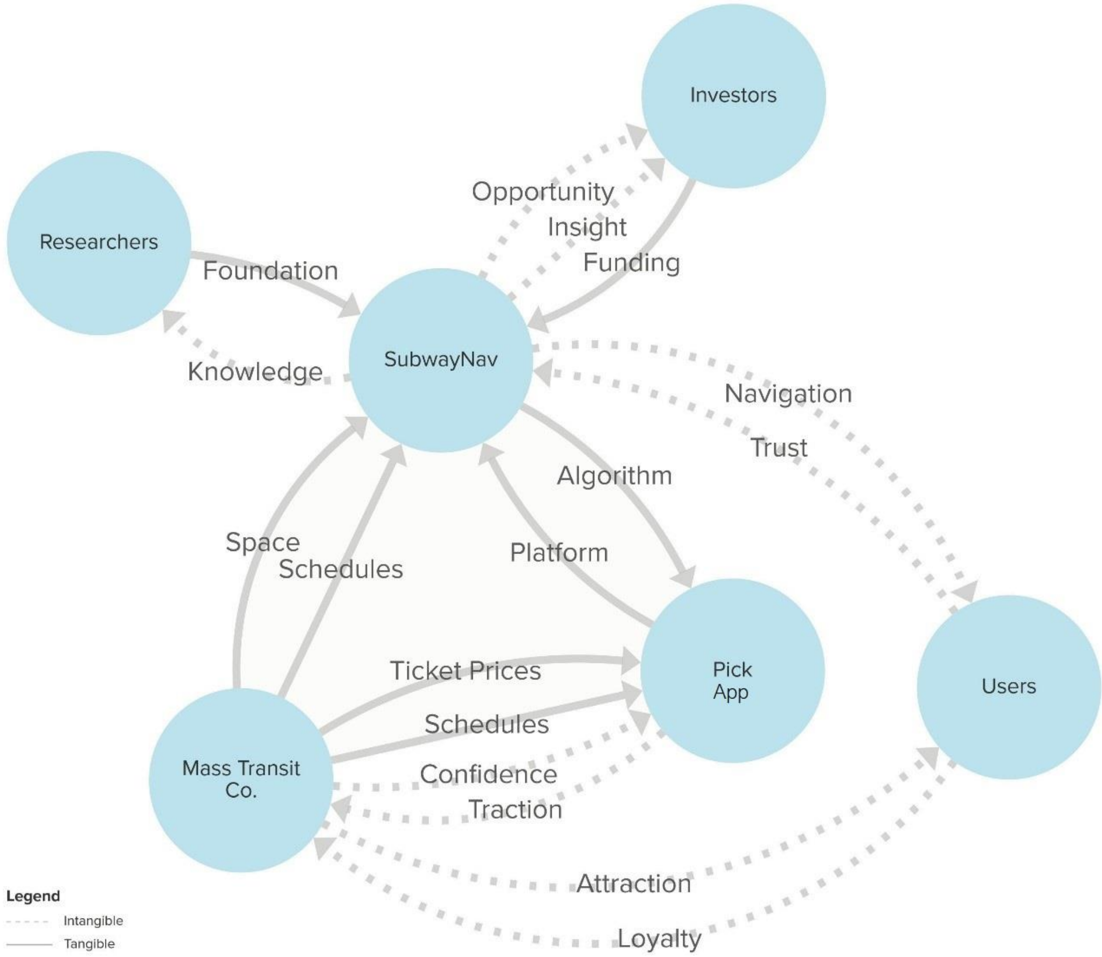


Figure 6 – Value Network

In Figure 6 the main entities are represented with the exchanged tangible and intangible resources. The following tables will address the description of each entity and each of the traded valuables.

Entity	Description
SubwayNav	The services and features provided by the subway navigation solution.
PickApp	Application that will integrate the developed subway location algorithm.
Researchers	The researchers involved in the conceptualization and assessment of the solution. These provide support while studying the usefulness of smartphone sensors for subway navigation.
Users	The primary clients who would use the product to navigate within the subway.
Mass Transit Company(ies)	The primary collaborators that would rely on SubwayNav to improve the quality of experience in subway networks. These companies are responsible for the vehicles and subway station services.
Investors	Potential financial supporters of the project.

Table 15 – Value Network Entities

Relationship	Valuables	Description
SubwayNav/Users	• Trust • Navigation	SubwayNav's main purpose is to navigate users in the subway and notify of any key situations. In exchange for the quality information, users place their trust on the product, which translates into referrals and regular usage.

Relationship	Valuables	Description
SubwayNav/Researchers	• Knowledge • Foundation	Researchers provide a foundation for the solution in the form of a proof of concept, receiving knowledge regarding its results.
Mass Transit Company(ies)/Users	• Trust • Attraction	Collaborating companies would increase attractiveness through improved user experience, gaining trust from visitors.
SubwayNav/Mass Transit Company(ies)	• Space • Schedules	Mass transit or public transportation companies supply their space for testing, installation and usage of the product, as well as information regarding train schedules in case they are required for the location algorithm.
SubwayNav/Investors	• Insight • Opportunity • Funding	Any investor would provide supporting funds to cover costs for its opportunity and insight into the market potential.
SubwayNav/PickApp	• Algorithm • Platform	Ideally, the algorithm implemented for SubwayNav should be integrated in the upcoming app PickApp which is offered as a platform to expose

Relationship	Valuables	Description
		the algorithm to customers and other environments.
Pick/Mass Transit Company(ies)	• Traction • Confidence • Ticket Prices • Schedules	Ubirider's[14] aim with Pick is to allow users to purchase transportation tickets and view information regarding the transports all in a single mobile app. Therefore, it requires ticket prices, schedules and confidence in exchange for the increased traction it brings with its convenient features.

Table 16 – Value Network Relationships

3.1.3 Perceived Value

A stakeholders' perceived quality and value is often crucial to determine the overall worth and potential of a product. Table 17 presents a list of benefits and sacrifices that can have an influence on the stakeholders' perception.

Scope/Domain	Product	Service	Relationship
Benefits	Alternative Solution Features Product Quality Easy to Interact Solutions	Reliability Responsiveness Efficiency Flexibility Technical Quality	Image Trust
Sacrifices	Costs	Costs	Time/Effort/Energy

Table 17 – Benefits and Sacrifices

- **Product**

The product refers to the final solution, resulting from the inclusion of the developed algorithm in an existent mobile system, which would, in turn, gain value as an alternative solution to subway network navigation. Its indoor location features could be used for many ends besides navigation, including statistics purposes, and would require no excessive actions or prior knowledge from users. Additionally, it could stand out in the market due to its quality, resulting from the adoption of good software engineering practices to maximize stability, precision and performance.

The main sacrifice would be the cost of overall maintenance and, in case it is required, the installation of a larger infrastructure.

- **Service**

Refers to the product's behaviour and quality of completed non-functional requirements.

The main purpose of SubwayNav is to use smartphone sensors to their fullest potential to detect changes in motion of a train and the position of users in stations, meaning that reliability and responsiveness are crucial benefits. The better the results, the more the product will be used.

The efficiency would be a result of the algorithm's balance of resource consumption and acceptable accuracy, which translates into a convenient and desirable application to install on a device.

The mere identification of a device's location and state underground makes the algorithm highly flexible in the sense it can open the door to a multitude of other services and adapt to changes over time through machine learning and low dependency on arbitrary constants. Such services and purposes include statistics, marketing and location within other vehicles or buildings.

Technical quality of the implementation and design is an important factor due to concern over service availability, precision and responsiveness.

- **Relationship**

The relationship relates to the perception clients take from the product and service.

To ensure lasting success, the final product should reflect a positive image of the company and its partners, particularly of expertise in the business, and gain the public's trust through the quality of its products, marketing campaigns and customer relations. However, time, effort and energy are necessary to give off a proper impression.

3.1.4 SWOT Analysis

Strengths	Weaknesses
○ Technical Experience; ○ Possession of quality work equipment and software; ○ Possession of sensor data recorder; ○ Solid market state awareness.	○ Insufficient gathered data; ○ Low signal processing experience; ○ Low Machine Learning experience; ○ Limited test locations.
Opportunities	**Threats**
○ Integration of diverse sensors in smartphones; ○ Negligible national competition; ○ Increasing number of tourists and mobile transportation users; ○ Higher preoccupation with environment.	○ Tech giants' involvement; ○ People may prefer driving to taking public transportation; ○ Improving other subway station services may render the solution obsolete or inconvenient.

Table 18 – SWOT Analysis

3.1.5 Business Model Canvas

The Business Model Canvas, is a strategic management tool widely used to quickly represent a company's business model, highlighting its key drivers and serving as a foundation for later specification *[32]*.

<table>
<tr>
<td rowspan="2">Key Partners
• Mass transit companies.</td>
<td>Key Activities
• Design and development of subway navigation mobile app;
• Collect data for research on inertial smartphone sensors.</td>
<td rowspan="2">Value Proposition
• Provide a low-cost mobile solution that guides and locates users within subway networks with smartphone sensors, reducing delays and mistakes.</td>
<td>Customer Relationships
• Online Support
• Telephone</td>
<td rowspan="2">Customer Segments
• Tourists;
• Subway users.</td>
</tr>
<tr>
<td>Key Resources
• Research Team;
• Test Devices.</td>
<td>Channels
Awareness:
• Digital Marketing
• Public Advertising
Purchase:
• App Stores</td>
</tr>
<tr>
<td colspan="3">Cost Structure
• Marketing;
• Server Hosting;
• Equipment.</td>
<td colspan="2">Revenue Streams
• Commissions</td>
</tr>
</table>

4 Solution Evaluation

This chapter describes the method used to evaluate the developed proof of concept, called the Qualitative Evaluation Framework (QEF).

4.1 QEF

The QEF is a tool for incremental and quantitative measurements of the solution's quality that uses a multidimensional foundation to calculate the percentage of completeness of the system.

Each dimension is divided into factors and these factors are divided into requirements, usually separating the functional from non-functional ones. The requirements have a specific weight to indicate their importance and priority, and an associated evaluation metric, which allows to accurately measure how much of the requirement has been achieved.

Factor	Weight {2, 4, 6, 8, 10}	Requirement	Fullfilment [0,100]
Stop Detection	10	SD01 – Detect Train in Motion	0
	10	SD02 – Detect Arrival at Station	0
	6	SD03 – Detect Train Direction	0
	10	SD04 – Notify User on Detection	0
	4	SD05 – Detect Unexpected Stop	0
Indoor Location	10	IL01 – Detect Exit/Entrance	0

Table 19 – Functionality QEF Dimension

Factor	Weight {2, 4, 6, 8, 10}	Requirement	Fullfilment [0,100]
Maintenance	10	AM01 – Applications does not store sensitive data	0
	6	AM02 – The algorithm must be scalable	0

Table 20 – Adaptability QEF Dimension

4.2 Assessment Methodology

Due to the binary detection of states, the requirements will be assessed through calculations of precision and recall, given by the following formulas:

$$Precision = \frac{truePositives}{truePositives + \boldsymbol{falsePositives}}$$

$$Recall = \frac{truePositives}{truePositives + \boldsymbol{falseNegatives}}$$

The precision indicates the likelihood of correctly detecting a state while recall refers to the algorithm's sensitivity, that is, out of all relevant states that must be detected, how many are in fact detected.

Alerting on detection for every situation guarantees 100% recall but reduces precision if some detections are not relevant, so an acceptable balance between the two variables is ideal.

All values will be collected from test results once the proof of concept is developed, where positives correspond to detections and negatives to the absence of detections.

4.3 Hypothesis

Seeing as most indoor location solutions do not distinguish situations within a moving vehicle such as the metro, the goal of SubwayNav is to prove that locating users and detecting their state in underground networks using smartphone sensors is just as reliable as solutions that rely on infrastructure, or better.

So, the null hypothesis can be defined as:

$$Precision_{Sensors} >\cong Precision_{Infrastructure}$$

And the alternative hypothesis becomes:

$$Precision_{Sensors} \ll Precision_{Infrastructure}$$

5 Development

5.1 Project Plan

SubwayNav is a 6 month-long project, divided into 6 main work packages, each with specific tasks to accomplish (Figure 7). Every week, the state of the project was reviewed in short meetings to define the next course of action, allowing for slight changes in the plan if justified.

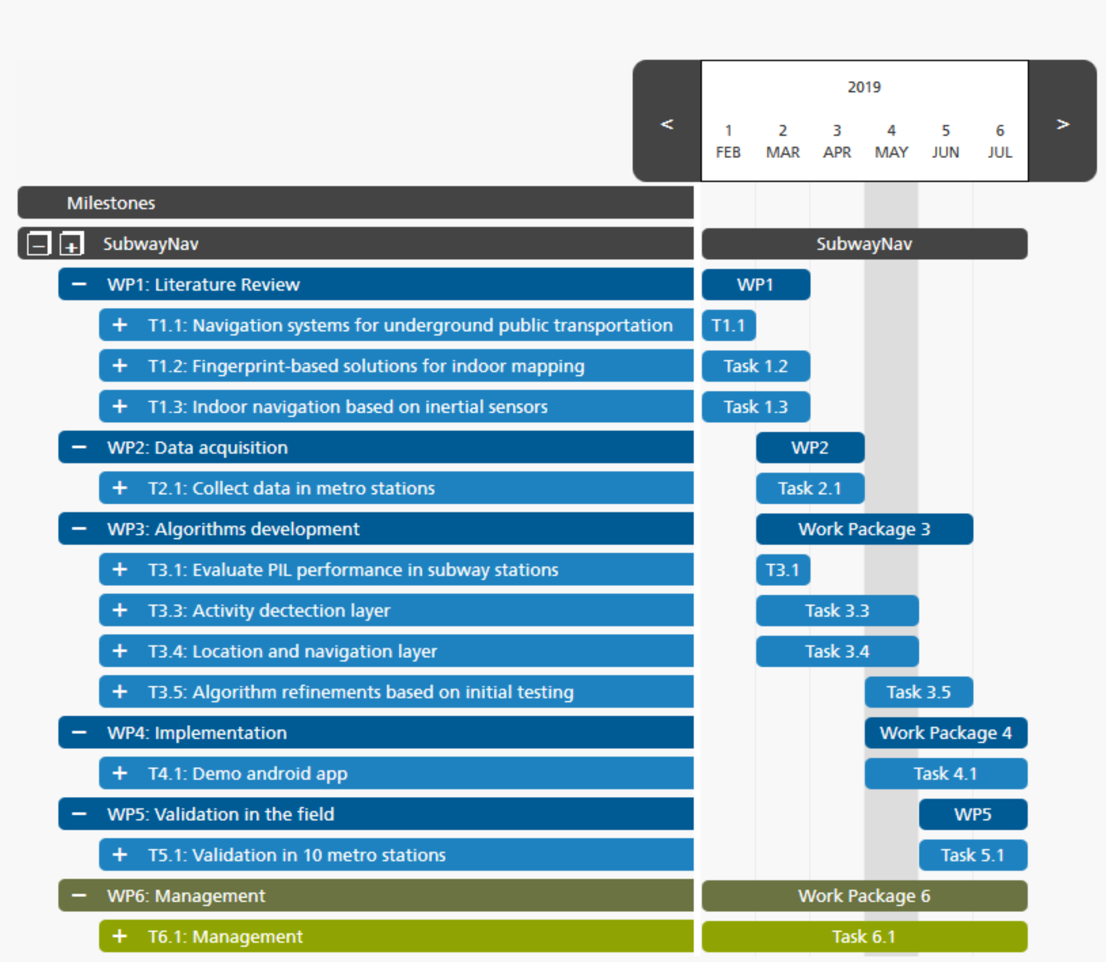


Figure 7 – Project Plan Graph

The literature review is reflected in the State of Art chapter and further decisions regarding the development of the algorithm.

The following work packages are described in the next chapters.

5.2 Activity Detection Layer

5.2.1 Data Acquisition

In order to acquire environment data, a protocol was developed to check possibly impactful (independent) variables, but also guide other researchers to contribute to the project or review it.

Annex 1 includes the protocol in full, which describes its purpose, enumerates the testing scenarios and sensors to acquire data with, the recording procedure and a trajectory example.

The data was recorded using Fraunhofer AICOS's own Recorder app, currently able to record from a wide range of sensors, including inertial sensors and the barometer, as well as annotate events such as a metro arrival or departure. Data was acquired by 3 young adult participants and with 4 distinct Android devices, 2 of which had a functioning barometer. In total, 34 routes were recorded in line D in Porto and 14 were recorded in both Green and Red lines in Lisbon, where routes are mostly underground. Recordings from Lisbon had no barometer data.

Prior to the algorithm's development, the data was plotted and analysed to make any changes to the acquisition procedure, particularly when should events be annotated.

5.2.2 Algorithm Development

Once acquired a significant number of recordings, several experimentations were made using a script written in Python. This script calculates and selects, from the gathered data, key features which, in turn, are used test various digital filters and multiple machine learning models. The results of predictions made with these models allow for the selection of the best performing classifiers to use in a simplified simulation of the final proof of concept.

As displayed in Figure 8, the pipeline begins by reading data and running a selection of signal filters [33], including:

- Moving average filter;
- Butterworth Low-pass filter;
- Butterworth High-pass filter;
- Butterworth Band-pass filter.

All filters are used separately with many parameters regarding order and cut-off frequency value(s), and each result, including the original data, is used for testing. Then, the data containing every route is reshaped for different window sizes and reformatted to run a feature extraction and selection tool (tsfresh [34]), based on which intervals were annotated as 'Stopped' and 'Moving'. This tool calculates numerous features for time-series, as well as allow the addition of custom features. The features are selected by tsfresh according to their p-value, using the Benjamini Hochberg procedure [35]. Optionally, to reduce the number of features while maintaining or improving desired accuracy of a model, mlxtend's SFS (Sequential Feature Selector) function [36] is used to perform forward feature selection.

Now, a variety of classification-type scikit-learn [37] estimators (i.e. classifier) is trained with the features calculated for most routes, leaving at least one route out for testing. Additionally, each of the classifiers' hyper-parameters are tuned using a Random Search Cross Validation method to achieve even better results.

Once trained, every classifier is tested in a simulation of the mobile app, that is, stops will be counted for each of the routes and both Precision and Recall are calculated. A Leave-One-Out strategy was adopted after first resorting to cross validation, to assure data of different routes are not mixed.

Thus, all the best performing window sizes, filters, features and classifiers are extracted from the results to be used in the developed prototype. Additionally, the outputs of crucial phases are saved in case of error, interruption or intent to reuse.

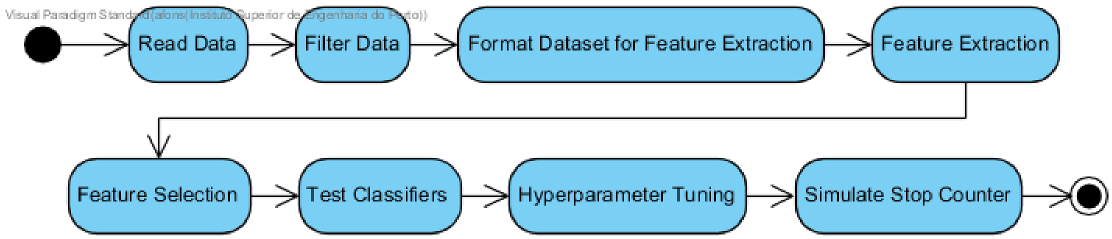


Figure 8 – Activity detection layer – activity pipeline

5.2.3 Refinements, Testing and Model Selection

Initial attempts used over a hundred features so, to limit the processing required to calculate every selected feature, SFS was used to reduce features to a maximum of 12.

Additionally, the confusion matrix alone was not enough to evaluate each classifier's performance. Therefore, a separate score of precision and recall was calculated from the stop counter simulation's results. Since predictions alternated, a state of "stopped" or "moving" is confirmed whenever a certain number of equal predictions succeeded one another. Taking the length of the windows into account as well, a minimum of 3 predictions was chosen to avoid both incorrect and missed increments.

The pipeline exports two best performing classifiers, both with features selected for accelerometer, magnetometer and gyroscope data, but only one with additional barometer features.

Due to time constraints as well as underperformance of the filters, several window sizes and certain classifiers, tables 21 and 22 show the settings that were thoroughly tested, including the results of the final test. Only 2 and 4 second windows were tested, with 50% overlapping values on the latter case.

Sensors	AdaBoost	DecisionTree	RandomForest
Acc., Mag., Gyro. (191 Stops)	Precision: 96.9% Recall: 82.1%	Precision: 82.8% Recall: 63.4%	Precision: 79.2% Recall: 68%
Acc., Mag., Gyro., Bar. (121 Stops)	Precision: 96.6% Recall: 95.8%	Precision: 87.9% Recall: 78.5%	Precision: 95.1% Recall: 96.6%
Bar. (121 Stops)	Precision: 95.1% Recall: 88.3%	Precision: 93.5% Recall: 90.9%	Precision: 95.3% Recall: 92.7%

Table 21 – Last Results for 2s Window No Overlap

Sensors	AdaBoost	DecisionTree	RandomForest
Acc., Mag., Gyro. (111 Stops)	Precision: 90.9% Recall: 82.5%	Precision: 86% Recall: 67.8%	Precision: 83.9% Recall: 66.9%
Acc., Mag., Gyro., Bar. (109 Stops)	Precision: 95.8% Recall: 85.3%	Precision: 92.1% Recall: 75.2%	Precision: 89.3% Recall: 77%
Bar. (109 Stops)	Precision: 97.2% Recall: 94.5%	Precision: 96.9% Recall: 85.5%	Precision: 93.5% Recall: 90.9%

Table 22 – Last Results for 4s Window 50% Overlap

Based on the results, the selected model was the AdaBoostClassifier [38], an ensemble model, with a DecisionTree as a base estimator, using a 2 second window without overlapping values, and no applied filter. This classifier is adequate because it is not influenced by class imbalance unlike those discarded in earlier phases. The barometer-only model was discarded due to underperformance.

5.3 Implementation

To put the trained models to use in the mobile proof of concept, they are first converted from Python to Java using the sklearn-porter library [39], making them compatible to run in an Android app.

To calculate features from sensor data as in the pipeline in real time, Chaquopy [40] is used to run python scripts in Android, then the result is passed to the respective classifier to obtain a prediction. Since the classifier is bound to change predictions despite the real situation not changing, the state is confirmed based on repeated predictions and the duration of state when switching to another. So, each stop is counted whenever the moving state lasts long enough to be considered a trip between stations and when 3 consecutive “Stopped” predictions are made.

With a functional stop counter, we compare the duration between two stops to those provided by Google, which are approximations, to detect whether the correct vehicle was taken. A reasonable margin is used, selected empirically from average and standard deviation calculations from the real data acquired initially.

In order to navigate users, the starting station is detected using the Google Places API [41] once a destination is chosen. After confirming the current location, the number of stops and line changes is calculated and displayed. Once the stop counter reaches the penultimate stop, the user is warned to get off at the next station.

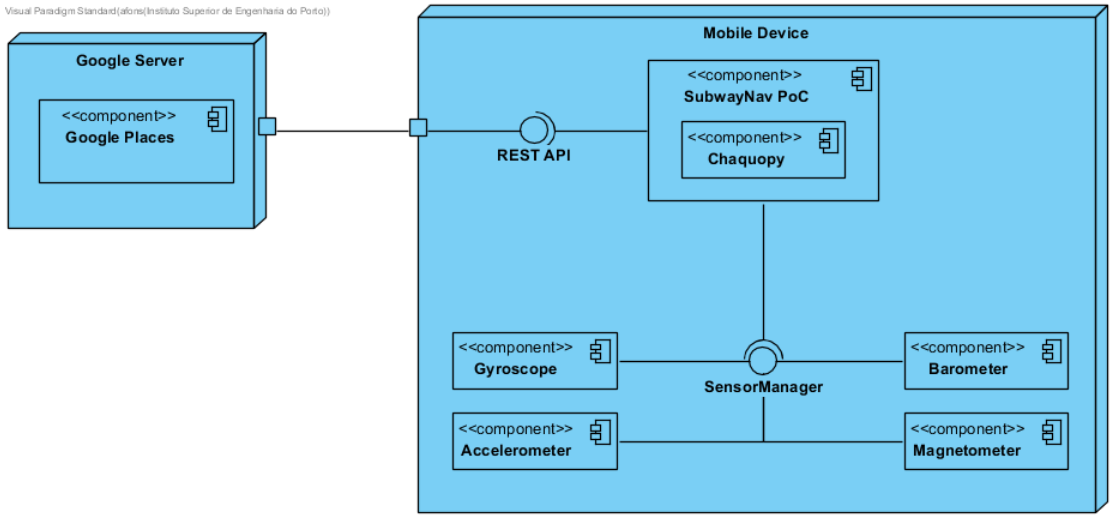


Figure 9 – Final Deployment Diagram

Figure 9 represents the final architecture of the developed prototype, which depends on data received from the sensors as well as the Google Places API. Should the API fail, other options are available that count stations without providing navigational information. However, if the sensors fail or register anomalous values, the system does not function properly.

The main steps of the algorithm are represented in Figure 10, describing the flow of the stop counter functionality. At the start, the UI thread listens to new sensor values and stores them in a Map. Once 2 seconds have passed, it uses all sensor data in an asynchronous task to calculate every feature, make a prediction and update the UI if all conditions meet.

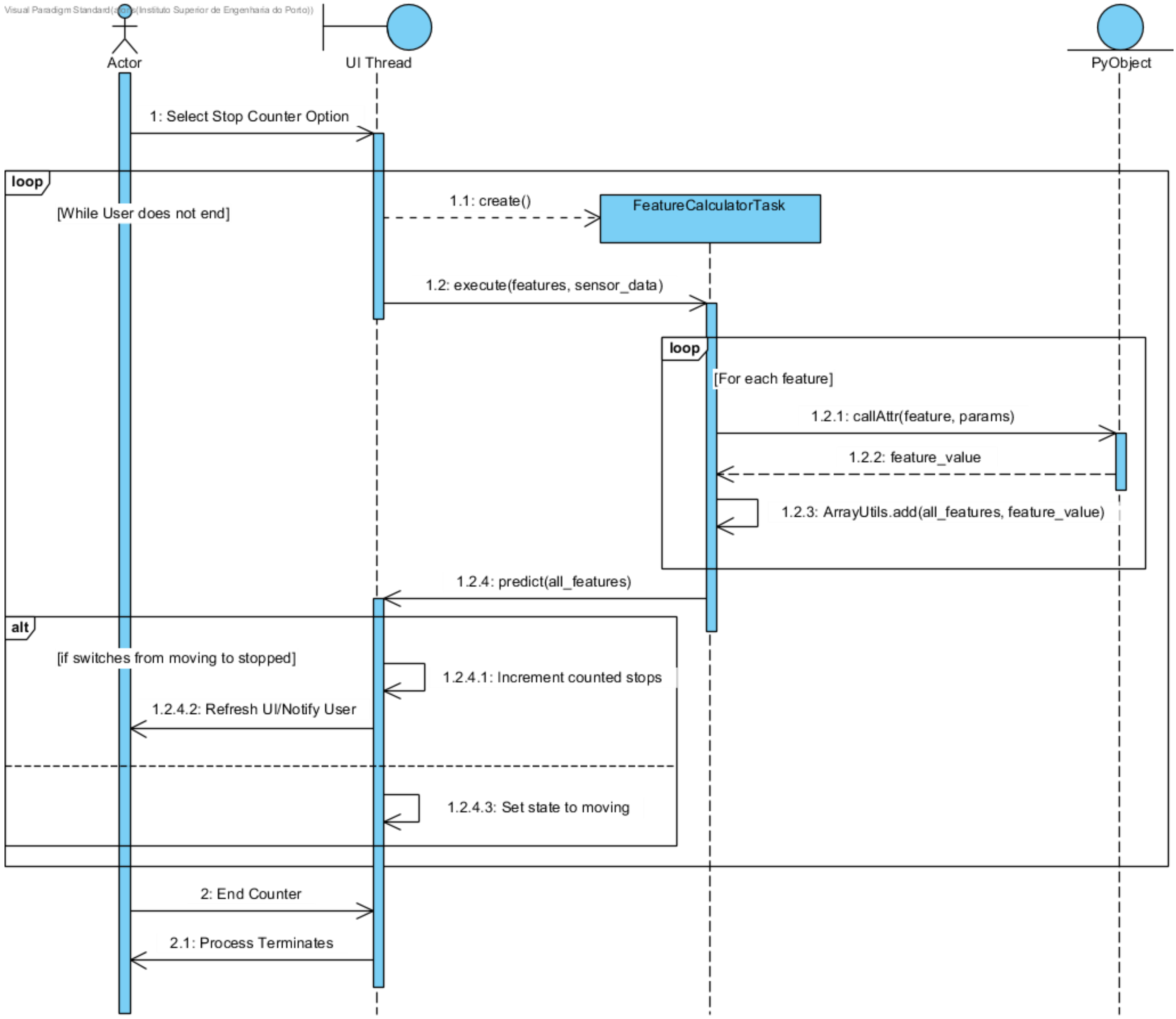


Figure 10 – Stop Counter Sequence Diagram

6 Testing and Results

The proof of concept was tweaked and improved based on preliminary testing to improve performance and fix bugs. Afterwards, following a simple test protocol described in Annex 2 – Test Protocol, testing was carried out involving 3 participants and 4 distinct Android devices, all of which had a functioning barometer. Participants were all young adults with no condition that heavily influences results.

Tests mainly covered line D of the Porto metro network, each trip spanning 5 to 7 underground stops unlike all other lines that are mostly on the surface. Both ways were tested as well for good measure, lasting an average of 9 minutes each way, along with every described scenario. Figure 11 illustrates an example of a route highlighted in pink which was commonly travelled by the testers.

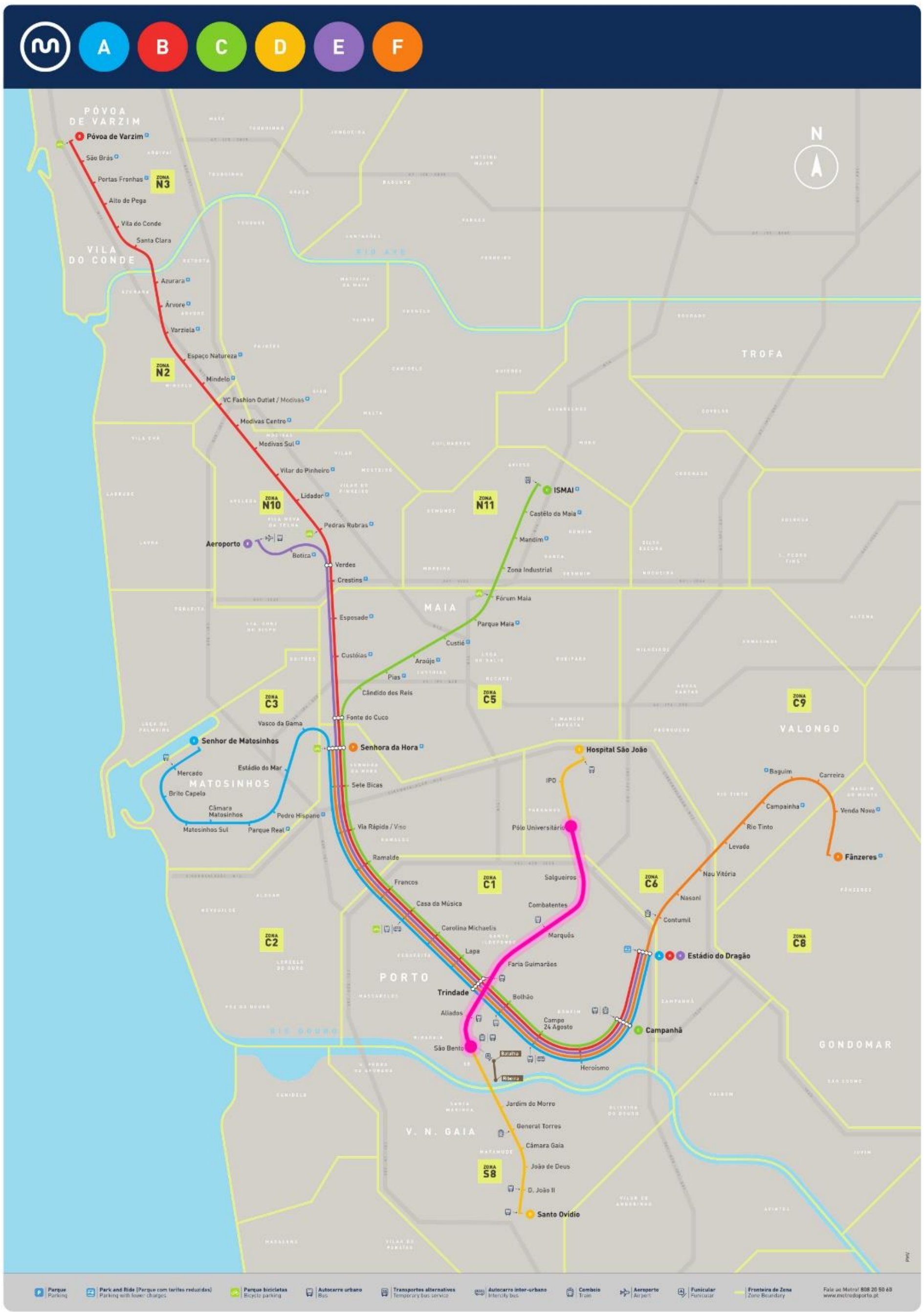


Figure 11 – Example of test route. Extracted from *[42]*

Promising results were obtained as shown on figures in Annex 3, with and without barometer, respectively. In these results, the prediction of the algorithm, not the estimator, is compared with the ground truth which was annotated manually during testing.

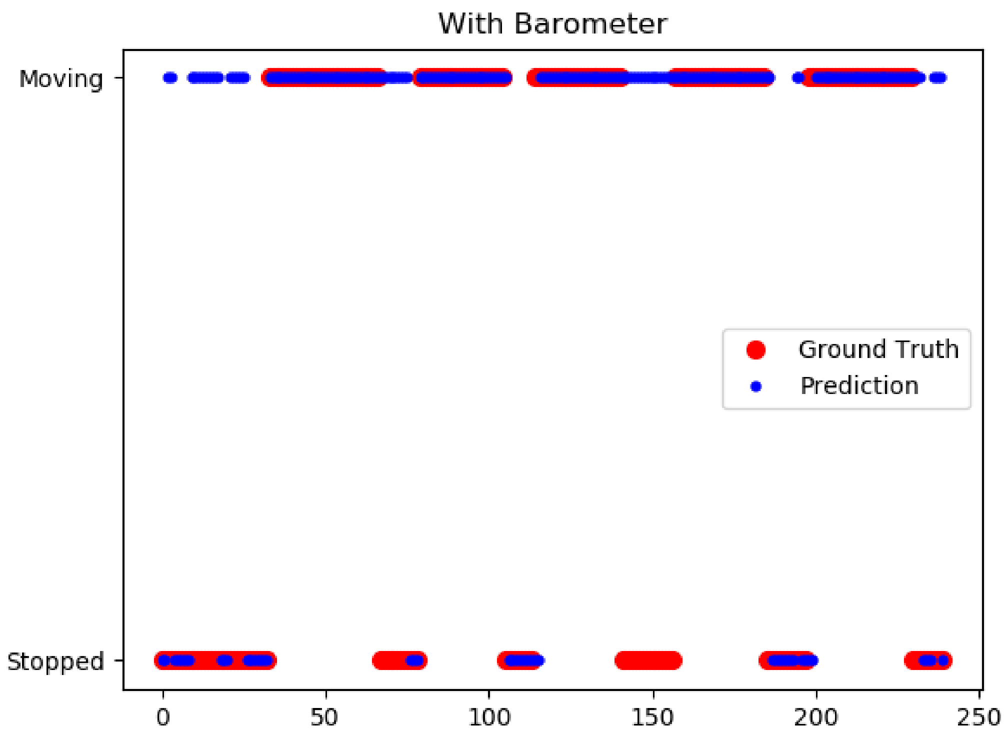


Figure 12 – Test Results using Barometer

In Figure 12, one of the test results is illustrated, showing the state on the Y axis and the prediction number on the X axis. The exemplified route has 5 stops excluding the starting station, one of which was not counted, as indicated on the 4th red line of dots without any overlapping blue dots. Instead, the output of the algorithm remained as "Moving".

Without the additional validations added on top of the predictions, the state would alternate repeatedly in specific scenarios. For instance, when running downstairs in a metro station, the model would predict "Moving", and sometimes predict "Stopped" while on a moving vehicle. So, instead of relying solely on individual predictions of the model, the final algorithm was implemented with validations such as ignoring short intervals while "Moving", thus leading to better results.

Despite the expected delays, if the right stations are counted before warning the user to get off at their destination then the functionality is considered a success.

Sensors	Total Stops	True Positives	False Positives	False Negatives
Mag., Acc., Gyro., Bar.	72	63	0	9
Mag., Acc., Gyro.	50	42	0	8

Table 23 – Final Test Results

Sensors	Precision (%)	Recall (%)	F_1 Score (%)
Mag., Acc., Gyro., Bar.	100	88	93
Mag., Acc., Gyro.	100	84	91

Table 24 – Final Test Result Metrics

The results in Tables 21 and 22, as impressive as they may be, indicate the likelihood of overfitting, considering that not enough testing or data acquisition was done in any environment besides the Porto metro system. This also resulted in several false negatives during uncommon situations such as a second metro departing while the current one is arriving.

In total, using the barometer, the algorithm yielded 100% Precision and 88% recall over 72 stations in the Porto subway network, meaning every counted stop was correct despite missing some. Whereas, without the barometer, results also reached 100% Precision and 84% over 50 stations.

It is also important to note that only lines in Porto were used to test the solution due to insufficient testing volunteers in other locations, even though all scenarios were covered.

7 Conclusions

7.1 Objectives achieved

The planning elaborated by Fraunhofer for the system consisted of many iterations. A standard research workflow was completed from a beginning investigation phase to a test and assessment phase. In this chapter, achieved objectives regarding the solution and adopted practices are addressed following the test phase.

7.1.1 Solution

Table 25 presents the degree of completeness for each requirement during the development of the project and their corresponding work plan.

Factor	Weight {2, 4, 6, 8, 10}	Requirement	Fullfilment [0,100]
Stop Detection	10	SD01 – Detect Train in Motion	100
	10	SD02 – Detect Arrival at Station	100
	6	SD03 – Detect Train Direction	100
	10	SD04 – Notify User on Detection	100
	4	SD05 – Detect Unexpected Stop	0
Indoor Location	10	IL01 – Detect Exit/Entrance	100
Maintenance	10	AM01 – Applications does not store sensitive data	100
	6	AM02 – The algorithm must be scalable	50

Factor	Weight {2, 4, 6, 8, 10}	Requirement	Fullfilment [0,100]
	10	AM03 – Support mobiles with and without barometers	100

Table 25 – Completed QEF table

As observed, the highly prioritized tasks were completed, albeit the suspicions of overfitting to the Porto metro system. To achieve 100% of fulfillment, both recall and precision should exceed 80%.

Unexpected stops are considered in the algorithm but, due to no such scenarios occurring during testing, it is not considered complete. The algorithm is also scalable and easily updatable for Android devices, although the task is half complete because it does not operate in iOS devices.

In total, the final percentage of completeness calculated by QEF is 94%.

7.1.2 Technology & Methodologies

Since all used technologies were unfamiliar to me, the study and the extensive practice translated into significant knowledge acquisition.

Proper application of machine learning practices was key to achieving promising results while avoiding subtle mistakes that influence the training and prediction of the models. For instance, excessive parameter tuning and insufficient data of particular situations would cause overfitting.

Should the project require further improvements, a solid foundation is available to guarantee reliable results as well as take multiple parameters into account, all while saving execution time because of data persistence and available configurations.

7.2 Future Work

Even though the main objectives were achieved, and promising results were obtained, there were changes in the initial plan which, in future iterations, could be considered to improve the prototype. Accurate localization within a station, for instance, was dropped due to time constraints.

In total, the main improvements to consider are:

- Localization within a station
- Further field testing
- Adapt to different environments/Reduce overfitting
- Unit and integration tests with interest of methodologies of continuous integration and delivery.
- Code optimization

7.3 Final Remarks

7.3.1 Personal

Despite the limitations of the technology and lack of data in other subway networks, I consider myself quite satisfied with the tasks accomplished, as well as the skills, habits and lessons learned.

However, for the time being, it is likely that starting with native languages or the python library Tensorflow [43] for mobile development might be more time-efficient and scalable, respectively.

Design patterns or proper UI development practices were not the main focus, but experience with research and machine learning problems is and will be highly valuable in the present and near future.

7.3.2 Solution

The solution met most requirements set by the customer, with valuable capacity for future upgrades and results which exceeded expectations.

Initially, there was little experience in using most technologies and frameworks, such as Chaquopy and sklearn, so, in the final stages, time was devoted to optimizing and cleaning older code snippets.

Through practice, it was noted why Python machine/deep learning frameworks continue to gain popularity. These technologies have made it possible to develop a highly accurate system without requiring high resource consumption from the devices running the application.

Considering their potential to solve everyday problems and automate/simplify existing procedures, this technology is expected to be used more in the future.

Paper/Solution	Method	Results
Tracking Motion Context of Railways Passengers by Fusion of Low-Power Sensors in Mobile Devices [18]	Accelerometer + Magnetometer	Precision: 80% Recall: 82%
Learning to detect the Subway Station Arrival for Mobile Users [20]	Accelerometer + GSM	Precision: 83% Recall: 90%
SubwayPS [19]	Accelerometer + Gyroscope	Accuracy: 85.7%
Estimate a User's Location Using Smartphone's Barometer on a Subway [22]	Barometer	Accuracy: 86% Movement Detection, 58% Stops
Snips [21]	Barometer	Accuracy: >90%

Paper/Solution	Method	Results
SubwayNav	Accelerometer + Gyroscope + Magnetometer + Barometer	Precision: 100% Recall: 88%
SubwayNav	Accelerometer + Gyroscope + Magnetometer	Precision: 100% Recall: 84%

Table 26 – Final Solution Comparison

As observed in Table *26*, all analyzed solutions' results are compared with those obtained with SubwayNav based on precision and recall. Accuracy was not calculated but compared with precision because it better suits the context, one state (moving in a metro) is specific while the other applies to any other situation.

So, because the results are superior to the analyzed approaches, the potential and reliability of SubwayNav was demonstrated successfully.

Bibliography

[1] "Alojamento turístico com crescimento mas em desaceleração - 2018." Instituto Nacional Estatística, 02-Aug-2019.

[2] "Uma Estratégia para o Melhor Destino do Mundo." [Online]. Available: http://www.turismodeportugal.pt/pt/Media/Kit_Informativo/Paginas/Uma-Estrat%C3%A9gia-para-o-Melhor-Destino-do-Mundo.aspx. [Accessed: 12-Feb-2019].

[3] J.-P. Rodrigue, "Transportation and the Environment," in *The Geography of Transport Systems*, 4th ed., New York, 2017.

[4] "Resolução do Conselho de Ministros 173/2018, 2018-12-14," *Diário da República Eletrónico*. [Online]. Available: https://dre.pt/web/guest/pesquisa/-/search/117384027/details/normal?q=expans%C3%A3o+metro. [Accessed: 27-Aug-2019].

[5] "Transporte de passageiros continua a aumentar - 1.º Trimestre de 2019." Instituto Nacional Estatística, 06-Jun-2019.

[6] "What is GNSS?," 01-Mar-2016. [Online]. Available: https://www.gsa.europa.eu/european-gnss/what-gnss. [Accessed: 01-Feb-2019].

[7] "Garmin | What is GPS?" [Online]. Available: https://www8.garmin.com/aboutGPS/. [Accessed: 01-Feb-2019].

[8] G. Gruman, "Beacons are harder to deploy than you think," *InfoWorld*, 15-Sep-2015. [Online]. Available: https://www.infoworld.com/article/2983166/bluetooth/beacons-are-harder-to-deploy-than-you-think.html. [Accessed: 12-Feb-2019].

[9] "Engineering against all odds, or how NYC's subway will get wireless in the tunnels," *TechCrunch*. .

[10] "How Does Google Maps Work?," *MakeUseOf*. [Online]. Available: https://www.makeuseof.com/tag/technology-explained-google-maps-work/. [Accessed: 13-Feb-2019].

[11] "What is Google Maps? - Definition from WhatIs.com," *WhatIs.com*. [Online]. Available: https://whatis.techtarget.com/definition/Google-Maps. [Accessed: 13-Feb-2019].

[12] "About Us | Moovit," *Moovit - The world's #1 transit app*. [Online]. Available: https://www.company.moovit.com/about. [Accessed: 13-Feb-2019].

[13] "Smartphone sensor fusion and indoor positioning." [Online]. Available: http://www.indoornavigation.com/wiki-en/smartphone-sensor-fusion-and-indoor-positioning. [Accessed: 10-Feb-2019].

[14] "Ubirider: urban mobility, the smart way," *Ubirider: urban mobility, the smart way*. [Online]. Available: https://ubirider.com/. [Accessed: 06-Feb-2019].

[15] Blaz, "Indoor Positioning Systems based on BLE Beacons - Basics," *Locatify*. .

[16] "Inertial Sensors," *Dr. Kostas Alexis*. [Online]. Available: http://www.kostasalexis.com/inertial-sensors.html. [Accessed: 11-Feb-2019].

[17] K. Khoshelham and S. Zlatanova, “Sensors for Indoor Mapping and Navigation,” *Sensors*, vol. 16, no. 5, May 2016.

[18] T. Higuchi, H. Yamaguchi, and T. Higashino, “Tracking motion context of railway passengers by fusion of low-power sensors in mobile devices,” in *Proceedings of the 2015 ACM International Symposium on Wearable Computers - ISWC ’15*, Osaka, Japan, 2015, pp. 163–170.

[19] T. Stockx, B. Hecht, and J. Schöning, “SubwayPS: towards smartphone positioning in underground public transportation systems,” in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - SIGSPATIAL ’14*, Dallas, Texas, 2014, pp. 93–102.

[20] “Learning to Detect the Subway Station Arrival for Mobile Users,” *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/299682447_Learning_to_Detect_the_Subway_Station_Arrival_for_Mobile_Users. [Accessed: 16-Jan-2019].

[21] J. Dureau, “Underground location tracking,” *Snips Blog*, 22-Mar-2016. .

[22] S. Hyuga, M. Ito, M. Iwai, and K. Sezaki, “Estimate a user’s location using smartphone’s barometer on a subway.” MELT@SIGSPATIAL 2015, 2015.

[23] “An In-depth Explanation of the Venturi Effect and its Applications,” *ScienceStruck*. [Online]. Available: https://sciencestruck.com/explanation-applications-of-venturi-effect. [Accessed: 16-Jan-2019].

[24] Locatify, “Precise Indoor Positioning is finally here thanks to Ultra-wideband RTLS,” *Locatify*. .

[25] R. B. B. dos Santos, “Human Crowdsourcing Data for Indoor Location Applied to Ambient Assisted Living Scenarios,” Nov. 2018.

[26] V. Guimarães *et al.*, “A motion tracking solution for indoor localization using smartphones,” *ResearchGate*, 2016. [Online]. Available: https://www.researchgate.net/publication/310498312_A_motion_tracking_solution_for_indoor_localization_using_smartphones. [Accessed: 12-Feb-2019].

[27] “Saaty, T.L., 1980. ‘The Analytic Hierarchy Process.’ McGraw-Hill, New York.” .

[28] I. Khan, S. Khusro, S. Ali, and J. Ahmad, “Sensors are Power Hungry: An Investigation of Smartphone Sensors Impact on Battery Power from Lifelogging Perspective,” *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/317494054_Sensors_are_Power_Hungry_An_Investigation_of_Smartphone_Sensors_Impact_on_Battery_Power_from_Lifelogging_Perspective. [Accessed: 19-Jan-2019].

[29] “Front End Innovation - FEI.” [Online]. Available: http://www.frontendinnovation.com/fei. [Accessed: 26-Jan-2019].

[30] A. Martikainen, “Front End of Innovation in Industrial Organization,” *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/324208591_Front_End_of_Innovation_in_Industrial_Organization. [Accessed: 26-Jan-2019].

[31] V. Allee, “Value network analysis and value conversion of tangible and intangible assets,” *J. Intellect. Cap.*, vol. 9, no. 1, pp. 5–24, Jan. 2008.

[32] “The 20 Minute Business Plan: Business Model Canvas Made Easy,” *Alex Cowan*. [Online]. Available: https://www.alexandercowan.com/business-model-canvas-templates/. [Accessed: 06-Feb-2019].

[33] Saurabh, “Band Pass-Stop, High Pass and Low Pass Filter | Full Explaination,” *Electronics For You*, 22-Dec-2018. .

[34] “GitHub - blue-yonder/tsfresh: Automatic extraction of relevant features from time series:” [Online]. Available: https://github.com/blue-yonder/tsfresh. [Accessed: 07-Jul-2019].

[35] Y. Benjamini and D. Yekutieli, “The control of the false discovery rate in multiple testing under dependency,” *Ann. Stat.*, vol. 29, pp. 1165–1188, 2001.

[36] “Sequential Feature Selector - mlxtend.” [Online]. Available: http://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector/. [Accessed: 07-Jul-2019].

[37] “scikit-learn: machine learning in Python — scikit-learn 0.21.2 documentation.” [Online]. Available: https://scikit-learn.org/stable/. [Accessed: 07-Jul-2019].

[38] A. Desarda, “Understanding AdaBoost,” *Towards Data Science*, 17-Jan-2019. [Online]. Available: https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe. [Accessed: 12-Jul-2019].

[39] D. Morawiec, *Transpile trained scikit-learn estimators to C, Java, JavaScript and others.: nok/sklearn-porter*. 2019.

[40] “The easiest way to use Python in your Android app,” *Chaquopy*. [Online]. Available: https://chaquo.com/chaquopy. [Accessed: 12-Jul-2019].

[41] “Overview | Places SDK for Android,” *Google Developers*. [Online]. Available: https://developers.google.com/places/android-sdk/intro. [Accessed: 12-Jul-2019].

[42] “Metro do Porto - A Vida em Movimento.” [Online]. Available: https://www.metrodoporto.pt/. [Accessed: 04-Aug-2019].

[43] “TensorFlow,” *TensorFlow*. [Online]. Available: https://www.tensorflow.org/. [Accessed: 04-Aug-2019].

Annex 1 – Data Acquisition Protocol

- ## Purpose:

Acquire environment data during subway trips using smartphone sensors to check for patterns, and later identifying distinct situations based on the patterns and variations. Through these detections, the number of stations can be counted to better guide users to their desired destination.

- ## Scenarios:

  - Mobile device in hand
  - Mobile device in pocket
  - Standing
  - Sitting Down

- ## Sensors:

  - **Obligatory**
    - Accelerometer
    - Magnetometer
    - Gyroscope
  - **Optional**
    - Barometer
    - Microphone
    - Bluetooth
    - Location
    - Wi-Fi

- ## Trajectory Example:

  - **Porto**
    - Start in Pólo Universitário, line D, direction towards Santo Ovídio
    - Leave at Trindade (5th stop)
    - Reverse

  - **Lisbon**
    - Start in Alameda, green line, direction towards Cais do Sodré
    - Leave at Cais do Sodré (7th stop)
    - Reverse

- Steps:
    - Begin recording on metro arrival
    - Get in metro
    - Tap ‘SITTING’ or ‘STANDING’ buttons to annotate the current scenario
    - Tap ‘STATION DEPARTING’ button once when the metro accelerates and again when entering a tunnel
    - Tap ‘STATION ARRIVAL’ button once when the metro exits a tunnel and again when it comes to a full stop at a station
    - Tap ‘SUDDEN STOP’ on an unexpected full stop between stations, and ‘SUDDEN MOVEMENT’ once it resumes its course (2^{nd} tap is irrelevant)
    - Tap ‘OPEN SPACE’ button when the metro moves from the tunnel to an open space and again when entering the tunnel
    - End recording when leaving the metro

- Notes:
    - Routes and stops must be mostly, if not all, underground
    - Use various mobile devices to check for consistency
    - Create a group dedicated to acquisitions (e.g. “SubwayNav”) and name samples after the scenario and route (e.g. “Hand_Polo_Trindade”, “Pocket_Chiado_MPombal”)

Annex 2 – Test Protocol

Purpose:

Accompany and analyse the performance of the SubwayNav proof of concept application, in order to identify and correct bugs as well as judge the reliability of the solution.

Requirements:

- Permission to use Location & Read/Write to external storage
- WiFi or mobile data
- Android with working Accelerometer, Magnetometer and Gyroscope. Optionally, a working barometer.

Scenarios:

- Mobile device in hand
- Mobile device in pocket
- Standing
- Sitting Down

Steps:

- Begin outside of any station
- Select the Stop Counter option and, afterwards, the Pick Destination option
- Type and select an available station as a destination, preferably 5 to 10 stops.
- Confirm if a dialog is displayed with the correct station, retry until it is detected
- Follow displayed instructions, tracking the number of detected stops
- Important: Press "Annotate Movement" button whenever the vehicle is in motion **until** it stops again

Optional Steps:

- Take an incorrect vehicle, watch for a warning
- Go past the correct destination, observe an alert, informing of how many stops away the destination is now.

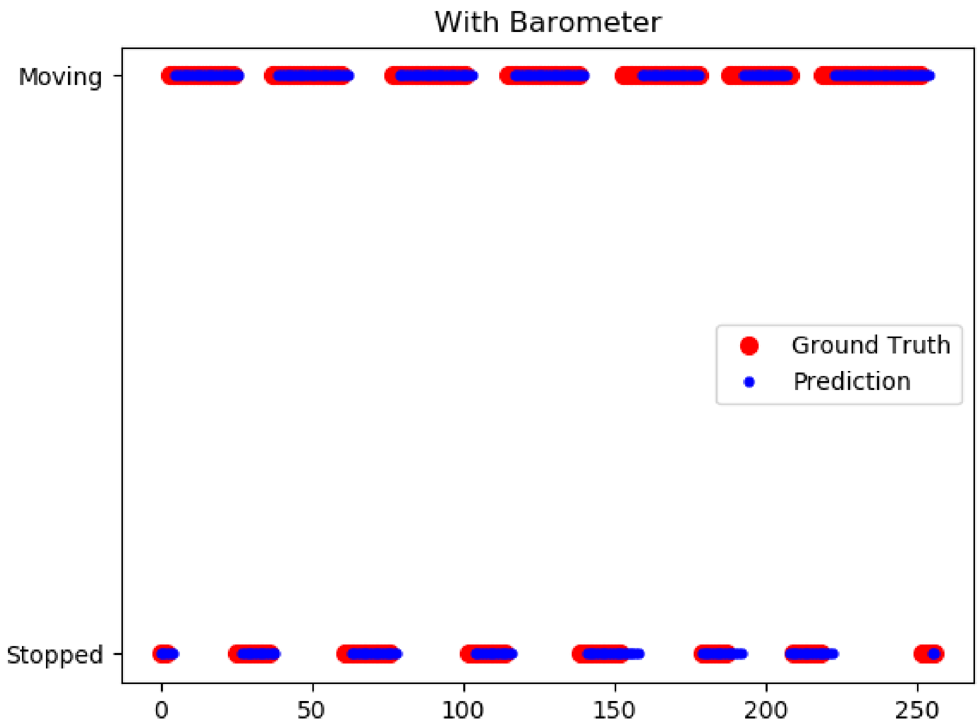
Notes:

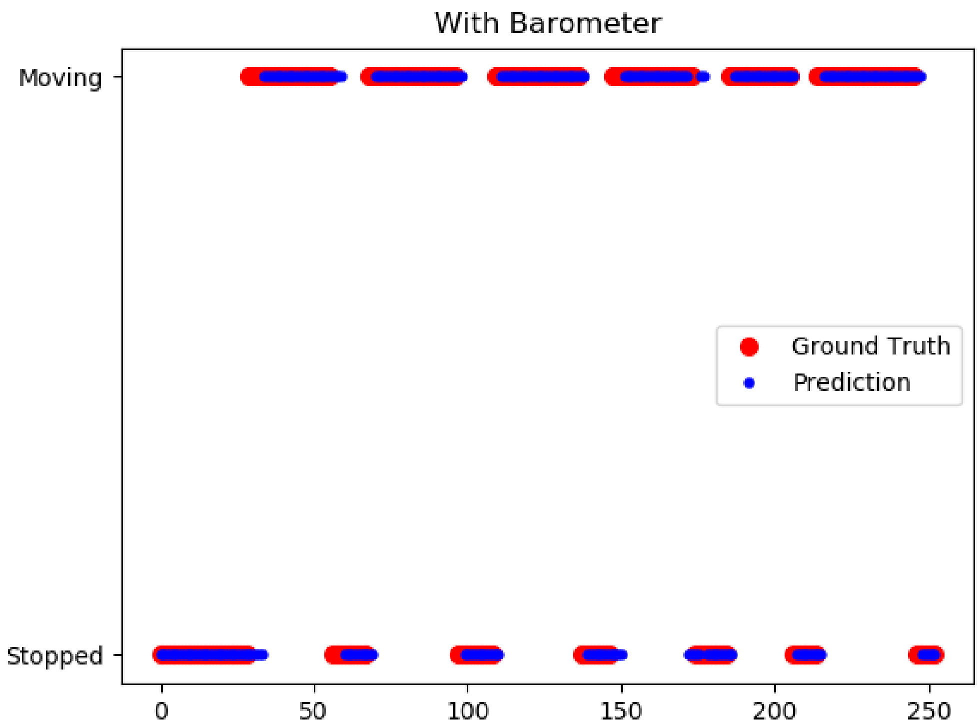
- Use various Android devices to check for consistency
- Act as natural as possible

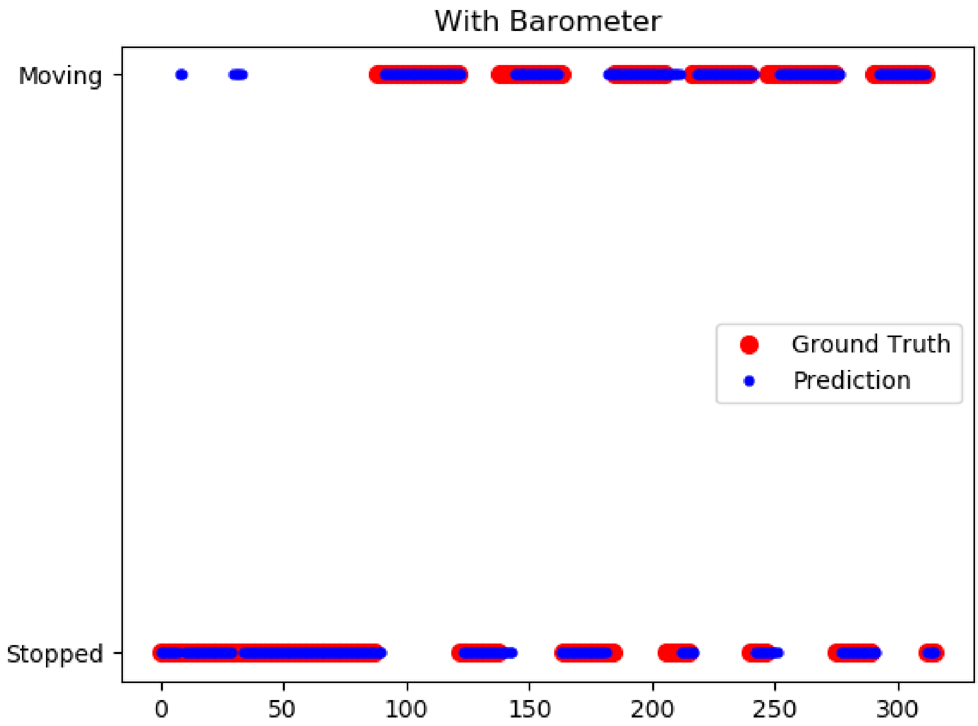
Annex 3 – Algorithm Results

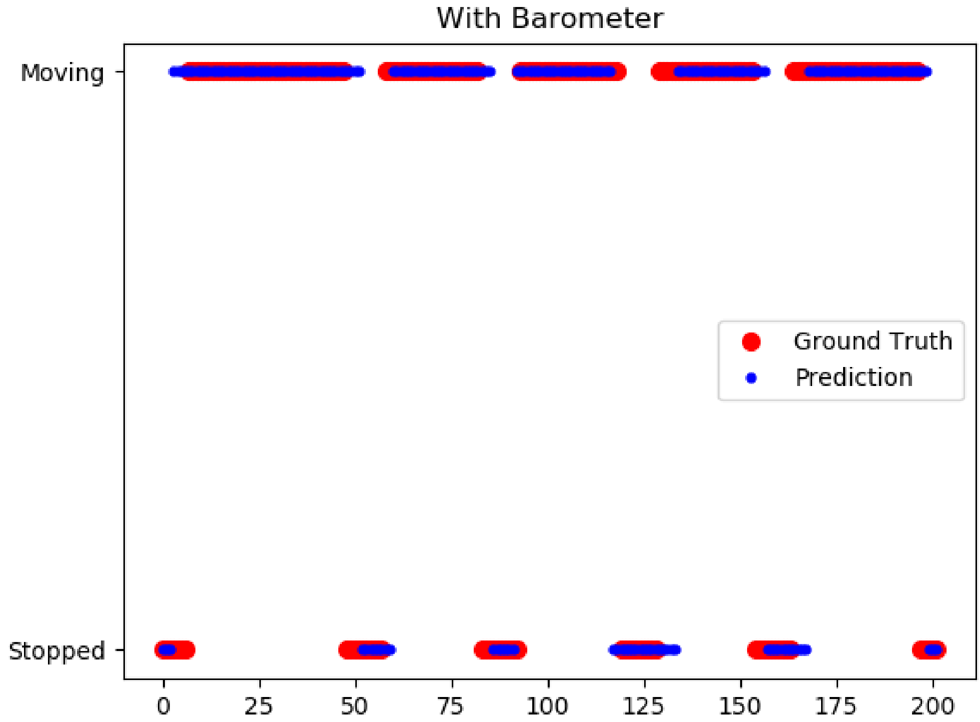
In this section, all results from the SubwayNav prototype field tests are illustrated, comparing the truth as annotated by participants in red with the algorithm's prediction in blue. The user's state on the Y axis and the prediction number on the X axis.

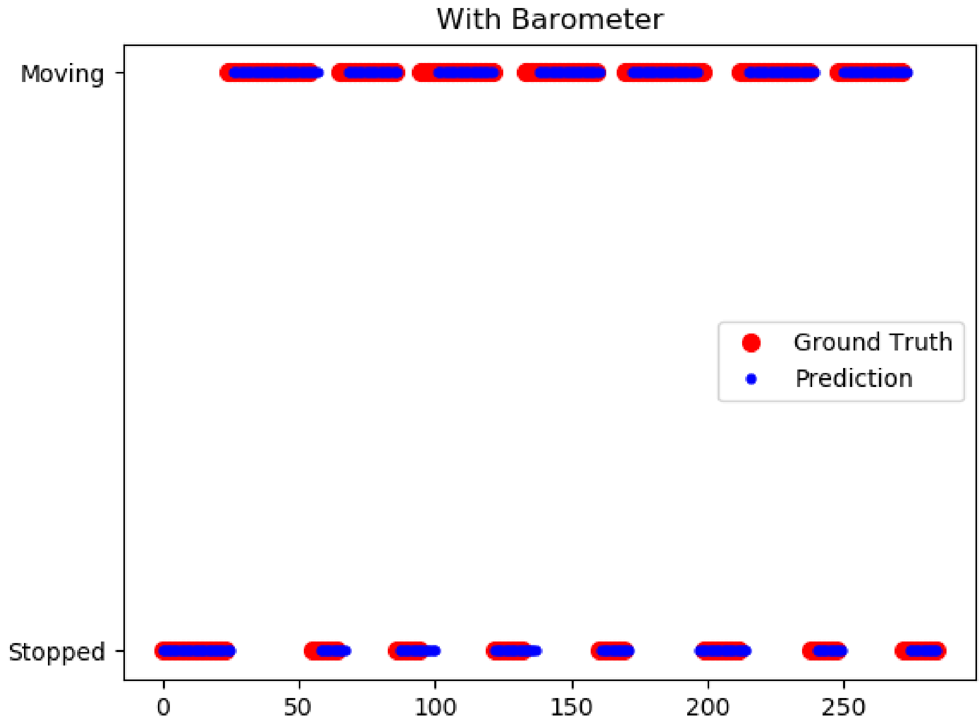
With Barometer:

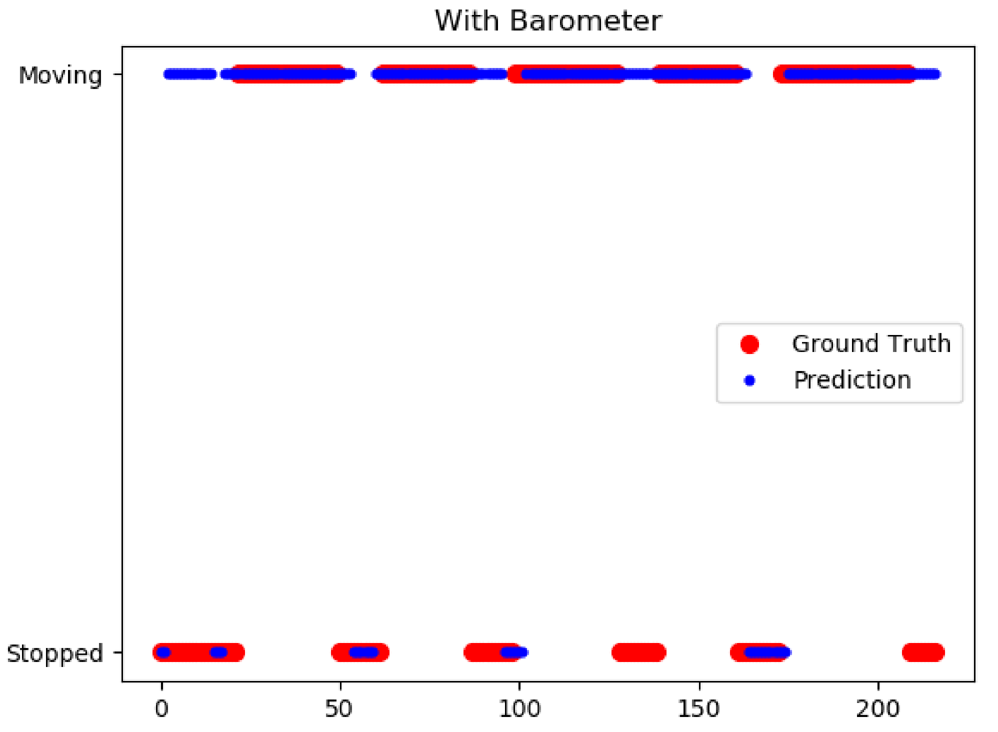



With Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200
250

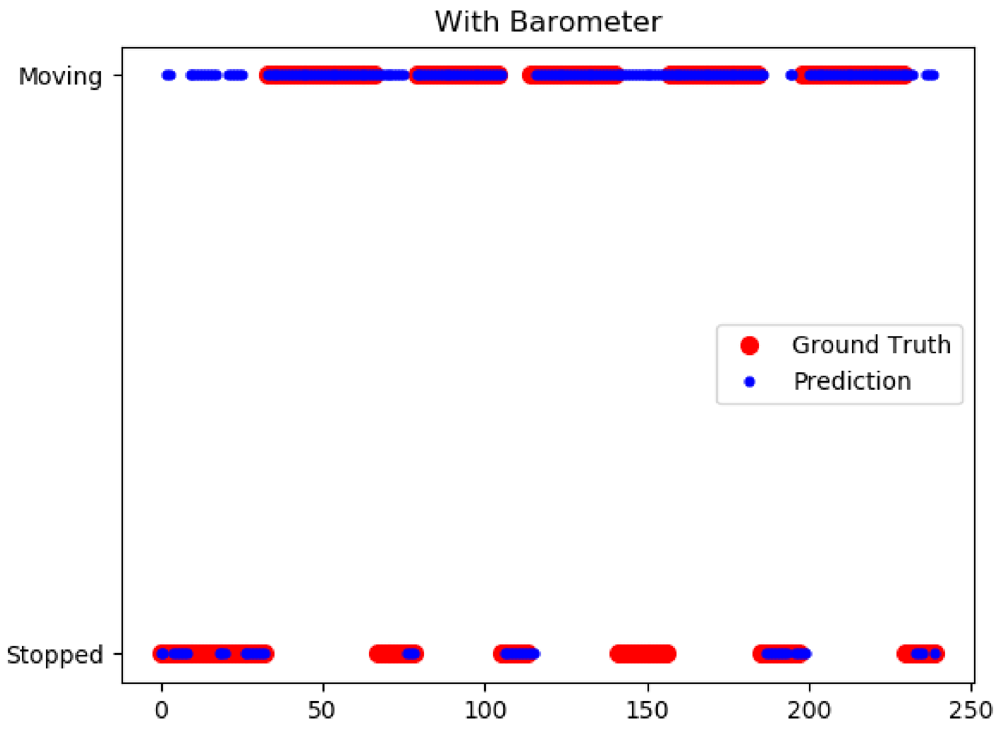

With Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200
250
300


With Barometer
Moving
Stopped
Ground Truth
Prediction
0
25
50
75
100
125
150
175
200

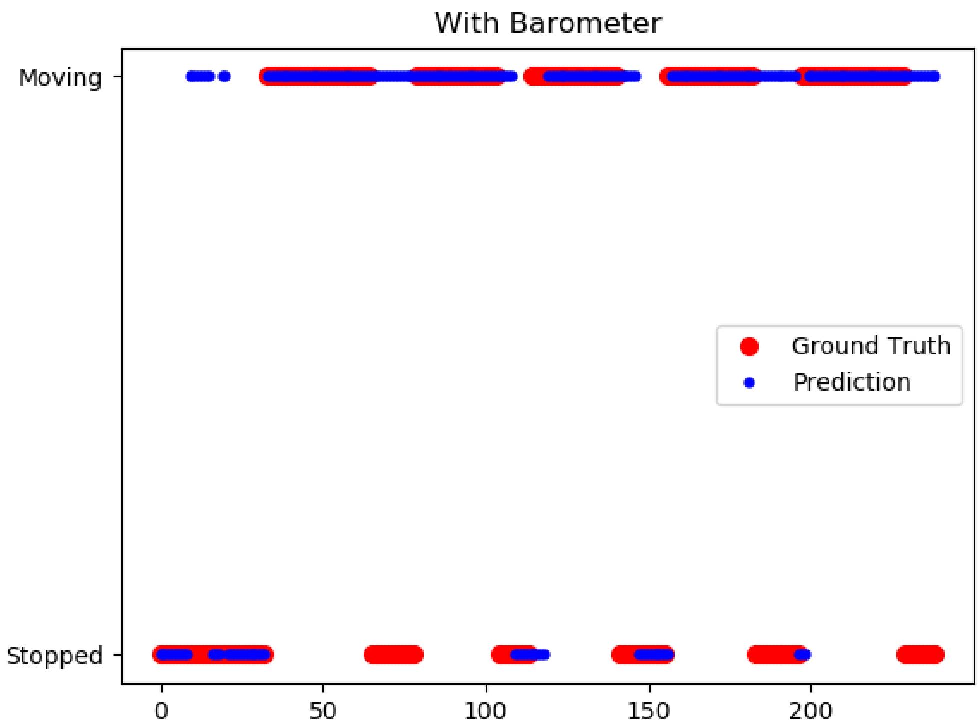

With Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200
250

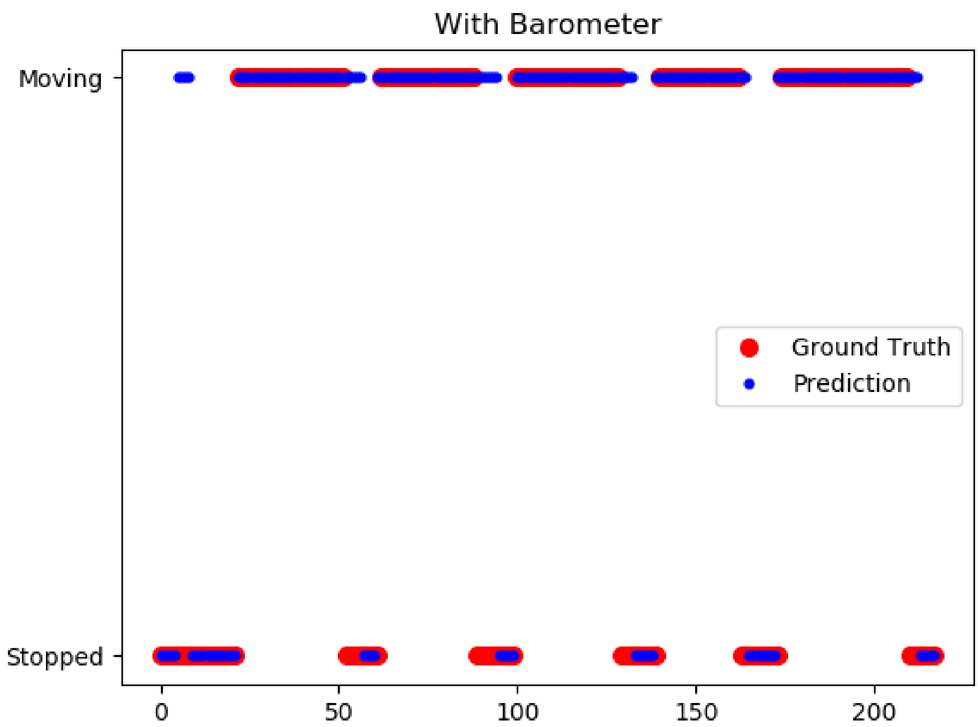


With Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200

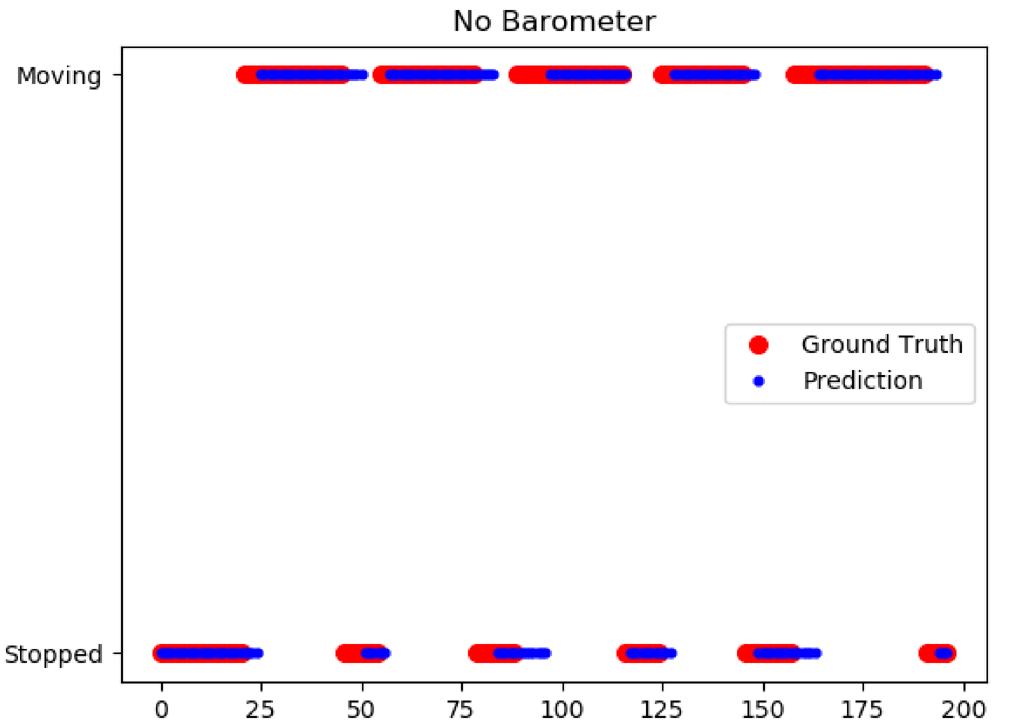


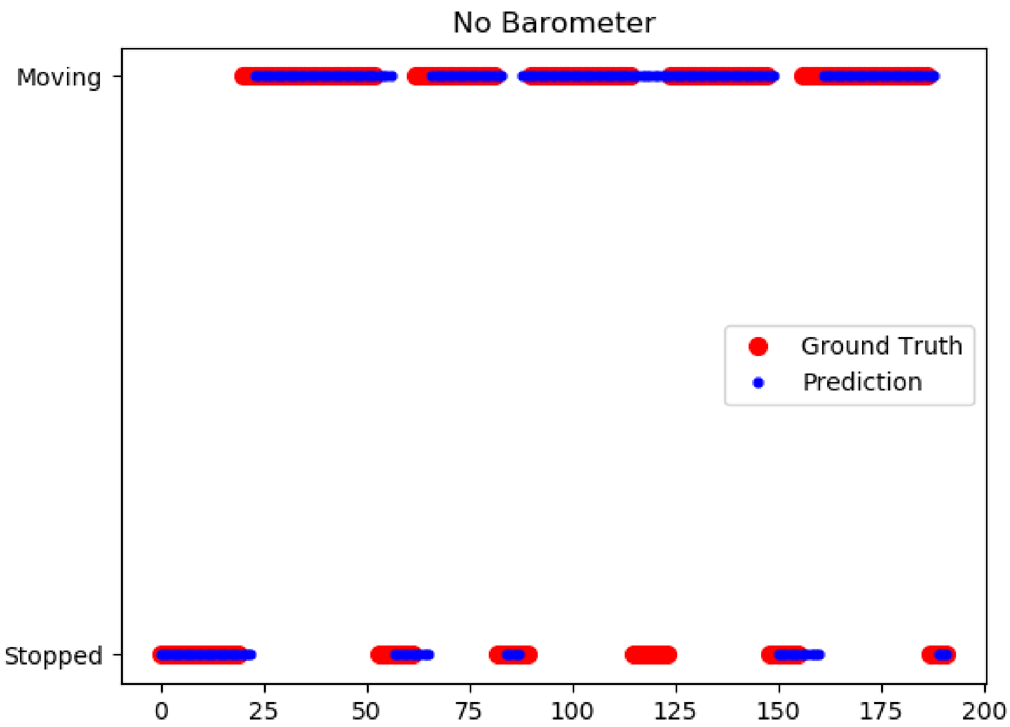
With Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200
250

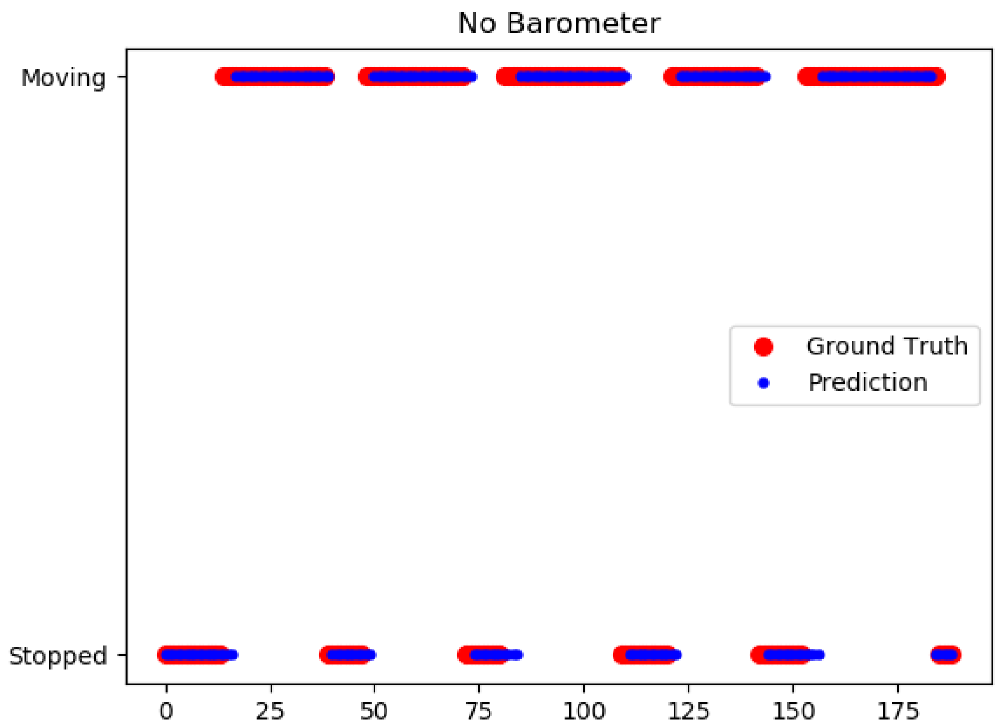

With Barometer
Moving
Stopped
0
50
100
150
200
Ground Truth
Prediction

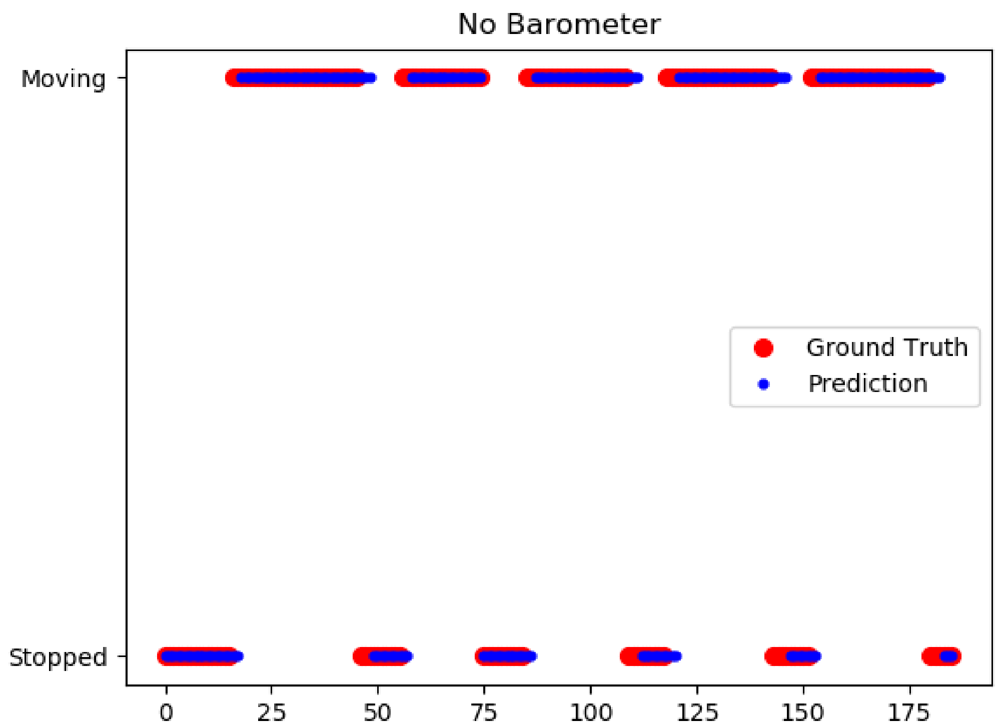

With Barometer
Moving
Stopped
0
50
100
150
200
Ground Truth
Prediction

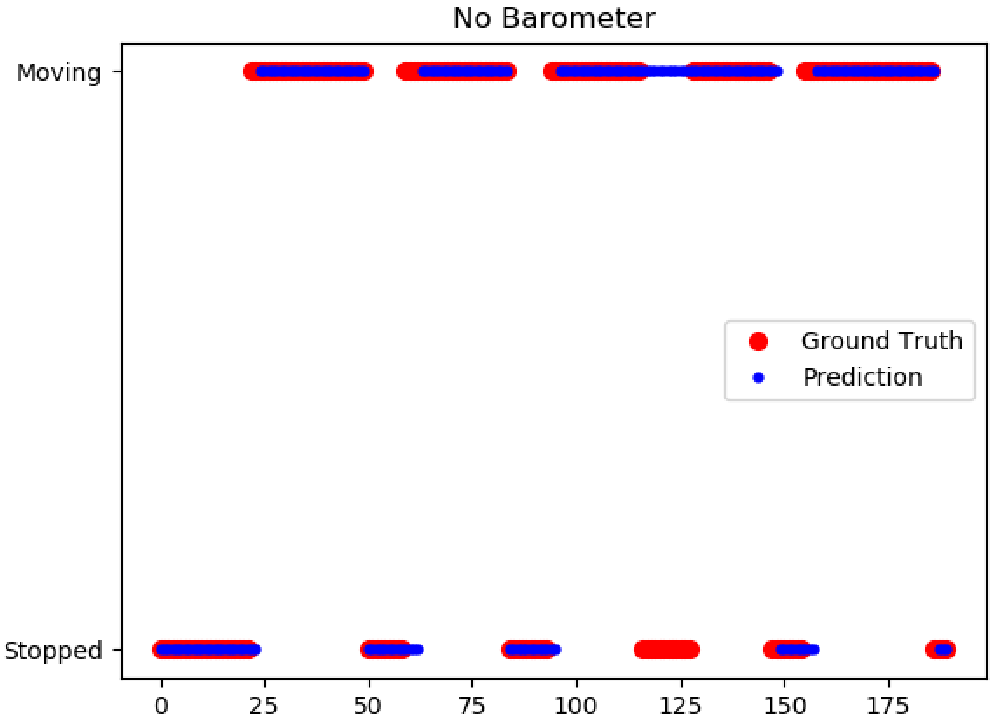
Without Barometer:

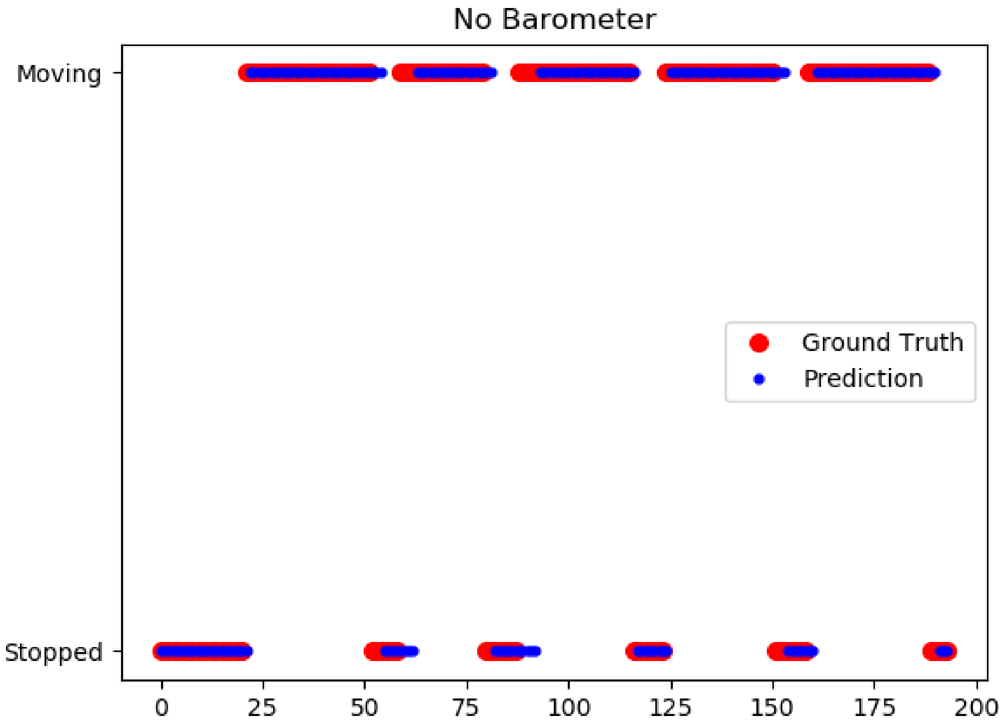


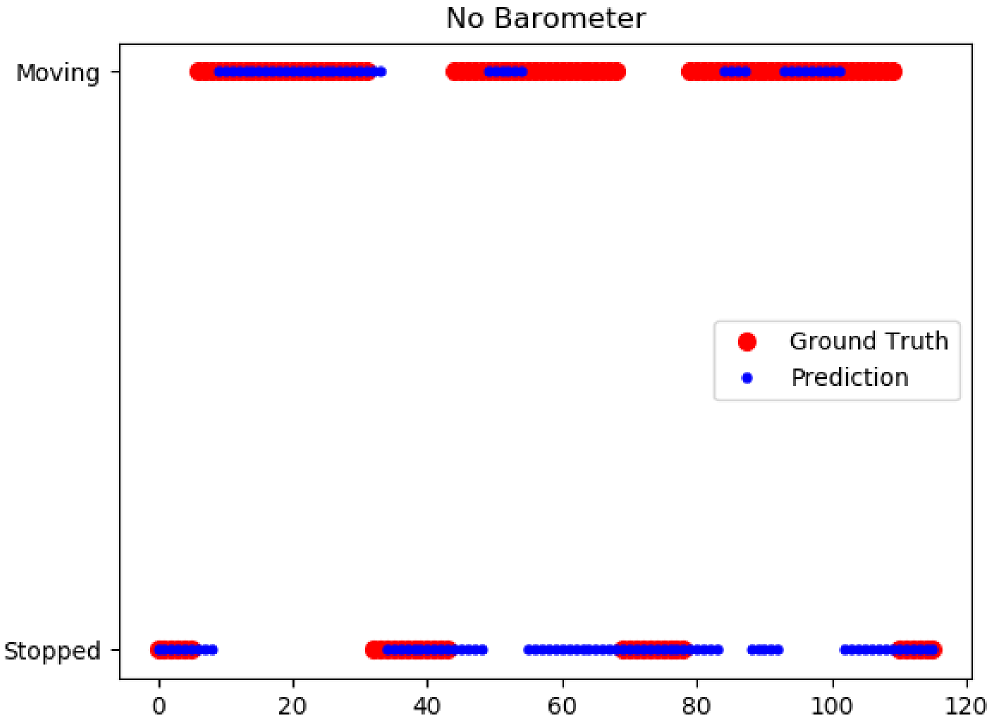


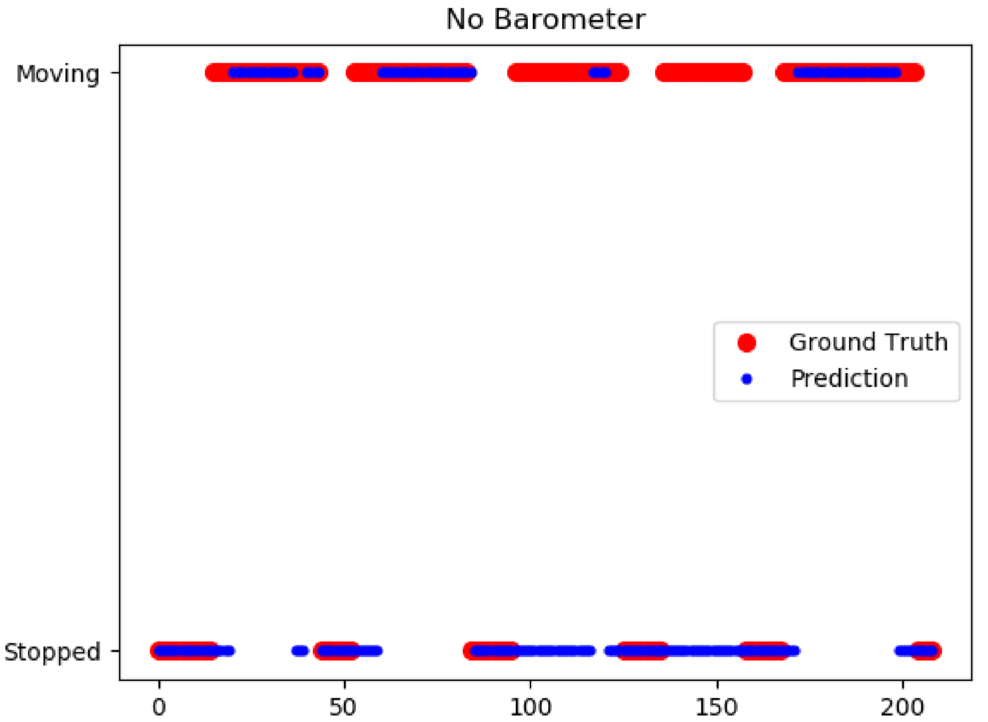

No Barometer
Moving
Stopped
0
25
50
75
100
125
150
175
Ground Truth
Prediction

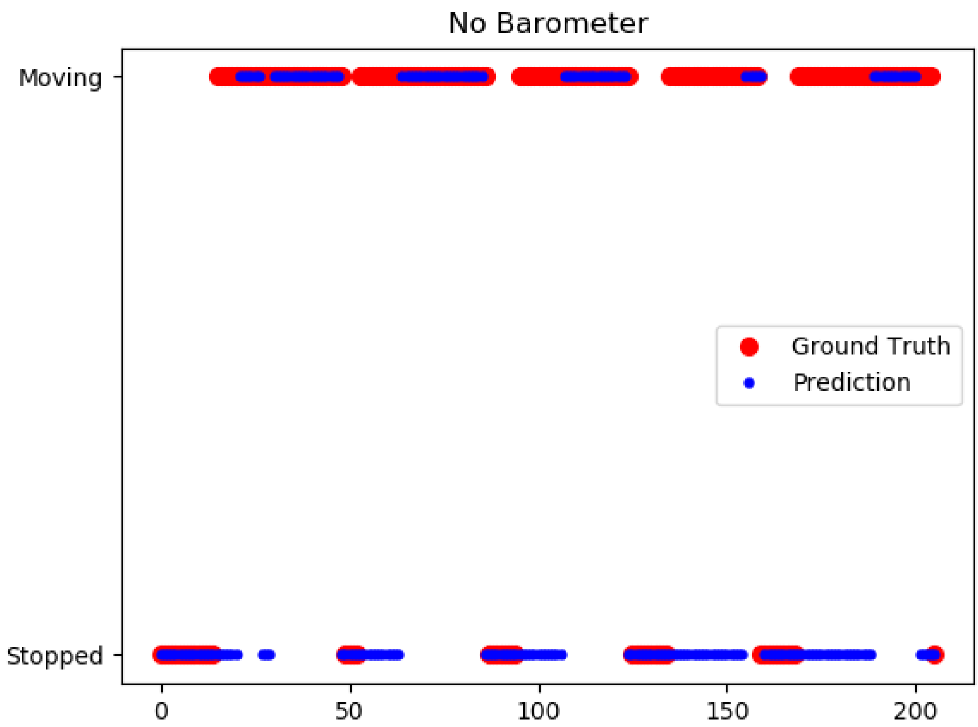

No Barometer
Moving
Stopped
0
25
50
75
100
125
150
175
Ground Truth
Prediction


No Barometer
Moving
Stopped
Ground Truth
Prediction
0
25
50
75
100
125
150
175


No Barometer
Moving
Stopped
Ground Truth
Prediction
0
25
50
75
100
125
150
175
200


No Barometer
Moving
Stopped
Ground Truth
Prediction
0
20
40
60
80
100
120


No Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200


No Barometer
Moving
Stopped
Ground Truth
Prediction
0
50
100
150
200