UNIVERSITAT JAUME I

DOCTORAL SCHOOL

BRNO UNIVERSITY OF TECHNOLOGY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF RADIO ELECTRONICS

EXPLOITING WIRELESS COMMUNICATIONS FOR LOCALIZATION: BEYOND FINGERPRINTING

SHORTENED DOCTORAL THESIS

AUTHOR	Ing. Tomáš Bravenec
ADVISORS	Dr. Michael Gould Carlson
	Dr. Joaquín Torres-Sospedra
	doc. Ing. Tomáš Frýza, Ph.D.
OPPONENTS	Dr. Adriano Moreira
	Dr. Enrique Quintana Orti
	Dra. Maria Cristina Rodriguez Sanchez
	Dr. Antonino Crivello
	Dr. Sergi Trilles Oliver

CASTELLON DE LA PLANA, October 2023

Keywords: *machine learning, indoor positioning, privacy, Wi-Fi, 802.11, data analysis, presence detection, occupancy estimation, memory optimization, radio map interpolation.*

ISBN 80-214-
ISSN 1213-4198

This dissertation is funded by the European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 813278 (A-WEAR: A network for dynamic WEarable Applications with pRivacy constraints, `https://www.a-wear.eu/`).

Exploiting Wireless Communications for Localization: Beyond Fingerprinting.
© by Tomáš Bravenec 2023. This work is licensed under CC BY 4.0.

Contents

1 Introduction

This chapter introduces the background information and motivations for this dissertation and the A-WEAR *project. It also contains the definition of objectives and research questions, as well as the outline of the thesis.*

1.1 Motivation

This dissertation is one of the 15 doctoral theses within the four years long H2020 Marie Skłodowska-Curie *Innovative Training Network* (ITN)/*European Joint Doctorate* (EJD) called *A network for dynamic WEarable Applications with pRivacy constraints* (A-WEAR). The project connects five universities: *Tampere University* (TAU), Finland; *Brno University of Technology* (BUT), Czech Republic; *Universita Mediterranea di Reggio Calabria* (URC), Italy; *University "Politehnica" of Bucharest* (UPB), Romania; and *Universitat Jaume I de Castellon* (UJI), Spain, to provide post-graduate education, supervision, and training for the 15 early stage researchers. All topics of the project are closely related to wearable applications, be it regarding privacy, security, localization, communication, or health applications.

The central question of this thesis is privacy in indoor positioning and localization. The problem of indoor positioning has grown rapidly over the last couple of years. The radio signal characteristics and the ways radio signal propagates throughout the environment is one of the main reasons wireless networks are popular in the indoor positioning and indoor navigation community [1].

Because of the nature of wireless communications, the data transfers can be exploited by a third party. That makes it easy for adversaries to capture the packets. As such this thesis explores the possibilities of passive capture of management frames of the Wi-Fi protocol to find possibilities for user tracking without the knowledge of the users.

Furthermore, this thesis dives into the very important field of algorithm optimizations. Since memory is limited in mobile and *Internet of Things* (IoT) devices, the *Machine Learning* (ML) optimizations in this thesis focus on memory. Also, specific *Radio Map* (RM) improvements through interpolations are explored.

1.1.1 Wearable Technologies

Wearable technology has become increasingly popular over the last several years, and the market for wearable technology is growing rapidly. As the name suggests, wearable devices can be worn by humans. There are three main categories of how close the wearables are to the human body [2]: *near-body* (for example smartphones), *on-body* (smart watches, smart rings, fitness trackers, earbuds, etc.), and *in-body* (implants).

Apart from the classification based on placement, wearable functions can be split into several sections: *health monitoring* (measuring heart rate, electrocardiograms, blood pressure, oxygen saturation levels, etc.), *fitness tracking* (counting steps, traveled distance and in case of outdoor activities location information or burned calories), and *lifestyle tracking* (sleep, stress, water, and calories income).

The data captured by all of the sensors and the transfer between devices are protected by encryption. However, radio signals are still a place of weakness. This is mainly due to the employment of wireless technologies, as these can be used for breaching the location privacy of users [3].

1.1.2 Technologies for Indoor Positioning Systems

There are many wireless technologies with the possibility to be employed in *Indoor Positioning System* (IPS) [4]. Two of the more common technologies, mainly due to their widespread adoption in smartphones, are Bluetooth [5] and Wi-Fi [6]. Less common are technologies like *Ultra-Wideband* (UWB) [7], which due to its higher cost [8], is yet to gain widespread adoption. Other technologies used in indoor positioning are ZigBee [9], visible light [10], millimetre wave radar [11] and others using computer vision [12] or dead reckoning [13]. Computer vision and dead reckoning are two technologies capable of working without infrastructure changes, as neither requires any beacons. However, both rely on the environment itself. Computer vision is used to analyze the environment using the images captured by the phone camera [14] and dead reckoning [15] using incremental location estimation from a known *Reference Point* (RP). Each technology possesses its own advantages and disadvantages [16].

1.1.3 Privacy in Indoor Positioning Systems

The ubiquity of wireless interfaces in most user devices, like Wi-Fi and Bluetooth, raises concerns about potential privacy breaches through these common networks. The issue is particularly pronounced with Wi-Fi management frames, which lack encryption. These concerns include:

- Non-anonymized *Media Access Control* (MAC) addresses in probe request frames can serve as unique identifiers for tracking devices.
- The *Preferred Network List* (PNL) can be used for fingerprinting and potentially revealing user locations.
- Some probe request fields may directly identify the device owner, such as the device name in the *Wi-Fi Protected Setup* (WPS) field.

This privacy issue is just the tip of the iceberg, as user presence detection via Wi-Fi management frames is relatively straightforward. However, achieving precise user localization requires the collaboration of at least three *Access Points* (APs), a more complex endeavor compared to *Global Navigation Satellite Systems* (GNSS) and IPS, which necessitate tailored solutions for diverse indoor layouts and environments.

1.2 Objectives

In this section, based on the found research gaps found the Research Objectives (ROx) are defined:

- **RO1. Analyze the possibilities of passive presence detection and positioning in indoor environments.**
  The Wi-Fi communication protocol is the most widespread *Wireless Local Area Network* (WLAN) technology. People use it pretty much daily to connect to the internet at home, at work, or in public spaces, in part to save mobile data usage in the package provided by the cellular network carrier. Another factor is the use of Wi-Fi by our smartphones, not only for communication but also for coarse localization. This combined with the management frames opens the door to a possible breach of user privacy. The questions then are: *Do our devices leak data and if so, how? Are there ways adversaries could exploit this leak to track us without our knowledge?*
- **RO2. Balance the achieved accuracy and necessary compute requirements of IPS and ML algorithms.**
  Even though the computational capabilities of *User Equipments* (UEs) grows every year, the complexity of algorithms increases too. However, it should not need to do the same. This would result in a reduction in the processing time. By optimizing the algorithms we can reduce the processing time and subsequently have more time available either for other tasks or for saving energy by entering a low-power mode during idle. This is especially a good thing in embedded and wearable applications, in which battery life is an important factor. The question is: *What are the ways to preserve the accuracy of machine learning algorithms, while reducing the hardware requirements?*

2 Reproducible Research

This chapter contains the created software used in several publications as well as the description of the collected datasets.

2.1 ESP32 Probe Request Sniffer

In several of the publications used in this thesis, the packet sniffer based on an ESP32 *Microcontroller Unit* (MCU) is used for dataset collection. To simplify, the ESP32 firmware captures Wi-Fi management frames. Following the capture, using a filter it selects only probe requests and saves them in a standardized binary packet capture file compatible with several network traffic applications.

Probe requests do not contain a field with transmission time, and because of that, the ESP32 first connects to a Wi-Fi network to download current time from the *Network Time Protocol* (NTP) servers. After synchronizing the internal clock with the outside world, the ESP32 initializes the connection to the SD card and mounts the file system.

While the sniffing task runs, all received packets are checked for their type. If the packet is a probe request, the packet is saved in a file on the SD card. All other received packets are discarded.

2.2 Datasets

Using the previously mentioned ESP32 MCUs a couple of datasets of probe requests were collected in 2 different environments. In the university office the collection was done twice, during one week in 2021 and another during one month in 2023. Third dataset was collected in different environment, during an international conference IPIN. These datasets, due to the nature of probe requests, had to be anonymized, which was done by application of the SHA512 algorithm on all fields containing data about the user. Some of these fields were the *Service Set Identifier* (SSID) from the preferred network list, WPS fields that can contain the device model, name, and many others. The output of the SHA512 is 64 bytes long, however, only small sections of the hashing output were used. There are 2 reasons for that:

- MAC address field of the transmitter, has a fixed length of 6 bytes, to preserve compatibility with network analysis tools, the MAC address length was preserved,
- reducing the memory requirements, in case every anonymized field had a length of 64 bytes, the analysis of longer time intervals would be more intensive on the memory requirements of the system.

The MAC address requires special treatment during the anonymization process. To keep the possibilities for analysis intact, the first 3 bytes of the original MAC address were left unchanged because those are the most important for analysis.

2.2.1 Probe Request Dataset - IPIN

The probe request dataset collected at the 11th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2021) contains 390810 captured probe requests. The monitoring of the Wi-Fi started 38 minutes before the conference program began with a tutorial session, on Monday 29th November, 2021 at 08:22. The final probe request was collected on Thursday 2nd December, 2021 at 13:02, just a moment after the closing ceremony concluded. It was not possible to keep the sniffer working past the official end of the conference due to the preparation of the conference space for the following event.

The captured probe requests, in some cases, contained the user information. Some of the user information was real MAC addresses of their UE, SSIDs from the PNL the UE transmits in search of APs it connected in the past. In some cases, the UE transmitted even the name of the device and the user, which happened in the transmission of probe request with WPS field. After the capture ended, the anonymity of the data was ensured by anonymizing all fields of the probe request that can contain user-related information with the SHA512 algorithm.

The in-person conference event took place during the COVID-19 pandemic and was quite isolated. In the conference space, there was minimal presence of people not participating in the event. The only people in the proximity of the sniffer were the attendees of the conference, conference organizers, and hotel staff.

2.2.2 Probe Request Dataset - UJI 2021

This dataset was collected during one week starting on Thursday 9th December, 2021 and ending on Wednesday 15th December, 2021 in the GEOTEC office at UJI, Spain. This dataset was collected in a similar way to the probe request dataset captured at the IPIN 2021 conference using the ESP32-based probe request sniffer described in Section 2.1. The dataset contains in total 340360 probe requests.

Just like the probe request dataset collected at IPIN 2021, this dataset contained user information and it stays to say that the analysis was only approached from the implementation of Wi-Fi protocol, specifically to explore potential privacy issues in current implementations. The dataset was then anonymized in the same way as the dataset from IPIN 2021 by using SHA512 hashing algorithm over sensitive fields. Even though the originally collected data contains both real and randomized MAC addresses, it is not possible to match MAC addresses to specific individuals because the analysis was done over an anonymized version of the dataset, and additionally, data regarding the actual presence in the office or the building were not collected.

2.2.3 Probe Request Dataset - UJI Probes 2023

Same as the previous dataset UJI 2021, this dataset was also collected in the GEOTEC office at UJI, Spain. The probe requests were collected during the month of March, to capture apart from regular work weeks also the local holiday Magdalena 2023, during which the university was mostly closed. Magdalena 2023 started on Saturday 11th March, 2023 and ended a week later on Sunday 19th March, 2023.

During the entire collection time, probe requests were being sent at all times, be it during the day, or night, workday or weekends. The explanation for the probe requests captured at night can be all-in-one computers using Wi-Fi instead of a wired connection, phones, or IoT devices used for experiments in the office etc. The second noticeable thing is a peak in the transmitted probe requests happening every day around 05:00. This is due to the scheduled reboot of a Wi-Fi access point present in the office and devices searching for a network to connect to after being disconnected from it. Another noticeable thing is a short time period at night of the Sunday 26th March, 2023 with no captured probe requests. That is due to the switch to the Summer Time when the time changed from 02:00 to 03:00.

The dataset offers various potential use cases, including Wi-Fi signal stability evaluation by tracking *Received Signal Strength Indicator* (RSSI) values over time, presence detection and room occupancy estimation based on network traffic patterns and RSSI information, and the exploration of user privacy issues due to anonymized but potentially identifying data. Additionally, the dataset allows for the study of randomized MAC address recurrences and vulnerabilities in the probe request mechanism. However, capturing ground truth room occupancy in the GEOTEC office was challenging, so occupancy is categorized into levels.

2.2.4 RM Interpolation Dataset

This dataset was created for the work focused on balancing the accuracy, compute requirements, and time required for the data collection of RMs for fingerprinting approaches to indoor positioning. The dataset represents the radio environment map created in the office at UJI, Spain.

For the collection of this dataset, 5 ESP32 MCUs with sniffer firmware described above were used, and placed around the office. The first 4 were placed in each corner and the last 1 in the center of the room at approximately equal distance to the sniffers placed in the corners of the office. RSSI by nature fluctuates, and using more sniffers helps reduce the influence of the fluctuations in RSSI by having more samples.

3 Exploiting Wi-Fi Management Frames for Presence Detection

This chapter focuses on the detection of human presence in using passive monitoring of nearby Wi-Fi network traffic.

In Section 3.2,presence detection in a university office over one week is explored.

Section 3.3 is a case study conducted at international conference IPIN 2021. And it focuses on the occupancy analysis of the conference space.

The main focus of Section 3.4 is room occupancy from Wi-Fi management frames collected by the ESP32 MCU.

3.1 Motivation

Indoor positioning can also be just detecting a presence of people in the *Area of Interest* (AoI) or in the proximity of a *Point of Interest* (PoI). There are many reasons why presence detection is useful. Be it for power-saving purposes (smart lighting, ventilation, heating, or air conditioning control), safety (knowing someone is in the building during emergency situations), advertising, and other purposes.

Since most of the approaches require a new infrastructure, in the following sections, the focus is on presence detection employing Wi-Fi technology and passively sniffing management packets from the radio environment. Since Wi-Fi infrastructure is usually already in place, it is beneficial to use it to reduce the cost of presence detection systems.

3.2 Temporal Pattern Analysis Aided Tracking

Our devices are communicating with the surrounding world using standardized protocols. For instance, a device in a IEEE 802.11 network is uniquely identified by the MAC address, which is used in all the messages involving the device. The device probe request is a type of wireless frame used to gather information about Wi-Fi access points in the proximity of a device. These probe requests can be a major weak point of a Wi-Fi protocol since they allow for non-cooperative user tracking if the device does not use enough privacy measures such as MAC address randomization.

Tracking using Wi-Fi protocols can vary as they can be used to determine the past whereabouts of users, current presence, or both. The past locations of devices can be determined if the devices are transmitting the PNL (list of the networks the device was connected to in the past), which can be matched to the location using access point databases [17]. The current presence tracking can be done using a fingerprinting approach.

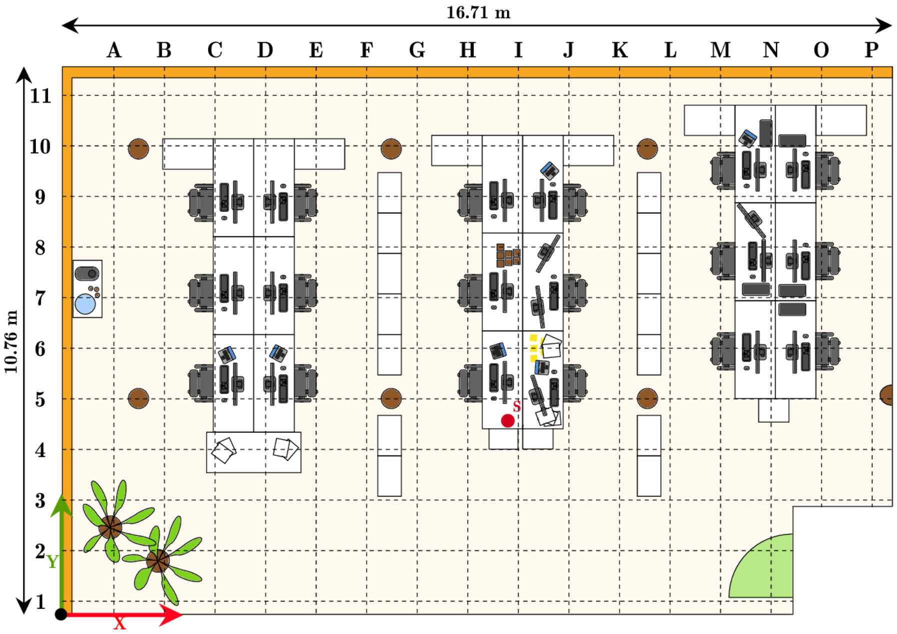


Fig. 3.1: Floor plan and location of sniffer in the office space of GEOTEC department at UJI, Spain.

3.2.1 Analysis

The used dataset was described in Section 2.2.2. The collection of probe requests was done at the GEOTEC office for 6 days of December 2021. The office is in the corner of the 5th floor and during peak times is occupied by about 15 researchers. The office space is visualized in Fig. 3.1. During that time, the sniffer collected 340 360 probe requests.

In the past, the tracking of mobile devices using only probe requests was not very difficult as there were several factors that made the identification of a single device fairly straightforward. These include non-randomized MAC addresses, consecutive Sequence Numbers, common time differences between 2 probe requests, or] Information Element.

MAC addresses: Even though MAC addresses cannot be used effectively to locate most modern devices, they can still be used to identify a device during a single scan.

Sequence Numbers: The incremental nature of sequence numbers allow for another opportunity to easily identify packets coming from a single device.

Information Elements: There can be various data, starting with supported transfer speeds, and information about the vendor of the wireless chip inside of the device, and even a device name. All of this can be used for fingerprinting.

Preferred Network Lists: Knowing all probe requests coming from a single device for compilation of Preferred Network List. By using sets with each SSID represented only once, it is possible to use set similarity: $p = \frac{\mathrm{set}(A)\,\textbf{and}\,\mathrm{set}(B)}{\mathrm{set}(A)\,\textbf{or}\,\mathrm{set}(B)}$ to match same devices together despite using different MAC address.

Device Identification: During the identification process there is a check if the MAC addresses of 2 separate instances are the same. If the MAC addresses are randomized

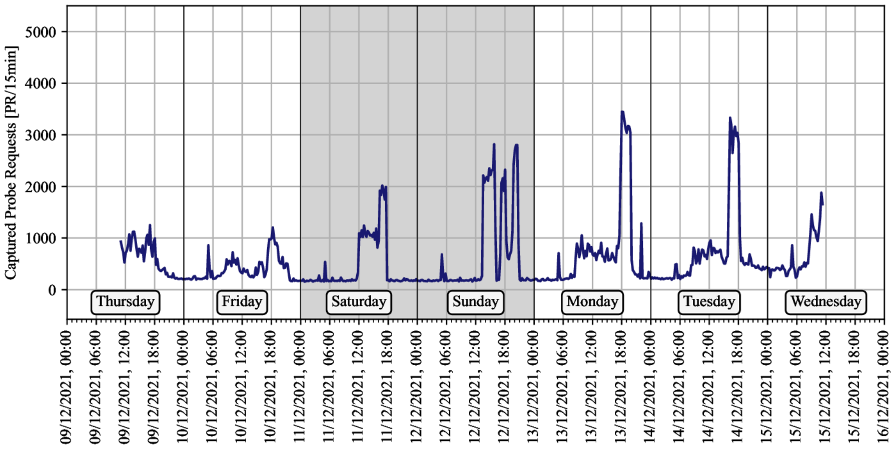


Fig. 3.2: Density of captured Probe Requests over time in the office space of the GEOTEC department (amount of probe requests grouped in 15-minute clusters).

or different from each other, the presence of the WPS field is checked. In case of its inclusion, the *Universally Unique IDentifier-Enrollee* (UUID-E) field can be used in place of MAC address. In case the WPS field is not included and MAC addresses are not matching, similarity using the information elements and PNLs is used.

Temporal Pattern Analysis: One of the more difficult parameters to mask for a single device sending multiple probe requests is the time difference between 2 Wi-Fi scans. This can be achieved by considering scan instance appearances of one device and clustering them together based on time. Then, the overlay similarity between clusters was compared.

3.2.2 Results

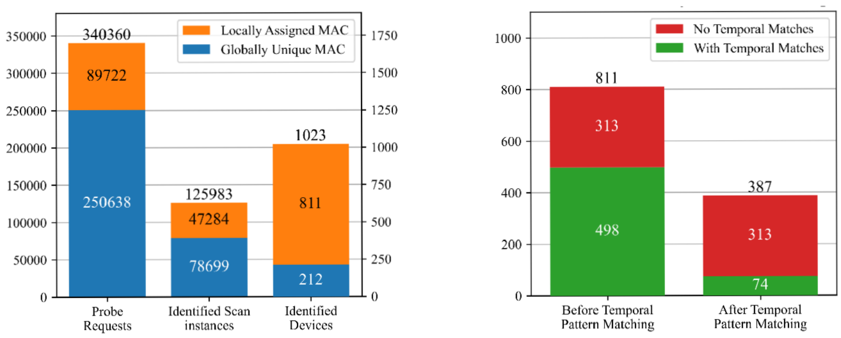
Throughout the week, the sniffer captured 340 360 probe requests. The distribution of probe requests, is shown in Fig. 3.2. From the distribution is clearly visible that some devices in the office are running without interruptions.

From the 340 360 probe requests collected at the office, identified in total 125 983 scan instances. As a follow up 1023 devices were identified, as is represented in Fig. 3.3a.

For devices that do not randomize their MAC addresses, the tracking is very effective. The reason is that the unique identifier is the MAC address, which never changed. As presented in Fig. 3.3a, 212 devices did not use MAC randomization, and the example of presence in time for 8 such devices is shown in Fig. 3.4a.

The identification of devices randomizing MAC addresses is more complicated. The results can be seen on 8 devices using randomized MAC address in Fig. 3.4b. Even with the more complicated identification, the analysis of user presence is still possible.

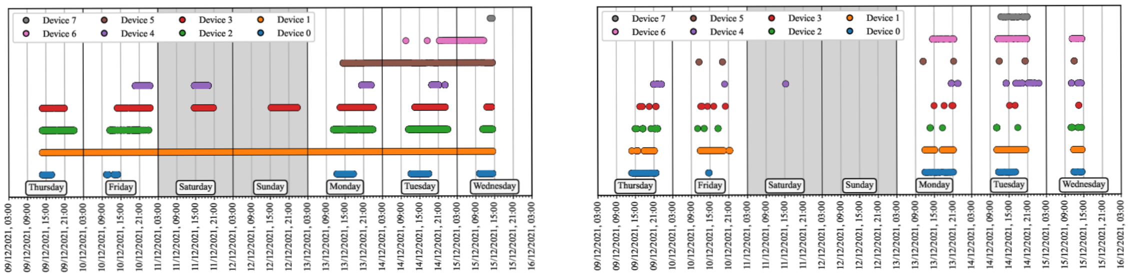
The instance matching is not 100 % accurate and, it can misidentify a single device as several devices. The number of devices with locally assigned MAC address before and after temporal pattern matching can be seen in Fig. 3.3b. The probe request transmission patterns were quite closely matching each other, as can be seen in Fig. 3.5.



(a) Randomized MAC addresses in probe requests, identified scan instances and devices.

(b) Identified devices with locally assigned MAC address before and after temporal pattern matching.

Fig. 3.3: Dataset information and device identification in the office space of the GEOTEC department.



(a) Occurrence of devices identified by the usage of globally unique MAC address.

(b) Occurrence of devices identified despite the use of MAC randomization.

Fig. 3.4: Occurences of devices in the GEOTEC department.

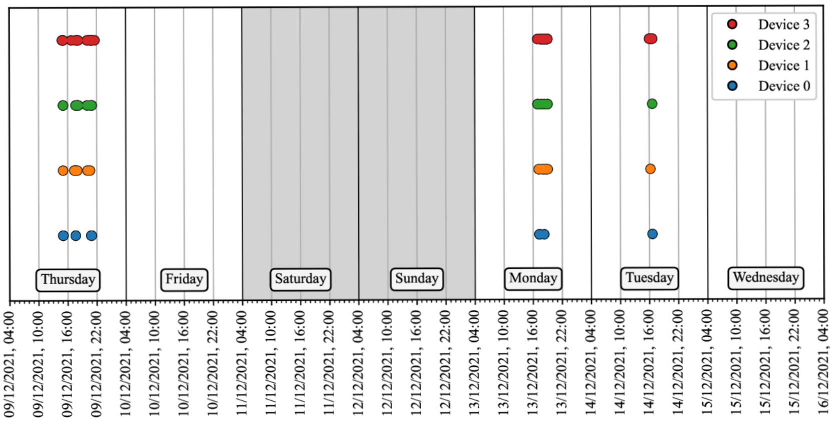


Fig. 3.5: Occurrence of single device misidentified as multiple devices, later identified as a single device through the similarity in temporal patterns in the office space of the GEOTEC department.

3.3 Presence Analysis with Wi-Fi Probe Requests

The attendance at the international conference IPIN 2021 provided an idea for a case study. As the conference focuses on indoor positioning and indoor navigation, the case study focused on presence detection during the conference.

3.3.1 Analysis

The conference took place in Lloret de Mar, Spain in Evenia Olympic Congress Centre from 29 November to 2 December 2021. The only people present around the hotel lobby and near the session rooms from the beginning to the end of the conference were attendants of the conference, conference organizers, hotel employees, and cleaning staff.

The entire conference space was around the lobby, with hotel rooms and hotel restaurants being far enough to not pose interference and capture probe requests from unwanted sources

3.3.2 Results

The results are split into several categories. At first, the presence of users in the proximity of the sniffer is evaluated. This follows by tracking the presence of individual users based on the probe request frames. For this, the fingerprinting of information elements introduced in Section 3.2 is used.

Presence Detection

From Fig. 3.6, it is visible that during every session, the presence of users was increased. Quite a lot of people also left the range of the sniffer to go into the hotel restaurants for lunch. From the figure, it is also visible which keynote or session group (IPIN 2021 had 4 parallel session tracks) was more interesting to the participants of the conference.

The Tuesday social event (Networking in the Kitchens) took place mostly out of the range of the probe sniffer in one of the hotel's restaurants. After the event, some of the participants stayed for further socializing, which can be seen on the small local peak right after the event ended.

One of the noticeable trends is also the drop in the amount of captured probe requests during coffee breaks. This indicates people leaving the area either to get some fresh air outside of the hotel lobby, use the restroom, or go to their hotel rooms.

Analysis of User Presence with Global MAC Address

Since it is possible to identify probe requests using their globally unique identifier, their identification is very simple. At the IPIN 2021 conference, 28.62 % of identified scan instances used their globally unique MAC address. There were identified 229 devices

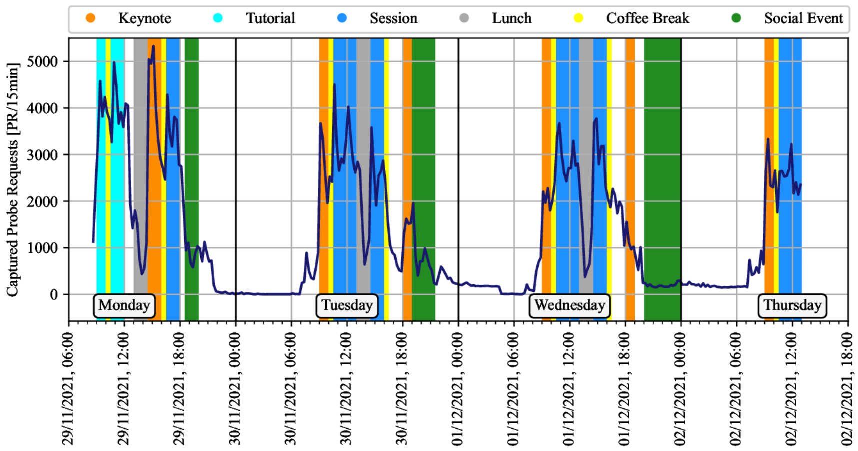
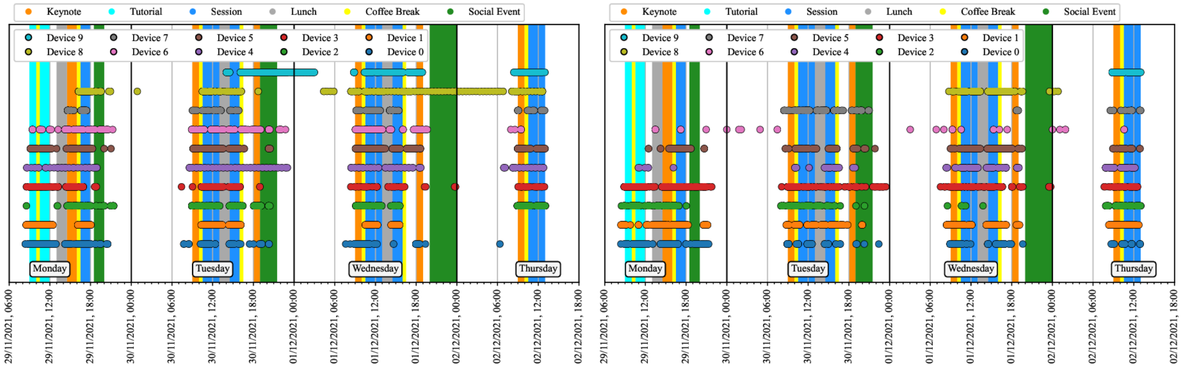


Fig. 3.6: Density of captured Probe Requests correlated with the program of the IPIN 2021 conference (amount of probe requests grouped in 15-minute clusters).



(a) Repeated occurrences of devices identified by the usage of globally unique MAC address

(b) Recurrent identification of the same devices despite using locally assigned MAC address.

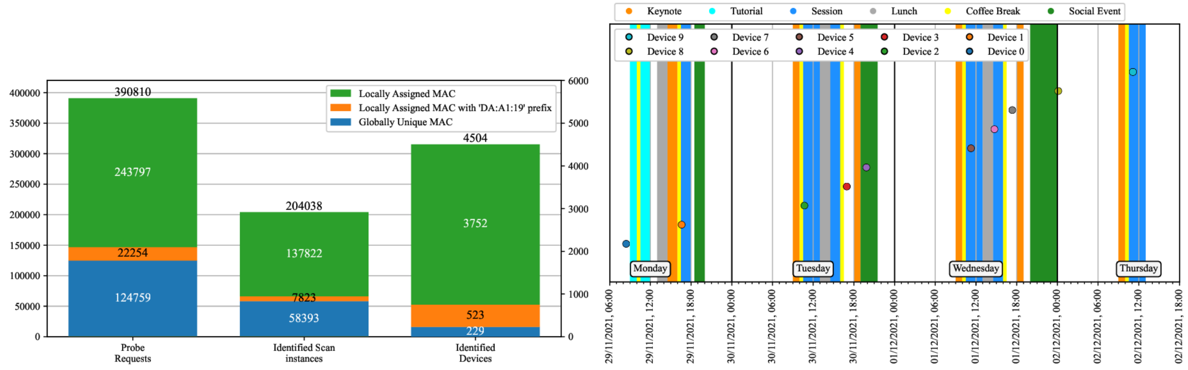
Fig. 3.7: Recurrent identification of same devices at the IPIN 2021 conference.

without MAC address randomization. This data can be seen in Fig. 3.8a. The temporal presence of 10 devices using their real MAC address in the conference space is in Fig. 3.7a.

Analysis of User Presence with Local MAC Address

Out of the captured probe requests, 68.08 % (266 051) were using locally assigned MAC addresses. There were 7823 individual scan instances using *DA:1A:19* prefix. After matching these instances together there were identified 523 devices using the *DA:1A:19* MAC address prefix. These data are presented in Fig. 3.8a with the comparison to the number of devices with a fully randomized MAC address and with a globally unique one.

After identifying individual scan instances, 3544 MAC addresses appeared less than 10 times. On the other hand, 296 devices with fully randomized MAC addresses showed up more than 10 × , which made them easily identifiable despite, as can be seen from 10 example devices in Fig. 3.7b.



(a) Randomized MAC addresses in probe requests, identified scan instances, and distinguished devices at the IPIN 2021 conference.

(b) Detected devices without identified recurrences in time at the IPIN 2021 conference.

Fig. 3.8: Randomized MAC addresses, recognized scan instances, distinguished devices and devices without recurrences at the IPIN 2021 conference.

Single Occurrence of Devices in Time Domain

From Fig. 3.8a, it is visible that the number of identified devices is still really high for just 3 full days in a conference space. The reason for this can be a good implementation of MAC address randomization, the transmission of reduced information elements in the probe requests, and omitting the transfer of SSIDs from the saved PNL. Representation of unmatched devices is shown in Fig. 3.8b with 10 examples.

3.4 Room Occupancy Detection Using Wi-Fi Probes

The work exploring room occupancy detection was presented [18] at the international conference MAREW 2023.

3.4.1 Analysis

The data were collected during five working days in three defined-sized rooms and scenarios: office, laboratory, and meeting room. At first, the signal strength distribution in all three locations was recorded to get the RMs throughout the rooms. Therefore, it was possible to accurately remove non-interesting devices from the analyzed data.

Room occupancy is analyzed within a given time interval, the length of which can be set according to the intended application. Each probe request is tested against its RSSI value and the device's MAC address is obtained from the probe request. The occurrence of each unique MAC address is accumulated within the time interval. After the selected interval, the frequency of MAC addresses is assessed. The targeted estimation of room occupancy is equal to the number of MAC addresses more frequent than the threshold.

Tab. 3.1: Experimental results of occurrence estimation.

Room	Captured [-]	RSSI threshold [dBm]	Considered [-]	MACs [-]	RMSE [-]
Office	143 871	−56	4761	149	0.208
Lab	265 797	−63	29 827	4126	0.797
Meeting	281 661	−66	12 262	3765	3.389

3.4.2 Results

To support the algorithm's functionality for estimating room occupancy, three datasets of probe requests were created, collected during five days between October 3rd and 7th, 2022 at the Department of Radio Electronics at the Brno University of Technology, Czechia.

Within the entire measurement period, a total of 143 871, 265 797, and 281 661 probe requests were recorded in the office, laboratory, and lecture room, respectively.

The testing was carried out during normal university operations, such as laboratory exercises, lectures, and meetings, and was attended by more than 100 people. Due to the randomization of MAC addresses a total of 8040 unique addresses were captured during the recording of probe requests.

To assess the error rate of the proposed algorithm, the actual occupancy of the rooms was recorded. The differences between the results of the proposed detection algorithm were compared to the actual occupancy using *Root Mean Squared Error* (RMSE). The summarized data, the selected parameters, and the resulting estimation error are shown in Table 3.1.

The results in the minimal, small, and medium scenarios, obtained by the proposed method (Estimated) and by real observations (Measured), are shown in Fig. 3.9. It can be seen that the number of mobile devices or persons in the office was most often equal to one. The resulting low RMSE error means a high degree of agreement between the proposed algorithm and reality. For small- and medium-scale scenarios, where larger rooms and numbers of people are considered, RMSE is 0.797 for the laboratory and 3.389 for the lecture & meeting room. However, the difference is not large and confirms the suitability of the used method.

3.5 Summary

The presence detection and room occupancy analysis are summarized in the following paragraphs:

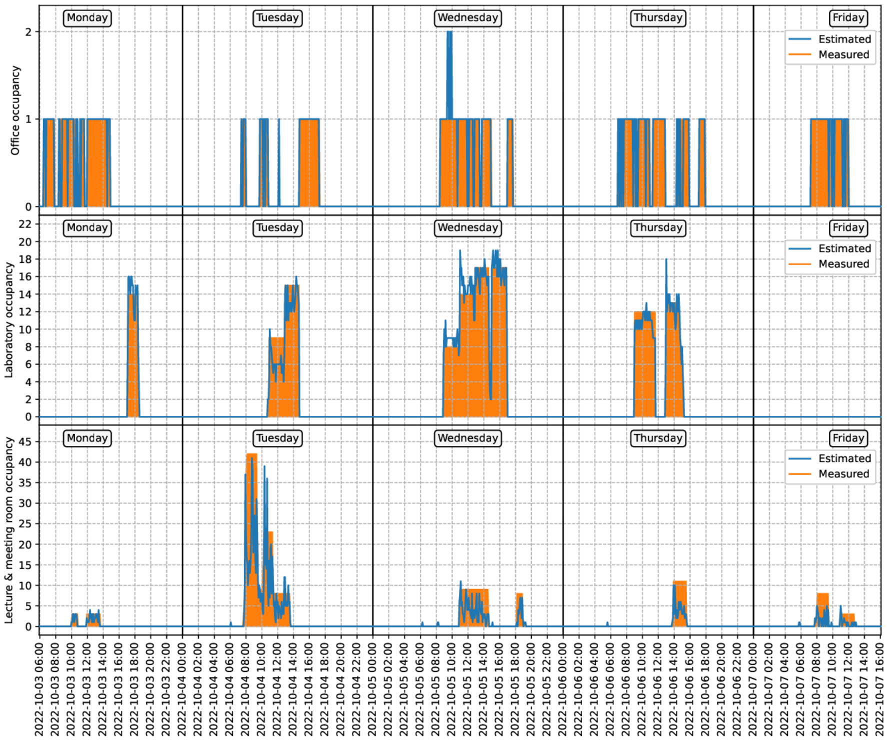


Fig. 3.9: Estimated and measured occupancy in minimal-scale office, small-scale laboratory, and medium-scale lecture & meeting room scenarios.

- At first, the presence detection and tracking of humans in the office was presented. This was achieved by sniffing Wi-Fi management packets from the radio environment, which was done in a completely passive way, that prevented any nearby device from detecting the sniffer. The collected frames were then used for temporal analysis and tracking by employing fingerprinting of data available in the unencrypted packets.
- Similar to the presence detection in the office, the case study evaluating the presence of conference participants near the session rooms was conducted. The conference IPIN 2021 was chosen as the place to conduct this study. From the gathered data, it was possible to again prove that non-cooperative tracking using Wi-Fi is possible and improvement of privacy-related measures in the near future is necessary.
- Finally, the room occupancy estimation was done using the same management frames of Wi-Fi. The ability to estimate the occupancy of rooms is going to be an important function of smart buildings, to reduce the energy requirements for air conditioning, lighting, and air circulation. The use of existing infrastructure is also a good step in the integration into smart building systems.

4 Balancing Accuracy and Complexity

This chapter focuses on the optimizations in machine learning approaches to reduce computational complexity while preserving or increasing the accuracy of machine learning computing at the edge and in positioning algorithms.

Section 4.2 describes a possibility to reduce the memory requirements of Convolutional Neural Networks.

In Section 4.3, the possibilities of using interpolation techniques to find a balance between the size of Radio Map, the time required to create Radio Maps, and the computational complexity of the final Indoor Positioning System are presented.

4.1 Motivation

At first, the look into the reduction of memory requirements is explored. For this, neural networks are taken into account, as models can have hundreds to thousands of MB.

Secondly, the balance between accuracy and processing performance, while enhancing the original data is explored. The data pre-processing can improve results in many ways.

4.2 Reducing Memory Requirements by Lowering Data Precision

The high computing requirements that come with some machine learning algorithms, as well as the need for large amounts of *Random Access Memory* (RAM) had been the main reason to explore the ways to balance the requirements and accuracy.

4.2.1 Analysis

A couple of popular *Convolutional Neural Networks* (CNNs) for image classification with very different architectures were selected for testing the influence of Half-Precision weights on their accuracy: AlexNet [19], GoogLeNet [20], Inception V3 [21], ShuffleNet V2 [22], and MobileNet V2 [23].

The tested networks were modified to use a quantized 8-bit integer. The networks have not been retrained as only post-training quantization was done, which compared to fully quantization-aware training might produce worse results. Post-training quantization was used because, unlike with the use of quantization-aware training, the use of post-training quantization, as the name suggests, is applied to the already trained network.

Testing itself was done in two parts. First of all, the memory footprint of the neural network was compared before and after. And second, the accuracy was tested on the classification of 1000 classes present in the ImageNet dataset [24].

Tab. 4.1: Comparison of Single-Precision, Half-Precision, and Quantized Integer Influence on the Size of Networks Weights.

	Single-Precision (32-bit) Size [MB]	Half-Precision (16-bit) Size [MB]	Quantized Integer (8-bit) Size [MB]
AlexNet	244.4	122.2	68.5
GoogLeNet	52.2	26.2	13.1
Inception V3	109.0	54.6	24.0
ShuffleNet V2	9.3	4.7	2.4
MobileNet V2	14.3	7.2	3.6

Tab. 4.2: Comparison of Single-Precision, Half-Precision, and Quantized Integer Data Type Influence on Top-1 and Top-5 Error.

	Single-Precision (32-bit)		Half-Precision (16-bit)		Quantized Integer (8-bit)	
	Top-1 Error [%]	Top-5 Error [%]	Top-1 Error [%]	Top-5 Error [%]	Top-1 Error [%]	Top-5 Error [%]
AlexNet	43.963	21.008	43.967	21.019	43.965	21.004
GoogLeNet	30.159	10.405	30.184	10.392	30.171	10.444
Inception V3	22.439	6.312	22.418	6.300	30.551	11.456
ShuffleNet V2	30.721	11.696	30.754	11.698	31.917	12.741
MobileNet V2	28.351	9.644	28.364	9.642	68.890	46.584

4.2.2 Results

From the results of data type conversion in Table 4.1, it can be seen that reducing the precision of the network's weights had reduced the size to half and quarter, by using half precision and quantized 8-bit integer respectively.

The accuracy did not suffer using Half-Precision at all. The difference in the accuracy of the networks using Half-Precision weights did not get over 0.04 % (20 images). A little worse results were achieved while using a Quantized Integer. Some networks were not affected, while some suffered a big loss in accuracy. The results are in Table 4.2.

The performance is not directly comparable due to the support of different data type operations in *Central Processing units* (CPUs) and *Graphical Processing Units* (GPUs). The average frame rate achievable on the NVIDIA Jetson Nano using PyTorch for each network on both the CPU and GPU is presented in Table 4.3.

AlexNet: AlexNet is an old architecture with the largest weight file and lowest accuracy of the tested networks. While using Half-Precision weights, the accuracy dropped, but not very much, only by 0.004 % to 0.011 %. Even with weights in quantized 8-bit integer data type, the network performed pretty much the same.

The network managed to get great real-time performance with almost 190 processed frames per second using weights in Single-Precision format. While using Half-Precision, the memory bandwidth was halved and the performance increased to 221 FPS.

GoogLeNet: Just like AlexNet, the difference was minimal with just 12 more images classified incorrectly for the Top-1 error and only 6 more in the top 5. The same goes

Tab. 4.3: Comparison of Single-Precision, Half-Precision, and Quantized Integer Influence on the Average Frames per Second on NVIDIA Jetson Nano.

	GPU		CPU	
	Single Precision (32-bit) [images/sec]	Half Precision (16-bit) [images/sec]	Single Precision (32-bit) [images/sec]	Quantized Integer (8-bit) [images/sec]
AlexNet	189.7	221.5	0.9	1.7
GoogLeNet	24.1	26.4	0.5	2.2
Inception V3	14.9	16.7	0.3	1.4
ShuffleNet V2	26.4	26.7	2.0	6.5
MobileNet V2	33.2	34.1	2.6	9.5

for weights in quantized 8-bit integer, the accuracy of GoogLeNet did not suffer and the difference was lower than 20 incorrect classifications.

GoogLeNet gained on average about 2.3 frames per second while using weights and input data in 16-bit floating point format. That is an increase in performance of 10 % as seen with AlexNet.

Inception V3: The most interesting result was delivered by the network Inception V3, which provided better results in Top-1 error by 0.021 % and in Top-5 error by 0.012 % while using Half-Precision weights. Quite surprisingly, due to the better accuracy while using Half-Precision weights, Inception V3 suffered in terms of accuracy when using fixed-point arithmetic. The difference being about 8 % (3913 images) in Top-1 and 5 % (2481 images) in Top-5 error.

Just like the networks AlexNet and GoogLeNet, the performance of inference on the GPU while using Half-Precision weights increased about 10 %, which makes a difference of 1.5 frames per second, which is achieved due to the decrease in memory bandwidth.

ShuffleNet V2: Keeping with the trend, using Half-Precision weights had minimal effect. The quantized 8-bit integer weights have a small impact on the accuracy of the ShuffleNet V2, as both the Top-1 and Top-5 error increased by 1.2 %.

Even though it is small network, the achieved frame rate on CPU with fixed point weights still can not compete with inference on the GPU, but achieved the best performance to accuracy ratio on the CPU with an average of 6.5 FPS.

MobileNet V2: is a very small and accurate network. The accuracy stays almost the same while using Half-Precision weights. However it is the most sensitive network to weights using 8-bit fixed point format. The accuracy suffered the most with correct classification of less than 35 % of all validation images.

The inference performance did not increase almost at all. That is due to the network being quite small and the memory bandwidth is still very small compared to bigger networks. However, it proved to be the most efficient for inference on the CPU. The conversion of weights into fixed point did not prove to provide good results in terms of accuracy. At almost 10 FPS, it is still the fastest result while not using GPU acceleration.

4.3 Interpolation of Radio Maps for Indoor Positioning

In the office, the ESP32 with Probe Request sender firmware emitted 50 probe requests at each RP, which were all collected by the 5 ESP32 MCU boards *S1-S5* used for data collection.

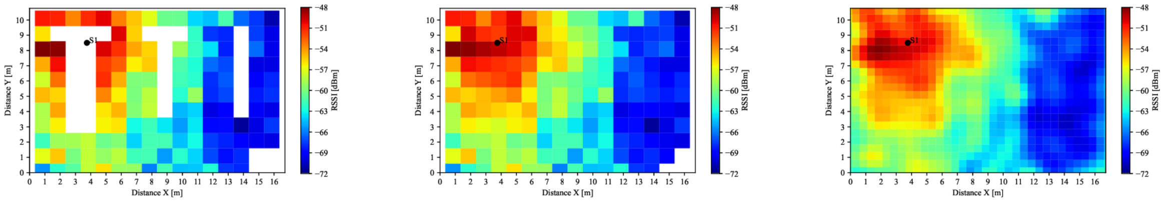
4.3.1 Analysis

In the RM processing, several algorithms were used to achieve an approximation of data samples at hard-to-reach RPs and to acquire a higher density of RPs with approximated data in these locations. For the sake of simplicity, the visualizations in the following sections are only for the mean of all values of ESP32 sniffer *S1*. The algorithms that were used for data processing were linear interpolation and *Gaussian Process Regression* (GPR).

Linear Interpolation: At first, all measurements in a 1 m grid were considered for an approximation of missing values, for which, linear interpolation was used. The RM with the *Measured Data* (MD) by the sniffer *S1* is in Fig. 4.1a while RM with approximated data using is in Fig. 4.1b. In Fig. 4.1a and 4.1b, the centers of the cells are aligned in 1 m grid, while in Fig. 4.1c, the grid is 0.5 m.

Gaussian Process Regression: Using Gaussian Process Regression [25], a model representing the radio space is created. To use GPR, the selection of covariance function is required. From Table 4.4, it is visible that the differences between positioning accuracy achievable by RMs with different covariance functions are negligible. The covariance function *Squared Exponential* (SE) with fixed length scale was chosen, as it provided the lowest 95$^{\text{th}}$ percentile in most variations of RM enhancements.

The usage of IPS has issues in situations employing incomplete data, with lower accuracy around locations with missing data points. Unlike using just linear interpolation, GPR approximates data with a machine learning model, which means that in short distances from the edge of the measured RM it is possible to extrapolate the data by passing the model coordinates that are outside of the room boundaries. The extrapolation can be seen using the top view in Fig. 4.1c.



(a) RM with measured only data.

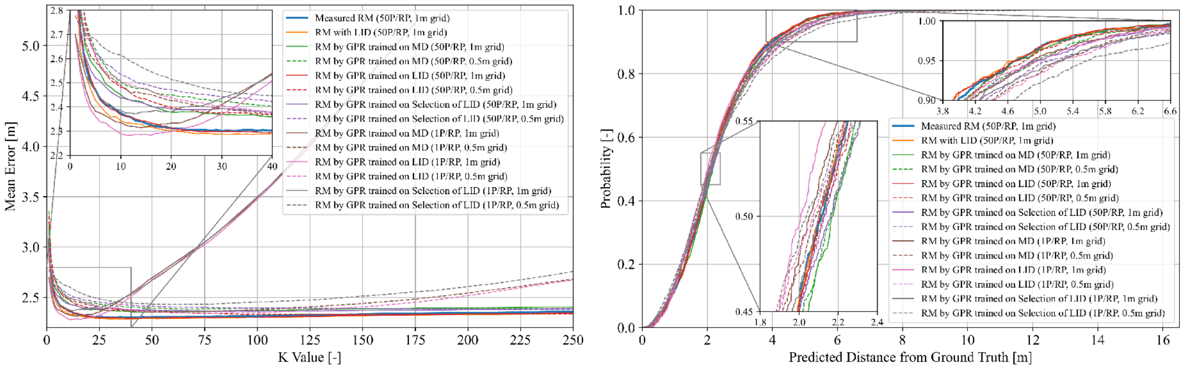
(b) RM with interpolated data.

(c) GPR approximation of RM.

Fig. 4.1: Visualization of RSSI RM captured by *S1* (ESP32 placement) compared with RM with interpolation of missing values in unreachable RPs.

Tab. 4.4: Comparison of difference in 95th percentile of positioning accuracy employing RMs created using different covariance functions. In bold is highlighted the best performing covariance function for each RM.

	$\frac{Samples}{RP}$	RM Grid	SE	SE Fixed	Matern	RQ	SE+Matern	SE+RQ	Matern+RQ
RM	[-]	[m]	[m]	[m]	[m]	[m]	[m]	[m]	[m]
Measured RM	50	1.0	**4.73**	**4.73**	**4.73**	**4.73**	**4.73**	**4.73**	**4.73**
RM with LID	50	1.0	**4.68**	**4.68**	**4.68**	**4.68**	**4.68**	**4.68**	**4.68**
RM by GPR trained	50	1.0	4.81	**4.72**	4.85	4.85	4.85	4.79	4.79
on MD	50	0.5	**4.86**	4.98	5.08	5.26	5.08	5.24	5.26
RM by GPR trained	50	1.0	4.73	4.71	**4.66**	4.70	**4.66**	4.72	4.70
on LID	50	0.5	4.89	**4.85**	5.05	5.10	5.05	5.11	5.12
RM by GPR trained	50	1.0	**4.91**	4.92	4.94	5.02	4.94	5.02	5.05
on Selection of LID	50	0.5	5.24	**5.20**	5.28	5.21	5.28	5.36	5.47
RM by GPR trained	1	1.0	5.01	**4.87**	5.13	5.05	4.99	4.99	4.99
on MD	1	0.5	5.34	**5.27**	5.33	5.38	5.62	5.33	5.62
RM by GPR trained	1	1.0	4.95	4.95	4.93	4.95	4.92	**4.91**	**4.91**
on LID	1	0.5	5.34	5.31	5.35	5.30	**5.30**	5.43	5.37
RM by GPR trained	1	1.0	5.23	5.23	5.36	5.18	6.05	5.18	**5.15**
on Selection of LID	1	0.5	5.69	5.69	5.54	5.52	8.49	5.51	**5.26**



(a) Comparison of RM processing on the mean positioning accuracy of IPS depending on k of kNN algorithm.

(b) Comparison of RM processing on the CDF of of positioning error, highlighting median and 95th percentile.

Fig. 4.2: Mean positioning accuracy and CDF for selected k.

4.3.2 Results

The testing of the influence of RM interpolation was done in 3 ways. From the perspective of accuracy, processing speed, balance between the time necessary for gathering data, and finally processing time and accuracy.

Selected values of k for each RM are in Table 4.5. The validity of the selection can be checked by looking at Fig. 4.2a. From Fig. 4.2a, it is visible that the values selected by the equation match with the k with which each RM achieved the best mean accuracy.

Tab. 4.5: Overview of accuracy and normalized performance achieved with final IPS based on *k*NN.

	$\frac{Samples}{\text{RP}}$	RM	Selected	Reference	MAE	Median	75th	95th	RMSE	Normalized
Final IPS Input RM		Grid	k	Samples		Error	percentile	percentile		Time
	[-]	[m]	[-]	[-]	[m]	[m]	[m]	[m]	[m]	[-]
Measured RM	50	1.0	90	6390	2.32	2.12	2.92	4.68	2.62	1.00
RM with LID	50	1.0	90	7940	2.30	2.10	2.93	4.68	2.60	1.25
RM by GPR trained	50	1.0	90	8800	2.37	2.17	3.07	4.74	2.68	1.38
on MD	50	0.5	250	40 800	2.39	2.16	3.12	4.94	2.73	6.52
RM by GPR trained	50	1.0	90	8800	2.30	2.11	2.95	4.72	2.61	1.39
on LID	50	0.5	250	40 800	2.34	2.13	3.00	4.82	2.66	6.39
RM by GPR trained	50	1.0	90	8800	2.37	2.16	3.02	4.95	2.72	1.39
on Selection of LID	50	0.5	250	40 800	2.38	2.13	3.08	5.11	2.78	6.35
RM by GPR trained	1	1.0	13	176	2.32	2.05	2.93	4.90	2.66	0.03
on MD	1	0.5	35	816	2.38	2.09	3.08	5.24	2.79	0.13
RM by GPR trained	1	1.0	13	176	2.29	2.01	2.89	5.01	2.65	0.03
on LID	1	0.5	35	816	2.39	2.07	3.10	5.26	2.81	0.13
RM by GPR trained	1	1.0	13	176	2.37	2.10	3.03	5.28	2.77	0.03
on Selection of LID	1	0.5	35	816	2.46	2.04	3.26	5.58	2.96	0.13

Influence of Processed RMs on Positioning Accuracy

The best accuracy was achieved by GPR generated RMs with 1 sample per RP, trained on MD and *Linearly Interpolated Data* (LID). However, the accuracy started dropping with the value of k being higher than 16. That is due to the nature of *k-Nearest Neighbors* (*k*NN) with low number of features. The most consistent results were gained by using LID. As expected, the lowest accuracy was achieved by using samples collected at every 2nd RP. On the other hand, the difference is only about 10 cm worse than using all of the MD. To visually show the distribution of the error, Fig. 4.2b presents the *Cumulative Distribution Function* (CDF) for each of the RMs.

Compute Requirements Based on RM Complexity

To evaluate the compute performance depending on the RM used for *k*NN based IPS, the evaluation run times were normalized to the baseline performance of RM created out of only MD. The normalized performance results are in Table 4.5.

The results, depend on the number of samples in a given RM. This means the RMs with RPs spread out in a 0.5 m grid, instead of 1 m grid, contains approximately 500 % more samples. The performance in these cases is much slower. Following the same pattern, the RMs with just 1 sample per RP achieves the best run time performance, thanks to the small size of such models. the time required for computation resulted in a fraction of the time required for any RM with 50 samples per RP.

Tab. 4.6: Approximation of time required for RM collection in the office acquired using cross-multiplication.

RM type	Grid [m]	RPs [-]	Approximated Time [hh:mm]
Accesible RP	2.0	41	00:38
	1.0	**142**	**02:13**
	0.5	497	07:45
All RP	2.0	47	00:44
	1.0	173	02:42
	0.5	656	10:14

Reductions in RM Collection Time

The baseline MD with 142 RP took 2 hours and 13 minutes to collect, For a relatively small space like the used office, this is a very time-consuming task. Using this collection time, the time required to collect data in a 0.5 m grid, or in 2 m grid can be estimated. The comparison of the time required to create baseline RM, and approximations through cross-multiplication is in Table 4.6.

4.4 Summary

The work done in the optimizations of machine learning algorithms and interpolations of RMs is summarized by the following paragraphs:

- The first look into the optimizations of ML was done by employing the reduction of data types used for the storage of neural network models. The used approach does not require retraining and by using a Half-Precision floating point data type the difference in accuracy is negligible. The use of fixed point data types proved to be more difficult, and for better results, it would need retraining to adapt weights with the data type constraints in mind.
- Second look into optimizations, was more specifically targeted at indoor positioning using fingerprinting of RSSI to create RMs. As a side effect using one approach to interpolation resulted in a negligible drop in accuracy, while the time required for predictions dropped to mere percents of the location predictions using the originally collected data. This approach drastically reduced both memory and computational requirements.

5 Conclusions

In this Chapter, the conclusions are drawn and the research questions asked in Chapter 1.

RQ1. ***Do our devices leak data and if so, how? Are there ways adversaries could exploit this leak to track us without our knowledge?***

During the first scenario based in the office, the probe requests of the Wi-Fi protocol were used to evaluate the privacy-related measures implemented in Wi-Fi. Temporal pattern matching was also introduced as a way to find devices hiding behind MAC address randomization by exploiting the appearances over time.

At a 2021 conference, probe requests were used to study non-cooperative tracking using Wi-Fi. Despite MAC address randomization, devices could still be tracked using probe information, revealing the inadequacy of randomization for privacy and the need for better measures.

By analyzing probe requests, experiments accurately estimated room occupancy, offering potential for energy-efficient smart buildings and increased safety in crises.

In summary Wi-Fi networks need more than just MAC address randomization. And even though MAC randomization helps, it is far from being enough and it can give users a false sense of security.

RQ2. ***What are the ways to preserve the accuracy of machine learning algorithms, while reducing the hardware requirements?***

The reduction of computational requirements by using lower precision data types for neural network weights was also explored. Shifting from 32-bit to 16-bit floating points doesn't need retraining and reduces storage size by half without affecting accuracy. Further reductions to 8-bit fixed point data types may vary in accuracy, requiring additional testing.

Apart from the memory requirements for ML algorithms, data augmentation specific to the indoor positioning field was also evaluated. Specifically the use of GPR for interpolation of RMs for fingerprinting approaches to indoor positioning. In this work, the most important results were achieved by creating a RM with a single sample per RP. In this case, the complexity of the dataset dropped drastically from 6390 samples to just 176. This means the single sample was created by a combination of the information from 50 samples belonging to the original RM. The computation speed dropped to just 3 % of the original RM, while the drop in accuracy was negligible in just a few cm. This provides a massive improvement in both memory requirements to store the RM, as well as in the speed of prediction of locations using this RM.

To summarize, 2 possible approaches to optimizations were introduced and proven to preserve the accuracy levels, while requiring much less storage space. The computational performance can also be increased, however, in some cases, it might need hardware with support for specific operations.

6 References

[1] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, “Kernel-based positioning in wireless local area networks”, *IEEE transactions on mobile computing*, vol. 6, no. 6, 2007.

[2] A. Ometov *et al.*, “A survey on wearable technology: History, state-of-the-art and current challenges”, *Computer Networks*, vol. 193, 2021.

[3] E. Fenske *et al.*, “Three Years Later: A Study of MAC Address Randomization in Mobile Devices and When it Succeeds”, *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, 2021.

[4] S. Subedi and J.-Y. Pyun, “A Survey of Smartphone-Based Indoor Positioning System Using RF-Based Wireless Technologies ”, *Sensors*, vol. 20, no. 24, 2020.

[5] A. Basiri *et al.*, “Indoor location based services challenges, requirements and usability of current solutions”, *Computer Science Review*, vol. 24, 2017.

[6] P. Roy and C. Chowdhury, “A Survey on Ubiquitous WiFi-based Indoor Localization System for Smartphone Users from Implementation Perspectives”, *CCF Transactions on Pervasive Computing and Interaction*, vol. 4, no. 3, 2022.

[7] L. Flueratoru *et al.*, “High-Accuracy Ranging and Localization With Ultrawideband Communications for Energy-Constrained Devices”, *IEEE Internet of Things Journal*, vol. 9, no. 10, 2021.

[8] J. Kunhoth *et al.*, “Indoor positioning and wayfinding systems: a survey”, *Human-centric Computing and Information Sciences*, vol. 10, no. 1, 2020.

[9] Z. Y. Dong, W. M. Xu, and H. Zhuang, “Research on ZigBee Indoor Technology Positioning Based on RSSI”, *Procedia Computer Science*, vol. 154, 2019.

[10] Á. De-La-Llana-Calvo *et al.*, “Weak Calibration of a Visible Light Positioning System based on a Position-Sensitive Detector: Positioning Error Assessment”, *Sensors*, vol. 21, no. 11, 2021.

[11] J. A. Paredes *et al.*, “A Gaussian Process Model for UAV Localization Using millimetre Wave Radar”, *Expert Systems with Applications*, vol. 185, 2021.

[12] S. Yang *et al.*, “An Improved Vision-based Indoor Positioning Method”, *IEEE Access*, vol. 8, 2020.

[13] A. Riady and G. P. Kusuma, “Indoor Positioning System Using Hybrid Method of Fingerprinting and Pedestrian Dead Reckoning”, *Journal of King Saud University-Computer and Information Sciences*, 2021.

[14] A. Morar *et al.*, “A Comprehensive Survey of Indoor Localization Methods Based on Computer Vision ”, *Sensors*, vol. 20, no. 9, 2020.

[15] Y. Wu *et al.*, “A Survey of the Research Status of Pedestrian Dead Reckoning Systems Based on Inertial Sensors”, *International Journal of Automation and Computing*, vol. 16, 2019.

[16] R. F. Brena *et al.*, “Evolution of Indoor Positioning Technologies: A Survey”, *Journal of Sensors*, vol. 2017, 2017.

[17] WiGLE, *WiGLE: Wireless Network Mapping*, 2022. [Online]. Available: `https://wigle.net/`.

[18] T. Fryza, T. Bravenec, and Z. Kohl, “Security and Reliability of Room Occupancy Detection Using Probe Requests in Smart Buildings”, in *2023 33rd International Conference Radioelektronika (RADIOELEKTRONIKA)*, IEEE, 2023.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, *Communications of the ACM*, vol. 60, no. 6, 2017.

[20] C. Szegedy *et al.*, “Going Deeper with Convolutions”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[21] C. Szegedy *et al.*, “Rethinking the Inception Architecture for Computer Vision”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[22] N. Ma *et al.*, *ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design*, arXiv preprint arXiv:1807.11164, 2018. arXiv: `1807.11164 [cs.CV]`.

[23] M. Sandler *et al.*, “MobileNetV2: Inverted Residuals and Linear Bottlenecks”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[24] J. Deng *et al.*, “Imagenet: A Large-Scale Hierarchical Image Database”, in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Ieee, 2009.

[25] F. Pedregosa *et al.*, “Scikit-learn: Machine learning in Python”, *the Journal of machine Learning research*, vol. 12, 2011.

7 Appendix

7.1 Author's CV

Tomas Bravenec received the M.S. degree in electronics and communications from Brno University of Technology, Czechia in 2019. He is currently pursuing the joint-Ph.D. degree at University Jaume I Spain and at Brno University of Technology, Czechia, working as an Early Stage Researcher (ESR) within the A-WEAR project. His research interests include machine learning, indoor localization, and privacy and security issues related to wearable applications.

7.2 Abstract

The field of Location-based Services (LBS) has experienced significant growth over the past decade, driven by increasing interest in fitness tracking, robotics, and eHealth. This dissertation focuses on evaluating privacy measures in Indoor Positioning Systems (IPS), particularly in the context of ubiquitous Wi-Fi networks. It addresses non-cooperative user tracking through the exploitation of unencrypted Wi-Fi management frames, which contain enough information for device fingerprinting despite MAC address randomization. The research also explores an algorithm to estimate room occupancy based on passive Wi-Fi frame sniffing and Received Signal Strength Indicator (RSSI) measurements. Such room occupancy detection has implications for energy regulations in smart buildings. Furthermore, the thesis investigates methods to reduce computational requirements of machine learning and positioning algorithms through optimizing neural networks and employing interpolation techniques for IPS based on RSSI fingerprinting. The work contributes datasets, analysis scripts, and firmware to improve reproducibility and supports advancements in the LBS field.