University of Hradec Králové
Faculty of Informatics and Management
Department of Information Technologies

MASTER'S THESIS

Radio Fingerprint Acquisition Using Smartwatch

Author: Bc. David Sucharda

Study programme: Applied Informatics

Supervisor: Ing. Pavel Kříž, Ph.D.

Hradec Králové April 2018

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a uvedl jsem všechny použité prameny a literaturu.

Declaration

I declare that I have elaborated this thesis independently and listed all the sources and literature.

Hradec Králové day 26th of April 2018 Bc. David Sucharda

Poděkování

Rád bych zde poděkoval Ing. Pavlu Kříži, Ph.D. za odborné vedení práce, podnětné rady a čas, který mi věnoval.

Thanks

I would like to thank to Ing. Pavel Kříž, Ph.D. for professional guidance, incentive advices, and the time he gave me.

Anotace

Název práce: Sběr rádiových fingerprintů pomocí chytrých hodinek

Diplomová práce se zabývá možnostmi sběru rádiových otisků (fingerprintů) za pomoci chytrých hodinek. Tyto otisky se používají k lokalizaci uvnitř budovy. Hlavním cílem této práce je prozkoumat možnosti sběru otisků a návrh aplikace která bude tento sběr umožňovat. V první části práce je potřeba zjistit, jestli je tento sběr na hodinkách vůbec možný. V další části je zpracování aplikace na mobil a hodinky. A jako poslední část této práce je sběr otisků a jejich analýza. Jeden z osobních cílů je zpracovat tuto aplikaci aby byla co nejvíce uživatelky přívětivá.

Annotation

The Master's thesis deals with possibilities of collecting radio fingerprints with the help of smart watches. These prints are used in indoor localization. Main aim of this thesis is to explore possibilities of fingerprint collection and creation of application that will allow it. First part is to figure out if this collection is even possible using smart watch. Next part deals with creation of such application not only for watch but also for the phone. And at the end part there is testing of fingerprint collection and data analysis. One of the personal goal is to make this application as user friendly as possible.

Content

List of figures

List of tables

1 Introduction

As the technology evolves it unlocks more and more possibilities. Just few years back there were no smart watches or phones but at this time they are important part of our lives. As they evolve there is the need for them to have more functions and features. One of them is to locate it's position on the map. This information is very useful since it can prevent people from getting lost, figuring out path to drive, used by military and countless more cases.

Finding out such position is possible using Global Navigation Satellite System (GNSS). Multiple implementations of this system exist such as GPS, GLONASS or Galileo. All of these systems provide location using sufficient number (at least four) of satellites [1, 2]. GNSS solution requires clear line of sight between satellites and the receiving device because signal is not able to pass through buildings. That makes it the main reason why it cannot be used for indoor localization.

There are multiple approaches to find out location inside the building. They can be divided into three main types. First, using wireless signal ranging approach with multiple kinds of data such as Time of Arrival (ToA). Second, using special equipment such as active bats (Ultrasonic). Final, based on Signal Strength Fingerprint Maps (SSFM), in which first part is to collect signal strengths from the environment and construct fingerprint maps. They are then used to match with current signal to obtain the location [3].

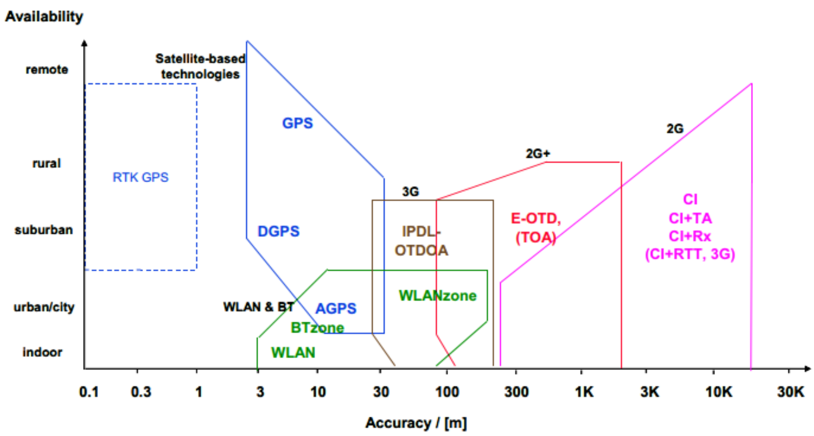

Figure 1.1: Comparison of Positioning Technologies (source: [4])

In addition to these types there are also multiple algorithms used in indoor localization. Some of them are location fingerprinting, triangulation, proximity and dead reckoning [5]. Description of few algorithms can be found in chapter 2.

This thesis is focused on method using radio signal strength (RSS) fingerprinting collecting data from bluetooth, wireless and cellular networks.

1.1 Goals of this thesis

Main goal of this thesis is to explore possibilities of fingerprint acquisition using smartwatch technology. The first question that needs to be answered is if this can be done. Is smartwatch capable of RSS data collection? And the answer to this question is yes since smartwatches have the similar specifications as low-end smartphones.

One of the goals for this thesis is to create an application for Android phone and wear device which handles fingerprint collection. Problem with smartwatches is their diversity in operational system because a lot of watch creators build their own custom systems which can complicate things. Luckily there is new system from Android called Wear 2.0 and it is basically port of Android system to wearable devices.

And final goal is to test created application and figure out if it's data are useful for indoor localization or not.

1.2 Reason for selection of this topic

The reason behind selection of this topic is rather simple. I was introduced to Android during my studies at the University but it was not any deep knowledge so I decided to go for a study abroad to deepen my knowledge. Part of that study was to work for a company where we developed rather technical heavy Android application. It's core part was using multiple APIs but it was focused only on a singe device. So next thing I wanted try was working with multiple kinds of devices and since Android Wear 2.0 is rather new I wanted to test it out. So the main reason is to get more experienced with Android and as a developer.

2 Localization techniques

This chapter describes most common techniques and methods for localization. Most of these approaches have multiple implementations and can be also used in parallel. Fingerprinting for example can be used to increase accuracy of other methods.

2.1 Triangulation

Methods based on Triangulation use geometric properties of triangles to determine target position. This can further be divided into Lateration and Angulation [6]. There are multiple sources of data these methods can use such as distance estimation between device and specific transmitters, measurements of the signal propagation-time (TOA: Time Of Arrival and TDOA: Time Difference of Arrival[7]) and the direction of received signal (AOA: Angle of Arrival[8]) [9].

2.1.1 Lateration

Lateration refers to the technique of determining position based on distance measurements that are calculated using specific devices that know their own position. Mainly used types of Lateration and are Trilateration and Multilateration.

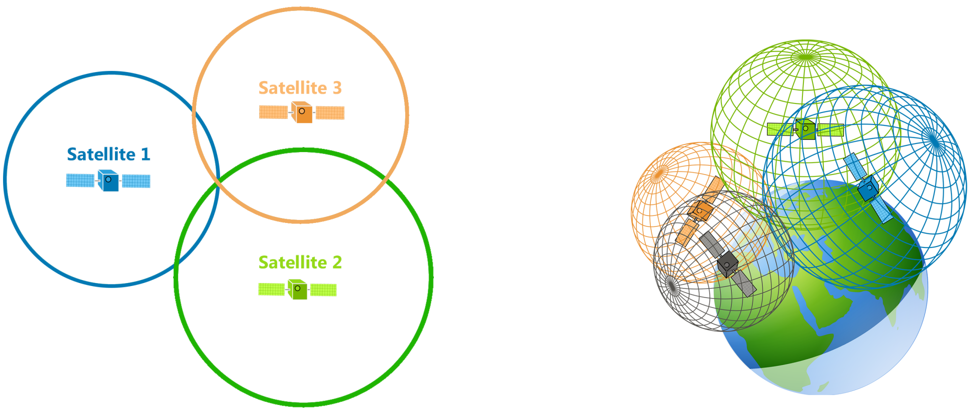

Figure 2.1: 2D and 3D Trilateration (source: [10])

Trilateration uses distance measurements from at least three devices in particular as "tri" in the name suggests [6]. Figure 2.1 illustrates usage of Trilateration in 2D and 3D environment. While working in 2D plane will result with only one specific location point. Moving to the 3D plane can create a problem because signal is send in a sphere which could result in more than one position. That is the reason why some systems use at least four signal sources, example of such system is GPS [2]. Advantage is easy implementation and simple calculations. One down side of this approach is that all devices must have synchronized clock [6].

Multilateration also known as hyperbolic positioning is using Time Difference of Arrival (TDoA) instead of Time Of Arrival (ToA) used in previous case. This approach involves the intersections of hyperbolas rather than circles as shown in Figure 2.2. Main advantage of this method is that only receiving devices must have synchronized clock instead of all [12]. Multilateration was developed for tracking aircraft position and it is widely used.

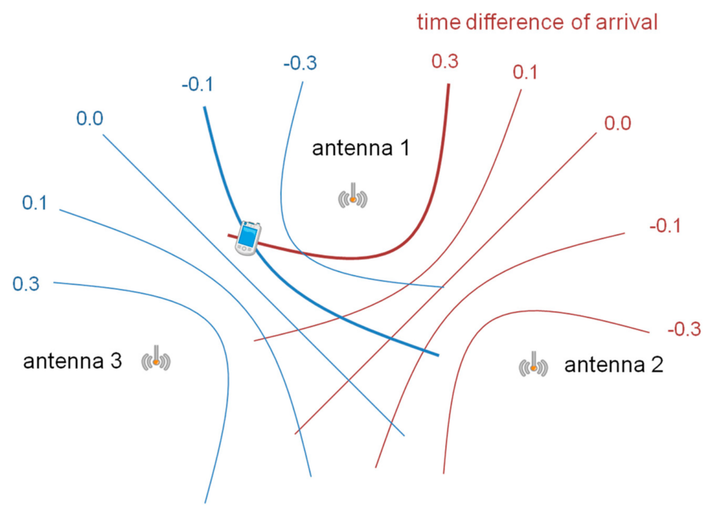

Figure 2.2: Multilateration (source: [11])

Note: At this time term Multilateration is not as strict as it used to be. It can now refer Lateration with more than three devices.

2.1.2 Angulation

This technique uses Angle of Arrival (AoA) of radio signals to determine location. It uses highly directional antennas or antenna arrays. Same as Lateration these antennas are placed in known location and basic AoA requires at least two of them to determine position on 2D plane but more of them can be used to improve accuracy [6]. That makes it an advantage over

Trilateration. Second advantage of this approach is no need for synchronization between devices.

There are few disadvantages of this approach since it needs complex hardware setup due to the use of antennas. Other problem is with multipath locations since it can cause signal reflection making it not useful for indoor localization. And final one to mention is the decrease of accuracy when mobile target moves further from the antennas [13, 14].

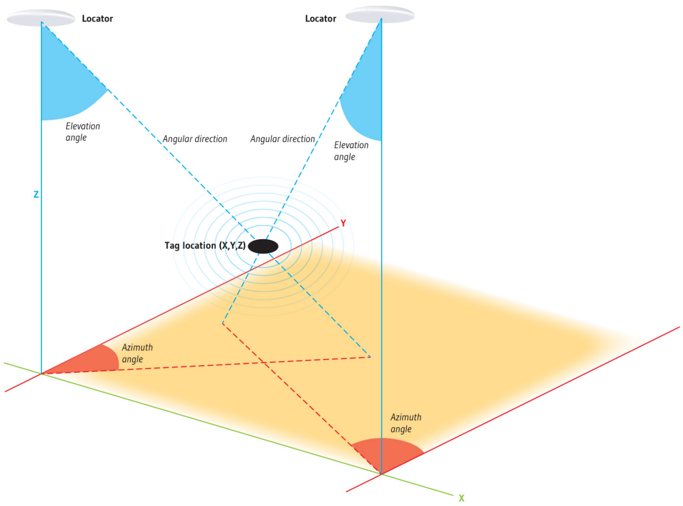

Figure 2.3: 3D location using AoA from Quuppa Intelligent Locating System (source: [15])

2.2 Fingerprinting

This method is a part of Signal Strength Fingerprint Maps (SSFM) type. Main point of this approach is using previously recorded data to figure out device location. Hence fingerprint term in the name. There are multiple kinds of radio signal sources such as bluetooth, wireless or cellular devices that can be recorded.

Fingerprinting has two main phases where the first one is fingerprint maps construction also called offline phase. They are created by collecting Received Signal Strength (RSS) and optional extra features in known locations. All these values are saved in the database and it is called fingerprint map. The second phase is localization itself also known as online phase where the device measures RSS values and compares them with fingerprint maps to approximate position using suitable method [3, 16]. Most used algorithms or methods to approximate position are [9]

- probabilistic methods,
- k-Nearest Neighbors,

- neural networks,
- support vector machine,
- smallest M-vertex polygon.

There are multiple advantages of this approach and the most important is that it does not need any additional or specialized hardware. Next one is no need for time synchronization between the stations. Both of these advantages make it simple and cost effective method for localization. On the other hand building of the map is very time consuming and needs heavy calibration. It is also susceptible to changes in environment such as people presence, object movement or relative humidity [9, 17].

2.3 Proximity

Proximity detection also known as connectivity based positioning calculates only approximate location. Position is determined by cell of origin (CoO) method with known position and limited range [6]. Specific device location is based on cell of the connected device ("associated access" point in Wi-Fi 802.11 systems) as shown in Figure 2.4 [18].

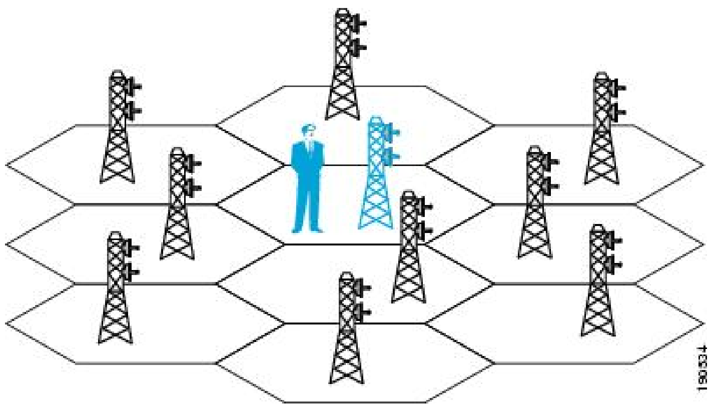

Figure 2.4: Cell of Origin (source: [18])

Primary advantage of this approach is very easy implementation and no need for complicated algorithms and thus making calculations fast. However, for various reasons devices can be associated to cells that are not in close physical proximity. Such errors can happen for example in multi-floor buildings where floor cells overlap. There are additional methods that can be used to improve localization such as using received signal strength indication (RSSI), manual method (human search) or connecting to device with highest signal strength [6, 18].

2.4 Other techniques

Scene analysis is a pattern recognition method that uses features of a scene observed from a particular vantage point to draw conclusions about the location of the observer or of objects in the scene also [19]. This approach has been used in many applications, such as image and speech recognition, as well as location [20]. The advantage is that the location of objects can be inferred using passive observation and features. The disadvantage is that the observer needs to have access to the features of environment against which it will compare its observed scenes [19].

Dead Reckoning refers to a position solution that is obtained by measuring or deducing displacements from a known starting point in accordance with motion of the user [21]. Basically calculate new position based on starting point, travel distance and angle of movement. Because new position calculations are dependent on previous ones there is the need for high accuracy of data since it makes errors cumulative [22].

3 Related Work

Since this is not a novel idea there are some completed solutions and paper written about Fingerprint collection using multiple devices. This chapter will describe few selected solution close to this one for comparison.

3.1 Improving Precision by Using Multiple Wearable Devices

Focused on improving indoor localization using BLE-based fingerprinting with multiple devices [23]. Using combination of smart phone and wear should in this case prevent signal obstruction from human body and at least one of the devices should receive beacon signal. Unfortunately due to low BLE sensibility of wear devices, authors decided to supplement them with smartphone, in this case with Nexus 5 running Android 4.4.

This paper proposes calculating medians using 800 millisecond tests where user can move at most one meter. For all medians at the same position is calculated average and variance, based on which a normal distribution is used to model the potential variation of RSSI. One thing to note is that fingerprint maps are built based on facing direction. There are five main scenarios tested

- P1: device held in hand where body does not obstruct its LOS path to all beacons,
- P2: device is put in breast pocket and LOS may be obstructed to some beacons,
- P3: user holds smartphone in hand and wears a smartwatch on one wrist,
- P4: one device is put in the breast pocket and the other is on one wrist,
- P5: one device is in the breast pocket, and two other devices, each on one wrist.

Firstly, four of these scenarios were tested in a 15x8 meters entrance hall with four deployed beacons in the corners and ten measurement positions. In this location using multiple devices can improve position precision and reduce error by 57%. Table 3.1 shows mean errors for previously mentioned cases where using more devices improves localization. However there is one position where case P4 will result with higher error then P3 due to building structure and signal obstruction for nearest beacons.

Scenario	Mean error (m)
P1	2.36
P2	1.71
P3	0.96
P4	0.41

Table 3.1: Mean errors at first location (sources: [23])

Secondly, all of previously mentioned scenarios were tested in a conference room with unified ceiling and desks near the walls. This location is used to investigate the impact of beacon density to position precision. Three following combinations of beacons are used

- A: four beacons at the corners,
- B: combination A with one beacon at the center,
- C: combination B with four more beacons at the sides of the room.

Using directional maps at this location resulted in increase of maximum localization error, which was not expected. This error can increase even more when using higher count of beacons and occurs mostly when testing at the edges of the room. Mean error on the other hand shows an improvement, higher with more beacons used.

	Mean error (m)		
Scenario	A	B	C
P1	2.05	2.05	1.76
P2	1.84	1.49	0.90
P3	1.36	1.09	0.63
P4	1.24	0.78	0.23
P5	0.80	0.39	0.07

Table 3.2: Mean errors at second location (sources: [23])

In summary, position error can be improved by three aspects: using more devices, using directional map or increasing the number of beacons. There are two main conclusions of this paper. First, confirmed precision degradation when testing near the edge of the room with obstructed signal from nearest beacon. Second, human body does not greatly change radio signal and device can receive reflected signals with sufficient strength.

3.2 SmartFix

An Indoor Locating Optimization Algorithm for Energy-Constrained Wearable Devices called SmartFix [24]. The main goal of this paper is improve energy consumption efficiency for wearable-based indoor localization using WiFi fingerprints. Single real-time experiment of energy consumption was run and split into two main parts: computation of location and collection of fingerprints. According to this experiment, energy consumption caused by collection is occupies 99% of localization algorithm.

This paper proposes novel indoor localization strategy, SmartFix, that can cooperate with existing indoor localization technologies based on WiFi. It enhances accuracy of such algorithm with a little extra energy cost of calculation but a large save of signal collection energy cost. Aided with machine-learning algorithm, it obtains the relative features given the trajectories of users in certain areas and modify the localization results. SmartFix can save up to 70% of energy while achieving the same localization accuracy when compared to the original fingerprint method.

To test this new system it was implemented with prototypes of TinyLoc [25], MoLoc [26] and basic WiFi fingerprint method using k-neighbor algorithm. TinyLoc is more focused on energy efficiency than location accuracy. In contrast SmartFix analyses the history trajectory of people in given area to improve locating results. It refers to user motion features, SmartFix modifies locating results to achieve satisfying accuracy. MoLoc also leverages user motion by collecting trajectory patterns using device built-in sensors.

All previously mentioned prototypes were deployed with and without SmartFix to test its power consumptions. It only needs single real-time RSS signal in the locating phase to guarantee excellent energy saving performance. Figure 3.1 shows energy consumption of on-time locating on two specific devices: HTC one and Moto 360.

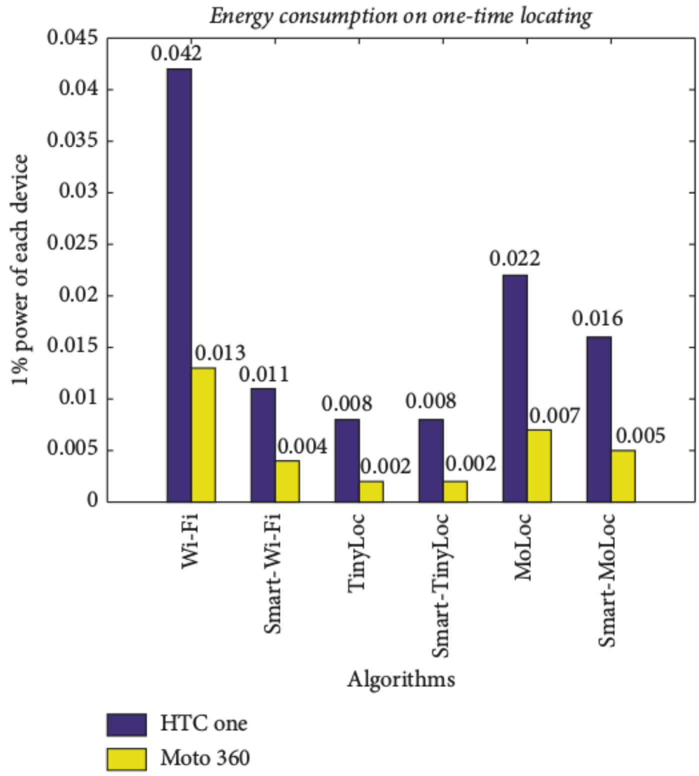

Figure 3.1: Energy consumption based on prototypes (source: [24])

This paper proposed an tested new localization technology, SmartFix, which main focus is to improve energy efficiency for wearable devices. According to experiment, the probability of error within 2 meter can be reached in 80% of cases. Meanwhile, energy consumption is 35% lower than that of MoLoc with the same accuracy, and SmartFix obtains best accuracy with minimal energy cost.

3.3 SmartWatch vs. SmartPhone

http://www.eislab.fim.uni-passau.de/files/publications/2017/1570325956.pdf

4 Android

This chapter will provide information about Android and Wear 2.0 technology. Why it was developed and what are the differences between previous version and other wear technologies.

Android is a Linux based operating system for mobile and wear devices developed by Google. The main selling point of this system is that it's open-source project, meaning everyone can access the code and modify it as they wish. Android was mainly developed for mobile phones but in time moved beyond and at this time is implemented into all kinds of wear devices, tablets, televisions and even refrigerators or cameras [46].

4.1 Android system structure

Android is created as a stack, meaning there are functional modules stack on top of each other from Linux core over native libraries to applications as shown in Figure 4.1. Android maintains complete software stacks to enable device creators to run and modify Android for their specific hardware. To support these modifications and testing every release has multiple "code lines" to separate stable versions from experimental work [28].

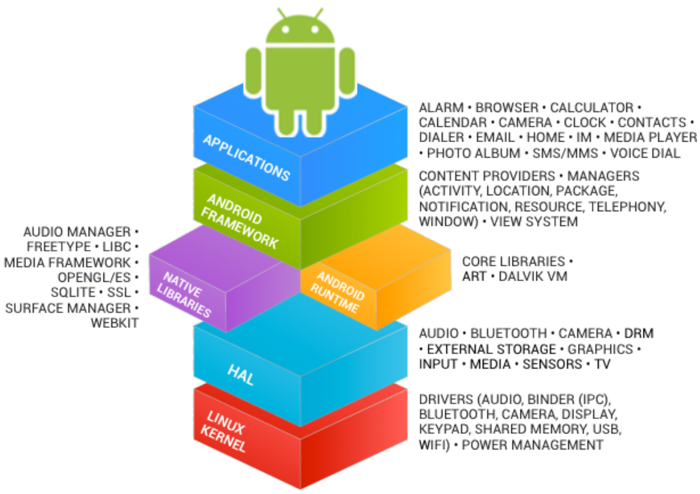

Figure 4.1: Android stack (source: [28])

There are multiple versions of Android system at this time and every single one has it's own version, code name and API level. Version codes are number identifications of a specific system version. Highest levels of these numbers are grouped into code names that are ordered alphabetically. As an example versions 8.0.0 and 8.1.0 have the same code name called Oreo. Finally API level is number identification for compatibility of specific application and it will be compared to API level of device Android system [28, 29].

The highest part of the Android system stack are applications which extend device functionality and are written primarily in Java programming language [33]. These application are packaged into `.apk` file, which is a zip archive, containing all application files like Java classes, layouts, images and more. The most important file is `AndroidManifest.xml` that contains all meta-data about the application, such as permissions, package name, used components, versions and so forth. These application can be shared, for a nominal fee, via official market called Google Play. At the end of 2017 there were over three and a half million applications available in Google Play Store [33, 35, 36].

Android is platform designed to be open source and free which makes it easy to create malicious applications. These application can bypass existing security and steal sensitive data, use telephony services or even gain control over the device [34]. Android has multiple ways to protect against such applications one of the most notable ones are Android Permission Framework and Google Play Protect [33].

4.2 Wear technologies

Interactive wearable, as an example smartwatches, is a new part of mobile computers. Wear devices are categorically different from phones or tables in term of usage, design and user interfaces (UI). According to the app design guidelines by major vendors, users interact with wearable devices frequently throughout daily use. Each interaction is short, often less than 10 seconds, and is dedicated to simple tasks [30].

Important thing to note is that there are multiple kinds of wear devices from smart watches, wristbands, cameras or even glasses [31]. Based on report from Gartner technology research, conducted in 2017, most used wear devices were Bluetooth headset, wristbands and smartwatch [32]. Thanks to their small size wear devices are ideal to use for hands-free communication and health monitoring.

One problem with this diversity is hardware and software compatibility. Every device creator can create their own operating system for specific wear device and it can be difficult to develop custom applications for it. To avoid such problems this thesis is focused only on smartwatch with Android Wear operating system.

There are three main points to note with watch devices. First, small battery capacity that can be almost ten times smaller then in typical smartphone. Second, point is display with around forty-times less pixels which completely changes properties. Final, scaled down CPU with high efficiency [30]. Last two points are main parts of lowering power consumption of smartwatches but even with these cuts high-end watch devices can have really small battery life only in matter of few days or just hours.

4.2.1 Android Wear

Android Wear is a version of Android OS tailored to small-screen wearable devices. There are not too many changes from system for smartphone but one of the main differences can be seen in UI since system had to be adjusted for watch size [37]. Due to scaled down processing power of watches Android Wear wirelessly offloads data to the smartphone for heavy computing tasks, e.g., voice recognition [38].

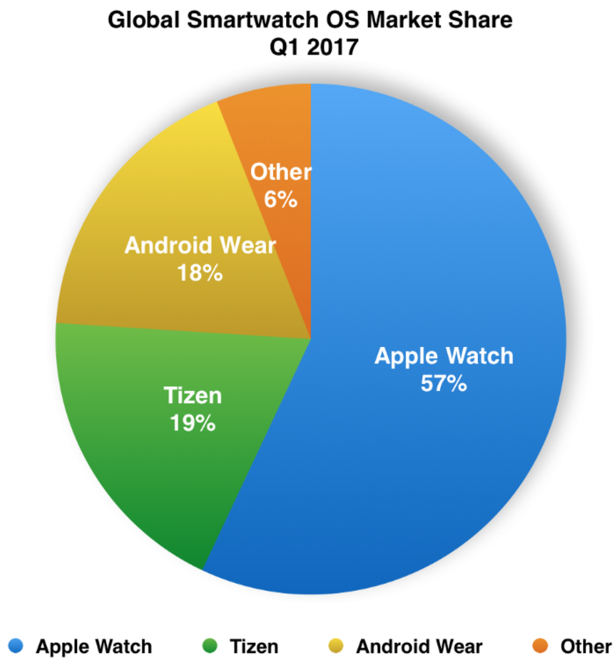

Figure 4.2: Smartwatch OS market share (source: [41])

Android Wear is one of the most popular smartwatch systems but comes with it is own set of problems. Most notable and annoying one is being unable to pair wear with specific mobile device. It can be caused by combination of things such as system compatibility, custom hardware or phone type but it is more common then it should be [39]. Having phone connected to Android Wear can also cause phone to drain battery way faster. One thing to note is that smartwatch can be connected to only single device and connecting to another requires factory reset. Few other problems can be update issues, notifications not coming through to the watch, not being able to connect to Wi-Fi and system crashes [40].

Even with all these problems it is popular system and in early 2017 it got it's biggest update yet. New version 2.0 brought numerous improvements and features and few of the most notable ones will be described in this section [42–44].

4.2.1.1 Standalone applications

This feature is crucial change and it means that watch applications do not need mobile phone to function. Before this version it was needed to have connected phone with Android Wear support to use applications. Being forced to have Android phone proved as an obstacle for users without one since they could not use any applications on the watch [42, 43].

Since application can now work without phone there should be a way to install them directly. Thankfully part of this update is also standalone Google Play Store where users can browse apps that are designed specifically for the watch [43]. Part of this feature is also enabling watch to use wireless and cellular networks on their own since most standalone applications require this feature. And final part of this update was improving and securing communication with phone. This is now done via Wearable Data Layer API that is used in almost all Google applications and it is also pretty easy to use as a developer [42].

4.2.1.2 UI improvements

Part of new Android Wear version is implementation of Android's Material design guidelines [45]. It has much more "mature" look and darker design for reducing battery drain [43]. It is completely focused on Wear devices and supports both round and square screens with new re-design of application launcher [42].

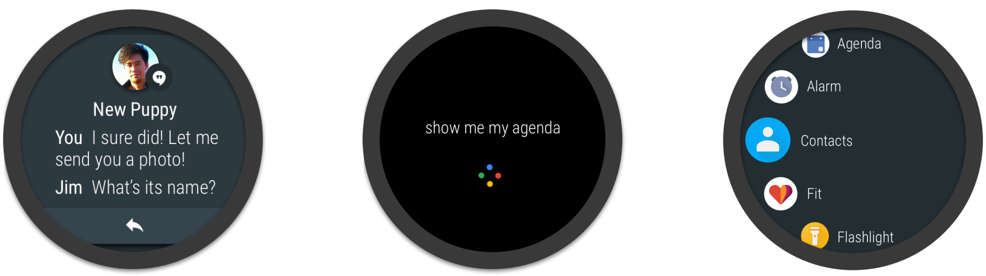

Figure 4.3: Wear design examples (source: [45])

Android is also trying to catch up with Apple's watchOS and make default watch display, also called watch faces, much more useful. Users can add different widgets of any containing data of any application to the watch faces [42]. This ensures quick access to the data user deems important [44]. All this data displays also match design of currently selected watch face and after clicking it will direct right into the application [43].

Figure 4.4: Wear watch faces (source: [42])

4.2.1.3 Google Assistant

Google Assistant is basically voice controlled smart assistant same as for example Amazon's Alexa, Apple's Siri or Microsoft's Cortana. There are multiple tech sites that run benchmarks of these systems [47–50] and they do not seem to be that different so there is no need to buy one over the other. These systems can pull information that you need or want and they track where you work, sports you like, your schedule, stuff that might interest you and much more [46]. With the update of Wear 2.0 this feature is now available on smartwatches [42, 43].

4.3 Other wear technologies

Tizen, Pebble, Apple, ...

5 Application design and implementation

This chapter describes all important information about created application. First is hardware and software used for developing and testing of the application. Second is structure and description of core parts used in the application.

5.1 Hardware

There are three main Hardware components used and those are smartphone, smartwatch and BLE beacons. Both smart devices must support scanning of Bluetooth Low Energy beacons that can be done with Bluetooth 4.0 and higher. Secondary requirements are Wi-Fi, GSM and LTE modules to support more data types than just BLE beacons.

5.1.1 Smartphone

Main part of the application will be developed and tested on Redmi Note 4 from Chinese company Xiaomi. It is running customized version of Android 6.0 called MIUI. Even thought system was customized, in core it is still Android so there are no problems in that regard [52]. This phone has Bluetooth 4.1 with LE support so main requirement for the hardware is met. This device also met all the secondary requirements with Wi-Fi 802.11 a/b/g/n, GSM, and LTE support like most modern smartphones would [51].

One interesting thing about Xioami smartphones is their locked bootloader. This prevents user from any manual updates but most importantly from factory reset. To unlock it owner has to create an account in Xiaomi website and put a request to unlock bootloader. This request is usually processed within two weeks and there is no actual guarantee of it being approved. User is later notified via sms about the result of this request.

5.1.2 Smartwatch

As already mentioned this thesis is using smartwatch with support of Android Wear 2.0 which makes it harder to select proper wear device since there not so many options at this time. There was around twenty of watches with 2.0 system and only five of them were selected to closer inspection based on few articles [53–55].

Watch	BLE / Wi-fi	Czech Republic	Problems
LG W280 Sport	Yes / Yes	No	Battery life is one day or less. Too big in size.
LG W270 Titanium Style	Yes / Yes	Yes	Battery life is one day or less.
Huawei Watch 2	Yes / Yes	Yes	First update can take a long time. Slight Bluetooth pairing issues.
Polar M600	Yes / Yes	Yes	Polar support complains. Phone synchronization issues. GPS location malfunctions.
ASUS ZenWatch 3	Yes / Yes	No	Strap breaks fast. AW 2.0 update can break the watch. ASUS support complains.

Table 5.1: Smartwatch comparison (sources: [56–60])

There were funds only for one watch device out of five which is displayed in Table 5.1 with selection parameters. First requirement for the wear device is support of BLE and Wi-Fi which all have. Second information considered was being able to buy it in Czech Republic (CR) since it is easier for prices, shipping and warranty. Only three of five devices were sold in CR at that time so others are out of the question. Final decision was made based on extensive research of customer reviews in shopping sites (Amazon, CZC, Heureka, Alza), wear official websites [56–60] and other tech sites [53–55]. Finally, selected device is Huawei Watch 2 since there were not too many problems in reviews and other requirements were met.

Initial setup of the wear device was composed of two main parts. First one was update wear system that took about one to two hours. Second task was copy Google account into the wear device where problem was discovered. Copying of accounts from Redmi Note 4 ti the watch hanged nd never completed. To fix this problem another smartphone was used to copy the account but as it was already mentioned only single device can be connected to smartwatch and connecting a new one requires factory reset that would remove all the data. So there was the need to pair wear with phone without factory reset which was handled via debugging following this [61] article.

5.1.3 BLE beacons

Beacons are small devices that can be easily placed in almost any environment. The only thing they do is send their information packets using bluetooth and are used in museums, airports and as of late in indoor localization [63]. Beacons have their own battery that can last around a year without charging because of the new Bluetooth Low Energy (BLE) standard that consumes little to no energy [62].

There are multiple beacon manufacturers with their own quality, signal strength, battery life and other hardware differences. There are also multiple software (packet) specifications for beacons meaning that data they send have different format but they does not mean other systems cannot see them. Beacons are usually platform independent and work with multiple platforms [62, 63].

Figure 5.1: Parts of Estimote beacon (source: [64])

Another important thing to note it that Beacons are not connected to the Internet and do not collect any data from devices around. Meaning all the data have to be processed in another device, most commonly a smartphone, and it also makes Beacon safe to use because there is no need to worry about sensitive data theft [63].

5.2 Software

Software part for smart phone and wear uses Android system which was already described in Chapter 3. This section will on the other hand provide basic information about libraries, technologies and systems used in the application.

5.2.1 AltBeacon Library

Since Android core does not allow scanning for BLE beacons this feature must be implemented in a library extending system features. There are multiple solutions that can be used to scan for beacons such as Estimote SDK [65] which was already used in previous thesis [66]. To change things up BLE beacons are found via AltBeacon Library [67].

Since at the time there was no open and inter-operable specification for proximity beacons, Radius Networks has created the AltBeacon specification as a proposal to solve this problem. It is an open and free specification for BLE beacons with focus to create an open, competitive market for implementation of these devices [68].

Basic configuration of this library can scan for beacons based on AltBeacon standard. As a second type of beacons it supports Eddystone which is Google's open source beacon format and calculation of range between the devices which will help with localization. Luckily this library function can be easily modified to support different kinds of beacons by just one following line of code [67, 69].

```
beaconManager.getBeaconParsers().add(
   new BeaconParser().setBeaconLayout("m:2-3=0215,i:4-19,
      i:20-21,i:22-23,p:24-24"));
```

Code example 5.1: Enable all beacons

5.2.2 Database

Database is needed to keep all the Fingerprint data for calculations and there are two types of databases used for this application. First one is SQLite database that is used in Android application to save Fingerprints. This database is default and most used solution in Android applications and there is no need to use other ones. Another type of database used is Couchbase which is implemented on `beacons.uhk.cz` server to keep all Fingerprint data on one place for multiple applications.

5.2.2.1 Couchbase database

Since SQL based databases can be complex to implement, scale and consuming more data storage space there was a drive to create so called NoSQL databases. Their most important feature is not having a scheme, that makes them easy to scale, and easy to replicate between multiple devices. There are around 225 NoSQL databases at this time and Couchbase is one of them [71].

Couchbase is distributed, document-based database with its own querying language called N1QL. It is a database focused on simple server configuration and easy use for clients and with built in caching layer and distribution system it does not require any changes in the application. There can be either one server instance of Couchbase or multiple of them can be connected and create a database cluster that holds all the data in multiple location (nodes) [72].

Since SQL is used for decades and it became standard for working with data in databases this approach was extended for usage with JSON and called N1QL. It has all features of the SQL with only difference of being focused on JSON files that has no structure and it is used in Couchebase [73].

There is a version of Couchbase for mobile called Lite that provides APIs to work with the documents and synchronize with the server instance. Problem of this solution is that Lite version does not implement N1QL and maps data as document views that have to instanced and usually use more data, at least on Android.

5.2.2.2 SQLite database

SQL (Structured Query Language) is a standard language for storing, manipulating and retrieving data in databases. It is a type of Relational Database that means all data is saved into tables with rows and columns [**?**]. These tables usually have set amount of rows with specific names that protect from adding wrong data as en example you cannot add data `Person(name, surname, eye color)` into table `Person(name, surname)` because there is no column named `eye color` in the table.

Advantages of these databases are structured data which makes calculation faster but uses more storage space. Data can be only saved once since they can be connected to each other. It supports complex queries for creating, reading, updating and removing data (CRUD) and better security with user and table management. Some disadvantages of this system can be with complexity and inflexibility of database scheme because it can be hard to setup and it does not allow other data then is defined in the tables [74].

Since SQL with all the features can consume a lot of hardware resources for a smartphone that is not as fast as a server Android decided to implement SQLite. SQLite has the following noticeable features: self-contained, serverless, zero-configuration, transactional [75].

- Serverless = does not need second process for the server.
- Self-Contained = requires minimal support from operating system.

- Zero-configuration = no need for installation or any configuration.
- Transactional = data are protected against failed changes (application crashes, power failure, ...).

5.2.2.3 Comparison

Both of these database solutions were tested on Android and SQL proved as a better solution. Not only it takes less storage space it is also able to load data faster. As Table 5.2 shows SQL lite takes less data space and is almost three times faster in loading all the documents.

Database type	Data size	Loading speed (315 documents)
SQLite	15MB	23 second
Couchbase without views	31MB	65 seconds
Couchbase with views	91MB	65 seconds

Table 5.2: Couchbase vs SQLite (sources: [56–60])

As shown SQL comes up better but it does not support any data synchronization so there must be custom solution created. If Couchbase Lite would have faster query time it would be preferable solution but it is not at this time.

5.2.3 TileView

There are multiple ways to display map image in Android but the main ones have problems while displaying a big image because they will run out of memory. So it is usually better to try custom library or widget created specifically for displaying big image. There is multiple 2D and 3D libraries to display such an image but most of them display image as a whole meaning one big image file.

There is another way introduced by Google Maps where map is displayed using small images so called `tiles`. Image is cut into small pieces which is better for device memory management than one big image. Depending on how big the cuts are it can also display multiple zoom levels keeping the map quality high with high and low levels.

5.3 Application structure

This section describes structure of the applications and since there are two parts of this application (mobile and wear) both of them will be described.

5.3.1 Mobile application

Mobile application is the bigger and more complex part that handles scanning, map display and synchronization with the server.

5.3.1.1 Activities

Activities are visible screens of the application. They serve as the entry point for a user's interaction with an application, and are also central to how a user navigates within an application (as with the Back button) or between apps (as with the Recents button) [29].

Main activity for this application is called `ScanActivity`. It displays map with proper building, floor and found Fingerprint in that specific location. It enables users to scan for new beacons and thus create new Fingerprints. Map and controls are implemented using TileView widget that was described in previous section.

5.3.1.2 Model

Model package contains most important classes of the application and it's split into four parts called: adapters, configuration, database and tasks.

Adapters

Adapters information.

Configuration

Configuration classes contain settings of the application. It uses Android's shared preferences that are key-value (xml) files to save small amount of application data. Each SharedPreferences file is managed by the framework and can be private or shared. One thing to note is that these files are not encrypted so it is not good to save user sensitive data in them.

Database

Database package contains all classes needed for communication between application and SQL database. First part of this package are Fingerprints classes that correspond with tables and columns of the database. Second part are so called database helpers that handle communication between the application and SQLite database such as tables creation, selects, inserts, updates and deletions.

Tasks

Tasks contain the most complex classes of this application and that is the Fingerprint scanner that has multiple sub-classes to parse all the necessary data from the scan. Tasks also contain connection with the server for data synchronization and Fingerprint transfers.

5.3.1.3 Utilities

Utilities contain classes not fit to any other category or classified as support ones which usually means classes for animations or warning messages but since this application communicates with the wear device this package also contains classes that handle such communication.

5.3.2 Wear application

Wear application is simplified version of its smartphone counterpart. It handles only scanning, display of scanning progress and sending the data to the phone.

6 Testing and data analysis

This chapter goal is to show application testing, data collection and analysis.

6.1 Data collection

6.2 Analysis

7 Conclusion

7.1 Application improvements

Literature

[1] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger and Elmar Wasle. *GNSS – Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007 [cited 2018-01-10], ISBN 9783211730171.

[2] AviationChief. *Global Navigation Satellite System (GNSS) Global Positioning Satellite (GPS) System* [online]. AviationChief.Com, 2017 [cited 2018-01-15]. Available at: `http://www.aviationchief.com/gps-system.html`

[3] Xinglin Piao, Yong Zhang, Tingshu Li, Yongli Hu, Hao Liu, Ke Zhang and Yun Ge. *RSS Fingerprint Based Indoor Localization Using Sparse Representation with Spatio-Temporal Constraint* [online]. National Center for Biotechnology Information, 2016 [cited 2018-01-14], Available at: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5134504/`

[4] Stéphane Beauregard and Harald Haas. *Pedestrian Dead Reckoning: Basis for Personal Positioning* [online]. School of Engineering and Science International University Bremen, 2006, Available at: `http://ave.dee.isep.ipp.pt/~lbf/PINSFUSION/BeHa06.pdf`

[5] Gabriel Deak, Kevin Curran and Joan Condell. *A survey of active and passive indoor localisation systems*. In: *Computer Communications*. Elsevier, 2012 [cited 2018-01-11], Volume 35, Issue 16, ISSN: 0140-3664.

[6] Zahid Farid, Rosdiadee Nordin, and Mahamod Ismail. *Recent Advances in Wireless Indoor Localization Techniques and System* [online]. School of Electrical, Electronics & System Engineering, University Kebangsaan Malaysia (UKM), 2013 [cited 2018-01-15], Available at: `http://downloads.hindawi.com/journals/jcnc/2013/185138.pdf`

[7] Shweta Singh, Ravi Shakya and Yaduvir Singh. *Localization techniques in wireless sensor networks* [online]. Department of Computer Science, Ideal

Institute of Technology, Ghaziabad, 2015 [cited 2018-01-15], ISSN: 0975-9646, Available at: https://pdfs.semanticscholar.org/6299/85defbf9cc1a937a1b88c9c2a893552e3d89.pdf

[8] Paweł Kułakowski, Javier Vales-Alonso, Esteban Egea-López, Wiesław Ludwin and Joan García-Haro. *Angle-of-arrival localization based on antenna arrays for wireless sensor networks* [online]. In: *Computers & Electrical Engineering*. Elsevier, 2010 [cited 2018-01-15], Volume 36, Issue 6, Pages 1181-1186. Available at: http://ai2-s2-pdfs.s3.amazonaws.com/17c6/0e17c4e72cc3fd821e12169c1c2ca7736bd4.pdf

[9] Pavel Kriz, Filip Maly, and Tomas Kozel. *Improving Indoor Localization Using Bluetooth Low Energy Beacons* [online]. In: *Mobile Information Systems*. Hindawi Publishing Corporation, 2016 [cited 2018-01-15], Volume 2016, Article ID 2083094. Available at: https://www.hindawi.com/journals/misy/2016/2083094/abs/

[10] GISGeography. *Trilateration vs Triangulation – How GPS Receivers Work* [online]. GISGeography.com, 2018 [cited 2018-01-15]. Available at: http://gisgeography.com/trilateration-triangulation-gps/

[11] Kenjirou Fujii, Yoshihiro Sakamoto, Wei Wang, Hiroaki Arie, Alexander Schmitz and Shigeki Sugano. *Hyperbolic Positioning with Antenna Arrays and Multi-Channel Pseudolite for Indoor Localization* [online]. MDPI AG, Basel, 2015 [cited 2018-01-15]. Available at: http://www.mdpi.com/1424-8220/15/10/25157/htm

[12] David Munoz, Frantz Bouchereau Lara, Cesar Vargas and Rogerio Enriquez-Caldera. *Position Location Techniques and Applications*. Elsevier Science Publishing Co Inc, 2009 [cited 2018-01-15], ISBN: 9780080921938. Available at: http://www.mdpi.com/1424-8220/15/10/25157/htm

[13] Group 891: Wireless Location. *ANGULATION: AOA (Angle Of Arrival)* [online]. DEPARTMENT OF ELECTRONIC SYSTEMS, Aalborg University, 2010 [cited 2018-01-15]. Available at: http://kom.aau.dk/group/10gr891/methods/Triangulation/Angulation/ANGULATION.pdf

[14] Jais, M. I., Ehkan, P., Ahmad, R. B., Ismail, I., Sabapathy, T., and Jusoh, M. *Review of angle of arrival (AOA) estimations through received signal strength indication (RSSI) for wireless sensors network (WSN)* [online]. In: Computer,

Communications, and Control Technology (I4CT), 2015 International Conference on. IEEE, 2015, [cited 2018-01-16], p. 354-359. Available at: https://www.researchgate.net/profile/Phaklen_Ehkan/publication/283476641_Review_of_angle_of_arrival_AOA_estimations_through_received_signal_strength_indication_RSSI_for_wireless_sensors_network_WSN/links/564106b008aebaaea1f6d6e5/Review-of-angle-of-arrival-AOA-estimations-through-received-signal-strength-indication-RSSI-for-wireless-sensors-network-WSN.pdf

[15] Quuppa Oy. *Quuppa Intelligent Locating System* [online]. 2018 [cited 2018-01-16]. Available at: http://quuppa.com/technology/

[16] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N. Padmanabhan. *Indoor Localization Without the Pain* [online]. In: Proceedings of the sixteenth annual international conference on Mobile computing and networking, 2010 [cited 2018-01-16], Available at: http://dl.acm.org/citation.cfm?id=1860016

[17] Xiaoyang Wen, Wenyuan Tao, Chung-Ming Own, and Zhenjiang Pan. *On the Dynamic RSS Feedbacks of Indoor Fingerprinting Databases for Localization Reliability Improvement* [online]. Sensors, 2016 [cited 2018-01-16], Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5017443/

[18] Cisco. *Wi-Fi Location-Based Services 4.1 Design Guide - Location Tracking Approaches* [online]. Cisco, 2018 [cited 2018-01-16], Available at: https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Mobility/WiFiLBS-DG/wifich2.html

[19] Jeffrey Hightower and Gaetano Borriello. *Location systems for ubiquitous computing* [online]. Computer, 2001 [cited 2018-01-17], 34.8: 57-66. Available at: http://www.csd.uoc.gr/~hy439/lectures11/hightower2001survey.pdf

[20] COOK, B., et al. *Location by scene analysis of wi-fi characteristics* [online]. Relation, 2009 [cited 2018-01-17], 10.1.119: 6216. Available at: http://www.ee.ucl.ac.uk/lcs/previous/LCS2006/2.pdf

[21] Levi, R.W. and Judd, T. *Dead reckoning navigational system using accelerometer to measure foot impacts* [online]. Google Patents, 1996 [cited 2018-01-17]. Available at: https://www.google.com/patents/US5583776

[22] Z. Zhou, T. Chen, L. Xu. *An Improved Dead Reckoning Algorithm for Indoor Positioning Based on Inertial Sensors* [online]. In: Advances in Engineering Research, 2015 [cited 2018-01-17]. ISBN: 978-94-62520-71-4. Available at: `https://www.atlantis-press.com/proceedings/eame-15/22314`

[23] NAKAJIMA, Naoki, et al. *Improving Precision of BLE-based Indoor Positioning by Using Multiple Wearable Devices* [online]. In: Adjunct Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing Networking and Services. ACM, 2016 [cited 2018-03-26]. p. 118-123. Available at: `https://dl.acm.org/citation.cfm?id=3004041`

[24] WANG, Xiaoliang; XU, Ke; LI, Ziwei. *SmartFix: Indoor Locating Optimization Algorithm for Energy-Constrained Wearable Devices.* In: Wireless Communications and Mobile Computing, 2017 [cited 2018-03-26]. Available at: `https://dl.acm.org/citation.cfm?id=3004041`

[25] W. Xiaoliang, X. Ke, Y. Zheng, and Z. Ge. *Tinyloc: Indoor localization for energy-constrained wearable devices* [online]. In: Chinese Journal of Computers, 2016 (Chinese), [cited 2018-03-26]. Available at: `http://www.cnki.net/kcms/detail/11.1826.TP.20161106.1649.002.html`

[26] SUN, Wei, et al. *MoLoc: On distinguishing fingerprint twins* [online]. In: Distributed Computing Systems (ICDCS), 2013 IEEE 33rd International Conference on. IEEE, 2013 [cited 2018-03-26]. p. 226-235. Available at: `http://ieeexplore.ieee.org/abstract/document/6681592/`

[27] Marziah Karch. *What Is Google Android?* [online]. Lifewire, 2017 [cited 2018-01-17]. Available at: `https://www.lifewire.com/what-is-google-android-1616887`

[28] Android. *Android Open Source Code* [online]. Android.com, 2018 [cited 2018-01-17]. Available at: `https://source.android.com/`

[29] Android. *Android Developers* [online]. Android.com, 2018 [cited 2018-01-17]. Available at: `https://developer.android.com/index.html`

[30] Renju Liu and Felix Xiaozhu Lin. *Understanding the Characteristics of Android Wear OS* [online]. In: Proceedings of the 14th Annual International Conference

on Mobile Systems, Applications, and Services. ACM, 2016. p. 151-164. Available at: https://athena.smu.edu.sg/mobisys/backend/mobisys/assets/paper_list/pdf_version/paper_12.pdf

[31] Samuel Gibbs. *10 most influential wearable devices* [online]. Guardian News, 2017 [cited 2018-01-18]. Available at: https://www.theguardian.com/technology/2017/mar/03/10-most-influential-wearable-devices

[32] Gartner, Inc. *Gartner Says Worldwide Wearable Device Sales to Grow 17 Percent in 2017* [online]. Gartner, Inc., 2017 [cited 2018-01-18]. Available at: https://www.gartner.com/newsroom/id/3790965

[33] Bahman Rashidi and Carol Fung. *A Survey of Android Security Threats and Defenses* [online]. JoWUA, 2015, [cited 2018-01-19]. Available at: https://www.researchgate.net/profile/Bahman_Rashidi2/publication/282365848_A_Survey_of_Android_Security_Threats_and_Defenses/links/560ec06908ae6b29b499a51f/A-Survey-of-Android-Security-Threats-and-Defenses.pdf

[34] Parvez Faruki, Ammar Bharmal, Vijay Laxmi, Vijay Ganmoor, Manoj Singh Gaur and Mauro Conti. *Android Security: A Survey of Issues, Malware Penetration and Defenses* [online]. IEEE Communications Surveys and Tutorials, 17(2), pp. 998-1022, 2015, [cited 2018-01-19]. Available at: http://dx.doi.org/10.1109/COMST.2014.2386139

[35] Statista. *Number of available applications in the Google Play Store from December 2009 to December 2017* [online]. Statista, 2018, [cited 2018-01-19]. Available at: https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/

[36] AppBrain. *Number of Android applications* [online]. AppBrain, 2018, [cited 2018-01-19]. Available at: https://www.appbrain.com/stats/number-of-android-apps

[37] LIU, Xing, et al. *Characterizing Smartwatch Usage In The Wild* [online]. In: Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services. ACM, 2017 [cited 2018-01-19]. p. 385-398. Available at: https://pdfs.semanticscholar.org/0cc2/4bccc3067ed688e576603bc6bab0e5e1b1db.pdf

[38] Liu, Renju, and Felix Xiaozhu Lin. *Understanding the Characteristics of Android Wear OS* [online]. In: Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services. ACM, 2016 [cited 2018-01-19], p. 151-164. Available at: https://athena.smu.edu.sg/mobisys/backend/mobisys/assets/paper_list/pdf_version/paper_12.pdf

[39] Sandra Henshaw. *Android Wear Problems (and Solutions!)* [online]. Tiger Mobiles Limited, 2016 [cited 2018-01-20]. Available at: https://www.tigermobiles.com/2016/01/android-wear-problems-and-solutions/

[40] Simon Hill. *10 of the worst Android Wear problems, and how to fix them* [online]. Designtechnica Corporation, 2017 [cited 2018-01-20]. Available at: https://www.digitaltrends.com/wearables/android-wear-problems/

[41] ADNAN F. *Tizen overtakes Android Wear in smartwatch market share* [online]. SamMobile, 2017 [cited 2018-01-20]. Available at: https://www.sammobile.com/2017/05/11/tizen-overtakes-android-wear-in-smartwatch-market-share/

[42] Paul Lamkin. *Android Wear 2.0: Ultimate guide to the major smartwatch update* [online]. Wareable, 2017 [cited 2018-01-20]. Available at: https://www.wareable.com/android-wear/android-wear-update-everything-you-need-to-know-2735

[43] Elyse Betters and Chris Hall. *Android Wear 2.0: What's new in the major software update for watches?* [online]. Pocket-lint Limited, 2017 [cited 2018-01-20]. Available at: https://www.pocket-lint.com/smartwatches/news/google/139007-android-wear-2-0-what-s-new-in-the-major-software-update-for-watches

[44] Chris Martin. *Android Wear 2.0 news: release date and features* [online]. Tech Advisor, 2017 [cited 2018-01-20]. Available at: https://www.techadvisor.co.uk/new-product/google-android/android-wear-2-3640616/

[45] Android Developers. *Designing for Android Wear* [online]. Android, 2018 [cited 2018-01-20]. Available at: https://developer.android.com/design/wear/index.html

[46] Elyse Betters. *What is Google Assistant, how does it work, and which devices offer it?* [online]. Pocket-lint Limited, 2018 [cited 2018-01-20]. Available at:

https://www.pocket-lint.com/apps/news/google/137722-what-is-google-assistant-how-does-it-work-and-which-devices-offer-it

[47] DAN MOREN. *Alexa vs. Siri vs. Google Assistant: Which Smart Assistant Wins?* [online]. Tom's Guide, 2017 [cited 2018-01-20]. Available at: https://www.tomsguide.com/us/alexa-vs-siri-vs-google,review-4772.html

[48] Digital Trends Staff. *Virtual assistant comparison: Cortana, Google Assistant, Siri, Alexa, Bixby* [online]. Digital Trends, 2017 [cited 2018-01-20]. Available at: https://www.digitaltrends.com/computing/cortana-vs-siri-vs-google-now/

[49] Brian Heater. *Comparing Alexa, Google Assistant, Cortana and Siri smart speakers* [online]. TechCrunch, 2017 [cited 2018-01-20]. Available at: https://techcrunch.com/2017/10/08/comparing-alexa-google-assistant-cortana-and-siri-smart-speakers/

[50] Joe Hindy. *Google Assistant vs Siri vs Bixby vs Amazon Alexa vs Cortana – Best virtual assistant showdown!* [online]. Android Authority, 2017 [cited 2018-01-20]. Available at: https://www.androidauthority.com/google-assistant-vs-siri-vs-bixby-vs-amazon-alexa-vs-cortana-best-virtual-assistant-showdown-796205/

[51] GSMArena. *Xiaomi Redmi Note 4 - Full phone specifications* [online]. GSMArena, 2018 [cited 2018-01-21]. Available at: https://www.gsmarena.com/xiaomi_redmi_note_4-8531.php

[52] XiaomiMobile. *Xiaomi Redmi Note 4 LTE* [online]. XiaomiMobile, 2018 [cited 2018-01-21]. Available at: https://xiaomimobile.cz/xiaomi-redmi-note-4-pro-lte-global.html?search_query=note+4&results=16

[53] Android Authority Team. *Best Android Wear watches (old version)* [online]. Android Authority, 2017 [cited 2018-01-21]. Available at: https://www.androidauthority.com/best-android-watches-572773/

[54] James Peckham. *Best Android Wear watch 2018: our list of the top Google OS smartwatches* [online]. TechRadar, 2017 [cited 2018-01-21]. Available at: http://www.techradar.com/news/wearables/every-android-wear-smartwatch-in-the-world-today-1288283

[55] Michael Simon. *Best Android Wear watches of 2017* [online]. PCWorld, 2017 [cited 2018-01-21]. Available at: https://www.pcworld.com/article/3209668/android/best-android-wear-watches-of-2017.html

[56] LG Electronics. *LG Watch Sport™ - AT&T* [online]. LG Electronics, 2018 [cited 2018-01-22]. Available at: http://www.lg.com/us/smart-watches/lg-W280A-sport

[57] LG Electronics. *LG Watch Style* [online]. LG Electronics, 2018 [cited 2018-01-22]. Available at: http://www.lg.com/us/smart-watches/lg-W270-Titanium-style

[58] HUAWEI Technologies Co. *HUAWEI WATCH 2* [online]. HUAWEI Technologies Co., 2018 [cited 2018-01-22]. Available at: http://consumer.huawei.com/en/wearables/watch2/specs/

[59] Polar Electro. *Polar M600* [online]. Polar Electro, 2018 [cited 2018-01-22]. Available at: https://support.polar.com/e_manuals/M600/Polar_M600_user_manual_English/Content/technical-specifications.htm

[60] ASUSTeK Computer Inc. *ASUS ZenWatch 3* [online]. ASUSTeK Computer Inc., 2018 [cited 2018-01-22]. Available at: https://www.asus.com/us/ZenWatch/ASUS-ZenWatch-3-WI503Q/specifications/

[61] Adam Conway. *How to Pair Android Wear Watches to New Phones without Factory Resetting* [online]. xda-developers, 2017 [cited 2018-01-22]. Available at: https://www.xda-developers.com/pair-android-wear-without-factory-reset/

[62] Locatify. *Indoor Positioning Systems based on BLE Beacons – Basics* [online]. Locatify, 2015 [cited 2018-01-27]. Available at: https://locatify.com/blog/indoor-positioning-systems-ble-beacons/

[63] Patrick Leddy. *10 Things About Bluetooth Beacons You Need to Know* [online]. pulsate, 2015 [cited 2018-01-27]. Available at: http://academy.pulsatehq.com/bluetooth-beacons

[64] The Estimote Team Blog. *Reality matters — Preorder for Estimote Beacons available, shipping this summer* [online]. Estimote, Inc., 2013 [cited 2018-01-27]. Available at: http://blog.estimote.com/post/57087851702/preorder-for-estimote-beacons-available-shipping

[65] *Estimote SDK for Android* [online]. Github, 2018 [cited 2018-01-22]. Available at: https://github.com/Estimote/Android-SDK

[66] Radek Brůha. *Pokročilé metody rádiové indoor lokalizace* [online]. Univerzita Hradec Králové, 2017 [cited 2018-01-22]. Available at: https://theses.cz/id/jss047

[67] *Android Beacon Library* [online]. Radius Networks, 2018 [cited 2018-01-22]. Available at: https://altbeacon.github.io/android-beacon-library/

[68] *AltBeacon* [online]. AltBeacon, 2018 [cited 2018-01-22]. Available at: http://altbeacon.org/

[69] *Eddystone format* [online]. Google Developers, 2018 [cited 2018-01-22]. Available at: https://developers.google.com/beacons/eddystone

[70] *NOSQL Databases* [online]. NoSQL, 2018 [cited 2018-01-27]. Available at: http://nosql-database.org/

[71] *NOSQL Databases* [online]. NoSQL, 2018 [cited 2018-01-27]. Available at: http://nosql-database.org/

[72] Brown, Martin C. *Getting Started with Couchbase Server: Extreme Scalability at Your Fingertips* [online]. O'Reilly Media, Inc., 2012 [cited 2018-01-27]. Available at: https://books.google.cz/books?hl=cs&lr=&id=5xu33G9LGkMC&oi=fnd&pg=PR5&dq=Couchbase&ots=nrw7O3HiVh&sig=6qmpVJLxwdOK9RRZsICHUfsD2wI&redir_esc=y#v=onepage&q&f=false

[73] *What is N1QL?* [online]. Couchbase, 2018 [cited 2018-01-27]. Available at: https://www.couchbase.com/products/n1ql

[74] *Explain Relational Database Management System (RDBMS)* [online]. W3Schools, 2016 [cited 2018-01-23]. Available at: http://whatisdbms.com/explain-relational-database-management-system-rdbms/

[75] *What Is SQLite* [online]. SQLite Tutorial, 2018 [cited 2018-01-23]. Available at: http://www.sqlitetutorial.net/what-is-sqlite/

Attachments

1. CD with application
   a) Application data
2. Public Github repository with application and thesis text.
   `https://github.com/Del-S/WearNavigation`

Univerzita Hradec Králové
Fakulta informatiky a managementu
Akademický rok: 2017/2018

Studijní program: Aplikovaná informatika
Forma: Prezenční
Obor/komb.: Aplikovaná informatika (ai2-p)

Podklad pro zadání DIPLOMOVÉ práce studenta

PŘEDKLÁDÁ:	ADRESA	OSOBNÍ ČÍSLO
Bc. Sucharda David	Masarykovo náměstí 3, Nová Paka	I1500697

TÉMA ČESKY:

Sběr rádiových fingerprintů pomocí chytrých hodinek

TÉMA ANGLICKY:

Radio Fingerprint Acquisition Using a SmartWatch

VEDOUCÍ PRÁCE:

Ing. Pavel Kříž, Ph.D. - KIKM

ZÁSADY PRO VYPRACOVÁNÍ:

Cíl: Navrhnout a implementovat aplikaci pro mobilní telefon a chytré hodinky na platformě Android Wear 2.0, pomocí které bude možné změřit rádiové fingerprinty souběžně pomocí mobilního telefonu a hodinek. Vyhodnotit a porovnat přesnost indoor lokalizace s použitím existujících algoritmů využívajících rádiové fingeprinty.

Osnova:
1. Úvod
2. Indoor lokalizace s pomocí rádiových fingerprintů
3. Platforma Android Wear 2.0
4. Analýza a návrh
5. Implementace
6. Testování, zhodnocení výsledků
7. Závěr

SEZNAM DOPORUČENÉ LITERATURY:

https://www.hindawi.com/journals/misy/2016/2083094/
https://www.microsoft.com/en-us/research/project/radar/
https://developer.android.com/wear/index.html
http://www.sciencedirect.com/science/article/pii/S1877050912005170
https://link.springer.com/article/10.1007%2Fs13218-017-0496-6
http://www.mdpi.com/1424-8220/17/6/1299
http://www.mdpi.com/1424-8220/17/6/1339
http://www.mdpi.com/1424-8220/17/8/1789

Podpis studenta: .... **Datum:** 15. 11. 17

Podpis vedoucího práce: .... **Datum:** 15. 11. 17