RSSI based indoor localisation of multi-floor buildings using a Weighted k-Nearest Neighbor approach

Submitted in partial fulfillment for the degree Master of Science

Sander van Wickeren
11060999

Master Information Studies
Data Science University of Amsterdam
The Netherlands

19-08-2020

	Internal Supervisor	**Internal Supervisor 2**	**Internal Supervisor 3**
Name	dr. Maarten Marx	dr. ing. Cristian Rodriguez Rivero	ing. Edwin Steffens
Affiliation	UvA, IVI	UvA, IVI	UvA, IVI
Email	M.J.Marx@uva.nl	c.m.rodriguezrivero@uva.nl	E.H.Steffens@uva.nl

	External Supervisor	**External Supervisor 2**
Name	Jasper Kaizer MSc	Tom Lokhorst MSc
Affiliation	Q42	Q42
Email	jasper@q42.nl	tom@q42.nl

ABSTRACT

Bluetooth Low Energy (BLE) fingerprinting uses a database that connects the locations and RSSI (Bluetooth Received Signal Strength Indicator) values of beacons, to estimate indoor positions. RSSI suffers from attenuation and reflection due to environmental properties, influencing the performance. The aim of the study is to find to what extent a Weighted k-Nearest Neighbor (WkNN) fingerprinting approach works better in a practical environment than the proximity room-based approach currently used in the old, multi-floor Rijksmuseum building. Through an analysis of existing RSSI filters, distance metrics, data representations and environmental factors, is examined which variables create the best performing model, whether a single model or fingerprint database is possible for all devices, and how the absolute error could be reduced. In addition, this study provides a public data set on which other researchers can test and compare their models. The results suggest that all devices benefit from using Sørensen distance in combination with a normal or exponential data representation, that a single model and fingerprint database can be used, at the expense of a small increase in error, and that the Moto G5 achieved the lowest absolute mean error of 4.30 meters on the Iphone 6S fingerprint database, with 96.5% of the floors correctly predicted. The proposed system does not perform significantly better than the current system at room level, but can be more accurate within a room. The findings are limited by manually determined locations in the data, the usage of only deterministic filters and the difficult comparison with the literature and the current system. Future research should look into standardisation of reporting the performance, including environmental properties like beacon density, to make comparisons more meaningful.

KEYWORDS

BLE, RSSI, Distance metrics, Filters, Indoor localisation, Weighted k-Nearest Neighbor, WkNN, Human influences

1 INTRODUCTION

Museums have started to use mobile technologies and sensors widely, facilitating the visits and providing an enhanced visitor experience [37, 41]. This also applies to the Rijksmuseum, one of the most visited museums of the Netherlands. As a national institute, they offer a representative overview of Dutch historical art and important aspects of European and Asian art [1]. With the Rijksmuseum app, visitors can follow guided audio tours, search for additional art information and look through an extensive art gallery. In order to guide people from beginning to the end of the tour, low-energy Bluetooth beacons have been used. These small devices broadcast Bluetooth signals containing an identifier, which get recognised by the app and converted to a location. This technique is referred to in literature as proximity. However, a couple of constraints from the Rijksmuseum and the Bluetooth technology cause the localisation quality of users to be inconsistent. Firstly, the signals broadcasted by beacons interfere because of: the wooden construction parts between different rooms and floors [23], WiFi interfering with Bluetooth signals [3, 43], human bodies weakening the signals [17, 21], interference by reflections [31] and environmental changes, like the density of people, positioning of furniture [9, 16] and multipath propagation. The interference can even be used to detect people without them wearing any Bluetooth devices [8]. Secondly, beacons are placed at sub-optimal positions because the Rijksmuseum requires that no distractions should be made from the objects on display, and that the technology should therefore not be visible for the public [43]. This also prevents the usage of other localisation techniques like WiFi [18] or Near Field Communication (NFC) [33]. Thirdly, smartphones of different brands and types each have distinctive sensors, which means that one phone receives fewer signals than others in certain locations. Fourthly, the environment is constantly changing, at one point there are many people in the same room, while at other times it is a lot quieter. In addition, exhibits in the rooms can also change, which affects the signals.

The current Rijksmuseum app has been developed and maintained by Q42, a company that builds apps, websites, connected devices, games, robots, voice assistants, AI and VR products [20]. Initially, the location of users was determined by a third party API, which identified the room based upon the strongest signal. Next to the signal strength, the API provides an estimated distance towards the beacons in meters. Q42 improved the strongest signal localisation by applying two heuristic rules to reduce misinterpreted locations. Due to the earlier mentioned interference and uncertainty in the signals and the fact that the distance between two beacons can be about the same, it is possible that one moment beacon A is seen as nearest and the next moment beacon B. To prevent the app user from regularly experiencing a change of location, because the distance between two beacons is approximately the same, the first rule applies a 'preference bonus' of 2.5 meters to the received distance from the last nearest beacon. For example when the app registered A as the last location and new data have been measured that beacon A is 6 meters away and beacon B is 5 meters away, the app keeps location A, because $Min(6 - 2.5, 5)$ results in A.

The second rule is intended to prevent switching between floors by using the travel time between the last known location A and the measured location B. When the distance between the two locations in meters is not reachable within the time between the two measurements (assuming people walk 2 meters per second), the app keeps location A.

Although the rules improved the localisation, the heuristic approach is far from optimal because the problem has not been addressed at the direct source and the locations are still flawed. Existing research often use a single device to test performance, have an optimal beacon layout, are only tested in controlled lab environments or do not incorporate multiple floors in a single model. This study fills the gap by applying state of the art fingerprinting techniques on multiple devices in a practical environment and providing a public data set for other researchers to apply and test algorithms on. Answering the question: *To what extent improves a Weighted k-Nearest Neighbor algorithm the localisation of people in multi-floor buildings where beacons are sub-optimally positioned?*

Additionally, this research answers which filters are the most promising to reduce signal fluctuations caused by the environment, to what extent humans influence the signals, which data representations and distance metrics perform best and to what extent it is possible to use a single model for all devices.

The study has been executed in couple of steps. The literature section provides an overview of the possible filters, distance metrics, data representations, various applied kNN fingerprinting methods and other models. A conclusion is drawn which filter is the most promising and how fingerprinting can be applied. To test the performance of the Weighed kNN, Android and iOS apps have been built to record and store the signal strength of the available beacons in combination with the real location. The collected data is extensively described. Then, the algorithm is explained, the optimal parameters are found and an error analysis is executed. Finally, the proposed model is compared to previous literature and the current system and a conclusion is drawn as to whether the proposed approach is an improvement.

2 LITERATURE REVIEW

Indoor localisation is often based on monitoring the Received Signal Strength Indicator (RSSI) using mobile devices. The RSSI is used for both Wifi and Bluetooth Low Energy (BLE) localisation solutions, operating on the license-free 2.4 GHz band and sharing the same propagation characteristics. BLE uses forty channels, each 2 MHz wide, of which only three are used as advertising channels to save energy for the scanner (named 37, 38 and 39). The 37 connection channels allow two parties to connect and share data with each other. Figure 1 shows a comparison between the BLE and Wifi channels.

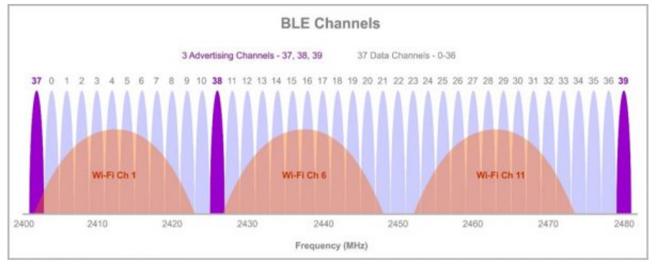

Figure 1: WiFi (red) and Bluetooth (purple) channels and their width within the 2.4 GHz band. The advertising channels are colored dark purple [2].

The localisation solutions follow three main directions. Firstly, proximity, where the localisation is based on the location of the beacon with the strongest signal. Secondly, trilateration, where localisation is based on the intersection of distances between the receiver and multiple beacons. The distance between each beacon is calculated by converting the RSSI into meters using a Log-Distance Path Loss model. These distances can be used to create a ranging circle around the known locations of the beacons. Calculating the intersection of these circles gives an approximation of the location. Trilateration is different from triangulation. The former is using distances, while the latter is using angles. Trilateration can also be applied on time characteristics or on the direction of signals, such as respectively Time Difference of Arrival (TDOA) and Angle of Arrival (AOA) [24]. Thirdly, fingerprinting, existing of two phases. The offline phase, where the actual locations are stored in a database together with the RSSI value for each beacon. And the online phase, where a measurement is compared to the fingerprint database. The closest fingerprints, calculated with for example the Euclidean distance, determines the approximated location. Using a kNN approach, the true location of the closest fingerprint is the estimated location with $K = 1$, When $K > 1$, the estimated location is an average of the K fingerprints.

While trilateration is more time efficient than fingerprinting, because creating a database beforehand takes a lot of time, one of the disadvantages of a trilateration approach is the problem of converting the RSSI signals into distances. The non-linear relation between distance and RSSI makes the readable range of signal strength narrow [5]. Madigan et al., nuances that in a laboratory setting the RSSI signals decay approximately linearly with log distance [29], but interference in a real environment makes this ratio more difficult to realise. Moreover, the variables of the Log-Distance Path Loss model needs to be adjusted for every beacon, especially in the Rijksmuseum where beacons are hidden away behind different materials, each with their own effect on the RSSI and possibly the distance model [23]. Another disadvantage of trilateration is that in most of the cases the result is an area of possible locations instead of a single point [10]. One way to mitigate this is by selecting the point located in the middle of the area. This approach is referred to as centroid localisation.

2.1 Fingerprinting

RADAR was one of the first systems applying the offline and online phase on radio frequencies to determine indoor positions. During the offline phase a search space for the kNN algorithm is constructed, referred to as the fingerprint database. In the online phase the database is used to infer the location of new data in real time. Bahl & Padmanabhan found, while developing RADAR, that kNN performed best in comparison with the propagation model and the strongest signal localisation, with an error distance of 2.94, 4.3 and 8.16 meters respectively [4]. Zang et al., performed a study comparing kNN, support vector machine and neural networks and concluded that a kNN-regression model was generally the best candidate for the algorithm [47].

One effect that occurs while using RSSI signals, which is related to the interference problems stated in the introduction, is that the ranging error increases as the distance increases. Or in other words, the further the beacon, the less accurate the reading [11, 16]. This effect can be incorporated with the kNN by applying a weight. The closer the beacon to the receiver, the more influence it has on the approximated location. So, the location is no longer determined by the closest fingerprint, but a weighted combination of the real location from the K closest fingerprints.

Orujov & Maskeliunas compared the localisation performance of proximity, trilateration, weighted centroid and a hybrid approach, combining fingerprint kNN and the weighted centroid, and found that the latter approach performed best [32].

Dawes & Chin did an extensive research to indoor localisation based on WLANs RSSI fingerprinting by comparing various kNN approaches and probabilistic methods [12]. The study showed a couple of findings. Firstly, probabilistic methods have a similar performance to kNN approaches, but perform better in environments where the beacon placement is sparse. In general, raising the beacon density results in higher performance, but also a higher cost. Secondly, calculating the Euclidean distance using only the beacons that are in range during the measurement yield a better performance than using all the beacons. Thirdly, including the compass direction of the devices to the kNN resulted in one of the worst scores with an error distance of 6.2 meters. And finally, the weighted kNN approach was the most accurate with an error distance of 2 meter. The weight was calculated based on the rule that the closest location is weighted K times more than the least closest location. So, when for example $K = 3$, the weights would be $\frac{3}{6}$, $\frac{2}{6}$ and $\frac{1}{6}$.

2.1.2 Multi-floor buildings.

While many research focus on localisation within rooms on the same floor, the localisation within multi-floor buildings adds an extra layer of complexity. Determining the floor can be approached from multiple angles. The floor can be approximated by taking the floor of the closest fingerprint, by a majority vote of the K nearest fingerprints or by using relative barometer readings. Torres-Sospedra et al., found that the performance of the approaches is depending on the environment, but that although differences are small, using the majority vote resulted overall in the highest accuracy [38].

2.1.3 Distance metrics and data representations.

For determining the closest fingerprint, the Euclidean distance is the most used approach. However, there are many more ways to calculate distances between measurements and fingerprints. Torres-Sospedra et al., compared 53 distance metrics and four representations of the data in combination with a kNN on the *UJIIndoorLoc* database [39]. This database contains fingerprints of WiFi access points of multiple multi-floor buildings. They found that the usage of an exponential or powed representation of the data could improve the performance. For example, the use of an exponential data representation in combination with the Euclidean distance led to an error reduction of 1 meter. The best score on the exponential data representation was found using Neyman X^2, while the other data representations performed best in combination with Sørensen as distance measure.

Both data representations use equation (1), where *min* equals the weakest signal found minus one in decibel-milliwatts (dBm). The value of beacons that are not found is set to zero. Torres-Sospedra et al. used equation (2) to convert to the exponential, and equation (3) to convert to the powed data representation. They used an alpha (α) of 24 and set beta (β) to the mathematical constant e.

$$Positive_i(x) = (RSSI_i - min) \tag{1}$$

$$Exponential_i(x) = \frac{exp(\frac{Positive_i(x)}{\alpha})}{exp(\frac{-min}{\alpha})} \tag{2}$$

$$Powed_i(x) = \frac{(Positive_i(x))^\beta}{(-min)^\beta} \tag{3}$$

Machaj & Brida compared NN, kNN and WkNN in combination with seven different distance metrics. They found that the WkNN had the smallest error and that the performance of Manhatten, Euclidean, Minkowski and Sørensen distance were very similar [28]. Li et al., found that using Manhatten distance over Euclidean distance let to an improvement in combination with a WiFi based WkNN approach. The error distance was 0.48m and 0.64m respectively [25]. Although the performance of different distance metrics and data representations have been measured on WiFi based localisation systems, as both use RSSI as metric, this could mean that performance can be increased for BLE localisation systems as well.

2.2 RSSI interference

The Bluetooth standard combats interference and fading using frequency hopping, the process of switching the broadcast channels pseudo-randomly to transmit data in short chunks [7, 16]. However, this is not enough. As stated in the introduction, the RSSI fluctuates based on (changes in) the environment, multipath propagation and WiFi signals. To reduce the fluctuations and make the readings more consistent, filters are applied in order to smooth the signals. The filters are used in both the offline and the online phase. In the offline phase before the fingerprints are saved in the database, and in the online phase before calculating the closest fingerprints.

2.2.1 Smoothing interference.

A lot of research has been done to reduce the variance of the signals, increasing the precision of the localisation. But, what is the most promising way to filter interference from Bluetooth RSSI signals? The solutions can be divided into two types. Firstly, deterministic solutions, which represent the signal strength in values on a certain scale. One could think of moving averages, medians or an exponential moving average. Faragher & Harle found that applying the moving mean averages on a batch size of ten samples would be enough to get the best results [16]. Increasing the number of samples did not allow for any improvement, because earlier movements would still have a visible effect on the current position. In their more recent research they defined the moving median to be the most robust in comparison with the max, mean and raw values (although differences are small) [17]. Nakamura et al., researched the usage of exponential and non-exponential filters for detecting RSSI signals within a certain range in a short amount of time, and found that the window mean of an exponentially weighted moving average performed the best [30]. However, they ruled the well performing moving average out because it was susceptible to small-scale fading, meaning that rapid signal fluctuations impact the performance. Booranawong et al., argues that both exponential and non-exponential moving averages are viable for smoothing fluctuations, but adds that both have trouble filtering the effect of human movement on the signals [9]. A median approach could be less sensitive and reduce these fluctuations. Contreras et al., reduced the the maximum uncertainty of the RSSI from −55dBm to −5.7dBm by using the Butterworth algorithm in their experiment, but argue that the performance is a trade off between long filtering times and high uncertainty [11]. It required 143 samples to get reliable values, making it hard to use in phones, where only one sample per second is reported.

Secondly, probabilistic or Bayesian solutions which use mapped signal distributions [15]. A Kalman filter is a common technique to reduce fluctuations in data. It uses information from the previous states to be able to predict the next state and adjust new states accordingly. The Kalman filter is far from optimal because it assumes Gaussian distributed values, while RSSI fluctuations are not [26]. Moreover, Subhan et al., argue that the Kalman filter is limited because the properties of noise are often not known and it minimises the mean error only, causing the filter to fall short in its performance when there is a communication hole. [35, 36]. The extended Kalman filter (EKF) is more suitable, allowing non-linear data and models to be used. Zhuang et al., used the filter to improve BLE localisation [48].

One of the more recent filter algorithms for non-linear problems fall under the term Sigma-Point Kalman Filters (SPKF). These filters use novel deterministic sampling approaches to circumvent the need of computational complex Jacobian matrices used by the EKF [22, 42]. Khalil et al., proposed the Scaled Unscented Kalman Filter (SUKF) which outperformed the EKF in all their localisation experiments [22]. Paul et al., found that their Sigma-Point Kalman Smoother outperformed the EKF [34].

Subhan et al., used an extended gradient filter, which uses the time and RSSI differences to calculate a rate of change. The average of window size n containing the rate of changes of the measured beacons are used to predict the next RSSI value. The standard deviation of the predicted and measured value are used to calculate the filtered RSSI measurement. Results suggest that this implementation improved performance in comparison with a Kalman filter [35].

Zafari et al., compared the performance between a particle filter (PF), a combination between a Kalman filter and PF (PF-KF) and a combination between the extended Kalman filter and PF (PF-EKF), and found that the PF-EKF performed best [46]. In the survey from Zafari et al., is argued that although the server side system is energy efficient and accurate, it suffers from a considerable delay. Moreover, using a PF on a phone is not energy efficient and can reduce the battery live [45].

Other filtering techniques used for smoothing RSSI values like amongst others five-point triangular smoothing, Savittzky-Golay filter and the Lowess filter, are compared in a research by Deak et al. They found a median filter to produce the smallest mean error when filtering the fluctuations due the impact of the human body on the RSSI values [14].

Many filters have been used in literature for reducing fluctuations of RSSI signals. Filters from the Kalman family are often used as a probabilistic solution, while the median filter is the most promising as deterministic solution.

2.3 Test environments

Most of the referenced literature executed their tests in a controlled lab environment where beacons are optimally positioned, only one floor is used and humans and or obstacles do not interfere. This is also recognised by Canton et al., while analysing BLE fingerprint studies [10]. A real environment however, has to deal with these kind of influences. Often the product works in a controlled lab environment but then suffers performance degradation from the storm of interference from other 2.4GHz solutions in the field [44]. Or in the words from Lymberopoulos et al. Each system is usually evaluated in a custom, highly controlled environment making hard to draw conclusions about its performance and overhead in realistic conditions [27].

3 METHODOLOGY

3.1 Data collection and description

The data have been collected in the Rijksmuseum, with 4 floors and 30,000m^2 of space. Throughout the museum 320 Kontakt Smart Beacons SB16-2 are spread, which use Apple's Ibeacon protocol. The beacon's advertising packets contain an identifier, existing of three parts: An Universally Unique Identifier (UUID), a Major and a Minor. The UUID is used to distinguish between beacon networks, such that for example an app implementation only works in one building, and not in another with Bluetooth beacons installed. The Major is intended to identify and distinguish groups. For example each floor in a building could have its own Major value. Finally, the Minor is used to identify a single beacon. The combination of the Major and Minor value can be used to identify each unique beacon in a network. For all the beacons in its network, the Rijksmuseum has used a Major value of 1070, which means that only the use of the Minor as identification is sufficient.

In order to collect the data for multiple phones at the same time, two apps have been built. The goal of the first app was to select the real location on a scaled map of the Rijksmuseum (see Figure 13). GPS was not used to determine the location, because of the inaccurate readings indoors. Instead, the latitude and longitude of the real location was determined by placing markers on the map, estimating the position manually through the environment and the symmetrical properties of the building. For every location, the latitude and longitude together with a timestamp and the floor are saved to the device.

The goal of the second app was to record the Bluetooth RSSI, Major and Minor together with a timestamp for all beacons in range. Additionally, the compass direction of the devices from 0 to 360 degrees was recorded.

Both apps are built for Android 6 and higher, while the Bluetooth app is built for iOS 11 as well. The location app was running on a Samsung Galaxy S9. The Bluetooth app was running on an Iphone 6S, Moto G5, Samsung Galaxy S5 mini and an Imagineear MPtouch Interactive, which is the tour device from the Rijksmuseum. The Samsung Galaxy S9 was hand-held, placing markers accurately using a stylus pen. The other devices were stacked on each other and stored in a belly bag, while the Bluetooth app was active. The recorded locations are approximately six meters apart, each with ten seconds of RSSI data. During the collection few people were present.

To be able to compare the proposed with the current system, which is described in the introduction, an additional Imagineear device was used during creation of the test set. The device recorded the timestamps and rooms as reported by the current system.

After multiple days of collection, the data from the Bluetooth and Location app are combined based on the timestamp, creating three data sets per device. A training set, a validation set and a test set. The data is publicly available.

3.1.1 Limitations Android and iOS apps.

Both iOS and Android have recording limitations due to their built-in BLE classes. iOS only provides one measurement per second, while Android's standard implementation is inactive for around three seconds after being active for two seconds, reporting multiple measurements. This limits the amount of measurements within the used time frames. Moreover, there are some other differences between both operating systems. Firstly, iOS already applies a built in rolling window of twenty seconds, while Android does not. The rolling window average causes there to be a delay in measured and the actual RSSI value of the beacon. Secondly, iOS has less freedom in configuration and the source code is hidden as opposed to Android, where some configuration is possible and the code is mostly open source.

3.1.2 Training set.

The training set consists of 1817 recorded locations, from the ground floor to the third floor, spread over almost half of the museum, with coverage of approximately 14,500m^2. At every location, ten seconds of Beacon signals have been recorded. The collection took place over several days. Because the devices sometimes refused to record Bluetooth signals or ran into other problems, there were recovery moments on some days where the missed locations were re-recorded. Not every location could be replicated exactly, which means that one

Table 1: Properties of the training set. From left to right, the columns are *Number of measurements* (N_M), *Number of assigned locations* (N_L), *Number of unique assigned locations* (N_{UL}), *Number of unique beacons* (N_{UL}), *RSSI value range in dBm* ($RSSI_R$).

Device	N_M	N_L	N_{UL}	N_{UB}	$RSSI_R$
Imagineear	6125	2219	908	205	-56, -95
S5 Mini	29597	9713	1146	216	-49, -102
Moto G5	26943	9258	1389	219	-54, -103
Iphone 6S	129808	44898	1213	218	-53, -99

device can have certain locations, which the other devices do not have. Therefore, the number of unique recorded locations per device will not match 1817. Together, the devices recorded 219 unique beacons, indicating that there is on average one beacon every $66.2m^2$. Table 1, shows an overview of the training set, Table 4 displays a single row.

3.1.3 Validation set.

During the creation of the validation set, 169 locations spread over the ground, first and second floor have been recorded in a part of the museum. At every location three seconds of data have been collected. This three-second time window is based on a couple of reasons. Firstly, a very short time window would make it hard to determine the actual position during data collection precisely, increasing the error due to bad data recording. Secondly, due to the active and inactive times of Android Bluetooth measurements as mentioned in section 3.1.1, a shorter time span would make it harder to match the real location and measurements based on the timestamp. A bigger time window reduces the chance that no measurements were recorded at a location. Thirdly, a time span of ten seconds, as used during the offline phase, would produce a less realistic performance, because nobody waits ten seconds for each step during a walk in the museum. The three-second time window is a trade off between getting at least some signals at every recorded location, and being able to predict the location with only a few measurements. The quicker the correct location is determined, the better.

After the collection, for every phone, the location data is connected to the Bluetooth measurements. Together, the devices recorded 152 unique beacons. Because Android reports every measurement with a unique timestamp, while iOS bundles them and reports every second, all Android measurements within a one-second time window are combined. The median is used if there are multiple measurements from the same beacon.

Table 2 gives an overview of the data. Two things stand out. Firstly, the Iphone 6S has 488 locations, this is because the three-second time window allows one location to be linked to multiple measurements. Secondly, there is a big difference in the number of unique locations measured. The Imagineear and the Samsung have measured much less than half of the 169 locations successfully.

Table 2: Properties of the validation set.

Device	N_M	N_L	N_{UL}	N_{UB}	$RSSI_R$
Imagineear	182	35	35	92	-64, -90
S5 Mini	296	62	58	103	-67, -100
Moto G5	724	163	107	124	-69, -100
Iphone 6S	2367	488	164	143	-65, -98

Table 3: Properties of the test set.

Device	N_M	N_L	N_{UL}	N_{UB}	$RSSI_R$
Imagineear	293	50	50	148	-62, -93
S5 Mini	884	151	148	204	-55, -99
Moto G5	1229	226	164	198	-58, -98
Iphone 6S	5372	925	310	212	-59, -98

3.1.4 Test set.

For the test set, 314 locations from the ground to the third floor have been recorded. The locations are based on a walk through the museum during which almost all rooms were visited. Again, each location has a three-second time window and measurements for each Android device are aggregated to a one-second time frame. Together, the devices recorded 217 unique beacons. Table 3 shows an overview of the test set. Table 5 displays a single row.

The additional Imagineear device recorded 90 measurements over a time span of one hour. During collection, the device hung around the neck with a cord. The data contains a timestamp and the located room estimated by the current system.

3.2 Offline phase

The localisation method is divided into the offline and the online phase. The training set is used in the offline phase to construct a fingerprint database, where locations are matched with RSSI from the beacons. To mitigate the fluctuations found during the ten-second time frames, the median of the collected RSSI measurements per location has been used, creating the format as displayed in Table 6. The RSSI values are stored in dBm, and for the normal data representation, a value of 1000 is set if no signal was received from a beacon at the measured position. The reason for setting such a big number for missing signals, is that for example a value of 100 influences the performance with some of the distance metrics that use absolute differences. For the exponential or powed data representation a value equal to the weakest

Table 4: Second row of S5 Mini training set as an example of the structure.

Time	UUID	Major	Minor	RSSI	Direction	Latitude	Longitude	Floor
2020-06-03 11:59:33.084	e7826da6 ..	1070	147	-89	190.454	52.360	4.885	0

Table 5: First row of S5 Mini test set. The numeric columns represent beacon Minors.

Time	Direction	LAT	LONG	FLOOR	258	274	...	73
2020-06-24 14:17:57.061	38.185	52.360	4.885	0	-84.0	-91.0	...	NaN

Table 6: Example of the first row from the Iphone 6S fingerprint database with a normal data representation. The numeric columns represent beacon Minors.

LAT	LONG	FLOOR	183	198	...	272
52.360	4.885	0	-85	-86	...	1000

signal minus one is set. Leaving the missing values empty would result in math errors, when comparing a measurement in the online phase which contains a beacon which was not found during the offline phase.

In short, the fingerprint database for each of the phones is created as follows:

(1) Collect RSSI for each beacon in range at location L.
(2) Beacons not in range at L get a value of 1000.
(3) Calculate the median of the RSSI for each beacon.
(4) Store location L in combination with the median RSSI of each beacon in the database.

These steps are applied to every recorded location, creating four databases containing static deterministic fingerprints. The final step (for a non-normal data representation) is converting the RSSI values in the fingerprinting databases to the applied data representation, as described in section 2.1.3.

From Table 1, it becomes clear that the amount of measurements per location differs a lot. The ratio between the number of measurements (N_M) and the number of unique assigned locations (N_{UL}) is as low as 2.44 for Imagineear, and as high as 37.01 for the Iphone 6S.

This difference is also illustrated in Figure 2, where it becomes visible that Imagineear has only very few measurements in comparison with the Iphone 6S, which reached more than hundred measurements within ten seconds in some of the cases. Zooming into one of the locations, Figure 3 visualises the beacons and corresponding RSSI values found. The Iphone 6S recognises more beacons, from which

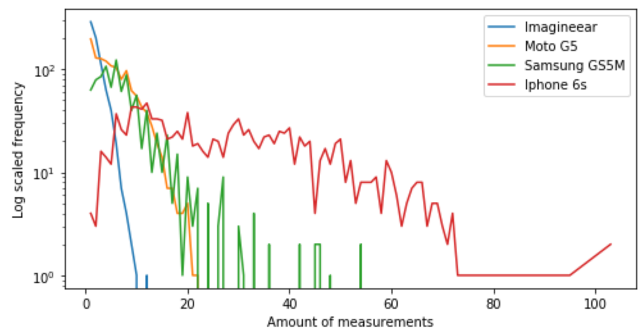

Figure 2: Visualisation of the occurrence rate of measurements per location (zero excluded) on a log scale, subdivided per phone. Imagineear has between one and ten measurements at one location, while the Iphone 6S reaches a maximum of more than a hundred measurements.

many in the weaker signal range. The Imagineear has fewer measurements, but has stronger signals. An aggregation on RSSI of the fingerprint database per device, supports this finding (Figure 4). Measurements from the Iphone 6S peak within the -95 to -90dBm range, while the Imagineear measurments peak within -80 to -75dBm. The differences could be caused by the varying sensor qualities in the phones and the limitations described in section 3.1.1. Disparities impose the difficulty of creating a single model that is equally functional for all phones.

3.3 Online phase

The test and validation sets are used in the online phase. During the online phase, unseen RSSI measurements are compared to the fingerprint database to test the performance. The location will be determined using the Weighted (centroid) kNN method, based on work from Kriz et al., and Gansemer,

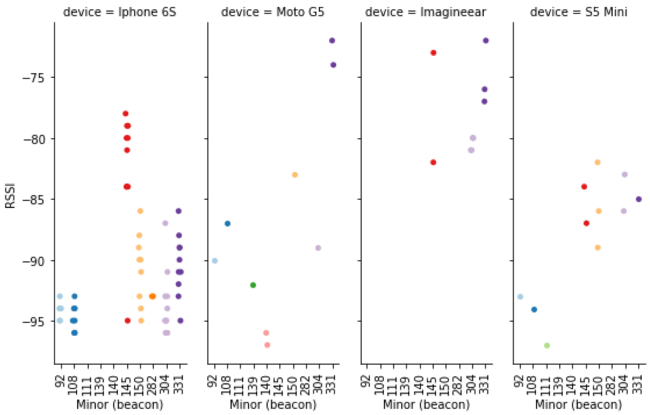

Figure 3: Visualisation of the differences in measurements between devices at one location. Each sub-diagram shows the measurement from the device mentioned up top. On the y-axis the RSSI in dBm, on the x-axis the Minor for each beacon recorded at the location. The colors across the sub-diagrams indicate the same beacons.

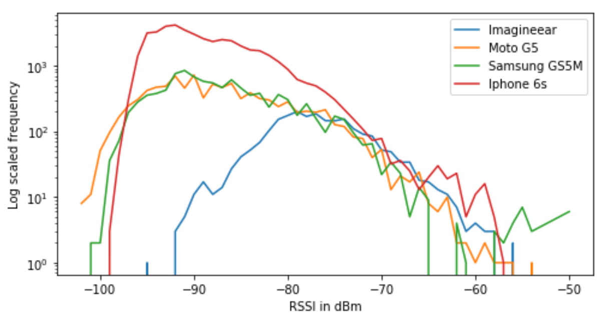

Figure 4: A line histogram of the log scaled occurrence rate of the RSSI in dBm, subdivided per phone. The Imagineear peaks at around -78dBm, while the other devices catch much weaker signals and peak at less than -90dBm.

[19, 24]. As opposed to the static fingerprint collection, the online phase is more dynamic. The usage of a median, mean or Kalman filter, to mitigate the interference can be used in combination with other statistics, such as time.

To estimate the position based on the nearest fingerprints, the distance between a new measurement m and the fingerprints $f_i = (f_1, f_2, ..., f_n)$ from the database is calculated. Measurement m exists of RSSI readings from multiple beacons that are in range at the recorded position. Only the N beacons that were in range during the measurement are used in the calculation of the distance towards the fingerprints. The other beacons are ignored. If the exponential or powed data representation is applied, the measurement is converted using equation (2) or (3), before calculating the closest fingerprints.

The Euclidean distance is one of the many possible metrics to determine the distance, and can be calculated for each ith fingerprint using the following formula:

$$Euclidean_{D_i} = \sqrt{\sum_{i=1}^{N} (m - f_i)^2} \tag{4}$$

The other distance metrics applied are Manhatten, Neyman and Sørensen and calculated using Equation (5), (6) and (7) respectively.

$$Manhatten_{D_i} = \sum_{i=1}^{N} |m - f_i| \tag{5}$$

$$Neyman_{D_i} = \sum_{i=1}^{N} (\frac{(m - f_i)^2}{m}) \tag{6}$$

$$Sorensen_{D_i} = \frac{\sum_{i=1}^{N} |m - f_i|}{\sum_{i=1}^{N} (m + f_i)} \tag{7}$$

Then, the database is sorted ascending based on the distance D_i. For each fingerprint the latitude, longitude, floor and the calculated distance are left $f_i = (x_i, y_i, z_i, D_i)$. Next, the predicted floor z_p is determined by a majority vote of each z_i from the K fingerprints with the smallest distance. If there is no majority, the furthest fingerprint is removed until there is a majority. The database is filtered again, to only contain the fingerprints with the floor z_p.

A weighted centroid approach between the K fingerprints is used to calculate the approximated position P. This way, the closer the beacon, the more influence it has on the location. A distance between a measurement and a fingerprint can be zero, but division by zero is not possible (equation (8)). To accommodate the math error, a distance of zero is replaced by a distance of one. The weight of the i nearest fingerprints (W_i), and the estimated location P are calculated as follows:

$$W_i = \frac{1}{D_i} \tag{8}$$

$$P = \frac{\sum_{i=1}^{k} (x_i, y_i) W_i}{\sum_{i=1}^{k} W_i} \tag{9}$$

In short, an approximated location in the online phase is determined as follows:

(1) Collect beacon RSSI measurements.
(2) Apply a filter to reduce fluctuations.
(3) Format and combine measurements within a predetermined time window into a single measurement.

(4) Calculate the distance between measurement and each fingerprint, using only the beacons which were in range during the measurement.
(5) Sort the database on distance ascending.
(6) Determine the floor by a majority vote of the K closest fingerprints.
(7) Filter the database to the determined floor.
(8) For the K nearest fingerprints, calculate the weight.
(9) Using the weights, estimate the position.

3.3.1 Model configurations.
As described in the literature section, many model configurations are possible. The usage of various filters and window sizes (median or mean), different distance metrics (Euclidean, Manhatten, Neyman or Sørensen), data representations (normal, exponential or powed) and the possible values for K. In addition, the configurations may also differ per device.

The baseline model in this study consists of no filter, the commonly used Euclidean distance, a normal data representation and a K of 3. In the next section is described how the best model for each of the devices will be determined.

3.4 Evaluation

The evaluation exists of three parts. A comparison between the models, a comparison with the literature, and a comparison with the existing in-app system.

3.4.1 Finding the best performing model.
The best performing model is determined based on the validation set. For each location the distance between estimated position and the real position will be calculated. The distance between two coordinates containing latitude and longitude is determined by using the geodesic distance. The amount of times a floor has been predicted successfully will be expressed as a percentage. Because the best model is a balance between a high floor percentage and a low mean distance error, a combination of the two should be used to determine the model with the least amount of error.

During the IPIN 2015 indoor localisation competition, a formula has been used where the final performance was calculated using the mean error and penalties for wrong predicted buildings and floors [40]. Because the Rijksmuseum exists of only one building, an adjusted version, without a penalty for buildings, will be used to calculate the performance.

$$Error = floor_{penalty} * floor_{error} + mean_{error} \tag{10}$$

In line with the competition, the $floor_{penalty}$ is set to four meters.

Because many configuration are possible, the best performing model is selected in a couple of steps. Each step is executed with the best performing model from the previous step.

(1) Find the best filter and window size configuration.
(2) Find the best data representation and distance metric.
(3) Find the best value of K.

These steps are applied to each of the devices, resulting into the configurations which produce the least amount of *Error*.

3.4.2 Literature.
To reduce bias and prevent overfitting, the final performance will be determined by using the test set. Based on the validation set, the three models with the lowest *Error* per device, the baseline and the best model across all the devices are selected. Then, the performance of those models on the test set is determined. Evaluation of the performance is tricky, because many evaluation measures do not encapsulate variations that occur in the real world [27]. Nevertheless, the most commonly used performance metrics will be used to evaluate the results. The mean, median, *Error*, RMSE, and the 95^{th} and 90^{th} percentile will be used to express the localisation performance. The floor estimation error will be expressed as a percentage. Then, a comparison will be made with existing literature and the error will be analysed to explain what the possible problems are for future research.

3.4.3 Existing in-app system.
As explained in the introduction, the current system is built within the Rijksmuseum app. In order to be able to compare the existing system and the proposed system, the data of the current system and the proposed system had to be collected at the same time, as described in section 3.1.4. Because the existing system classifies into a room, and the proposed system estimates long- and latitude positions, the latter has to be converted into a room. This will be done manually based on the scaled map of the Rijksmuseum. The estimated floor and location performance of both the current and proposed system will be analysed and presented in a table.

4 RESULTS

The results are divided into four sections. The first section describes the approach and results of the influence of humans on the RSSI. Then, the performance from different configurations on the validation set will be explained. The third section explores the performance on the test set, the source of the error and whether a general fingerprint database may be suitable. Finally, the last section evaluates the performance between the current and the proposed system.

Table 7: RSSI data for each beacon from the Night Watch room. The *Beacon Minor* (B_M) is a unique identifier, n is the amount of measurements and the RSSI of the starred B_M are significantly influenced by humans. On average there were 22.7 people present. The time span of the data is 186 seconds.

	Humans (μ = 22.7)		No humans	
B_M	n	μ (σ)	n	μ (σ)
12	184	-86.05 (2.82)	186	-86.30 (1.49)
206	29	-93.79 (2.21)	7	-94.14 (3.18)
229*	83	-93.42 (2.16)	67	-94.72 (0.87)
235*	121	-93.40 (1.23)	38	-94.45 (2.52)
236*	184	-84.53 (2.55)	186	-85.03 (0.82)
241*	175	-88.99 (2.25)	53	-94.70 (2.27)
244*	182	-86.95 (2.26)	183	-90.34 (1.99)
249*	179	-87.75 (2.95)	185	-86.94 (3.00)
All	1138	-88.19 (3.86)	906	-88.50 (3.92)

Table 8: RSSI beacon data Naval Power room. On average 10.3 people were present in the room. The time span of the data is 224 seconds.

	Humans μ = 10.3		No humans	
B_M	n	μ (σ)	n	μ (σ)
77*	223	-82.45 (1.21)	222	-88.01 (1.43)
91	13	-93.92 (0.95)	165	-94.69 (0.91)
218*	193	-90.31 (1.64)	221	-90.73 (1.21)
234*	221	-92.13 (1.18)	224	-91.45 (1.04)
306*	225	-83.19 (1.19)	225	-87.76 (1.50)
All*	875	-86.99 (4.52)	1057	-90.30 (2.71)

4.1 Human influences

As stated in the introduction, human bodies influence the RSSI through reflection and attenuation, which could result in a degraded localisation performance. This leads to the question: To what extent does the presence of humans influence the RSSI in the Rijksmuseum? To answer the question, data has been collected in two situations, for around three minutes, at a specific location. One situation with, and one situation without human presence. For collection of the data an Iphone 6S has been used, because it collects signals consistently per second. The first location was in the most visited Night Watch room (Appendix Figure A10). The second, was in a well attended adjacent room, called Naval Power (Figure A11). Respectively Table 7 and Table 8 give an overview of the data, showing the amount of beacons in range, the amount of measurements, the average RSSI and the standard deviation for both situations. The RSSI averages of the situation with and without humans will be compared to see whether or not there is a significant difference.

The results of Levene's test with $\alpha = 0.05$ (Table A14 and A15), show that only some of the beacons have equal variances. Therefore, Welch's t-test will be used, because unlike the Student's t-test, it does not have the assumption of equal variances. As stated in section 2.2.1, RSSI is not Gaussian distributed. Because the normality assumption becomes less important with a higher amount of samples, beacons with less than thirty values are not taken into account.

The results of Welch's t-test are presented in Table A16 and A17 from the Appendix. Table A16, shows that six of the eight collected beacons in the Night Watch room have a $p < 0.05$. For these beacons the RSSI averages are not equal between the situation when people are present and not present. From the other two beacons, the beacon with Minor 12 was not significant ($p > 0.05$), while beacon with Minor 206 did not have enough data to perform the test ($n < 30$). When the RSSI measurements of all beacons are combined, there is no significant difference ($0.074 > 0.050$).

Table 17, shows that four of the five collected beacons in the Naval Power room have a $p < 0.05$, for which the the RSSI averages between the two situations are not equal. The fifth beacon did not have enough data to perform the test ($n < 30$). When the RSSI measurements of all beacons are combined, there is a significant difference.

4.2 Performance on the validation set

As described in section 3.4, the best model for each of the phones will be determined in three steps. The first step was determining the filter and window sizes for each of the devices. Table 9 shows an overview of the first step. The mentioned *Error* in the table refers to equation (10), explained in section 3.4.1. The performance of the devices differ. The Imagineear does not seem to benefit from the use of a filter. A mean with a window size of two could be used, but bigger windows sizes increases the *Error*. The other devices do benefit from the use of a filter. Although the differences are very small, S5 Mini functions best with a median filter and a window size of three, Moto G5 with a mean filter and a window size of five, and Iphone 6S with a mean filter and window size of three. A mean filter and window size of two performs well for all the devices on average. Excluding the Imagineear, a median or mean filter with a window size of three performs best.

Table 10 shows an overview of step two. The four distance metrics and three data representations are combined with the filter that performed best in step one. The Imagineear has the lowest *Error* in combination with a normal data representation and either Euclidean or Manhatten distance. Combining Neyman distance and the exponential data representation results in the lowest mean. S5 Mini, Moto G5 and Iphone 6S perform uniformly with Sørensen distance and normal data representation as the best combination. Neyman is the

Table 9: Effect of filters and window sizes on the performance on the validation set, for each of the devices. All rows except *Floor* are displayed in meters. The best performance per row is bold.

	Filter	None	Median					Mean				
Device	Window size	-	2	3	4	5	6	2	3	4	5	6
Imagineear	3rd Q.	5.89	**5.77**	6.42	7.56	8.38	8.41	**5.77**	6.30	7.50	8.38	8.41
	Mean	**4.32**	4.52	5.21	5.57	6.59	7.16	4.52	5.19	5.61	6.65	7.18
	Floor (%)	100	100	100	100	100	100	100	100	100	100	100
	RMSE	**4.71**	5.58	6.20	6.43	7.51	8.63	5.58	6.09	6.44	7.53	8.62
	Error	**4.32**	4.52	5.21	5.57	6.59	7.16	4.52	5.19	5.61	6.65	7.18
S5 Mini	3rd Q.	8.11	**7.07**	8.20	8.53	8.47	8.52	**7.07**	8.21	8.70	8.27	8.27
	Mean	6.55	5.71	**5.50**	6.10	6.16	6.29	5.71	5.55	6.15	6.10	6.29
	Floor (%)	98.39	98.39	98.39	98.39	98.39	98.39	98.39	98.39	98.39	98.39	98.39
	RMSE	7.72	6.79	**6.38**	6.89	6.99	7.13	6.79	6.41	6.94	6.91	7.14
	Error	6.62	5.78	**5.56**	6.16	6.22	6.35	5.78	5.61	6.22	6.16	6.35
Moto G5	3rd Q.	10.24	8.17	7.46	7.36	7.62	7.51	8.17	7.41	**7.34**	7.49	7.50
	Mean	7.75	5.95	5.97	6.06	5.92	5.98	5.95	5.90	6.06	5.90	**5.87**
	Floor (%)	88.34	92.02	92.64	92.02	**93.25**	92.02	92.02	92.02	92.02	**93.25**	92.02
	RMSE	9.39	7.37	7.57	7.50	7.25	7.27	7.37	7.50	7.57	7.32	**7.15**
	Error	8.21	6.27	6.27	6.38	6.19	6.30	6.27	6.21	6.38	**6.17**	6.19
Iphone 6S	3rd Q.	5.90	5.70	5.67	5.70	5.92	5.96	5.70	**5.48**	5.71	5.81	5.90
	Mean	4.29	4.15	4.10	4.18	4.37	4.56	4.15	**4.06**	4.25	4.44	4.61
	Floor (%)	95.70	96.11	96.31	96.72	**96.93**	96.72	96.11	96.72	96.72	**96.93**	96.72
	RMSE	5.23	5.04	4.98	5.04	5.25	5.79	5.04	**4.90**	5.10	5.59	5.82
	Error	4.47	4.31	4.25	4.31	4.49	4.69	4.31	**4.19**	4.38	4.56	4.74
All devices	Avg. Error	5.91	**5.22**	5.32	5.61	5.87	6.13	**5.22**	5.3	5.65	5.89	6.12
Except Imagineear	Avg. Error	6.43	5.45	5.36	5.62	5.63	5.78	5.45	**5.34**	5.66	5.63	5.76

second best distance metric for the S5 Mini in combination with an exponential data representation, while Manhatten normal is second best for Iphone and Moto.

Figure A14 visualises the influence of K on the baseline and three best performing models (from step two) per phone. With the Imagineear, Euclidean (which is equal to the baseline), Manhatten and Sørensen normal follow a similar pattern. Manhatten normal outperforms the baseline with $K = 9$ and $K = 11$. Sørensen normal performs worse than the baseline. Neyman exponential quickly decreases in performance as K increases. All models have high fluctuations in the *Error* depending on the value of K. The peaks at for example $K = 13$ and $K = 21$, have in common that the correctly estimated floor percentage is off by a lot. A wrong estimated floor is both punished in calculation of the *Error* and increases the absolute error, due to only using fingerprints of the estimated floor. In combination with a low amount of validation points and absolute errors bigger than twenty meters, the mean performance is heavily influenced. A no filter, Manhatten normal with $K = 9$ model reaches the lowest *Error* of 3.82 meters.

With the S5 Mini, Sørensen normal, Neyman exponential and Sørensen exponential outperform the baseline. At $K = 5$, Sørensen exponential reaches the lowest *Error* of 4.96m. Sørensen normal results in the lowest *Error* for both the Moto G5 and Iphone 6S. With $K = 11$ and $K = 5$ respectively, scoring 5.15m and 3.86m. Overall, a K between three and nine, performs well across all devices.

Figure A15 visualises the cumulative distribution of the absolute error for the best performing models and K combinations together with the baseline, subdivided per device.

Figure 5, shows a comparison between the best performing models for each device. The models perform similar on the first 20%, but diverge as the percentage increases. 80% of the locations from the validation set have a maximal absolute error of around 7 meters, with the Iphone 6S having the least absolute error around 5.5 meters. From 80% to 100% the error increases rapidly, indicating that a number of locations are predicted far from the actual location.

4.3 Performance on the test set

For the S5 Mini, Moto and Iphone a general model has been defined. Imagineear is excluded because a filter affects the results for this device. Based on Table 9 and 10 the general model will exists of a mean filter with a window size of three, a normal data representation and Sørensen distance. Results

Table 10: Results of the different distance metrics and data representation on the validation set, for each of the phones. All columns except *Floor* are displayed in meters. The best results for each phone are bold.

Device	Filter (window size)	Distance metric	Data representation	3rd Q.	Mean	Floor (%)	RMSE	Error
Imagineear	No filter (-)	**Euclidean**	**Normal**	**5.89**	4.32	**100**	**4.71**	**4.32**
			Exponential	6.60	4.87	97.14	6.52	4.99
			Powed	6.65	4.27	97.14	10.41	4.38
		Manhatten	**Normal**	**5.89**	4.32	**100**	4.72	**4.32**
			Exponential	6.56	4.88	97.14	6.53	5.00
			Powed	7.00	6.85	97.14	10.42	6.97
		Neyman	Normal	6.87	4.62	97.14	5.21	4.74
			Exponential	6.37	**4.26**	97.14	5.05	4.37
			Powed	7.20	5.96	97.14	8.20	6.07
		Sørensen	Normal	6.65	4.27	97.14	4.83	4.38
			Exponential	6.59	4.61	97.14	5.76	4.72
			Powed	6.53	4.77	97.14	6.05	4.89
S5 Mini	Median (3)	Euclidean	Normal	8.20	5.50	98.39	6.38	5.56
			Exponential	7.39	5.55	**100**	6.53	5.55
			Powed	8.16	6.06	98.39	7.33	6.16
		Manhatten	Normal	7.34	5.59	98.39	6.35	5.65
			Exponential	7.98	5.77	**100**	6.71	5.77
			Powed	7.21	5.76	96.77	7.02	5.89
		Neyman	Normal	7.68	5.54	**100**	6.36	5.54
			Exponential	7.43	5.44	**100**	6.44	5.44
			Powed	8.14	6.31	95.16	7.38	6.51
		Sørensen	**Normal**	**6.65**	**5.38**	**100**	**6.14**	**5.38**
			Exponential	7.52	5.42	**100**	6.38	5.42
			Powed	8.02	5.85	**100**	6.93	5.85
Moto G5	Mean (5)	Euclidean	Normal	7.49	5.90	93.25	7.32	6.17
			Exponential	7.48	6.12	92.02	7.67	6.44
			Powed	8.32	6.79	88.96	8.42	7.23
		Manhatten	Normal	**7.07**	5.71	92.64	7.15	6.00
			Exponential	7.20	5.89	90.80	7.15	6.26
			Powed	8.23	6.93	88.96	8.77	7.37
		Neyman	Normal	7.65	5.82	**95.09**	7.13	6.02
			Exponential	7.68	6.30	92.64	7.84	6.60
			Powed	8.25	6.72	88.96	8.52	7.16
		Sørensen	**Normal**	7.20	**5.54**	92.02	**6.70**	**5.86**
			Exponential	7.42	6.05	92.64	7.41	6.34
			Powed	7.15	6.04	93.64	7.76	6.00
Iphone 6S	Mean (3)	Euclidean	Normal	5.48	4.06	96.72	5.10	4.19
			Exponential	5.47	4.12	96.72	4.98	4.25
			Powed	6.49	4.95	94.88	6.18	5.16
		Manhatten	Normal	5.24	4.00	97.54	4.79	4.10
			Exponential	5.29	4.11	96.93	4.91	4.23
			Powed	6.40	4.84	94.88	5.93	5.05
		Neyman	Normal	5.27	4.00	97.54	4.88	4.10
			Exponential	5.43	4.11	96.93	4.97	4.24
			Powed	6.76	5.15	93.85	6.38	5.39
		Sørensen	**Normal**	5.32	**3.93**	**98.36**	**4.76**	**4.00**
			Exponential	**5.22**	3.99	97.13	4.79	4.10
			Powed	6.10	4.48	95.49	5.49	4.66

Table 11: Best models on test set from all devices. *Floor* is a percentage, the other metrics are in meters.

Device	Model	Mean (std)	Median	3rd Q.	90^{th}	95^{th}	RMSE	Error	Floor
Imagineear	Sørensen, normal, no filter, $K = 7$	4.89 (3.49)	4.67	6.24	8.40	9.69	6.00	5.13	94.00
S5 Mini	Sørensen, exponential, median (3), $K = 7$	5.67 (4.13)	5.00	6.89	9.11	12.39	7.02	5.91	94.04
Moto G5	Sørensen, normal, mean (3), $K = 9$	5.29 (3.66)	4.79	6.87	8.47	11.78	6.43	5.73	88.94
Iphone 6S	Sørensen, exponential, mean (3), $K = 5$	4.56 (4.50)	3.55	5.54	8.20	10.33	6.41	4.70	96.43

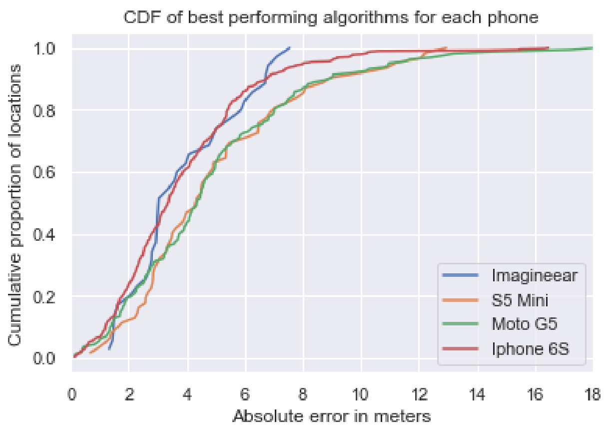

Figure 5: Cumulative distribution of the absolute error from the best performing models on the validation set, for each of the devices.

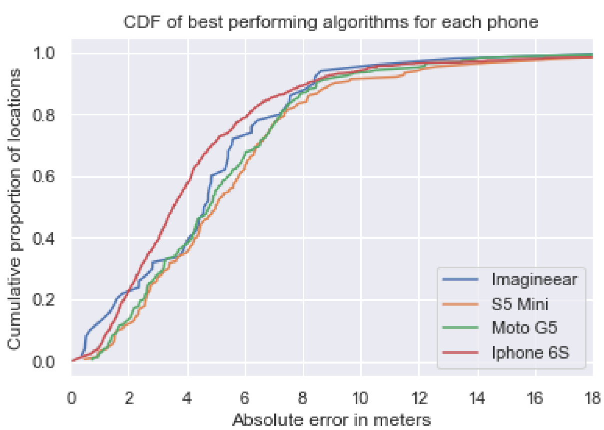

Figure 6: Cumulative distribution of the absolute error from the best performing models on the test set, for each of the devices.

on the validation set imply that the lowest *Error* is reached with a K between three and nine. Therefore, these values of K have been used for analysis on the test set. Figure A16 visualises the performance of the different models and K values per phone, including the baseline and the general model.

The selected models based on the validation set of the Imagineaar performed similar on the test set. A different data representation and distance metric does not perform much better than the baseline. On the test set a model with no filter, Sørensen normal, and a $K = 9$ produces the lowest *Error* of 5.13 meters

Both on test and validation, a median filter with a window size of three and Sørensen exponential performs best using the Samsung S5 Mini. With $K = 7$ this model produces an *Error* of 5.91 meters. Sørensen normal and the general model have overlapping lines, indicating that the usage of a median or mean filter does not make much of a difference in this case.

Although Sørensen normal performed best on the validation set of the Moto G5, the general model, which only differs by using a smaller filter size of three, performs best on the test set. With $K = 9$ the lowest *Error* is 5.73 meters.

On the validation set, the Iphone 6S best performing models have a mean filter with a window size of three, Sørensen distance and either a normal or exponential data representation. Where the usage of a normal data representation lead to the lowest *Error* on the validation set, the exponential data representation performs better on the test set. With $K = 5$ the lowest *Error* is 4.70 meters.

Figure A17 visualises the cumulative distribution of the absolute error for the best performing models and K combinations together with the baseline, subdivided per device.

Overall, comparing the results between the test and validation set, it becomes clear that the performance on test is about half a meter worse than on the validation set. The ranking of the models for each of the phones is comparable between both sets. Table 11 contains the performance metrics of the best model on the test set per device. The high standard deviation indicates a lot of uncertainty in the localisation. In Figure 6, the absolute error is plotted against the cumulative proportion of locations, to provide insight into the cause. For all devices 80% of the locations from the test set have a maximal absolute error of around 7.5 meters, with the Iphone 6S having the least absolute error of around 6 meters. Just like the validation set, from 80% to 100% the error increases rapidly, meaning that a number of locations

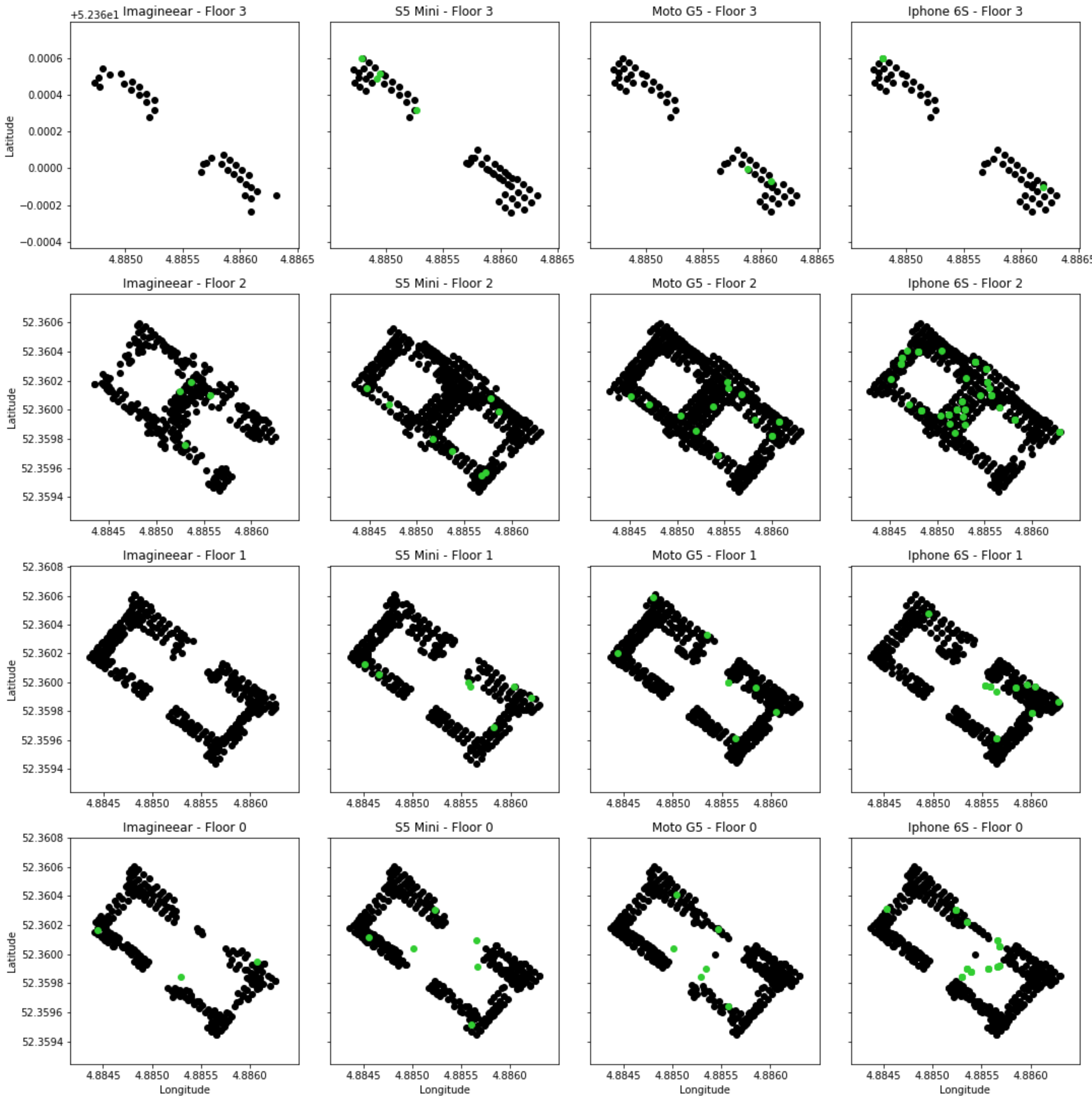

Figure 7: For each of the phones, the real position of wrong predicted locations with an absolute error bigger than 8 meters (green dots). The black dots represent the fingerprints from the database. Each column is a device, each row a floor.

are predicted far from the actual location, influencing the average performance.

4.3.1 Error analysis.

Results from both the validation and test set suggest that the absolute error increases rapidly in the top 20% worst predicted locations. Figure 7, visualizes the actual position of incorrectly predicted locations with an absolute error greater than 8 meters. At ground floor, all devices have trouble with locations with no fingerprints nearby (green dots without

surrounding black dots). Recall that the amount of green dots are related to the amount of test locations (see Table 3). While the test set contains locations in the atrium, this part of the Rijksmuseum was not recorded and therefore not present in the fingerprint database, causing big errors in the predictions. Some predictions of the Iphone 6S had an absolute error greater than thirty meters, with one even reaching forty meters. The lack of fingerprints also causes trouble at the first floor. All but Imagineear have green dots on east inside, which are stairs in the atrium leading to rooms of the first floor. These stairs are also not present in the fingerprint databases.
Furthermore, it is noticeable that, using the Iphone 6S, many locations in the middle of the second floor have an error of more than 8 meters. There may not be enough beacons active in this area, or the batteries of some of the beacons placed there may be empty. For example, during the influence of humans on RSSI experiment, described in section 4.1, some of the beacons that could be expected to be measured are not measured (see beacons with Minor 243 and 250 in Figure 10).

Figure A18 visualises the test set locations where the floor was incorrectly predicted. Again, positions in the atrium and at the stairs are among the errors due to the lack of fingerprints. The Moto G5 has a lot of errors in the north and south side of the building. This may be caused by an imbalance between fingerprints on the ground and first floor in these areas. Because more fingerprints are on the first floor than on the ground floor, this could influence the majority vote.

Figure A19 shows the precision-recall curves per floor from the best models on the test set. The Imagineear and S5 Mini have most problems with predicting the first floor (low precision), but do not predict the first floor when one of the other floors is correct (high recall). The Moto G5 and Iphone 6S have this problem with respectively the third and ground floor.

An analysis in why floors are predicted wrong while using the models with the smallest *Error*, implies that some wrong predictions may be prevented by using only the closest or a majority vote of three fingerprints to estimate the floor. To check if using smaller values of K for predicting the floor is beneficial, the best models on the test set have been used, applying a different K value for floor prediction and location prediction. Table A18 shows the results.

The localisation performance is influenced as well, because the fingerprint database is filtered to only contain the predicted floor. The Imagineear and the Moto G5 benefit from using a different K to predict the floor $K_F = 3$. For both devices the *Error* decreases, at the cost that the mean absolute error increases a little. The other two devices perform

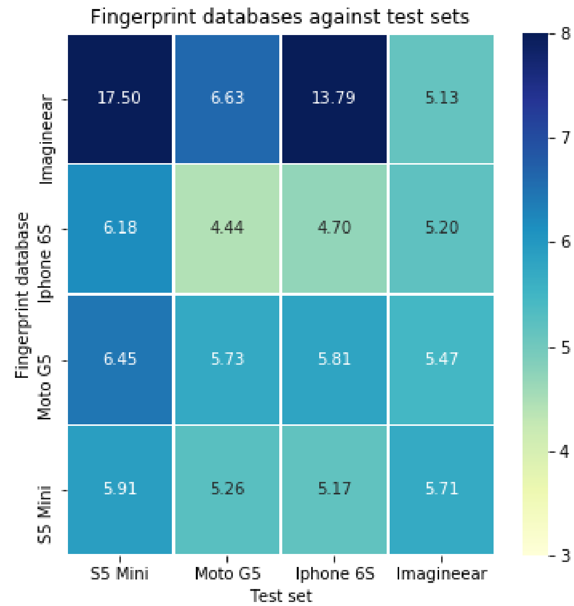

Figure 8: Performance of test sets on fingerprint databases of other devices using the best performing models. The numbers represent the *Error*.

worse on their best models.

4.3.2 Performance on other fingerprint databases.
Creating a fingerprint database for every phone is time consuming. So, to what extent is a fingerprint database from one device, suitable for other devices? Figure 8 shows the *Error* results when using the best models (see Table 11) against the fingerprint database of each device. From the figure, the following observations can be made:

- The high *Error* indicates that the Imagineear fingerprint database is not suitable for the other devices.
- The Moto test set performs better on the S5 Mini fingerprint database, and reaches the lowest *Error* of all models on the Iphone 6S fingerprint database of 4.44 meters.
- All devices but Moto, perform best on their own fingerprint database.
- The Iphone 6S fingerprint database could be a good candidate as a general database for the used devices.

4.3.3 Final performance.
After experimentation with the separation of K in section 4.3.1 and the possibility of using a single fingerprint database in section 4.3.2, better scores were achieved for Moto and

Table 12: The final scores of the performance metrics from the best models found on the test set. The Imagineear has a separate K for floor and location prediction, Moto G5 performs best on the Iphone 6S fingerprint database (I6S fpdb).

Device / Model	Mean (std)	Median	3rd Q.	90^{th}	95^{th}	RMSE	Error	Floor
Imagineear Sørensen, normal, no filter, $K = 7$, $K_F = 3$	4.93 (3.45)	4.67	6.24	8.40	9.69	6.02	5.01	98.00
S5 Mini Sørensen, exponential, median (3), $K = 7$	5.67 (4.13)	5.00	6.89	9.11	12.39	7.02	5.91	94.04
Moto G5 Sørensen, normal, mean (3), $K = 9$, I6S fpdb	4.30 (3.61)	3.79	5.29	7.02	9.01	5.62	4.44	96.46
Iphone 6S Sørensen, exponential, mean (3), $K = 5$	4.56 (4.50)	3.55	5.54	8.20	10.33	6.41	4.70	96.43

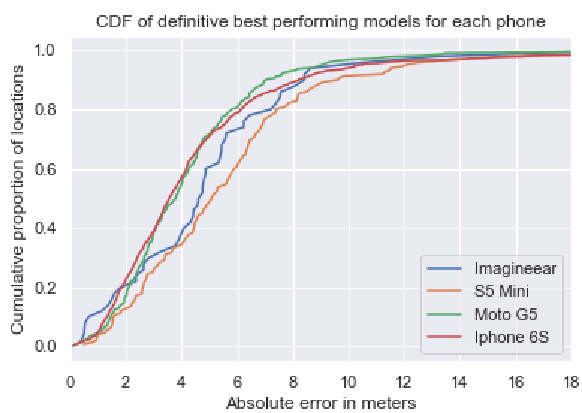

Figure 9: Cumulative distribution of the absolute error of the final best models found on the test set, for each of the devices.

Table 13: Correct and incorrect predicted locations and floors when using the current and the proposed system.

	Location prediction		Floor prediction	
Models	Correct	Incorrect	Correct	Incorrect
Current	23	12	35	0
Proposed	24	11	34	1

Imagineear on the test set. Table 12 shows the definitive performance of the best model found for each of the devices. Figure 9 visualises the cumulative distribution of the absolute error of the finals models from each of the devices.

4.4 Current and proposed system

Of the 50 locations in the test set, 35 were measured by both the current and proposed system. These locations have been compared and the results are shown in Table 13. When a comparison between the old system and the actual location was not unambiguous, it was seen as an incorrect prediction. When the proposed system predicted the location correct, but the floor wrong, this is also counted as an incorrect location prediction.

On the Imagineear both models perform similar. The current and proposed model predicted the room respectively 65.7% and 68.6% of the time correctly. The proposed model made one mistake in predicting the floor. Because of the small difference in performance, it cannot be said that the proposed model performs significantly better on room level than the current model.

5 DISCUSSION

The aim of this study was to examine to what extent a WkNN Bluetooth localisation approach can be applied in a real environment, what parameters are involved, and if it is workable with multiple devices. The Rijksmuseum is an old multi-floor building, where many people are present and restrictions apply on the positioning of the beacons. Therefore making it a challenging location. A comparison with their current system gives an indication of whether this approach can be seen as an improvement.

The introduction describes that people influence Bluetooth signals. The results from section 4.1 confirm that this also applies to the Rijksmuseum. This means that the performance of the model may be affected if the environment in which it is used is different from the environment used to create the fingerprint database. The strength of the influence on the absolute error will have to be examined in future research.

From the literature research two types of filters stood out, deterministic and probabilistic. Because of the scope of the paper, only deterministic filters have been tested. According to the results the mean and median filter both work well. The window size could be influenced by the fact that three seconds of measurements were collected at locations in the test set. Overall, the more measurements, the more accurate the RSSI for each beacon can be determined, the better the localisation performance. Torres-Sospedra et al., found in their experiments that without filtering, the exponential and powed data representation performed best, increasing the *success rate* by 2% and reducing the *positioning error* by one meter in comparison with a positive data representation [39]. Further, they found that Sørensen reached the best scores with a positive or powed representation, while Neyman X^2 worked best in combination with an exponential data representation. In this research, the results on the validation set show that in most of the times the normal data representation reached the lowest mean absolute error and highest floor success rate, followed by exponential and powed (Table 10). On validation, Sørensen in combination with a normal data representation performs best. On test, the Imagineear and Moto work best with Sørensen normal, while S5 Mini and Iphone 6S work best with Sørensen exponential.

Torres-Sospedra et al., attributes the success of Sørensen to the normalisation used in the calculation. It provides a distance value between zero and one, which can be considered as a degree or percentage of inequality that does not depend on absolute RSSI values [39].

Other insights and contributions can be summarised as follows. First, if filters and windows sizes are tuned according to the measurement frequency of the device, it is possible to use one general model for all devices at the cost of small *Error* differences. Unlike the others, the Imagineear did not benefit from using a filter, because successive measurements were too unrelated.

Second, although all tested devices, except Moto, perform best on their own fingerprint database, it is possible to use one fingerprint database for all devices.

Third, if no fingerprints are nearby the locations to be estimated, the method results in high absolute errors. The fingerprinting method has poor extrapolation of unexplored areas, emphasising the need of good data for creation of the database.

Fourth, using a different K for estimating the floor and location could decrease the *Error* at the cost of a small increase in the mean absolute error. Without using additional sensors, future research could investigate if a separate model for predicting the floor and location could improve both estimations. Additional sensors, such as the barometer, could be added to the model and potentially improve the floor estimation performance [6].

Fifth, in comparison with the literature overviews presented by Canton, Dawes et al., and De Blasio et al., the proposed model performs around two to four times worse based on the 90^{th} percentile [10, 12, 13]. However, as mentioned in section 2.3 and 3.4.2, a comparison between the models is tricky. There is no best solution applicable on every device in every situation. For example, the amount of environmental influences like the density of the beacons, the positioning of the beacons, the structure of the building, the presence of people, the used devices and the amount of measurements within a given time frame, each having their impact on the performance. It is unfair to compare models based on a beacon density of one every $66.2m^2$ to for example one every $20m^2$. In addition, the studies mentioned by Canton with beacon densities higher than one per $40m^2$ are influenced by interpretation. If for example only a corridor is tested, a much bigger area is counted as part of the experiment, skewing the results based on the surface.

As described by Lymberopoulus et al., future research should look in standardisation of the performance [27]. Although comparisons will probably always be difficult, a first step could be the usage of public data sets, on which different researchers can test their algorithms. A second step, could be a framework which can be used to compare systems more easily. This framework should support the reporting of the different model and environmental parameters.

Sixth, the proposed system does not perform significantly better than the current system if locations are converted to rooms. However, in most cases the proposed model can be more accurate, because it predicts the location within a room. Whether the system actually works better depends not only on performance, but also on the user experience, real-time performance and efficiency. These factors will have to be examined in future research.

Finally, the extensive data set is made publicly available, such that other researchers can apply different algorithms and approaches to improve the achieved scores.

5.1 Limitations

There are some limitations in this research. Firstly, the real position of the data during collection of the training, validation and test set is based on a rough guess. This means that a small part of the error found, is caused by the less accurate positioning precision. Secondly, during data collection the devices were placed stacked on each other and held in a belly bag. Both the positioning and the influence of other devices do not reflect a real scenario and may influence the performance. Thirdly, only deterministic filters have been tested, while a range of probabilistic filters are available. Fourthly, to test and validate properly, three seconds of data had to be recorded per location. This is a trade off between getting at least some signals at every recorded location, and being able to predict the location with only a few measurements, as would be the case in a real scenario. Fifthly, a comparison between the current and proposed system for the Rijksmuseum is not straightforward. Connecting the markers locations and timestamp to the timestamp and locations from the current system left space for interpretation, making the comparison less meaningful. Moreover, the small amount of locations reduces the generalisability, because some parts of the museum that are not included in this comparison, may influence the performance of the systems differently.

6 CONCLUSION

The goal of this research was to find to what extent the WkNN algorithm could improve the localisation of people in multi-floor buildings, where beacons are sub-optimally positioned. Based on the findings it can be concluded that the WkNN can localise people with more precision, but does not a significant better job in comparison with the existing system of the Rijksmuseum. In comparison with the literature it can be seen that it is less performing, but because of the difficult comparison of environmental variables it is hard to conclude whether it works better or worse.

As found in previous literature, people have a significant effect on the RSSI and could influence the positioning error. Based on the literature research, it emerged that there are many ways, either probabilistic or deterministic, to filter RSSI values. The effect of filtering depends on the amount of measurements available.

Among all devices, Sørensen distance outperformed Euclidean, Manhatten and Neyman with the usage of either a normal or exponential data representation. From the results can be concluded that it is possible to create one fingerprint database and one model for multiple devices at the cost of a small increase in the *Error*. By analysing the *Error*, the study emphasised the importance of a good data set for creating the fingerprint database. Future research could focus on separating the floor and location prediction, as this could potentially increase performance.

Bluetooth localisation does not have a one size fit all solution. Each building, beacon network and device has its differences. Future research should look into standardisation of reporting the performance, to get better insights in the best performing models in corresponding situations.

REFERENCES

[1] [n.d.]. Visie en missie van het Rijksmuseum - Organisatie. https://www.rijksmuseum.nl/nl/organisatie/visie-en-missie

[2] Accton. [n.d.]. *BLE Channels*. https://www.accton.com/Technology-Brief/ble-beacons-and-location-based-services/

[3] N Azmi, LM Kamarudin, Massudi Mahmuddin, Azizi Zakaria, AYM Shakaff, S Khatun, K Kamarudin, and MN Morshed. 2014. Interference issues and mitigation method in WSN 2.4 GHz ISM band: A survey. In *2014 2nd International Conference on Electronic Design (ICED)*. IEEE, 403–408.

[4] Paramvir Bahl and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, Vol. 2. Ieee, 775–784.

[5] Udana Bandara, Mikio Hasegawa, Masugi Inoue, Hiroyuki Morikawa, and Tomonori Aoyama. 2004. Design and implementation of a bluetooth signal strength based location sensing system. In *Proceedings. 2004 IEEE Radio and Wireless Conference (IEEE Cat. No. 04TH8746)*. IEEE, 319–322.

[6] Dipyaman Banerjee, Sheetal K Agarwal, and Parikshit Sharma. 2015. Improving floor localization accuracy in 3D spaces using barometer. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*. 171–178.

[7] SIG Bluetooth. 2010. Specification of the Bluetooth System-Covered Core Package version: 4.0.

[8] Apidet Booranawong, Nattha Jindapetch, and Hiroshi Saito. 2018. A system for detection and tracking of human movements using RSSI signals. *IEEE sensors journal* 18, 6 (2018), 2531–2544.

[9] Apidet Booranawong, Kiattisak Sengchuai, and Nattha Jindapetch. 2019. Implementation and test of an RSSI-based indoor target localization system: Human movement effects on the accuracy. *Measurement* 133 (2019), 370–382.

[10] Vicente Cantón Paterna, Anna Calveras Auge, Josep Paradells Aspas, and Maria Alejandra Perez Bullones. 2017. A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering. *Sensors* 17, 12 (2017), 2927.

[11] David Contreras, Mario Castro, and David Sánchez de la Torre. 2017. Performance evaluation of bluetooth low energy in indoor positioning systems. *Transactions on Emerging Telecommunications Technologies* 28, 1 (2017), e2864.

[12] Brett Dawes and Kwan-Wu Chin. 2011. A comparison of deterministic and probabilistic methods for indoor localization. *Journal of Systems and Software* 84, 3 (2011), 442–451.

[13] Gabriel De Blasio, Alexis Quesada-Arencibia, Carmelo R García, Jezabel Miriam Molina-Gil, and Cándido Caballero-Gil. 2017. Study on an indoor positioning system for harsh environments based on Wi-Fi and bluetooth low energy. *Sensors* 17, 6 (2017), 1299.

[14] Gabriel Deak, Kevin Curran, and Joan Condell. 2010. Filters for RSSI-based measurements in a Device-free Passive Localisation Scenario. *Image Processing & Communications* 15, 1 (2010), 23–34.

[15] Gabriel Deak, Kevin Curran, and Joan Condell. 2010. Wireless sensor networks-smoothing algorithms for RSSI-based device-free passive localisation. In *The Tenth International Conference on Information Technology and Telecommunications (IT&T 2010)*. 99–107.

[16] Ramsey Faragher and Robert Harle. 2014. An analysis of the accuracy of bluetooth low energy for indoor positioning applications. In *Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014)*, Vol. 812. 201–210.

[17] Ramsey Faragher and Robert Harle. 2015. Location fingerprinting with bluetooth low energy beacons. *IEEE journal on Selected Areas in Communications* 33, 11 (2015), 2418–2428.

[18] Toni Fetzer, Frank Ebner, Markus Bullmann, Frank Deinzer, and Marcin Grzegorzek. 2018. Smartphone-Based Indoor Localization within a 13th Century Historic Building. *Sensors* 18, 12 (2018), 4095.

[19] Sebastian Gansemer, Uwe Großmann, and Syuzanna Hakobyan. 2010. Rssi-based euclidean distance algorithm for indoor positioning adapted for the use in dynamically changing wlan environments and multi-level buildings. In *2010 International Conference on Indoor Positioning and Indoor Navigation*. IEEE, 1–6.

[20] Q42 info@q42.nl. [n.d.]. Q42: Technisch-strategisch internetbureau. https://www.q42.nl/

[21] Łukasz Januszkiewicz. 2018. Analysis of Human Body Shadowing Effect on Wireless Sensor Networks Operating in the 2.4 GHz Band. *Sensors* 18, 10 (2018), 3412.

[22] Laith Khalil and Peter Jung. 2015. Scaled unscented Kalman filter for RSSI-based indoor positioning and tracking. In *2015 9th International Conference on Next Generation Mobile Applications, Services and Technologies*. IEEE, 132–137.

[23] Tarmo Koppel, Andrei Shishkin, Heldur Haldre, Nikolajs Toropovs, Inese Vilcane, and Piia Tint. 2017. Reflection and transmission properties of common construction materials at 2.4 GHz frequency. *Energy Procedia* 113 (2017), 158–165.

[24] Pavel Kriz, Filip Maly, and Tomas Kozel. 2016. Improving indoor localization using bluetooth low energy beacons. *Mobile Information Systems* 2016 (2016).

[25] Changgeng Li, Zhengyang Qiu, and Changtong Liu. 2017. An improved weighted k-nearest neighbor algorithm for indoor positioning. *Wireless Personal Communications* 96, 2 (2017), 2239–2251.

[26] Guoquan Li, Enxu Geng, Zhouyang Ye, Yongjun Xu, Jinzhao Lin, and Yu Pang. 2018. Indoor positioning algorithm based on the improved RSSI distance model. *Sensors* 18, 9 (2018), 2820.

[27] Dimitrios Lymberopoulos, Jie Liu, Xue Yang, Romit Roy Choudhury, Vlado Handziski, and Souvik Sen. 2015. A realistic evaluation and comparison of indoor location technologies: Experiences and lessons learned. In *Proceedings of the 14th international conference on information processing in sensor networks*. 178–189.

[28] Juraj Machaj and Peter Brida. 2011. Performance comparison of similarity measurements for database correlation localization method. In *Asian Conference on Intelligent Information and Database Systems*. Springer, 452–461.

[29] David Madigan, E Einahrawy, Richard P Martin, W-H Ju, Parameshwaran Krishnan, and AS Krishnakumar. 2005. Bayesian indoor positioning systems. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, Vol. 2. IEEE, 1217–1227.

[30] Keiichi Nakamura, Masato Kamio, Tetsushi Watanabe, Shinsuke Kobayashi, Noboru Koshizuka, and Ken Sakamura. 2009. Reliable ranging technique based on statistical RSSI analyses for an ad-hoc proximity detection system. In *2009 IEEE International Conference on Pervasive Computing and Communications*. IEEE, 1–6.

[31] Jakub Neburka, Zdenek Tlamsa, Vlastimil Benes, Ladislav Polak, Ondrej Kaller, Libor Bolecek, Jiri Sebesta, and Tomas Kratochvil. 2016. Study of the performance of RSSI based Bluetooth Smart indoor positioning. In *2016 26th International Conference Radioelektronika (RADIOELEKTRONIKA)*. IEEE, 121–125.

[32] Farid Orujov and Rytis Maskeliunas. 2016. Comparative analysis of the indoor positioning algorithms using bluetooth low energy beacons. In *Proceedings of the 2016 International Conference for Young Researchers in Informatics, Mathematics and Engineering, ICYRIME 2016*. 53–57.

[33] Busra Ozdenizci, Vedat Coskun, and Kerem Ok. 2015. NFC internal: An indoor navigation system. *Sensors* 15, 4 (2015), 7571–7595.

[34] Anindya S Paul and Eric A Wan. 2009. RSSI-based indoor localization and tracking using sigma-point Kalman smoothers. *IEEE Journal of selected topics in signal processing* 3, 5 (2009), 860–873.

[35] Fazli Subhan, Salman Ahmed, and Khalid Ashraf. 2014. Extended gradient predictor and filter for smoothing RSSI. In *16th International Conference on Advanced Communication Technology*. IEEE, 1198–1202.

[36] Fazli Subhan, Halabi Hasbullah, Azat Rozyyev, and Sheikh Tahir Bakhsh. 2011. Indoor positioning in Bluetooth networks using fingerprinting and lateration approach. In *2011 International Conference on Information Science and Applications*. IEEE, 1–9.

[37] Anamaria Tomiuc et al. 2014. Navigating Culture. Enhancing Visitor Museum Experience through Mobile Technologies. From Smartphone to Google Glass. *Journal of Media Research-Revista de Studii Media* 7, 20 (2014), 33–46.

[38] Joaquín Torres-Sospedra, Antonio R Jiménez, Stefan Knauth, Adriano Moreira, Yair Beer, Toni Fetzer, Viet-Cuong Ta, Raul Montoliu, Fernando Seco, Germán M Mendoza-Silva, et al. 2017. The smartphone-based offline indoor location competition at IPIN 2016: Analysis and future work. *Sensors* 17, 3 (2017), 557.

[39] Joaquín Torres-Sospedra, Raúl Montoliu, Sergio Trilles, Óscar Belmonte, and Joaquín Huerta. 2015. Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems. *Expert Systems with Applications* 42, 23 (2015), 9263–9278.

[40] Joaquín Torres-Sospedra, Adriano Moreira, Stefan Knauth, Rafael Berkvens, Raul Montoliu, Oscar Belmonte, Sergio Trilles, Maria Joao Nicolau, Filipe Meneses, António Costa, et al. 2017. A realistic evaluation of indoor positioning systems based on Wi-Fi fingerprinting: The 2015 EvAAL–ETRI competition. *Journal of ambient intelligence and smart environments* 9, 2 (2017), 263–279.

[41] Henry Tsai and Kelvin Sung. 2012. Mobile applications and museum visitation. *Computer* 45, 4 (2012), 95–98.

[42] Rudolph Van Der Merwe et al. 2004. *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*. Ph.D. Dissertation. OGI School of Science & Engineering at OHSU.

[43] Industry Wire. 2014. Cisco bouwt 'onzichtbaar' netwerk in Rijksmuseum. https://www.emerce.nl/wire/cisco-bouwt-onzichtbaar-netwerk-rijksmuseum

[44] Ryan Winfield Woodings and Mark Gerrior. 2005. Avoiding interference in the 2.4-GHz ISM band. *Microwave Engineering Online* (2005).

[45] Faheem Zafari, Athanasios Gkelias, and Kin K Leung. 2019. A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials* 21, 3 (2019), 2568–2599.

[46] Faheem Zafari, Ioannis Papapanagiotou, and Thomas J Hacker. 2018. A novel Bayesian filtering based algorithm for RSSI-based indoor localization. In *2018 IEEE International Conference on Communications (ICC)*. IEEE, 1–7.

[47] Li Zhang, Xiao Liu, Jie Song, Cathal Gurrin, and Zhiliang Zhu. 2013. A comprehensive study of bluetooth fingerprinting-based algorithms for localization. In *2013 27th International Conference on Advanced Information Networking and Applications Workshops*. IEEE, 300–305.

[48] Yuan Zhuang, Jun Yang, You Li, Longning Qi, and Naser El-Sheimy. 2016. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors* 16, 5 (2016), 596.

A APPENDIX

A.1 Human influences

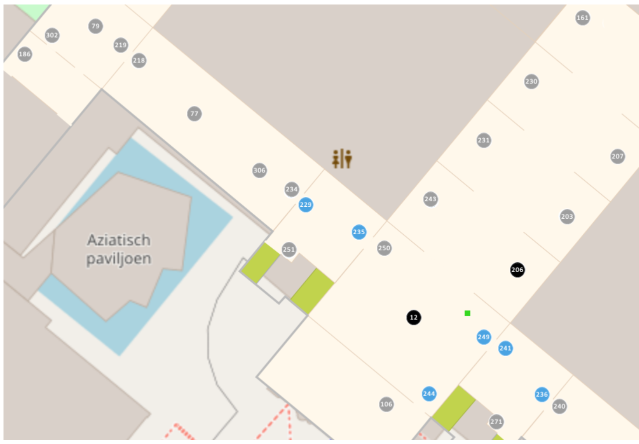

Figure 10: Visualisation of the beacons in range at the position measured in the Night Watch room (green square). The gray, blue and black circle represent beacons that are respectively not measured, significantly influenced by humans, and not influenced by humans. The numbers within the circles represent the beacon Minors. The beacon with Minor 206 is located on the first floor and is therefore not visible.

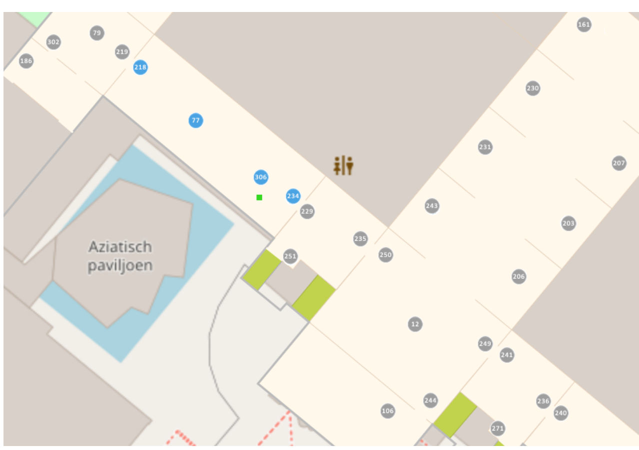

Figure 11: Visualisation of the beacons in range at the position measured in the Naval Power room (green square). The beacon with Minor 91 is located on the first floor and is therefore not visible.

Table 14: Result of Levene's test for the Night Watch Room. Five beacons are significant, indicating that the RSSI variations of these beacons differ from when people are present or not.

B_M	T-statistic	p
12	55.073	**0.000**
206	0.131	0.719
229	7.983	**0.005**
235	1.133	0.289
236	135.57	**0.000**
241	5.675	**0.018**
244	4.903	**0.027**
249	0.008	0.927
All	1.214	0.271

Table 15: Result of Levene's test for the Naval Power room.

B_M	T-statistic	p
77	13.988	**0.000**
91	0.002	0.969
218	18.075	**0.000**
234	0.099	0.753
306	6.687	**0.010**
All	449.884	**0.000**

Table 16: Result of Welch's t-test for each beacon from the Night Watch room. Six of the eight beacons have a significant difference comparing the average RSSI with and without people present.

B_M	T-statistic	p
12	1.028	0.305
206	-	-
229	4.977	**0.000**
235	2.458	**0.018**
236	2.530	**0.012**
241	16.056	**0.000**
244	15.238	**0.000**
249	-2.591	**0.001**
All	1.786	0.074

Table 17: Result of Welch's t-test for each beacon from the Naval Power room. Four of the five beacons have a significant difference comparing the average RSSI with and without people present.

B_M	T-statistic	p
77	44.292	**0.000**
91	-	-
218	2.941	**0.003**
234	-6.497	**0.000**
306	35.823	**0.000**
All	18.974	**0.000**

A.2 Apps

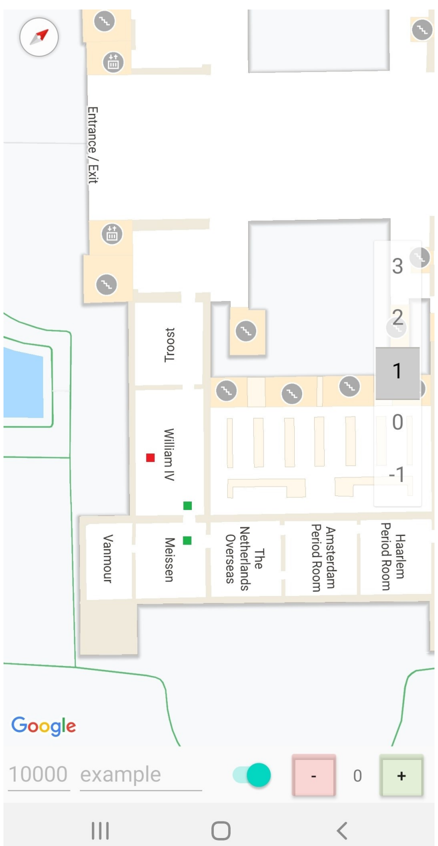

Figure 12: Screenshot of the location app. A Red dot appears after clicking on a certain spot. After 10 seconds the color of the dot changes to green to indicate that the time has passed.

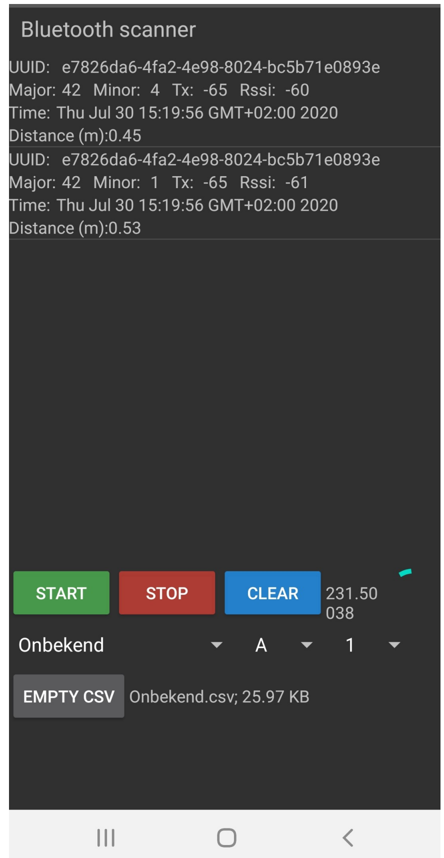

Figure 13: Screenshot of the Bluetooth recording app. All devices collected signals and saved them to a CSV file.

A.3 Results and error analysis

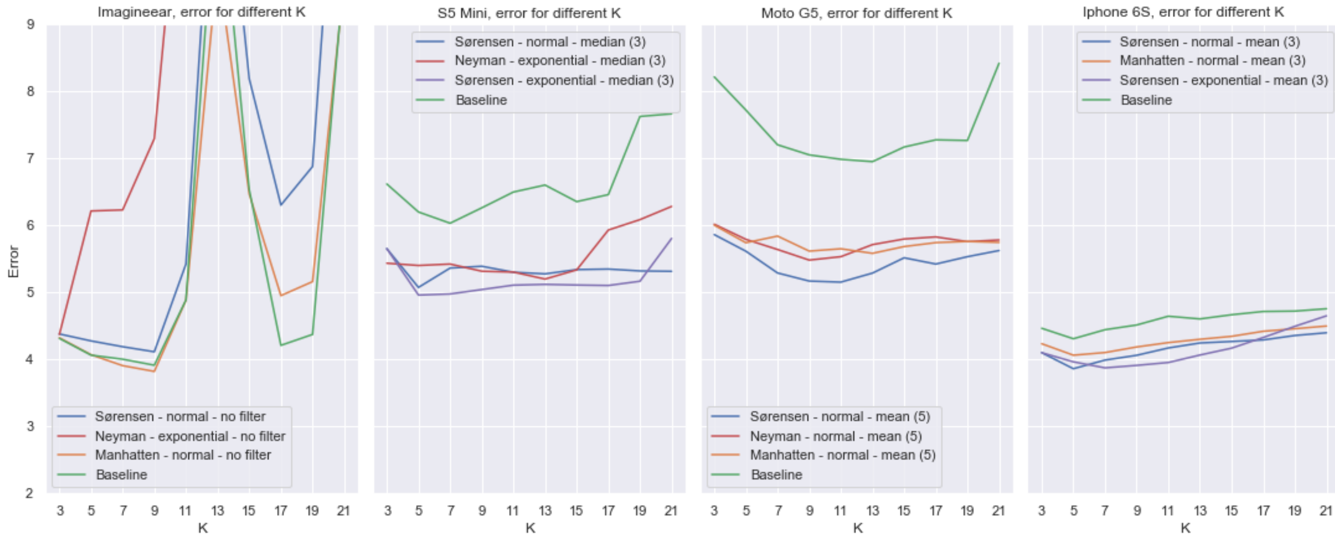

Figure 14: The *Error* found using the baseline and three best performing models per phone for different amounts of K on the validation set.

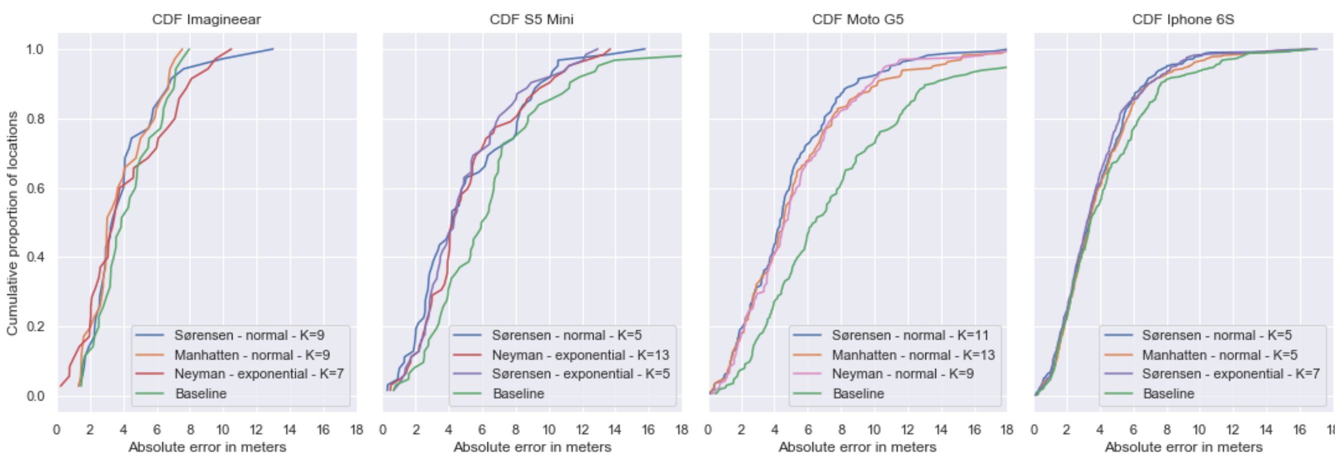

Figure 15: Cumulative distribution of the absolute error from the best performing models and K combinations, together with the baseline reached on the validation set for each of the devices.

Figure 16: The *Error* found using the baseline, the general model, and the three best performing models per phone for different amounts of K on the test set.

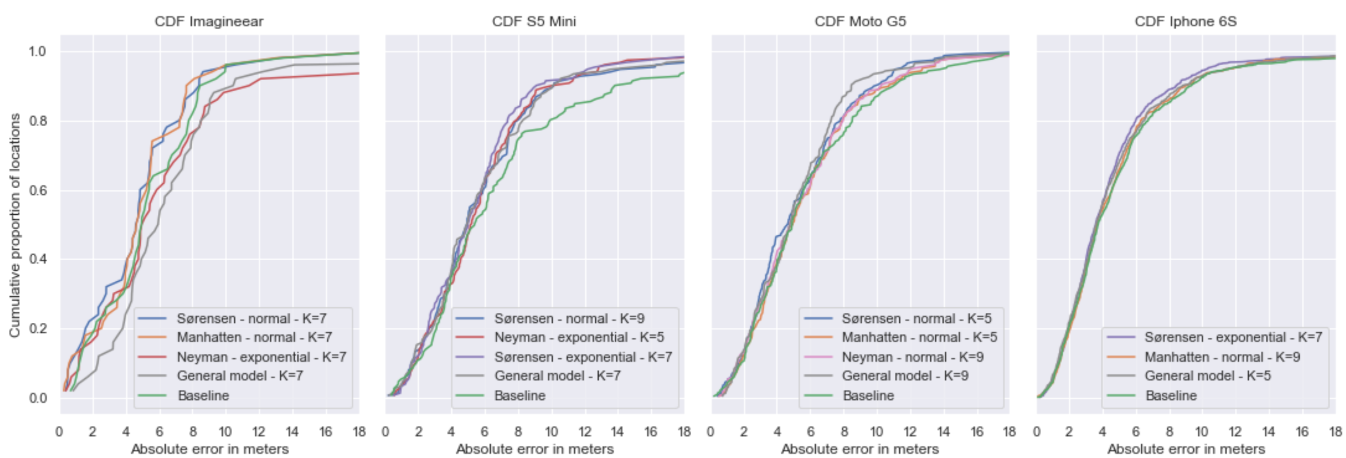

Figure 17: Cumulative distribution of the absolute error from the best performing models and K combinations, together with the baseline reached on the test set for each of the devices.

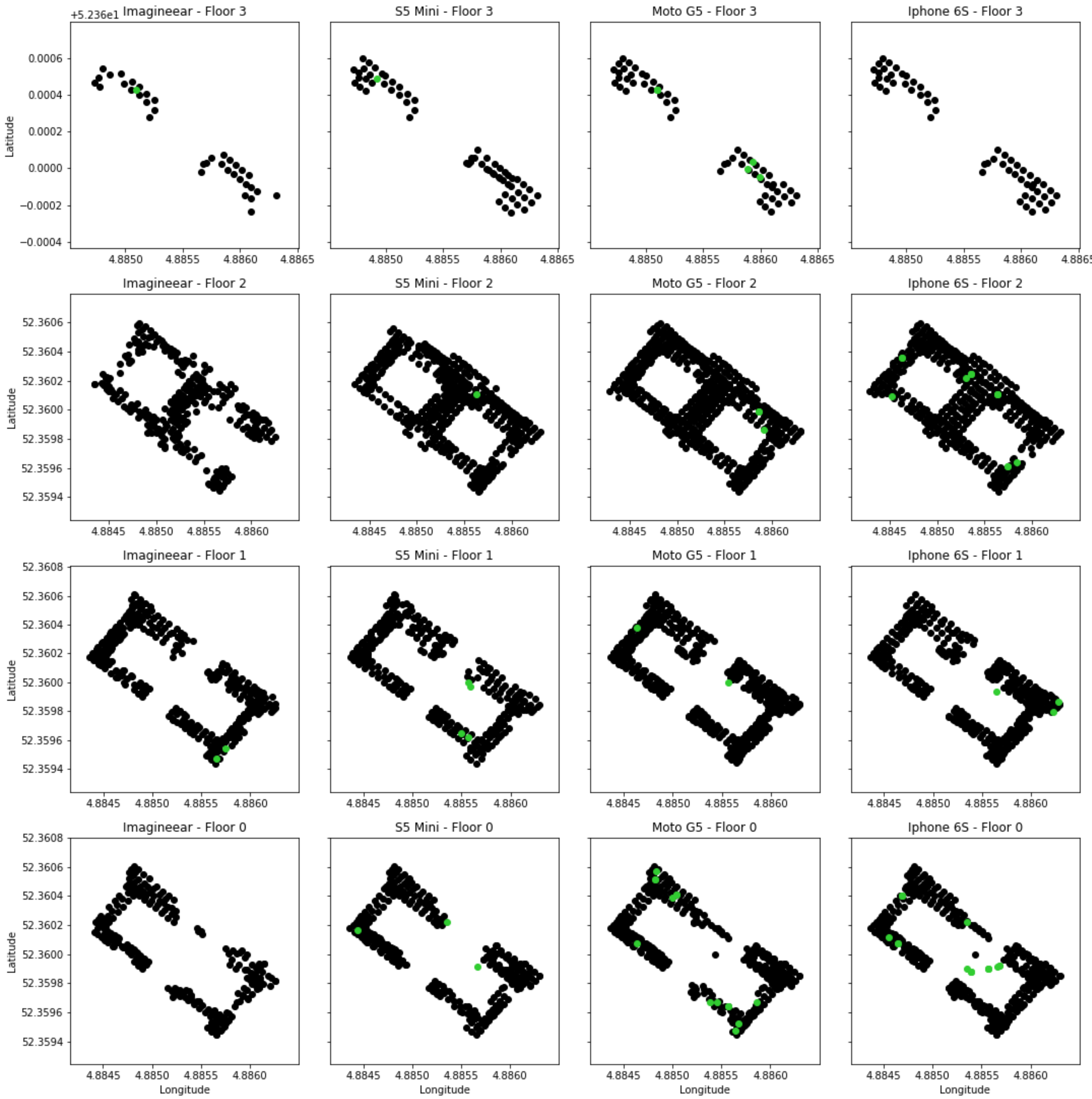

Figure 18: For each of the phones, the real position of predicted locations with a wrong predicted floor (green dots). The black dots represent the fingerprints from the database. Each column is a device, each row a floor.

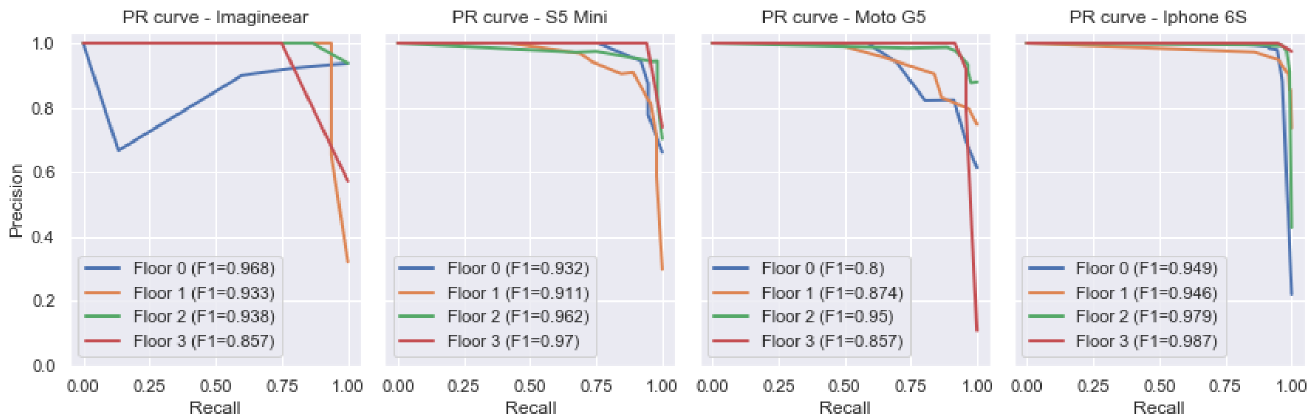

Figure 19: Precision-recall curves per floor, from the models with the lowest *Error* on the test set for each device.

Table 18: Effect on score metrics when separating K for floor and localisation prediction. The first value for K floor (K_F) per device is the same K used on the best performing model on the test set.

Device	K_F	Mean(std)	Median	3rd Q.	90th	95th	RMSE	Error	Floor
Imagineear	7	**4.89 (3.49)**	4.67	6.24	8.40	9.69	**6.00**	5.13	94.00
	3	4.93 (3.45)	4.67	6.24	8.40	9.69	6.02	**5.01**	**98.00**
	1	6.06 (8.53)	4.73	6.24	8.44	12.02	10.47	6.38	92.00
S5 Mini	7	5.67 (4.13)	5.00	6.89	**9.11**	12.39	7.02	**5.91**	**94.04**
	3	5.78 (4.50)	5.00	6.89	9.61	12.39	7.33	6.07	92.72
	1	**5.67 (3.96)**	5.08	6.92	9.61	**12.05**	**6.92**	5.96	92.72
Moto G5	9	5.29 (3.66)	**4.79**	**6.87**	**8.47**	11.78	**6.43**	5.73	88.94
	3	5.16 (3.14)	4.82	6.91	8.54	11.08	8.54	**5.57**	**89.82**
	1	**5.15 (3.04)**	4.82	6.95	8.64	**10.47**	8.64	5.63	88.05
Iphone 6S	5	**4.56 (4.50)**	3.55	5.54	**8.20**	**10.33**	**6.41**	**4.70**	**96.43**
	3	4.60 (4.58)	3.55	5.54	8.20	10.51	6.49	4.77	95.78
	1	4.64 (4.78)	3.55	5.63	8.43	10.54	6.67	4.87	94.38