Indoor Navigation Using Virtual Anchor Points

Eike Jens Hoffmann, Martin Werner, Lorenz Schauer
Mobile and Distributed Systems Group
Ludwig-Maximilians-Universität München
eike-jens.hoffmann@ifi.lmu.de, martin.werner@ifi.lmu.de, lorenz.schauer@ifi.lmu.de

***Abstract*—Indoor navigation and location-awareness are important and growing research areas due to the interest of mobile users in location-based services comparable to outside services. However, indoor positioning is a very hard task. Without installing dedicated hardware into a building, indoor positioning is a problem with inherent ambiguity. Using Wi-Fi signals to calculate location is a classical and successful approach for localizing mobile devices in buildings. With this paper, however, we provide support for proximity detection using Wi-Fi, which is considerably less ambiguous and still sufficiently useful for many indoor location-based services. Additionally, presence-based positioning has seen a boost in adoption due to the invention of cheap dedicated devices for localization including iBeacon and similar Bluetooth 4.0 beacons. We provide a concept called Virtual Anchor Point, which is a low-dimensional representation of the essence of a point in signal space providing a presence-based location-aware system.**

I. Introduction

The immense diffusion of modern mobile devices, such as tablets or smartphones, has involved a tremendous usage of mobile applications in recent years. A remarkable amount of them require access to the user's current location information in order to provide specific context-aware services. The benefit of such location-based services (LBS) is highly sensitive to the accuracy of the underlying positioning technique. While GPS is commonly used for outdoor scenarios, it neither operates well within buildings, nor does it meet the requirements for an accurate indoor positioning system. Hence, other techniques are required in this context in order to provide feasible indoor location based services (ILBS) such as indoor navigation applications.

A lot of research has been done with respect to this topic to provide adequate positioning information for ILBS. Several techniques and approaches have been investigated to improve important features of indoor positioning systems, e.g. accuracy, precision, cost, scalability. However, up to now, none of these investigations presents an overall solution which perfectly meets all requirements. For global adoption, wireless local area networks, commonly known as Wi-Fi, have been emerged as one of the most promising technique, due to the fact, that Wi-Fi infrastructures already exist in many buildings all over the world.

Nevertheless, accurate Wi-Fi based indoor positioning remains challenging, due to several aspects: first, received signal strengths of access points show high fluctuations within buildings (mainly caused by multipath propagation effects) rendering trilateration imprecise. Second, pattern matching approaches, e.g, the well-known Wi-Fi fingerprinting, require a lot of efforts for creating and maintaining radio maps. Third, time-of-flight approaches suffer from inaccurate timestamping, due to missing precise timer providing nanosecond resolution.

978-1-4799-8915-7/16/$31.00 ©2016 IEEE

In order to overcome the mentioned problems for Wi-Fi based positioning, special low-cost devices based on Bluetooth Low Energy (BLE) have been brought to market. The so-called beacons provide highly accurate and simple proximity detection, which is accessible in three threshold steps: far, near, and immediate. Furthermore, they are widely supported by common mobile operating systems, such as iOS and Android.

While accurate proximity detections becomes possible in the vicinity of such beacons, this technique suffers from low scalability, due to the small coverage range of about 10 meters per beacon. Hence, a seamless indoor navigation based on beacons is hardly realistic for large buildings, such as airports or shopping malls, due to the immense amount of beacons required.

With this paper, we propose a novel solution to this problem with a method that supports a proximity-like interface from time series of Wi-Fi readings. The basic idea is to detect certain locations inside an area of interest showing unambiguous patterns of receivable Wi-Fi signals. These locations are stored as Virtual Anchor Points (VAPs) and can be additionally used for indoor navigation in order to reduce the required amount of physical devices, like beacons. For scalability reasons, we present an efficient way for the detection of suitable VAPs from time series, using an embedding into a lower-dimensional Euclidean space. Common dissimilarity functions such as the Euclidean dissimilarity or the inverse of the Jaccard coefficient are statistically analysed and used for multidimensional scaling for embedding VAP candidates. Our methods and findings are evaluated with respect to the quality for proximity detection on VAPs using a real world Wi-Fi dataset recorded in our office building.

The most important thing to note is that the detection and location of Virtual Anchor Points is done in signal space alone. There is no need for any mapping between annotated locations, anchor points, and maps. Instead, the system just informs the service that a locally unique location has been traversed and these locations can then be mapped in another step.

The main contribution of this paper is an approach to identify locations where Virtual Anchor Points are non-ambiguously detected from a stream of Wi-Fi signals in a lower-dimensional model space.

The remainder of the paper is structured as follows: Section II reviews related work, Section III shortly recalls our indoor location-aware system based on time-series analysis, Section IV explains the methodology to find Virtual Anchor Points, Section V evaluates the methodology on a large dataset, and Section VI concludes the paper with some hints on future work.

II. Related Work

Indoor positioning techniques are well studied as the basis for indoor navigation. Liu et al. [1] present a detailed survey of common wireless indoor positioning techniques and systems. The most auspicious technique in this field is Wi-Fi, due to its' standardized usage and the overall existence in most public buildings. One of the most widely adopted approach to pervasive indoor positioning based on Wi-Fi is called fingerprinting. The first work into that direction might be RADAR proposing a weighted k-nearest-neighbor approach applied to RSSI readings in oder to infer the location of a mobile target [2]. Lately, this technique has been extended to include compass readings as a pre-filter improving the accuracy of the positioning process [3]. Furthermore, Wi-Fi has been investigated for time-of-flight indoor localization using off-the-shelf infrastructures [4], and also for an involuntary tracking of people in an area of interest [5]–[8].

Indoor location-aware systems have been proposed using a large range of methods. A general introduction to this area of research can be found in [9]. The most promising methods for pervasive location awareness include approaches based on image recognition [10], [11], GNSS inside buildings [12], and Wi-Fi [2], [3] signal strength. Most other approaches either need a dedicated infrastructure, detailed map information, or a personal calibration procedure such as in pedestrian dead reckoning.

Another upcoming technique for indoor location awareness is based on Bluetooth Low Energy. Bluetooth Low Energy is an extension to the Bluetooth standard providing Bluetooth services with a very low energy consumption. It has been used to create battery-powered beacons, which send out their own identification in regular intervals and can be used by recent smartphones to assess the proximity to the beacon. Commercially, Bluetooth beacons are promising as iPhones support this approach through Apple's iBeacon technology and Android supports beacons via Google Eddystone. Beacons serve as a cheap extension to Wi-Fi positioning and can overcome gaps in coverage. However, they still produce high cost, due to distribution and maintenance overhead.

The work presented herein is partly based on trajectory computing, which is the area of computing concerned with time series in spatial domains. A good overview to this area is given in [13]. This research area has first been applied to Wi-Fi positioning for destination estimation [14].

As a tool, this work uses multidimensional scaling (MDS), which can be described as follows: a distance matrix of locations is given to a system trying to find the best configuration of locations in a Euclidean model space of a fixed dimension by resembling the ratios of the given distances. In other words: the system finds a set of points ("configuration") from an Euclidean space such that the relative distances between points are preserved as much as possible.

Concretely, MDS minimizes the Kruskal stress

$$\sigma(X) = \sum_{i<j} \left(\omega_{i,j}(\delta_{i,j} - d_{i,j}(X))\right)^2$$

In this equation, $\delta_{i,j}$ denotes the distance between object i and j, $d_{i,j}(X)$ denotes the distance between point i and j in the model space, and $\omega_{i,j}$ weights the contribution of the distance between object i and j and can be used to model missing values by setting $\omega_{i,j} = 1$ for all available values and zero for missing distances. This is especially useful for RSSI-based fingerprints in buildings, which might have no access point in common making it impossible to measure any dissimilarity between them.

A commonly used algorithm for calculating MDS configurations is named SMACOF [15], [16]. The central tradeoff in MDS is between the number of dimensions and the quality of the embedding.

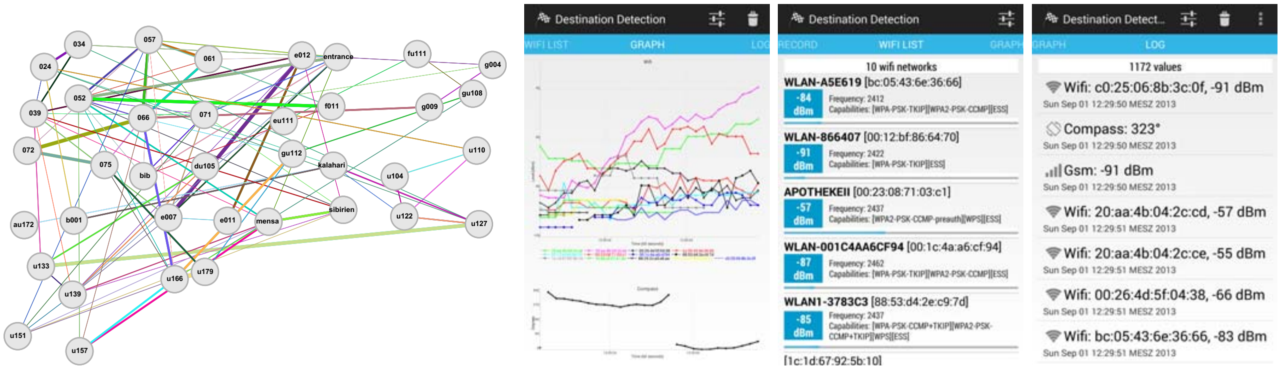
III. Navigation from Time Series of Sensor Readings

In a recent paper, we proposed a novel approach to Wi-Fi location awareness in which time series of signal readings are collected [14]. Additionally, these are sparsely labeled at important rooms (e.g., by asking the user for a label when the phone is not moving for a given time).

This creates a dataset of time series of signal strength readings for beacons of surrounding access points. The dataset used in this study is the same as we used for the original study. The modeled location labels and a two-dimensional embedding of these labels is depicted in Figure 1(a). This type of data has been collected by an Android application, which is depicted in Figure 1(b) showing part of such a time-series.

In this application, the dataset was segmented into pieces of trajectories and an incoming trajectory has been used to assess the most probable destinations of the mobile device. Using the Fréchet distance, the system was able to correctly predict the destination of the mobile user quite early and with high accuracy, a demo video is available on YouTube[1]. The Fréchet distance between two time series t_1 and t_2 is based on calculating the minimum length of a leash connecting a dog walking on t_1 and and his owner walking on t_2, both never going backwards. It is a true metric and especially well-behaved with respect to noise as it contains the noise level only once and does not sum up noise terms for different points in times. However, it is very susceptible to outliers, as it only remembers the minimum length of the leash. Variants for using the mean, median or another summary statistic of the leash length over time have been defined to alleviate this.

[1] https://www.youtube.com/watch?v=FHoHcRIK_j8



10 wifi networks

WLAN-A5E619 [bc:05:43:6e:36:66]
-84 dBm
Frequency: 2412
Capabilities: [WPA-PSK-TKIP][WPA2-PSK-CCMP][ESS]

WLAN-866407 [00:12:bf:86:64:70]
-91 dBm
Frequency: 2422
Capabilities: [WPA-PSK-TKIP][ESS]

APOTHEKEII [00:23:08:71:03:c1]
-57 dBm
Frequency: 2437
Capabilities: [WPA2-PSK-CCMP-preauth][WPS][ESS]

WLAN-001C4AA6CF94 [00:1c:4a:a6:cf:94]
-87 dBm
Frequency: 2462
Capabilities: [WPA-PSK-TKIP][WPA2-PSK-CCMP][ESS]

WLAN1-3783C3 [88:53:d4:2e:c9:7d]
-85 dBm
Frequency: 2437
Capabilities: [WPA-PSK-CCMP+TKIP][WPA2-PSK-CCMP+TKIP][WPS][ESS]

[1c:1d:67:92:5b:10]

1172 values

Wifi: c0:25:06:8b:3c:0f, -91 dBm
Sun Sep 01 12:29:50 MESZ 2013

Compass: 323°
Sun Sep 01 12:29:50 MESZ 2013

Gsm: -91 dBm
Sun Sep 01 12:29:50 MESZ 2013

Wifi: 20:aa:4b:04:2c:cd, -57 dBm
Sun Sep 01 12:29:51 MESZ 2013

Wifi: 20:aa:4b:04:2c:ce, -55 dBm
Sun Sep 01 12:29:51 MESZ 2013

Wifi: 00:26:4d:5f:04:38, -66 dBm
Sun Sep 01 12:29:51 MESZ 2013

Wifi: bc:05:43:6e:36:66, -83 dBm
Sun Sep 01 12:29:51 MESZ 2013

(a) Dataset of trajectories between labels

(b) App showing time series of Wi-Fi readings

Figure 1. Positioning System based on sparsely labeled time series [14].

IV. Method

The most important advantage of using RSS time series in indoor navigation is given by the fact, that training databases do not need a full spatial assignment: Only the start and end points of important paths must be labeled and, furthermore, these are often distant to each other. Hence, a simple positioning system with room level accuracy would already provide sensible labels for automatically extracting those trajectory annotations. Especially, the data can be used without standing in a specific location, waiting for a specific time, and rotating oneself with the measurement device, which is the time-consuming best practice for fingerprint-based positioning systems.

This, however, comes with a drawback of small support: The location can almost never be extracted in the online phase. Essentially, the system always only predicts the final location of the current trip. With this paper, however, we want to exploit all labels that have been given to the system as part of any trajectory and extract those in trajectories in the online phase. This extends the support of the system by a large amount.

In order to do that, we define the concept of Virtual Anchor Points. However, distant points in Wi-Fi signal space are often incomparable with each other due to not containing the same access points limiting the usefulness of fingerprints. In order to overcome this principal limitation, we project all fingerprints into a lower-dimensional Euclidean space in a way such that similarity of nearby points is preserved as much as possible and define Virtual Anchor Points as locations in this Euclidean model space.

With this paper, we want to exploit the existence of such special locations in the Euclidean model space in order to allow for a proximity-style positioning inside the query time-series. Therefore, we conduct a statistical analysis on how to choose the dissimilarity function in signal space, which dimensionality the model space needs to represent the signal space sufficiently, how weightings can be used to improve the separation of projection points and, finally, how to detect proximity to anchor points.

A. Building the Euclidean Model Space

First of all, we use Wi-Fi trajectories to extract promising, unambiguous locations, which we call Virtual Anchor Point candidates. The Wi-Fi trajectories used in this work contain timestamps associated with BSSIDs (i.e., the MAC address of the access point) and their respective signal strengths indicators. Furthermore, some timestamps include names of places in the building that are described as annotated places. In order to generate candidates, we use a time window of five seconds around an annotation to capture signal readings with respect to signal strength variations. Based on this window, an average signal strength of each measured BSSID is calculated as well as its respective standard deviation. Thus, for each annotation in the trajectories we have a set C' of quadruples (b, μ, σ, n) with b denoting the BSSID, μ and σ describing a normal distribution using mean and standard deviation and n denoting the number of samples.

Multiple distant time windows for the same location are then merged with the partioning algorithm of Chan *et al.* [17] and the resulting set C describes a candidate for a virtual anchor point.

For embedding these candidates in an Euclidean model space, a dissimilarity function is needed to create a dissimilarity matrix for the multidimensional scaling algorithm.

An obvious function is given by adapting the Euclidean distance to all BSSIDs two sets have in common. Given two anchor point candidates C_x and C_y, let I denote the set of pairs of mean values (μ_x, μ_y) of readings in C_x and C_y with the same BSSID. With these names, the Euclidean dissimilarity can be defined as

$$d_{\text{Eucl.}}(C_x, C_y) = \begin{cases} \infty, & \text{if } I = \emptyset \\ \sqrt{\sum_{(\mu_x,\mu_y)\in I} (\mu_x - \mu_y)^2}, & \text{else.} \end{cases} \tag{1}$$

Note that this definition is not a metric: let C_x contain two SSIDs S and T, C_y contain two SSIDs T and U, and C_z contain two SSIDs U and V. Then both $d_{\text{Eucl.}}(C_x, C_y)$

and $d_{\text{Eucl.}}(C_y, C_z)$ are finite, but $d_{\text{Eucl.}}(C_x, C_z)$ is infinite. This violates the triangle inequality $d_{\text{Eucl.}}(C_x, C_z) \leq d_{\text{Eucl.}}(C_x, C_y) + d_{\text{Eucl.}}(C_y, C_z)$

The Euclidean dissimilarity $d_{\text{Eucl.}}$ has the disadvantage that is does not take the size of the intersection $|I|$ into account. In order to incorporate this information as well, the dimension-normalised Euclidean distance uses this number:

$$d_{\text{DN-Eucl.}}(C_x, C_y) = \begin{cases} \infty, & \text{if } I = \emptyset \\ \sqrt{\frac{1}{|I|}\sum_I (\mu_x - \mu_y)^2}, & \text{else.} \end{cases} \tag{2}$$

Moghtadaiee and Dempster [18] proposed the Chebyshev metric for calculating distances in signal space, thus, we adapted this function for our use case, too:

$$d_{\text{Cheb.}}(C_x, C_y) = \begin{cases} \infty, & \text{if } I = \emptyset \\ \max_I \left(|\mu_x - \mu_y|\right), & \text{else.} \end{cases} \tag{3}$$

Since the signal strength measurements vary highly over time, we considered using a signal strength agnostic function, relying only the present and absent BSSIDs. The inverse of the Jaccard coefficient for similarity of sets results in a metric called the Jaccard metric:

$$d_{\text{Jacc.}}(C_x, C_y) = 1 - \frac{|I|}{|C_x| + |C_y| - |I|} \tag{4}$$

In contrast to all other functions presented above for comparing time surroundings, this dissimilarity is a true metric as it is able to deal correctly with empty intersections.

We perform the embedding into an Euclidean model space using multidimensional scaling with respect to all of these dissimilarity functions. In multidimensional scaling, the task is to find a set of locations in a given Euclidean space such that the Euclidean distance between those locations resembles the distance matrix given to the algorithm.

We use these locations in the Euclidean model space to identify the candidates, which are unique and unambiguous with respect to all other candidates.

B. Finding unique anchor points

The uniqueness of an anchor point candidate is expressed by the standard deviation of the set of measurements taken. Therefore, we need to map this standard deviation into the model space as well. We first calculate the mean standard deviation of a candidate C.

$$\bar{\sigma}_C = \frac{1}{|C|} \sum_{(b,\mu,\sigma,n)\in C} \sigma \tag{5}$$

For scaling this value into the model space, we calculate the mean ratio between the distances in signal space and model space.

$$\tau = \frac{1}{|\mathbf{C}|^2} \sum_{x\in\mathbf{C}} \sum_{y\in\mathbf{C}} \frac{d_{\text{mds}}(x,y)}{d_{\text{sig}}(x,y)} \tag{6}$$

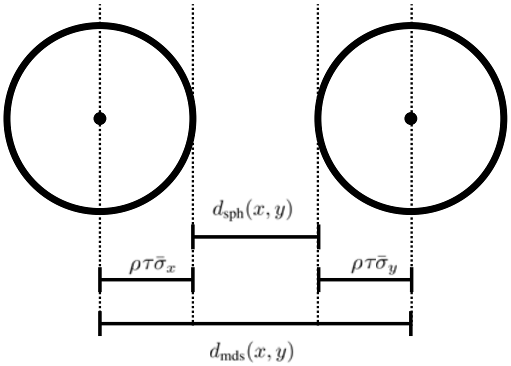


Figure 2. Sphere Distance

with $\mathbf{C}$ denoting the set of all candidates C.

Since we are interpreting the candidates as spheres in a low dimensional space and the mapping using τ is very coarse, we add a scaling parameter ρ and define the distance between two such spheres d_{sph} as illustrated in Figure 2 and expressed by the following equation:

$$d_{\text{sph}}(x,y) = d_{\text{mds}}(x,y) - \rho \cdot \tau \cdot (\bar{\sigma}_x + \bar{\sigma}_y) \tag{7}$$

This sphere distance can now be used to check if two candidates intersect each other, i.e. the sphere distance $d_{sph} \geq 0$. Using this definition, we define a Virtual Anchor Point as a Virtual Anchor Point candidate, which does not overlap another candidate.

$$x \in \mathbf{C} \text{ is VAP } \Leftrightarrow \forall y \in \mathbf{C}, y \neq x : d_{sph}(x,y) \geq 0 \tag{8}$$

After removing intersecting candidates in the set, the remaining candidates are true Virtual Anchor Points. This set is denoted as V_ρ depending on the scaling factor ρ used for creating the set. These points can now be used in the online phase for predicting the nearest Virtual Anchor Point to the current time series.

Figure 3 depicts an illustrative projection of Virtual Anchor Points in the model space generated with multidimensional scaling. Each Virtual Anchor Point has a radius from the scaled mean standard deviations of the observed signal strengths. Although, some seem to be quite close to each other, these are separated in other dimensions. Please note that the number of dimensions has to be derived from the dataset. This will be explained in detail in the evaluation.

C. Proximity detection

The Virtual Anchor Points identified during the offline phase can now be used to create a location-based service. Therefore, we compare the Wi-Fi access points within range m with the ones seen in the different Virtual Anchor Points V_ρ and predict the nearest anchor point. Consequently, the classification is done with a nearest neighbour classifier.

$$\text{argmin}_{v\in V_\rho}\{d(v,m)\} \tag{9}$$

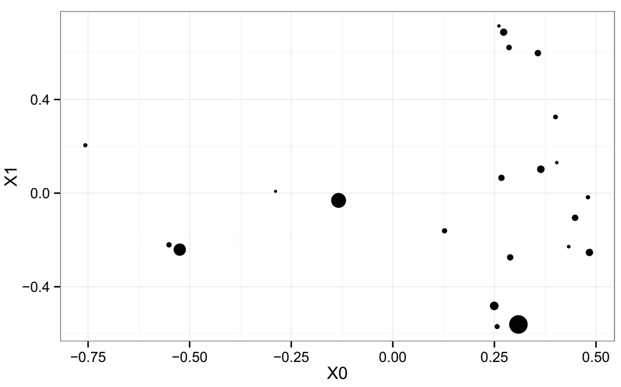


Figure 3. Illustrative example of Virtual Anchor Points in a two-dimensional projection

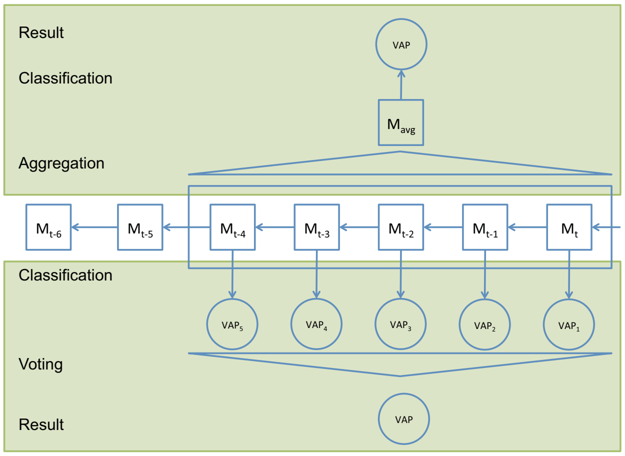


Figure 4. FIFO-queue used during the online phase with two different methods

Since Wi-Fi signals strengths vary highly at a single place [19]–[22], we evaluate several different methods for estimating the nearest virtual anchor point. In addition to the single measurement classification, we use a First-In-First-Out queue and predict the nearest anchor point based on the last five measurements. This is in accordance with creating the candidates, which are summed up of a time window of five seconds. Figure 4 shows the queue. Firstly, five measurements can be aggregated in two different ways with the union set or the intersection set of all received BSSIDs. Secondly, the measurements can be collected into means per access point or treated individually resolving the many classification results via a majority voting approach.

If the aggregation is done using the union set method, all observed BSSIDs are taken into account, whereas the intersection method calculates the set of BSSIDs that are found in each of the five measurements. In both cases the result set also includes the mean of each signal strength of a BSSID as well as its standard deviation.

Additionally, we reevaluate all four dissimilarities discussed above with respect to their suitability for the nearest-neighbour classifier within the different methods.

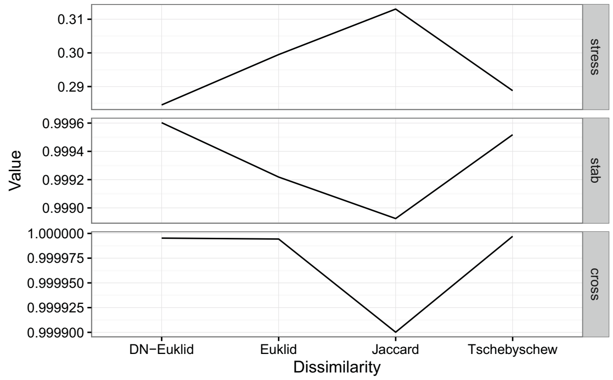


Figure 5. *Stress*, *Stability* and *Cross* values for different distance metrics when using candidates from the signal maximum and the annotated places method and embedding them in a two dimensional space

V. Evaluation

In this section, we concentrate on the questions of how to build the model space and how to extract meaningful anchor points from the given dataset type more concretely. We applied the previously described method to the dataset presented in [14]. The whole dataset consists of 9,245 fingerprints including 278 different Wi-Fi access points in a university building. Out of this dataset, 49 candidates for Virtual Anchor Points could be extracted and were used for optimizing the parameters for the multidimensional scaling using the SMACOF algorithm.

A. Building the Euclidean Model Space

First of all, we need to choose a suitable distance function for creating distance matrices between candidates. Therefore, we created distance matrices for various dissimilarity measures with all candidates and ran a MDS using default parameters.

Figure 5 shows three values: stress, stability, and cross validity from a Jackknife validation of the multidimensional scaling. The stress captures the mean difference between the distances given and the distances in the model and is independent of the Jackknife validation. The stability shows the ratio of Between to Total variance [23] and the cross value indicates the variance of different sampled configurations to the actual configuration with all data points. Basically, the goal is to find a distance metric with low stress, high stability, and high cross values. Consequently, we chose the dimension-normalised euclidean distance, which has the lowest stress value with ≈ 0.28 and cross and stability measures ≈ 1.00.

Next, we need to find the optimal dimensionality of the model space for embedding the dataset using MDS. The higher the dimension of the model space, the more degrees of freedom exist in order to embed points. Therefore, it is easier for the algorithm to find a configuration, which resembles the given dissimilarities. As a result, the higher the dimension, the lower the stress value and the stability and cross metrics become more important. Figure 6 shows a decreasing stress value with higher dimensionalities. Starting at ≈ 0.28 with two dimension, the stress can be reduced to ≈ 0.24 at five and more dimensions. At five dimensions both stability and cross validity, have a local maximum with ≈ 1.00. This finally led to the choice of using a model space with five dimensions for this dataset.

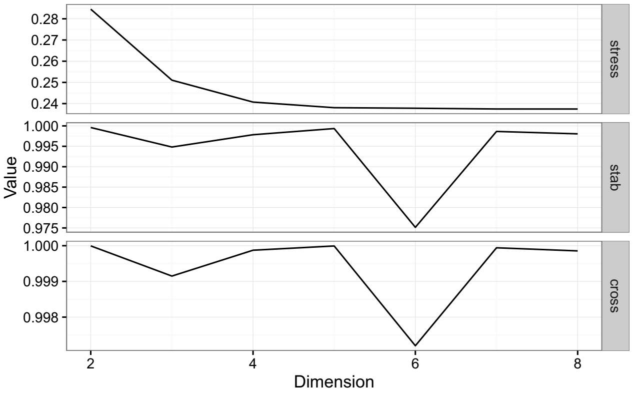


Figure 6. *Stress*, *Stability* and *Cross* values for different dimensionalities when using candidates from the signal maximum and the annotated places method

Multidimensionate scaling with the SMACOF algorithm allows to incorporate the reliability of each distance value into the process by specifying a weight matrix. From the assumption that the number of BSSIDs in the intersection is a determinant of the reliability of the distance, we used the size of the intersection as well as the Jaccard coefficient for weighting. Essentially, this leads to ignoring those cases, where the given distance matrices contain infinity and strengthening the influence of large intersections. The Jaccard weight matrix reduces the stress value to ≈ 0.21 with the same stability and a cross value of ≈ -32.57. This extreme cross validity is observed due to the different configurations in sampled weight matrices and can therefore be ignored.

In summary, the best approach turned out to run the MDS using a dissimilarity matrix based on the dimension-normalised Euclidean dissimilarity and a weight matrix filled with Jaccard coefficients between all candidates and embed this into a five dimensional Euclidean space. Thereby, we were able to reduce the stress by 25% in comparison to the default parameter stress.

A well-known visualisation for quality assessment of embeddings is given by Jackknife-plots [15], which show a two-dimensional projection of the configuration resulting from the MDS. Additionally, these plots show the variance that is introduced by the Jackknife validation leaving out some information.

Figure 7 shows a projection of the final configuration in the first two dimensions. On the one hand, there are candidates that are clearly separate from all other, on the other hand a cluster of candidates is formed in the lower right corner. Note that these candidates will be apart from each other with respect to another dimension.

B. Finding Unqiue Anchor Points

In order to find the candidates, which are separable in the five dimensional space, we needed to find a scaling factor τ to scale the mean standard deviations from the high dimensional signal space into the five-dimensional model space. In our case, τ was calculated to be approximately 3.12.

Choosing a scaling parameter of $\rho = 1.0$, 20 Virtual Anchor Points are filtered out of the 49 candidates. Increasing ρ to 1.5 leads to six anchor points, while decreasing it to 0.5 results in 41 anchor points.

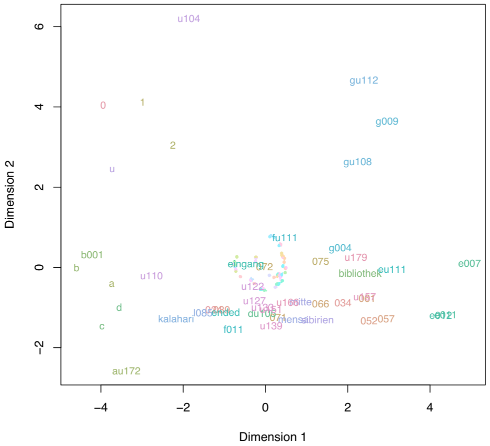


Figure 7. Jackknife plot showing the first two dimensions.

C. Quality of Proximity Detection

For predicting the nearest Virtual Anchor Point we use a nearest neighbour classifier, which implies the need for a suitable distance function. As ground truth, we use the given annotated places and compared the classification results to the nearest label in time from the trajectories. We evaluate the four dissimilarity functions as discussed above once again with respect to their performance within the classifier. Figure 8 shows that the Jaccard distance outperforms all other functions by more than 10 % with an average F1 value of 0.44. The signal strength based similarity functions result in a maximum F1 measure of 0.40 for the dimension-normalised Euclidean similarity and a lowest F1 measure of 0.34 for the Euclidean similarity.

This outstanding result of the Jaccard distance is probably due to the fact that this function does not take signal strengths into account, but is calculated based only on the present and absent BSSIDs. As near candidate points have already been rejected in a previous step, the overlap of BSSIDs has become a sufficient indicator of nearness.

With regard to the method, there is no clear winner. In combination with the Euclidean dissimilarity functions, the single method provides the best F1 values with a mean of 0.38. The intersection method is best when used with the Chebyshev dissimilarity. Together, they achieve a F1 value of 0.38. In case of the Jaccard function the best result is a F1 value of 0.46 with the union method, which is the overall best value. Therefore, we used this combination for all following evaluations.

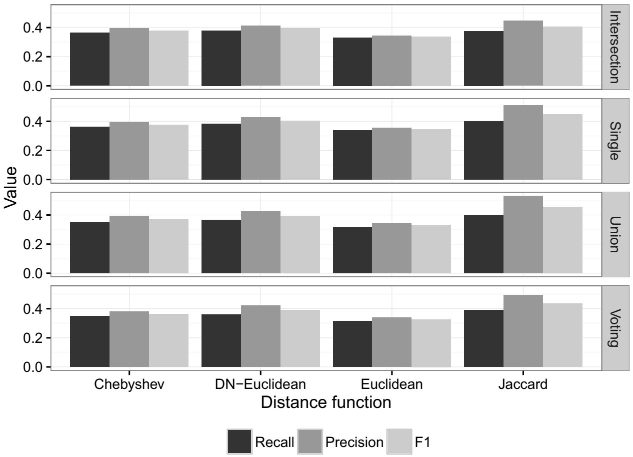


Figure 8. Evaluation of distance functions and classification methods with respect to recall, precision and F1-measure.

As a final step, we evaluate the impact of the scaling factor ρ on the classification results and measured the classification performance for different post-scaling factors $0 \leq \rho \leq 2$.

Figure 9 shows the three classification metrics precision, recall and F1-measure depending on ρ. Setting $\rho = 0$ implies a sphere radius of 0, thus all 49 candidates are considered Virtual Anchor Points. This results in a precision and a recall of 0.5, which is also true for $\rho < 0.3$. For $0.3 \leq \rho \leq 0.4$, the recall has a local maximum and therefore the F1 measure in this interval has a local maximum, too, since the precision remains almost unchanged. Starting at $\rho = 0.5$ the F1 measure is falling down to 0.36 at $\rho = 1.5$. While the recall is decreasing with higher values of ρ, the precision has local maxima of 0.53 at $\rho = 1.1$ and $\rho = 1.3$. Since there are fewer Virtual Anchor Points with higher values of ρ the probability of choosing the right anchor point by chance is much higher. Thus, a lower value of ρ is more suitable for real world applications. Therefore, we choose $\rho = 0.4$ for the dataset.

A closer look at the classification results of the single anchor points revealed that there are significant quality differences between the different anchor points. 22 out of 47 virtual anchor points can be classified with a precision ≥ 0.5 and only six anchor points show a precision of less than 0.25. Additionally, we investigated the false classifications in a confusion matrix and compared them to the spatial arrangement in the real world. As to be expected, most of the wrongly classified measurements were assigned to nearby anchor points with the most erroneous ones being just a few meters away from the next virtual anchor point. Since the true nearest Virtual Anchor Point of a measurement was derived from the next anchor point in time during the recording of the trajectories, our evaluation is a pessimistic estimation of our method. Hence, our evaluation can be seen as a lower bound of the classification performance with a mean F1-measure of 0.51 consisting of a recall of 0.50 and a precision of 0.52. A live demonstration of our system can be found on YouTube[2].

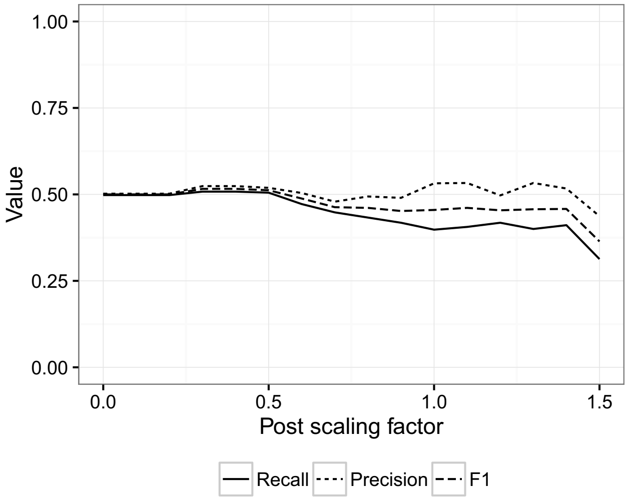


Figure 9. Evaluation of recall, precision and F1-measure for changing scaling factor ρ.

VI. Conclusion

With this paper, we have shown that it is possible to approximate the high-dimensional non-Euclidean space of RSSI readings in which every access point creates a new dimension and distance between points is only defined between the sets of access points visible in two different locations by a quite low-dimensional Euclidean model space.

First, this completes the distance relation: It is now possible to calculate a distance between any two fingerprints based on the assumption that the system stress minimization of the MDS process is sane.

As there is no mapping between spatial locations, RSSI readings, or the model space, we have redefined the positioning problem as a problem of finding locations in the Euclidean model space which are safely distinguishable from other locations. Here, different locations are extracted using the time domain of a time series of RSSI readings.

In a previous paper [14], we have shown how to use a simple labeling approach of important places (e.g., only the beginning and end of each trip through the building) in order to predict the next location in time.

With this paper, we sucessfully extended the coverage of the system for Virtual Anchor Points inside time series, even, where no labels have been modeled. An application can now

[2] https://www.youtube.com/watch?v=8ikqTxAJ8mk

use these Virtual Anchor Points for several tasks: it could ask users to manually assign locations in a map to these points or the system could just trigger beacon-like presence events for Virtual Anchor Points augmenting an existing or planned beacon infrastructure.

This approach poses an additional interface for indoor location-based services based on the time-series classification approach in which proximity information inside the time series is used. Additionally, this approach makes high-volume analysis and classification of indoor mobility data feasible by compressing high volume sequences of signal readings to low-volume sequences of Virtual Anchor Point sightings. This opens up indoor location analytics to scalable methods based on string similarity.

For future work, we pursue two directions: First, we are trying to extract the topology of the labels in an unsupervised manner by matching Virtual Anchor Points. Second, we want to revert the philosophy a bit: For now, we have used the subset of sufficiently distinguishable locations near labels to create Virtual Anchor Points. In future work, we want to extend this set of anchor points with points in signal space, which are sufficiently distinguishable from the current set of Virtual Anchor Points and which have a sufficient support in the time series set in the sense that enough people have observed these places. These could then be pushed out to the service asking users to assign labels to these additional locations.

In summary, this paper has shown that some geometry operations (nearness, nearest neighbors) in RSSI-based location-awareness systems can be performed inside a well-behaved, low-dimensional Euclidean space in which a lot of scalable algorithms (e.g., point indexing, time-series similarity search, etc.) are defined. Additionally, we can further simplify to series of discrete labels (e.g., the VAPs) opening up the application of a wide range of sequence analysis algorithms mainly designed for biological sequences or information retrieval.

REFERENCES

[1] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1067–1080, 2007.

[2] P. Bahl and V. Padmanabhan, "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, 2000, pp. 775–784. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=832252

[3] M. Kessel and M. Werner, "SMARTPOS: Accurate and precise indoor positioning on mobile phones," in *Proceedings of the 1st International Conference on Mobile Services, Resources, and Users, Barcelona*, 2011, pp. 158–163.

[4] L. Schauer, F. Dorfmeister, and M. Maier, "Potentials and limitations of wifi-positioning using time-of-flight," in *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*. IEEE, 2013, pp. 1–9.

[5] B. Bonné, A. Barzan, P. Quax, and W. Lamotte, "Wifipi: Involuntary tracking of visitors at mass events," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), IEEE 14th International Symposium and Workshops on a*, 2013, pp. 1–6.

[6] A. Musa and J. Eriksson, "Tracking unmodified smartphones using wi-fi monitors," in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, 2012, pp. 281–294.

[7] L. Schauer, M. Werner, and P. Marcus, "Estimating Crowd Densities and Pedestrian Flows Using Wi-Fi and Bluetooth," in *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). ICST, dec 2014, pp. 171–177. [Online]. Available: http://dl.acm.org/citation.cfm?id=2692983.2693006

[8] Y. Fukuzaki, M. Mochizuki, K. Murao, and N. Nishio, "Statistical analysis of actual number of pedestrians for Wi-Fi packet-based pedestrian flow sensing," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers - UbiComp '15*. New York, New York, USA: ACM Press, sep 2015, pp. 1519–1526. [Online]. Available: http://dl.acm.org/citation.cfm?id=2800835.2801623

[9] M. Werner, *Indoor Location-Based Services: Prerequisites and Foundations*. Springer, 2014.

[10] A. Mulloni, D. Wagner, I. Barakonyi, and D. Schmalstieg, "Indoor positioning and navigation with camera phones," *IEEE Pervasive Computing*, vol. 8, no. 2, pp. 22–31, 2009.

[11] M. Werner, M. Kessel, and C. Marouane, "Indoor positioning using smartphone camera," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN'11)*, 2011.

[12] M. B. Kjærgaard, H. Blunck, T. Godsk, T. Toftkjær, D. L. Christensen, and K. Grønbæk, "Indoor positioning using gps revisited," in *Pervasive Computing*. Springer, 2010, pp. 38–56.

[13] Y. Zheng and X. Zhou, *Computing with spatial trajectories*. Springer Science & Business Media, 2011.

[14] M. Werner, L. Schauer, and A. Scharf, "Reliable trajectory classification using Wi-Fi signal strength in indoor scenarios," in *2014 IEEE/ION Position, Location and Navigation Symposium - PLANS 2014*. IEEE, may 2014, pp. 663–670. [Online]. Available: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6851429

[15] J. D. Leeuw and J. Meulman, "A special jackknife for multidimensional scaling," *Journal of Classification*, vol. 3, no. 97, pp. 97–112, 1986. [Online]. Available: http://link.springer.com/article/10.1007/BF01896814

[16] J. D. Leeuw and P. Mair, "Multidimensional Scaling Using Majorization: SMACOF in R," *Journal of Statistical Software*, vol. 31, no. 3, 2009. [Online]. Available: http://escholarship.org/uc/item/9z64v481.pdf

[17] T. F. Chan, G. H. Golub, and R. J. LeVeque, "Updating formulae and a pairwise algorithm for computing sample variances," in *COMPSTAT 1982 5th Symposium held at Toulouse 1982*. Springer, 1982, pp. 30–41.

[18] V. Moghtadaiee and A. G. Dempster, "WiFi fingerprinting signal strength error modeling for short distances," *2012 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2012 - Conference Proceedings*, 2012.

[19] K. Kaemarungsi, "Distribution of WLAN received signal strength indication for indoor location determination," *2006 1st International Symposium on Wireless Pervasive Computing*, 2006.

[20] M. Kranz, C. Fischer, and A. Schmidt, "A comparative study of DECT and WLAN signals for indoor localization," in *2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, IEEE. IEEE, mar 2010, pp. 235–243. [Online]. Available: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=5466970

[21] K. Kaemarungsi and P. Krishnamurthy, "Analysis of WLAN's received signal strength indication for indoor location fingerprinting," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 292–316, apr 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2161001.2161224

[22] L. Chen, B. Li, K. Zhao, C. Rizos, and Z. Zheng, "An Improved Algorithm to Generate a Wi-Fi Fingerprint Database for Indoor Positioning," *Sensors (Basel, Switzerland)*, vol. 13, no. 8, pp. 11 085–11 096, 2013. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3812643/$\delimiter"026E30F$nhttp://www.ncbi.nlm.nih.gov/pmc/articles/PMC3812643/pdf/sensors-13-11085.pdf

[23] W. J. Heiser and J. Meulman, "Constrained multidimensional scaling, including confirmation," *Applied Psychological Measurement*, vol. 7, no. 4, pp. 381–404, 1983.