

REIHE INFORMATIK  
TR-2006-012

**COMPASS: A Probabilistic Indoor Positioning System Based on 802.11 and Digital Compasses**

Thomas King, Stephan Kopf, Thomas Haenselmann, Christian Lubberger, Wolfgang Effelsberg  
University of Mannheim  
– Fakultät für Mathematik und Informatik –  
Praktische Informatik IV  
A5, 6  
D-68159 Mannheim, Germany



# COMPASS: A Probabilistic Indoor Positioning System Based on 802.11 and Digital Compasses

Thomas King, Stephan Kopf, Thomas Haenselmann, Christian Lubberger, Wolfgang Effelsberg  
{king,kopf,haenselmann,lubberger,effelsberg}@informatik.uni-mannheim.de  
Department of Computer Science  
University of Mannheim

## ABSTRACT

Positioning systems are one of the key elements required by context-aware application and location-based services. This paper presents the design, implementation and analysis of a positioning system called COMPASS which is based on 802.11 compliant network infrastructure and digital compasses. On the mobile device, COMPASS samples the signal strength values of different access points in communication range and utilizes the orientation of the user to preselect a subset of the training data. The remaining training data is used by a probabilistic position determination algorithm to determine the position of the user. While prior systems show only limited accuracy due to blocking effects caused by human bodies, we apply digital compasses to detect the orientations of the users so that we can handle these blocking effects. After a short period of training our approach achieves an average error distance of less than 1.65 meters in our experimental environment of 312 square meters.

## Categories and Subject Descriptors

C.2.1 [Computer Systems Organization]: Network Architecture and Design—*Wireless communication*; I.5.1 [Pattern Recognition]: Models—*Statistical*

## General Terms

Algorithms, Design, Experimentation, Measurement

## Keywords

Positioning systems, location systems, 802.11, digital compass, ubiquitous computing, location-based services, context-aware applications

## 1. INTRODUCTION

During the last couple of years we have seen great improvements in downsizing computer hardware, increasing capacity of rechargeable batteries as well as the advent of wireless networks for the mass markets. These technologies allowed the manufacturers to build mobile devices that can be carried around and have the same

performance as traditional computers several years ago. The benefit of mobile devices can be leveraged by so-called *location-based services* or *context-aware applications*. Applications that act differently depending on the location or the context of the user or, even better, proactively offer location dependent information to the user are currently a hot topic in research and have been considered as a promising market.

Nowadays, the *Global Positioning System* [10] is the predominant outdoor positioning system. Although GPS works pretty well in outdoor scenarios it suffers from obstacles such as walls or roofs blocking the weak radio signals in indoor environments. So far, there is no prevailing indoor positioning system available. Many indoor positioning systems either require specialized hardware or provide poor accuracy. However, 802.11 based positioning systems really out to fill this gap because 802.11 compliant hardware is inexpensive, already available in many buildings and first research results show impressive accuracy for this kind of indoor positioning systems.

802.11 based indoor positioning systems usually rely on a so-called fingerprinting approach. Fingerprinting-based algorithms use a two stage mechanism: an *offline training phase* and an *online position determination phase*. During the offline phase, the signal strength distributions collected from access points at predefined reference points in the operation area (also called fingerprints) are stored together with their physical coordinate in a database. During the online position determination phase, mobile devices sample the signal strength of access points in communication range and search for similar patterns in the database. The closest match is selected and its coordinate is returned to the mobile device as a position estimate.

Users tend to carry mobile devices in front of themselves so that they can see the displays and interact with the devices by clicking on buttons or pressing keys. The human body consists to more than 50 percent of water and hence blocks the signal of 802.11 radios. Thus, the signal strength may vary significantly dependent on the user's orientation [2] and therefore degrade the performance of the system.

We propose a novel 802.11 based positioning system that is not affected by the orientation of the user. We utilize digital compasses to detect the orientation of the user, and based on this information, only fingerprints that share a similar orientation are selected during the offline phase to determine the user's position. Our probabilistic positioning algorithm uses only this subset of fingerprint data to determine the position of the user.

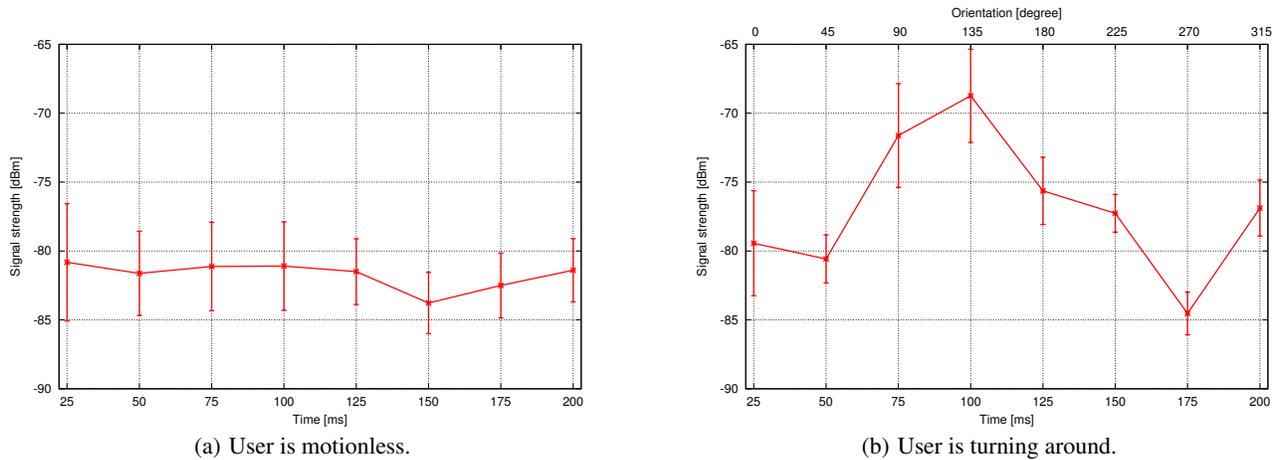


Figure 1: An example of the blocking effects caused by a human body.

As far as we know no mobile device is by default equipped with a digital compass. However, mobile devices can easily be enhanced with a digital compass because the latest generation of digital compasses is integrated into a single chip and its energy consumption is very low. Furthermore, digital compasses are cheaply available.

Our new approach leads to very encouraging results. We achieve highly accurate position estimates with a small average error distance. The precision of the results make location-based services as well as context-aware applications feasible in indoor scenarios. The user's orientation is further accessible as additional information in these applications.

The remainder of this paper is structured as follows: We present a summary of related work in the following section. In Section 3, we describe our positioning system in detail. The experimental environment is reported in Section 4 and the results of our experiments are presented and analyzed in detail in Section 5. Finally, we give a conclusion and outlook in Section 6.

## 2. RELATED WORK

Location-based services as well as context-aware applications primarily require positioning information of the user to be functional [7]. Existing indoor positioning systems can be divided into a group of systems that rely on specialized hardware and a set of systems that use off-the-shelf products. The former group comprises systems like the Active Badge project or the Cricket system. *Active Badge* utilizes badges that are capable of emitting infra-red signals [15]. Ceiling-mounted infra-red sensors detect these signals and a central unit interprets the sensor's data to determine the room where the user is located. A successor of Active Badge is the *Active Bat* project that used ultrasound time-of-flight information and multilateration to determine the position of users more precisely [16]. Similar to this project is the *Cricket* system that uses ultrasound and radio frequency receivers to locate specially equipped mobile devices [11] [12]. RFID-Tags are used in the *SpotOn* project to locate users by applying a simple radio propagation model [8]. In *EasyLiving*, cameras are used to detect users and to infer their location based on the captured images [4].

The latter group of indoor positioning systems uses off-the-shelf wireless network hardware (e.g. 802.11 compliant network hard-

ware). One of the first systems of this kind is the *RADAR* project [2] [1], but similar systems have been proposed by other researchers as well (e.g. [14] [13]). The authors of *RADAR* already experienced blocking effects of the human body to radio signals. However, they have not come up with a solution to mitigate these effects. While *RADAR* is a deterministic positioning system the systems most recently proposed have embraced probabilistic models to estimate the location of the user. Probabilistic approaches store the signal strength distribution from the access points and use probabilistic algorithms to estimate the most likely position of the user [6] [5]. *HORUS* is one of the projects that uses a similar probabilistic algorithm as our system [20] [17] [18] [19]. However, this project does not address the problem of blocking effects caused by the human body.

## 3. INDOOR POSITIONING

In this section, we present the 802.11 wireless channel characteristics and describe how our novel algorithm utilizes these properties to improve the overall positioning accuracy.

### 3.1 Wireless Channel Characteristics

The 802.11 [9] wireless network protocol is inexpensive and widely deployed in many offices, universities, and private properties. Nowadays, most modern laptop computers and digital personal assistants (PDAs) have built-in support of 802.11 and even most cellphone manufacturers provide cellphones with 802.11 support. 802.11 uses the so-called industrial, scientific, and medical (ISM) band at 2.4 Ghz or 5 Ghz depending on the substandard. However, predominant are the 2.4 Ghz based substandards called *802.11b* and *802.11g*, hence, we only focus on these standards.

Client-side hardware is capable of measuring the signal strength of access points in communication range to choose the best gateway to the wired network. For this purpose, the 802.11 specification defines a mechanism called *active scanning*. Active scanning is a simple request-response protocol: a client broadcasts a *ProbeRequest* packet and every access points in communication range replies with a *ProbeResponse* packet. Each time a client receives a *ProbeResponse* packet the network card measures the signal strength and forwards the power level value to the hardware driver. This information is accessible in all major operating system (e.g. Microsoft Windows and GNU/Linux) by user-space programs. Our position-

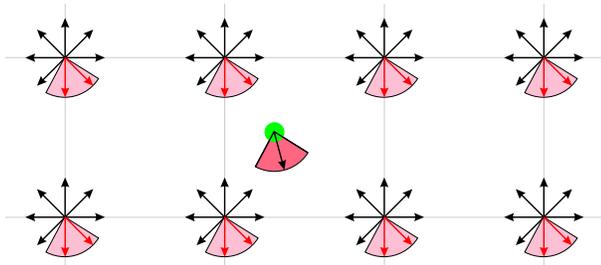
ing system is implemented as such a user-space program because it relies on the power level information provided by the network card.

2.4 Ghz is the resonance frequency of water so that the human body absorbs these radio signals. This is one of the reasons why it is difficult to predict the reception power measured at a mobile device for a given distance from the access point. For instance, a person who carries the mobile device may block the signal by standing in-between the access point and the mobile device's wireless network card.

We have analysed the blocking effect introduced by a human body to the reception power: The signal strength from an access point is sampled every 250 milliseconds while the user carries her mobile device in such a way that she is able to see the display. The access point is located five meters apart from the user. First, we measure the signal strength for 200 seconds while the person carrying the mobile device completely blocks the line of sight between the two peers. We aggregate the signal strength measurements for 25 milliseconds and calculated the average and standard deviation. As depicted in Figure 1(a) the reception power is relatively stable between 80 and 83 dBm and shows a standard deviation of less than 5 dBm. In a second step, we measure the signal strength every 45° while the person who is carrying the mobile device is turning around. The line in Figure 1(b) shows the blocking effect of the human body. The reception power increases nearly 15 dBm (see 135°) in case of a direct line of sight between the receiver and the access point. Even if the person only partially blocks the radio signals, the reception power still increases more than 5 dBm (see 90° and 225°). Furthermore, we see from the figure that the standard deviation increases if the line of sight is blocked by a human body (compare 45° and 135°).

### 3.2 Positioning Algorithm

The basic idea of our novel positioning algorithm is to sample the signal strength for selected orientations at each reference point during the offline phase and combine a subset of these values to histograms in the online phase, so that an orientation specific signal strength distribution can be computed and utilized to increase the accuracy of position estimates. Orientation specific distributions are calculated each time a user requests a position estimate. Only those signal strength histograms that show a similar orientation as the user are selected at each reference point (see Figure 2). These histograms are combined and used to compute a position estimate.



**Figure 2:** Each asterisk represents a reference point with eight signal strength histograms in an exemplary operation area. Only the signal strength histograms with a similar orientation as the user (green dot) are selected (red shaped areas) and used by the positioning algorithm.

To build a model of the real world, we overlay the operation area with a grid of  $q$  reference points spaced 1.0 meters apart. At each reference point we collect the access point's signal strength distribution for eight orientations. Formally, we model a finite *state space*  $S = \{s_1, \dots, s_n\}$  whereas we use  $n$  reference points  $R = \{r_1, \dots, r_q\}$  and the orientations  $O = \{0^\circ, 45^\circ, \dots, 315^\circ\}$  to define  $S = R \times O$ . Each state represents a certain two-dimensional coordinate with a certain orientation. The access point's signal strength distribution is aggregated by a bunch of measurements. Each measurement  $M = \{m_1, \dots, m_k\}$  contains the quantified reception power of  $k$  access points. Two different measurements may comprise a different number or set of access points. Furthermore,  $m_i = (p_i, id_i)$ ,  $i \in \{1, \dots, k\}$  is in fact a tuple containing an access point's reception power  $p_i$  ( $p_i \in \{0, 1, \dots, 255\}$ ) and unique MAC address  $id_i$  as a string. The MAC address is required to unambiguously identify a particular access point.

We use a reference vector  $\vec{\pi}$  to model the probability that a mobile device is located at a certain reference point. Therefore,  $\pi_i$ ,  $i \in \{1, \dots, q\}$  represents the probability that the user is located at reference point  $i$ . At the beginning, this vector is initialized with an uniform distribution by assigning  $\pi_i = 1/q$ .

Additionally, we need a definition for the *similarity of states*: Two states  $a$  and  $b$  represented by coordinates and an orientations  $o_a$  and  $o_b$ , respectively, are *similar* in respect of  $\alpha$  iff  $|o_a - o_b| \leq \alpha$ . This similarity measure is required by the online determination phase of our algorithm to select a subset of valid states.

#### 3.2.1 Offline Training Phase

During the offline phase, we train our algorithm by collecting a number of measurements at each state and store the data as histograms in a database. The optimal number of measurements is determined in Section 5.3.

#### 3.2.2 Online Determination Phase

In the online determination phase, the user takes one or more samples containing the signal strength measurements  $M_i$  and her orientation  $O_i$ ,  $i = 1, \dots, q$ . Our algorithm determines, based on these data, an accurate position (or reference point)  $r^*$ .

For a given threshold  $\alpha$ , only the states that are similar to the user's current orientation are considered. For each reference point  $i$  and access point  $j$  these similar states are merged by combining the histograms. We calculate the mean  $\mu_{i,j}$  and standard deviation  $\sigma_{i,j}$  of this merged and normalized histogram and use these aggregated values in the subsequent steps.

The probability to obtain the measurements  $M$  at reference point  $i$  can be expressed by the conditional probability:

$$P(M|r_i) = \prod_{a=1}^k P(m_a|r_i)$$

where

$$\begin{aligned} P(m_a|r_i) &= P((p_a, id_a)|r_i) \\ &= \frac{G_{i,id_a}(p_a) + \beta}{\sum_{m=0}^{255} G_{i,id_a}(m) + \beta} \end{aligned}$$

and

$$G_{i,id_a}(p_a) = \int_{p_a-1/2}^{p_a+1/2} \frac{e^{-(x-\mu_{i,id_a})^2/(2\sigma_{i,id_a}^2)}}{\sigma_{i,id_a}\sqrt{2\pi}} dx$$

$G_{i,id_a}(p_a)$  represents the probability to measure the signal strength value  $p_a$  of access point  $id_a$  at reference point  $i$ . This probability is calculated based on the distribution function determined by the average  $\mu_{i,id_a}$  and standard deviation  $\sigma_{i,id_a}$ . Furthermore,  $\beta$  is a small value that artificially adds noise to the probability calculation. During our experiments we set  $\beta = 10^{-44}$ .

These conditional probabilities are used to update the probability vector  $\vec{\pi}$  by applying Bayes' Rule [3]

$$\pi'_i = \frac{\pi_i * P(M|r_i)}{\sum_{j=1}^n \pi_j * P(M|r_j)}$$

for each reference points  $i$ .

Our analysis (see Section 5.2) shows that there is often more than one suitable candidate for a set of reference points. Surrounding reference points are usually considered highly probable if the user is located in the middle of these points. Therefore, the  $k$  most likely reference points are averaged to estimate the users' position  $r^*$  by calculating

$$r^* = \frac{1}{k} * \sum_{i=0}^k r_{max(i,\vec{\pi})}$$

where  $max(i, \vec{\pi})$  defines the  $i^{th}$  largest value of  $\vec{\pi}$ .

## 4. EXPERIMENTAL ENVIRONMENT

The subsequent subsections briefly describe our experimental environment.

### 4.1 Local Test Environment

We deployed our positioning system within the hallways of the A5, 6 building on the campus of the University of Mannheim. The operation area is nearly 15 meters in width and 36 meters in length, covering an area of approximately 312 square meters. The ground plan of the testing area is visualized in Figure 3. The large hallway in the left part of the map is connected by two narrow hallways that are separated by rooms such as archives and a kitchen.

We labeled the ground plan with markers which depict the grid of the reference points (orange dots) and the online measurement points (green dots). Additionally, the access points are marked by red squares.

### 4.2 Hardware

The test environment is equipped with five Linksys / Cisco WRT54GS and four Lancom L-54g access points. All access points support 802.11b and 802.11g. One Lancom and all Linksys access points are located on the same floor as our testing area, whereas three Lancom access point are located in other places inside the building. The exact position of the access points located inside the testing area is marked with red squares in Figure 3.

On the client, we used a Lucent Orinco Silver PCMCIA network card supporting 802.11b. We collected the signal strength samples on an IBM Thinkpad R51 running Linux 2.6.13 kernel and Wireless Tools 28pre.

To obtain the orientation of a user we used the Silicon Laboratories C8051F350 Digital Compass Reference Design Board. This device provides a USB-to-Serial bridge to access the data and can be powered by the USB electricity supply. We calibrated the compass in the middle of the operation area. In a closer area around the calibration point we measured a variation of  $1^\circ$ . However, variations up to  $23^\circ$  were rarely detected at a few points of the testing area. These measurement errors occurred always close to electromagnetic objects such as high voltage power lines and electronic devices.

### 4.3 Data Collection

The grid of reference points applied to the operation area includes 166 points with a spacing of 1 meter (see the orange dots in Figure 3). During the offline phase, the signal strength was measured at reference points for different orientations. We collected 110 signal strength measurements at each reference point and for each orientation. This leads to 146,080 measurements for the offline phase. We spent over 10 hours to collect all the data, however, we want to point out that for a productive deployment of this positioning system 20 signal strength measurements will be sufficient (see Section 5.3), cutting down the expenditure of time to less than 2 hours.

We randomly selected 60 coordinates and orientations for the online phase. The only condition to select a point inside the testing area as an online set point is that it is surrounded by four reference points. Again, we collected 110 signal strength measurements for each online set point, leading to 6,600 measurements in total. In Figure 3 the online set points are marked by green dots.

## 5. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of each element of our algorithm and discuss the impact of specific parameters on the performance of our positioning system. Afterwards, we compare the performance of our system with the performance of the RADAR approach.

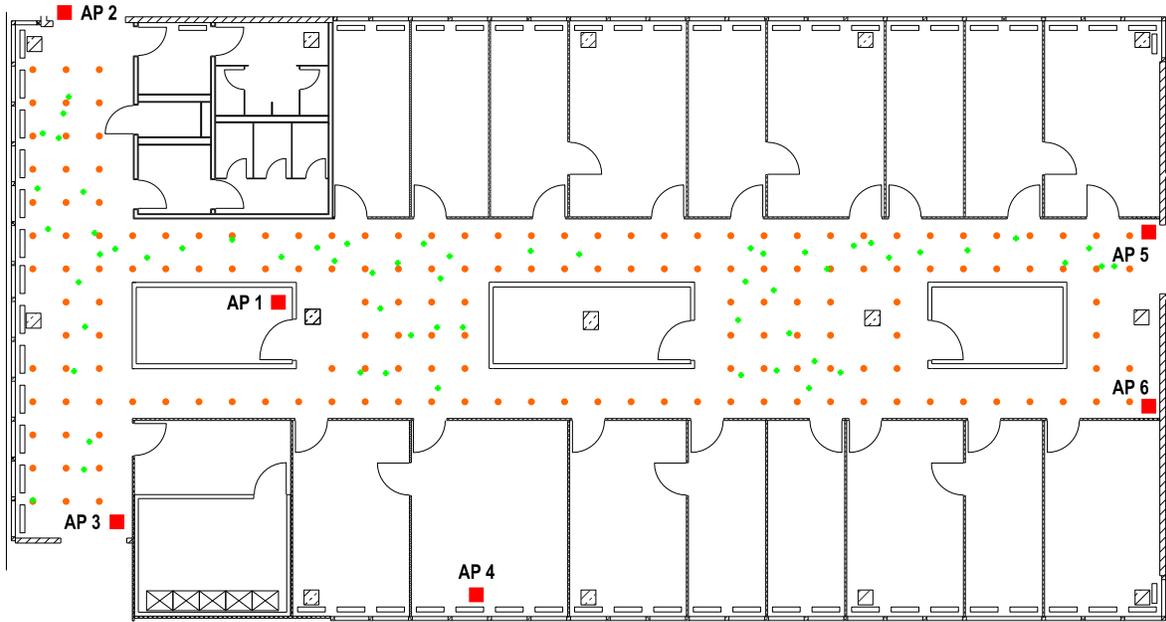
We developed a software application to analyze the performance of various positioning algorithms in different environments. The software logs every action and state (e.g. the signal strength) during the offline and online phase so that we can replay the scenario, and that we are able to change particular parameters we want to investigate. We used this software to simulate the behavior of the positioning algorithms.

To measure the performance of different positioning approaches we compute a cumulative distribution function (CDF) of the error distances between the estimated positions and the real positions of the user. Additionally, we calculate the average error distance.

### 5.1 Analysis of the Orientation

Our algorithm has two parameters that control the compass-specific behavior: The *number of orientations* considered during the offline phase for each reference point and the *angle  $\alpha$*  which determines similar states during the online phase. In the subsequent paragraphs, we study the effect of these parameters.

In our first experiment we show that the performance of our algorithms increases if signal strength distributions are sampled during



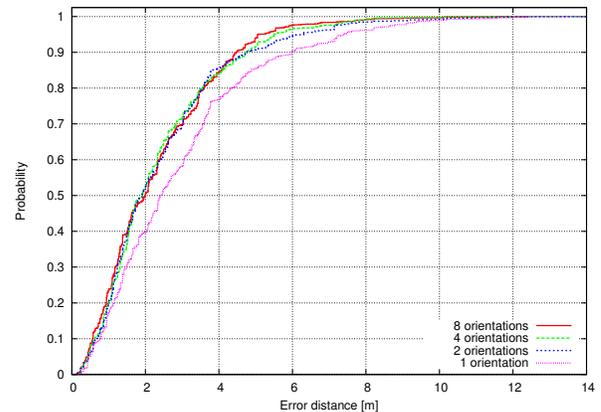
**Figure 3: Ground plan of the testing area. The orange dots represent the offline reference points and the green dots show the randomly selected online measurement points. The access points are marked by red squares.**

the offline phase for an increasing number of orientations. We selected four scenarios: In the first scenario only the signal strength distribution of one orientation is considered ( $0^\circ$ ) at each reference point. The second scenario takes two orientations into account ( $0^\circ$  and  $180^\circ$ ). Additionally, the signal strength distributions of the orientations at  $90^\circ$  and  $270^\circ$  are used in the third scenario and finally, the signal strength distributions of eight orientations are analyzed ( $0^\circ, 45^\circ, \dots, 315^\circ$ ).

For each scenario, to build the training dataset we randomly select 20 of the 110 already collected measurements for each reference point and each orientation. During the online phase, we randomly pick three of the 110 collected samples for each online set point. We set  $\alpha = 185^\circ$ , so that our algorithm is even able to proceed in the single orientation case. Furthermore, we disable the averaging part of our algorithm by setting  $k = 1$  so that the pure performance of orientation part of the algorithm is visible. We repeat the whole procedure 1000 times to achieve stable results.

Figure 4 shows the CDF of the error distance for these four scenarios (we cutted error distances larger than 14 meters to emphasize on the performance distinction of the different scenarios). The average error distance drops from 2.95 meters in the single orientation case down to 2.31 meters in the eight orientations scenario, resulting in an accuracy improvement of 22 percent. Furthermore, we see from the figure that each additional orientation leads to a diminishing marginal utility. For instance, the average error distance drops more than half a meter if a second orientation is added compared to an improvement of 4 centimeters in case four extra orientations are added to the four orientations scenario.

In a next step, we investigate the effect of angle  $\alpha$  on the error in the eight orientations scenario. We repeat the evaluation procedure as mentioned above, however, this time we change the value of  $\alpha$  stepwise from 25 to 100 degrees.



**Figure 4: Relation between number of orientations and error distance.**

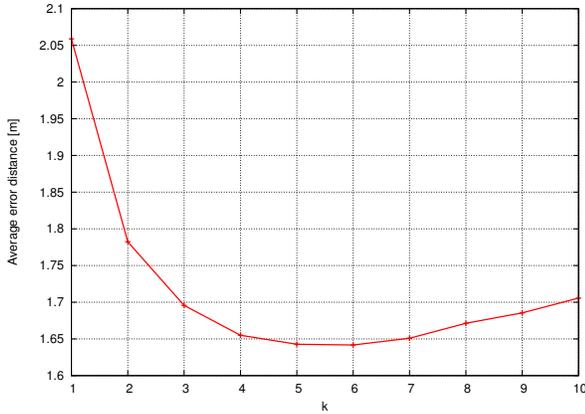
Our experimental analysis shows that the system achieves a minimal error distance for  $\alpha = 69^\circ$ . This can be explained by the fact that for small values of  $\alpha$  only one signal strength distribution is selected for each reference point. For instance, if the user's orientation is  $60^\circ$  and  $\alpha = 25^\circ$ , only the states with an orientation of  $45^\circ$  are considered, whereas a better approximation would be a combination of the signal strength distributions of the  $45^\circ$  and  $90^\circ$  orientations. The accuracy degrades rapidly for large values of  $\alpha$  because signal strength distributions for opposite orientations are also merged into the reference signal strength distribution, thereby corrupting the orientation specific characteristic of the signal strength distribution.

For  $\alpha = 69^\circ$  the average error drops to approximately 2.05 meters, improving the accuracy by 12 percent compared to  $\alpha = 185^\circ$ . To sum up, the usage of the compass-related components of our

algorithms reduces the average error distance to less than 2.05 meters, which corresponds to an accuracy improvement of more than 31 percent.

## 5.2 Averaging the Positions of the Most Probable Reference Points

In contrast to the previous section where we considered the most probable reference point as the position estimate, we now consider the average of the  $k$  most probable reference points as the user's position. In general, the user is surrounded by up to four reference points whose averaged coordinates represent a good approximation of the user's position.



**Figure 5: Impact of the averaged number of reference points ( $k$ ) on the positioning accuracy.**

We run an experimental analysis to find an optimal value of  $k$ . For this, we repeat the same experiment as mentioned above, but change the value of  $k$  from 1 to 10. As seen in Figure 5, the average error distance is minimal for  $k = 5$ . Smaller values of  $k$  increase the average error distance up to half a meter, where the error distance slowly increases for larger values of  $k$ . With a larger number of averaged reference points more distant points are selected resulting in averaged coordinate that is moving away from the real position of the user.

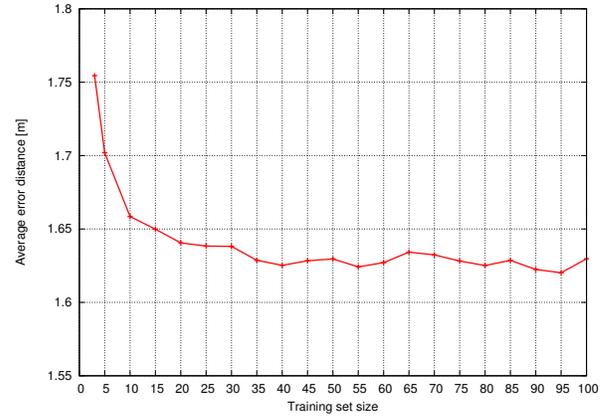
During our evaluation process, we also investigated other averaging algorithms (e.g. a weight-based averaging approach) but we could not find an algorithm that outperforms our simple averaging algorithm. The reason is that probability distributions differ significantly between different position requests and even a highly likely reference point might be over half a meter away from the user's real position. Hence, a simple averaging with surrounding reference points increases the accuracy.

From these results we see that the average error distance is reduced by more than half a meter due to the averaging.

## 5.3 Determination of the Training Set Size

The size of the training set is a critical parameter for the performance of the positioning algorithm. However, it determines the lower bound on the time needed to collect data for the fingerprinting database. It takes 250 ms to collect data for a single measurement with our network card. A reduction of the number of measurements required by the algorithm reduces the hours of work to collect data for the whole operation area significantly. On the other

hand, the collected signal strength distributions should be stable enough so that the positioning algorithm is able to accurately estimate the position of the user. We vary the training set size  $s$  between 3 and 100 to analyze how our algorithm performs with small training set sizes. For each number of  $s$  we randomly chose  $s$  readings out of the 110 signal strength measurements already collected in the data collection process and build the fingerprinting database. In the online phase we set  $\alpha = 69^\circ$ ,  $k = 5$ , and randomly select three measurements for each online set point as input for our positioning algorithm. We performed this experiment 1000 times for each value of  $s$ . The average error distance for each value of  $s$  is shown in Figure 6.



**Figure 6: Training set size versus average error distance.**

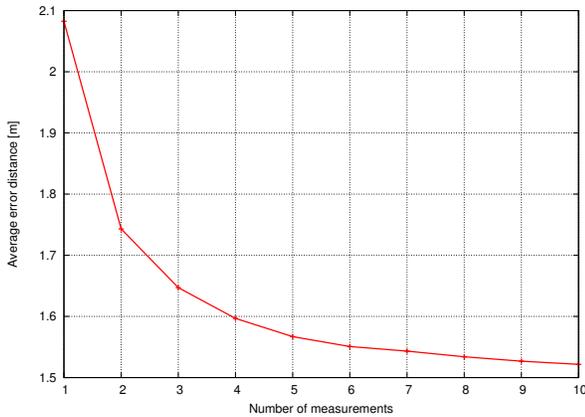
The graph shows that the accuracy increases more than 10 centimeters if the training set size is increased from 3 to 20 measurements. However, a further extension of the training set size only slightly improves the accuracy. Due to minor variations the average error distance oscillates around 1.625 meters for training set sizes larger than 30.

We have chosen a training set size of 20 for most of our experiments because a further extension barely improves the performance of our algorithm. Furthermore, the results in this paper should be easily achievable in productive environments where it is critical to minimize the time required to build the fingerprinting database.

## 5.4 Determination of the Online Set Size

In the following, we analyze the performance of our algorithm as we vary the number of measurements from which the positioning algorithm infers the user's position. Our hardware requires 250 ms to collect data for one measurement; the fewer measurements required to compute an accurate position estimate, the faster a position estimate is available. We repeat the same experiment as mentioned above, but this time we vary the number of measurements from 1 to 10 before a position estimate is finally computed. We perform this procedure 1000 times for each number of measurements. Our results are summarized in Figure 7.

The results show that an average error distance of less than 2.1 meters is achievable even if just one measurement is used. A second measurement decreases the error distance more than 25 centimeters (this corresponds to more than 17 percent). A third measurement improves the accuracy more than 6 percent or in other words nearly 10 centimeters. As we see from the graph any additional measurement further improves the accuracy, however, with a diminishing

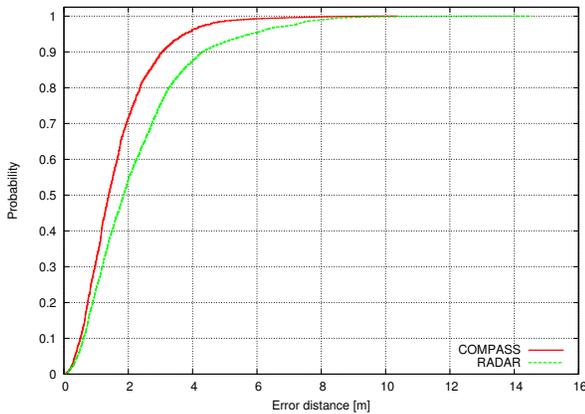


**Figure 7: Average error distance depending on the number of online measurements.**

marginal utility. In most of our experiments we use three measurements to compute a position estimate because three measurements prove to be a reasonable trade-off between accuracy and time.

## 5.5 Performance Comparison

In the previous sections, we studied the impact of the compass first, and separately investigated the effects of the different parameters afterwards. In this section, we compare the performance of our approach with the performance of the RADAR system [2]. We have chosen the *Multiple Nearest Neighbors* algorithm as proposed by Bahl et al. because this algorithm achieved the best performance in their paper.



**Figure 8: CDF of the performance of the COMPASS and RADAR system.**

To create equal conditions for both algorithms we use the data gathered from our experimental environment for the training and the position determination phase. We set the offline set size to 20 and randomly choose three readings for each online set point to compute the position estimates. This procedure is repeated 1000 times to receive stable results.

Figure 8 shows the CDF for both algorithms. The COMPASS system outperforms the RADAR approach: our algorithm is not only more accurate (e.g. the error distance in the COMPASS system is less than two meters in 70 percent of all measurements compared

to only 55 percent in case of RADAR); it also produces a smaller worst case error distance (less than 11 meters in contrast to more than 15 meters). On average, our system achieves an error distance of 1.65 meters in comparison to 2.26 meters in case RADAR is used. These results show the power of our new approach.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we presented COMPASS, a positioning system based on Wireless LAN and compasses. We have shown that the recognition of the user’s orientation can dramatically improve the accuracy of the positioning system. We used a probabilistic algorithm to compute a probability distribution over the reference points and presented a simple averaging algorithm to further improve the performance of the system.

We have deployed our positioning system in a building of the University of Mannheim to evaluate it in a real-world environment. In our experiments, we achieved an average error distance of less than 1.65 meters. We also investigated how different parameters of the algorithms influence the overall performance of the system. Finally, we compared the performance of our system with the performance of the RADAR system.

In the near future, we are going to work on a compass-enabled tracking system to track users while they are walking around. We believe, the user’s orientation will also help to improve the performance of this system because orientation is a well suited indicator to recognize turnarounds of users.

Based on our system, location-based services as well as context-aware applications that require highly accurate position information can now be build. Within our research group at the University of Mannheim, we are currently working on such novel applications.

## Acknowledgements

We would like to thank Alexander Biskop, Andreas Färber, and Daniel Kölsch for their great support collecting data for the experiments. Furthermore, the authors acknowledge the support granted by the *Deutschen Forschungsgemeinschaft* (DFG).

## 7. REFERENCES

- [1] P. Bahl and V. N. Padmanabhan. Enhancements of the RADAR User Location and Tracking System. Technical Report MSR-TR-2000-12, Microsoft Research, Microsoft Corporation One Microsoft Way Redmond, WA 98052, February 2000.
- [2] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proceedings of the 19th International Conference on Computer Communications (Infocom 2000)*, volume 2, pages 775–784, Tel Aviv, March 2000. IEEE.
- [3] T. Bayes. An Essay towards solving a Problem in the Doctrine of Chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418, November 1763.
- [4] B. Brumitt, B. Meyers, J. Krumm, A. Kern, and S. Shafer. EasyLiving: Technologies for Intelligent Environments. In *Second International Symposium on Handheld and Ubiquitous Computing*, pages 12–27, September 2000.

- [5] P. Castro, P. Chiu, T. Kremenek, and R. R. Muntz. A Probabilistic Room Location Service for Wireless Networked Environments. In *Proceedings of the 3rd International Conference on Ubiquitous Computing (UbiComp 2001)*, pages 18–34, Atlanta, Georgia, USA, 2001. Springer-Verlag, London, UK.
- [6] P. Castro and R. Muntz. Managing Context Data for Smart Spaces. *IEEE Personal Communications*, pages 44–46, October 2000.
- [7] G. Chen and D. Kotz. A Survey of Context-Aware Mobile Computing Research. Technical Report TR2000-381, Dartmouth College, Hanover, NH, USA, November 2000.
- [8] J. Hightower, R. Want, and G. Borriello. SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength. UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA, February 2000.
- [9] Institute for Electrical and Electronics Engineers, Inc. ANSI/IEEE Standard 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. <http://standards.ieee.org/getieee802/>, 1999.
- [10] E. Kaplan and C. Hegarty, editors. *Understanding GPS: Principles and Applications*. Artech House, Incorporated, second edition, December 2005.
- [11] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket Location-Support System. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom 2000)*, pages 32–43. ACM Press, 2000.
- [12] N. B. Priyantha, A. Miu, H. Balakrishnan, and S. Teller. The Cricket Compass for Context-Aware Mobile Applications. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom 2001)*, pages 1–14. ACM Press, 2001.
- [13] A. Smailagic, D. P. Siewiorek, J. Anhalt, D. Kogan, and Y. Wang. Location Sensing and Privacy in a Context Aware Computing Environment. *Wireless Communications, IEEE*, 9:10–17, October 2002.
- [14] J. Small, A. Smailagic, and D. P. Siewiorek. Determining User Location For Context Aware Computing Through the Use of a Wireless LAN Infrastructure. December 2000.
- [15] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. *ACM Transactions on Information Systems*, 10(1):91–102, January 1992.
- [16] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4:42–47, October 1997.
- [17] M. Youssef and A. Agrawala. On the Optimality of WLAN Location Determination Systems. Technical Report UMIACS-TR 2003-29 and CS-TR 4459, University of Maryland, College Park, March 2003.
- [18] M. Youssef and A. Agrawala. On the Optimality of WLAN Location Determination Systems. In *Communication Networks and Distributed Systems Modeling and Simulation Conference*, January 2004.
- [19] M. Youssef and A. Agrawala. The Horus WLAN Location Determination System. In *the 3rd international conference on Mobile Systems, Applications, and Services (Mobisys)*, pages 205–218, 2005.
- [20] M. Youssef, A. Agrawala, and A. U. Shankar. WLAN Location Determination via Clustering and Probability Distributions. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, page 143, Fort Worth, Texas, March 2003. IEEE Computer Society.