2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 30 Sept. - 3 Oct. 2019, Pisa, Italy

Multi-Modal Probabilistic Indoor Localization on a Smartphone

Frederike Dümbgen† Cynthia Oeschger‡† Mihailo Kolundžija† Adam Scholefield†
Emmanuel Girardin‡ Johan Leuenberger* Serge Ayer‡

† *School of Computer and Communication Sciences,*
Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland
‡ *School of Engineering and Architecture Fribourg (HEIA-FR),*
University of Applied Sciences and Arts Western Switzerland, CH-1700 Fribourg, Switzerland
* *Vidinoti, Passage du Cardinal 1, CH-1700 Fribourg, Switzerland*

***Abstract*—The satellite-based Global Positioning System (GPS) provides robust localization on smartphones outdoors. In indoor environments, however, no system is close to achieving a similar level of ubiquity, with existing solutions offering different trade-offs in terms of accuracy, robustness and cost.**

In this paper, we develop a multi-modal positioning system, targeted at smartphones, which aims to get the best out of each of its constituent modalities. More precisely, we combine Bluetooth low energy (BLE) beacons, round-trip-time (RTT) enabled WiFi access points and the smartphone's inertial measurement unit (IMU) to provide a cheap robust localization system that, unlike fingerprinting methods, requires no pre-training. To do this, we use a probabilistic algorithm based on a conditional random field (CRF). We show how to incorporate sparse visual information to improve the accuracy of our system, using pose estimation from pre-scanned visual landmarks, to calibrate the system online.

Our method achieves an accuracy of around 2 meters on two realistic datasets, outperforming other distance-based localization approaches. We also compare our approach with an ultra-wideband (UWB) system. While we do not match the performance of UWB, our system is cheap, smartphone compatible and provides satisfactory performance for many applications.

***Index Terms*—Smartphone localization, multi-modal systems, RTT-based ranging, conditional random fields**

I. Introduction

Despite decades of progress in indoor localization, no technology has achieved widespread use. This can be largely attributed to the fact that existing systems suffer from one or more of the following drawbacks: they

– require additional expensive dedicated infrastructure;
– require significant training/calibration;
– are not robust to changes of the environment;
– require the user to actively scan the environment; or
– are not smartphone-compatible.

For example, ultra-wideband (UWB) signals can provide accurate localization over a wide range of distances with a quick and stable system setup [14]. However, this modality requires dedicated infrastructure, and there is no evidence that it will become smartphone compatible in the near future.

Other solutions require only a smartphone and no infrastructure. For instance, cameras combined with inertial measurement units (IMU) can be used to scan the environment and, using methods called simultaneous localization and mapping (SLAM), infer an accurate 3D map and the user's location with centimeter-level precision [33]. However, it is not always convenient for the user to actively scan the surroundings; an ideal indoor localization system would be passive – meaning that it requires no active participation of the user. Most visual systems also require the environment to possess uniquely identifiable visual features, which is not always the case, in particular in exhibition rooms, lecture halls, and sterile environments such as hospitals. Finally, visual systems tend to break down when the environment is changed, causing the recorded maps to not match the reality anymore.

This work was supported by InnoSuisse grant number 25579.1 PFES-ES

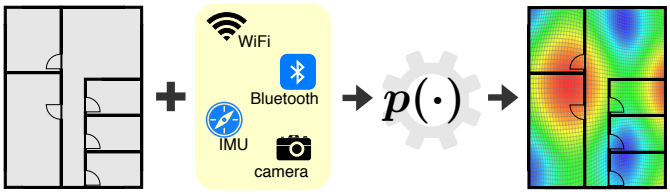


Figure 1. Overview of the proposed framework. We merge measurements from commodity systems of different types to localize a smartphone. Our solution requires no offline calibration or fingerprinting phase, uses little computing power and is entirely smartphone compatible.

In order to create a passive localization system, a third category of approaches use smartphone-compatible measurements, for example radio-frequency (RF) signals such as Bluetooth low energy (BLE) and WiFi. Since WiFi base stations are already prevalent and BLE beacons are inexpensive, these approaches can be deployed with reasonable infrastructure costs. In indoor environments, RF signals are heavily affected by multi-path, shadowing and fading [13]. Furthermore, synchronization issues, unknown latency times and unknown variations in antenna characteristics induce important measurement errors [26]. Therefore, the best-performing algorithms leveraging such signals use a "fingerprinting" phase, in which the signal's characteristics are extracted at a finite set of positions in the region of interest. These fingerprints are able to capture stationary non-line-of-sight (NLOS) and multipath effects. In the online localization phase, the recorded signal is matched with the fingerprint database, and—after possible combination with IMU data and map constraints—a position estimate is inferred [10], [29], [34].

In this paper, we present an indoor localization system that is smartphone compatible, easy to install and robust to environmental changes. The proposed system combines BLE beacons, WiFi hotspots, the smartphone's IMU data, and visual fingerprints recorded by its camera. The visual information is sparse in both time and space. In particular, we assume that certain visual landmarks—such as artworks in a museum, emergency exit signs, navigation maps, etc.—do not change over a foreseeable time and these landmarks can be "scanned" by the user at a handful of time points. At these moments, we obtain centimeter-level localization, which allows us to calibrate offsets in the other measurements.

For the localization algorithm, we use a conditional random field (CRF) to efficiently combine the different measurement modalities. This allows us to account for the expected relative accuracy of the different systems, and to obtain a distribution rather than a single estimate of the user's position.

In sum, the proposed framework provides robust localization that

does not break down when measurements or entire modalities are missing or unreliable. Furthermore, the localization is passive for the vast majority of the time. In terms of accuracy, we achieve mean localization errors of around two meters in a challenging indoor environment, without the need for any fingerprinting or prior calibration.

II. Related work

A. Measurement modalities

While an abundance of measurement modalities have been proposed for indoor localization [22], we focus on modalities that are smartphone compatible. Among these systems, WiFi, Bluetooth, IMU and images are most commonly available.

For the RF signals, the coarsely quantized and often unreliable Received Signal Strength Indicator (RSSI) is most commonly used. With the WiFi Round Trip Time (RTT) feature, known as IEEE 802.11mc FTM, more accurate distance measurements are now available [12]. With the publication of the new Bluetooth 5.1 standard, it is also expected that angular information will soon become more widespread [30]. Compared to WiFi, Bluetooth uses less power and is cheaper to deploy, however it has a shorter range of operation (1-5 meters compared to 50-100 meters for WiFi) and is typically only used for proximity detection.

It is widely known that the quality of RF signals degrades in challenging indoor environments. Whilst it is difficult to reduce this effect, estimating the accuracy of each measurement and processing this appropriately can significantly improve performance. In this regard, Xaio *et al.* [36] extract different RSSI-based features to identify NLOS conditions in Wifi-based localization and Li *et al.* [16] leverage the observation that NLOS signals have a higher variability than LOS signals. Bahillo *et al.* [3] use distances inferred from RTT measurements as constraints to improve RSSI-based distance estimates. Li *et al.* [15] propose new features extracted from the channel state information (CSI) which can be used to differentiate line-of-sight (LOS) from NLOS measurements. While CSI can greatly improve performance, it should be noted that, currently, it is not readily available on smartphones. In addition, most methods distinguishing LOS from NLOS rely on a training phase and are thus not suitable if the aim is to have little setup and calibration time.

Almost all smartphone-based systems leverage IMU data to improve performance. Since IMUs provide an estimate of the device's relative movement, both in terms of travelled distance and movement direction, integration can be applied to obtain position estimates; however, double integration of accelerometer data is notoriously inaccurate and step detection combined with estimated step length and direction is more commonly used [10], [21], [37].

On the vision side, object recognition and localization from visual features are well-studied topics and a review of these is beyond the scope of this paper. Therefore, we refer the interested reader to standard textbooks on multi-view geometry and computer vision [11], [18], [23].

B. Localization algorithms

Localization algorithms can be very broadly split into two categories: those that predominantly use geometric information—such as lateration—and those that learn how signals behave in the environment—such as fingerprinting.

Geometric methods have existed ab incunabulis and include standard lateration and angulation techniques. For multilateration, optimal and efficient solutions exist [5] and angulation has many parallels with the well-studied multi-view geometry.

In addition to methods that assume fixed anchors at known positions, geometric methods also exist for ad-hoc sensor networks [8], where a number of nodes can be simultaneously localized relative to each other. When distance measurements are obtained between the sensor nodes, Euclidean distance matrices (EDMs) provide a tool to denoise and complete missing measurements [9]. These ideas have also been extended to include both distance and angle measurements [2], [19].

Note that, in practical systems, these geometric methods are usually embedded in a more complex framework, often based on Kalman or particle filters, which produces smoother estimates across time and allows relative measurements from IMUs or similar modules to be incorporated [31].

For learning-based approaches, fingerprinting has seen widespread use. For example, Guimarães *et al.* [10] use WiFi fingerprint maps for coarse location estimation, and refine the estimate using magnetic fingerprints and IMU measurements. Shu *et al.* [29] use the same measurement modalities, but combine them in a particle filter. Since their system uses a bidirectional dynamic time warping method, it operates with a delay. Xiao *et al.* [34], on the other hand, propose a real-time localization system using similar modalities. Their probabilistic framework introduces features for WiFi fingerprints and IMU measurements to yield a probability distribution for the device's position.

The above approaches rely on fingerprint databases, which have to be collected and updated regularly in time-consuming offline calibration phases. Since the creation of these databases is the major bottleneck in terms of setup efforts, there have been multiple lines of research proposing to speed up this process, or to crowd-source the fingerprint creation and maintenance over time [24], [25], [35], [37]. The sensitivity of fingerprint maps to short-time variations in the environment however stays an unavoidable shortcoming of these methods. For instance, it was reported that a moving elevator was the main cause of fingerprint disturbances in the 2017 Microsoft Indoor Localization Competition [17]

The method proposed in this paper does not rely on fingerprint maps and therefore avoids the lengthy setup time. The mathematical framework is based on the conditional random field formulation provided by Xiao *et al.* [34], and extended to include both RSSI-based and RTT-based distance measurements. In [34], global orientation offsets are calibrated by simultaneously inferring the position over multiple time steps, which can introduce unwanted latency in the system. In contrast, we achieve calibration by using sparse visual features and therefore do not introduce delays.

III. Problem setup

Our goal is to find the location of a device at time instances $\{t_j\}_{j=1}^{T}$; we denote this estimate, at each time instant, by $\mathbf{y}[j] \in \mathbb{R}^3$. We assume that there are M_W WiFi access points and M_B Bluetooth beacons, from which we get distance and path loss measurements, respectively. From the IMU, we obtain an estimate of the device's orientation $\theta_{\text{IMU}} \in [0, 2\pi]$ and of the distance travelled since the last processing time, $l_{\text{IMU}} \in \mathbb{R}^+$, respectively. Finally, we sometimes obtain a pose estimate from visual scanning, denoted by $(\mathbf{v}, \theta_v) \in \mathbb{R}^3 \times [0, 2\pi]$. Note that we are only interested in the device's orientation in the x-y plane (its rotation around the z-axis), as we are treating objects moving mostly in a horizontal plane, in particular walking pedestrians.

To simplify notation, the observations at a given discrete time instant t_j are combined into a vector-valued observation variable Z_j. For example, if we have complete measurements at time instant t_j, then Z_j takes the form

$$Z_j = [d_1, \ldots, d_{M_W}, P_1, \ldots, P_{M_B}, \theta_{\text{IMU}}, l_{\text{IMU}}, \mathbf{v}^T, \theta_v]^T, \quad (1)$$

where $d_i \in \mathbb{R}^+, i = 1 \ldots M_W$ are the distance measurements from WiFi access points, and $P_k, k = 1 \ldots M_B$ are the signal strength measurements from the Bluetooth beacons. Each measurement is given by the median over a one-second time window. In practice we will only measure a subset of all modalities at each time instant.

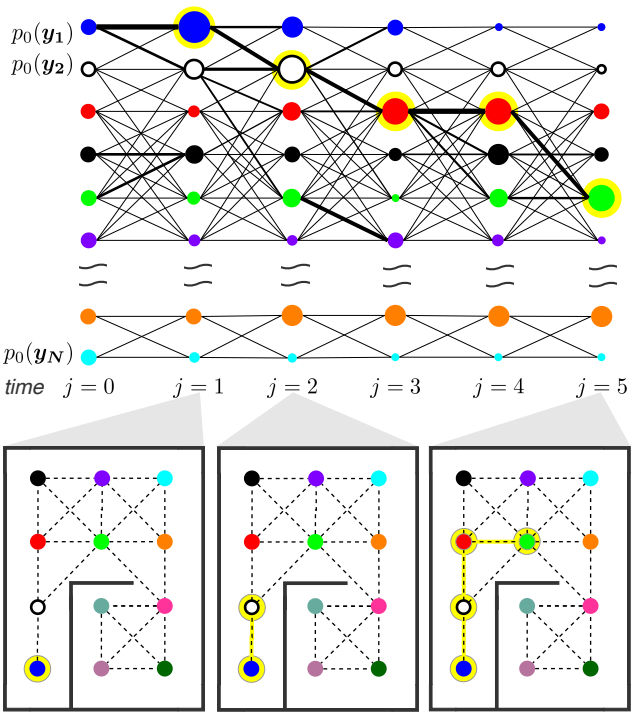


Figure 2. Visualization of the proposed CRF algorithm. At each time index j, the probability of each point (proportional to its size) is computed by considering the connected grid points and the transition probabilities (proportional to the line thickness). After each step, we pick the point with highest probability as our position estimate (highlighted in yellow).

Finally, we assume that the locations of Bluetooth beacons and WiFi access points, denoted by $\mathbf{a_m} \in \mathbb{R}^3, m = 1 \dots M_B + M_W$, remain unchanged throughout the experiment.

IV. Conditional Random Fields

We uniformly discretize the 3D domain, and denote the position of grid point $i = 1 \dots N$ by $\mathbf{y_i} \in \mathbb{R}^3$. In addition, Y_j denotes the latent or state variable at time j and in our framework contains the position estimate of the device.

At any time j, the goal is to maximize the conditional probability $p(Y|Z)$ of the state variables $Y = \{Y_1, \dots, Y_j\}$ given the observations $Z = \{Z_1, \dots, Z_j\}$.

Similar to [34], we use linear chain CRFs where the conditional probability function of states given observations can be represented by a product of potential functions:

$$p(Y|Z) \propto \prod_{j=1}^{T} \Psi(Y_j, Y_{j-1}, Z_j). \tag{2}$$

The potential functions $\Psi(Y_{j-1}, Y_j, Z_j)$ are, in turn, composed of $K > 0$ feature functions f_k:

$$\Psi(Y_j, Y_{j-1}, Z_j) = \exp\left(\sum_{k=1}^{K} \lambda_k f_k(Y_j, Y_{j-1}, Z_j)\right). \tag{3}$$

Here, each feature function characterizes the likelihood of transition from state Y_{j-1} to state Y_j given the observation Z_j. We use the parameters $\lambda_k \in \{0, 1\}$ to exclude certain feature types, and tune the relative importance of features through their own parameters, as outlined in the next section.

The location of the device is found sequentially, one step at a time, using the Viterbi algorithm. At any time step j, the algorithm computes, for all possible grid points $\mathbf{y_i}$, the probability $p_j(\mathbf{y_i}|Z_j)$ of the most likely sequence of states $(Y_1, \dots, Y_j)$ such that $Y_j = \mathbf{y_i}$. The main step of the Viterbi algorithm can be written as

$$p_j(\mathbf{y_i}|Z_j) = \max_{\mathbf{y_l}} p_{j-1}(\mathbf{y_l}|Z_{j-1}) \Psi(\mathbf{y_i}, \mathbf{y_l}, Z_j)\,. \tag{4}$$

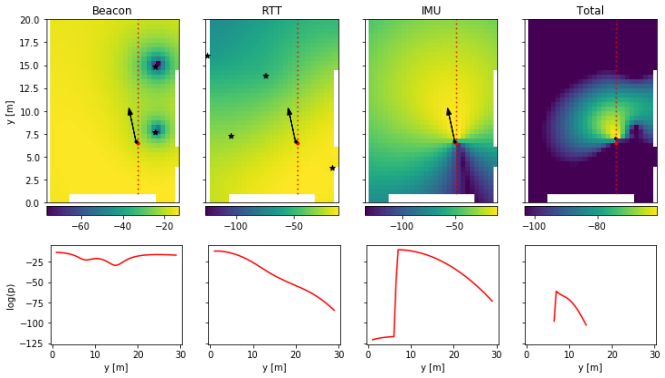


Figure 3. Probability maps inferred at one time instance for the different systems. The standard deviations are 1 dBm for Beacon RSSI, 2 m for IMU distance and 0.1 radians for IMU angle (the sum is shown), and 1 m for RTT distance measurements. The bottom row shows the 1D-cut of the distribution along the red dotted line. Zero probabilities are omitted for better readability.

The algorithm is visualized in Figure 2. Note that, as opposed to the original CRF formulation [34], our position estimate at time j depends only on the most recent measurements at time Z_j. This avoids introducing an unwanted delay in the system, and the smoothness of the solution is not compromised thanks to relative distance and angle measurements from the IMU system.

A. Modeling feature functions

The main challenge of the proposed framework is the design of transition probabilities $\Psi(Y_j, Y_{j-1}, Z_j)$. In the following, we introduce features based on WiFi RTT, Bluetooth RSSI and IMU data, and outline how to use visual measurements. All features are normalized such that their sum on the considered grid equals one before feeding them into (3).

1) RTT feature function: For the RTT feature function, we assume that distance measurements to anchors follow a Gaussian distribution. Namely, the measured distance d_m to anchor m follows the distribution $\mathcal{N}(d_m^*, \sigma_m^2)$, where $d_m^* = \|\mathbf{a_m} - \mathbf{y_i}\|$ is the true distance (conditioned on the device being at $\mathbf{y_i}$) and σ_m^2 is the measurement variance. We also assume that the measurements to different anchors are independent. This leads to the probabilities

$$p(Z_j|\mathbf{y_i}) \propto \prod_{m=1}^{M_W} \frac{1}{\sigma_m\sqrt{2\pi}} \exp\left(-\frac{(d_m - d_m^*)^2}{2\sigma_m^2}\right), \tag{5}$$

from which we deduce the feature function

$$f_1(Y_j, Z_j) = \sum_{m=1}^{M_W} \ln\left(\frac{1}{\sigma_m\sqrt{2\pi}}\right) - \frac{(d_m - d_m^*)^2}{2\sigma_m^2}. \tag{6}$$

2) RSSI feature function: With measures of received signal strength, such as RSSI, the distance distribution is not the same as for RTT. Let P_m be the power of the signal received from anchor m. We assume that, due to noise and interference, P_m follows a Gaussian distribution $\mathcal{N}(P_m^*, \sigma_m^2)$, where P_m^* is the expected (or ground truth) power and σ_m^2 is the power variance, both expressed in dBm.

The expected received signal power, P_m^*, is related to the true distance from the anchor d_m^* via the log-distance path loss model expressed by

$$P_m^* = T_m - 10\,n \log_{10} d_m^*, \tag{7}$$

where T_m is a constant and n is the path loss exponent. The same path loss model can be used to compute the distance d_m based on the received signal's power with

$$d_m = 10^{(T_m - P_m)/10\,n}. \tag{8}$$

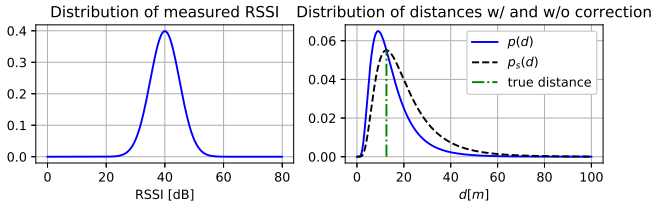


Figure 4. Illustration of how the normal distribution of measured RSSI values translates into a log-normal distribution of estimated distance, and how the latter looks after the correction given in (11).

Since P_m follows the normal distribution, the natural logarithm of the distance estimate d_m also follows a normal distribution, given by

$$p(\ln(d_m)\,|\mathbf{y_i}) \sim \mathcal{N}\left(S_m, \sigma^2_{m,d}\right), \tag{9}$$

where, to simplify notation, we introduced

$$S_m = \ln(10)\,\frac{T_m - P^*_m}{10\,n} \quad \text{and} \quad \sigma^2_{m,d} = \left(\frac{\ln(10)}{10\,n}\right)^2 \sigma^2_m.$$

Consequently, the estimated distance d_m conditioned on $Y_j = \mathbf{y_i}$ follows a log-normal distribution. Following the assumption that all anchor measurements at one time instant are independent, we can write

$$p(Z_j|\mathbf{y_i}) \propto \prod_{m=1}^{M_B} \frac{1}{d_m \sigma_{m,d}\sqrt{2\pi}} \exp\left(-\frac{(\log d_m - S_m)^2}{2\sigma^2_{m,d}}\right). \tag{10}$$

We note here that the sequential algorithm (4) chooses the state with highest posterior probability, at every step, or put differently, it finds the mode of the distribution $p(Y_j|Z_j)$. If we consider an RSSI measurement drawn from a normal distribution whose mean P^*_m equals the true power (without noise), then the true distance d^*_m equals the mean of the corresponding log-normal distribution of the distance. However, the mean distance does not correspond to the distance at which the PDF attains its maximum value.

In order to correct the discrepancy between the log-likelihood distribution and posterior maximization, we scale the PDF of the log-likelihood distribution by the distance and obtain

$$p_s(Z_j|\mathbf{y_i}) \propto \prod_{m=1}^{M_B} \frac{1}{\sigma_{m,d}\sqrt{2\pi}} \exp\left(-\frac{(\log d_m - S_m)^2}{2\sigma^2_{m,d}}\right). \tag{11}$$

By doing so, we get a probability density function $p_s(Z_j|\mathbf{y_i})$ whose mode equals the mean of the corresponding log-likelihood distribution $p(Z_j|\mathbf{y_i})$, which makes $p_s(Z_j|\mathbf{y_i})$ compatible with posterior maximization as a means to finding the position. This effect is visualized in Figure 4.

The corresponding feature function is then

$$f_2(Y_j, Z_j) = \sum_{m=1}^{M_B} \ln\left(\frac{1}{\sigma_{m,d}\sqrt{2\pi}}\right) - \frac{(\log d_m - S_m)^2}{2\sigma^2_{m,d}}. \tag{12}$$

3) IMU feature functions: We deduce a travelled distance l_{IMU} and direction θ_{IMU} from IMU measurements at time j as follows. First, we assume that from variations in the vertical acceleration we can identify step counts. We then calculate a step vector, obtained by averaging the product of an estimated step length with the direction for each step. To simplify, we fix the average step length throughout the experiments. Finally, we calculate the distance and orientation estimate as the norm and the angle of the averaged step vector.

Under a Gaussian noise assumption, the IMU features can be written as

$$f_3(Y_j, Y_{j-1}, Z_j) = \ln\left(\frac{1}{\sigma_l\sqrt{2\pi}}\right) - \frac{(l_{\text{IMU}} - \|\mathbf{y_i} - \mathbf{y}_k\|)^2}{2\sigma^2_l}, \tag{13}$$

$$f_4(Y_j, Y_{j-1}, Z_j) = \ln\left(\frac{1}{\sigma_\theta\sqrt{2\pi}}\right) - \frac{\Delta(\theta_{\text{IMU}}, \theta(\mathbf{y_i}, \mathbf{y}_k))^2}{2\sigma^2_\theta}, \tag{14}$$

where $\theta(\cdot)$ returns the angle between two states, and σ_θ, σ_l are the standard deviation of angle and distance measurements, respectively. The operator Δ returns the angular difference in the interval $[-\pi, \pi]$.

Experimental distributions for the four proposed feature functions, f_1 to f_4, are shown in Figure 3. To ensure a good balance, it is crucial to choose appropriate standard deviations for each of them. The choice of the standard deviations, shown in the Figure, is motivated in Section V-C.

4) Visual measurements: Our visual system, called Pixlive, uses AGAST, BRIEF and ORB [6], [20], [27], [28] to detect and create descriptors of feature points, which are compared to a database of known images. The system selects the best match and finally computes the homography and pose using multi-view geometry techniques [11]. We emphasize that these techniques are standard and not the focus of this paper.

Since the visual system provides centimeter-level accuracy, we treat measurements from it as "ground truth". Denoting the position estimate at time j by $\mathbf{v}$, we reset our probability map to a Gaussian distribution around the estimate, with a fixed low variance σ_v,

$$p(\mathbf{y_i}|Z_j) = \frac{1}{\sigma_v\sqrt{2\pi}} \exp\left(-\frac{\|\mathbf{y_i} - \mathbf{v}\|^2}{2\sigma^2_v}\right). \tag{15}$$

The variance should be chosen sufficiently small such that only few states around the visual estimate have a non-zero probability. For a 0.5 by 0.5 meter grid we found that σ_v=0.05 meters yielded this desired behavior.

B. Location inference

By applying (4) at time j, we obtain a probability map that contains the probability of the device being at each grid point. To obtain a single position estimate at each time, we pick the point of maximum probability, given by

$$\hat{\mathbf{y}}[j] = \underset{\mathbf{y}\in\{\mathbf{y}_i\}_{i=1}^N}{\arg\max}\; p_j(\mathbf{y}|Z_j). \tag{16}$$

In the pictorial example, shown in Figure 2, the obtained position estimates are highlighted in yellow.

We reduce the search space of the Viterbi algorithm (4) to grid points for which the probability at the previous time step $p_{j-1}(\mathbf{y_l}|Z_{j-1})$ was significant. With a threshold chosen low enough (in our case, 1e-10), this does not affect the result because these states are not realistic candidates, however it does speed up the position inference significantly.

An alternative approach to location inference is to recursively backtrack the position estimates. By keeping track of the most likely predecessor state (the *argmax* in (4)), we can reconstruct the sequence of states that lead to the current estimate. Experiments show that this approach can help to smooth the trajectory if one allows a small delay.

V. Processing and Calibration

A. Outlier rejection

As outlined in the previous section, we assume distance measurements from RTT anchors to be zero-mean Gaussian and Beacons to be log-normally distributed. In order to get closer to this assumption, we filter the raw measurement before feeding them into our algorithm.

The accuracy of raw BLE Beacon and WiFi RTT measurements are shown in Figure 5. The measurements are taken in a challenging environment with multi-path and shadowing. By plotting distance error vs. RSSI, we see that low RSSI measurements are correlated with high distance errors. For WiFi signals, which are generally stronger than BLE signals, the threshold at which the signal deteriorates is at around -65 dBm. For the lower-energy Bluetooth signals, it is at -90 dBm, which is close to the receiver sensitivity of standard BLE beacons. We found that rejecting measurements below this threshold was beneficial to the overall localization accuracy.

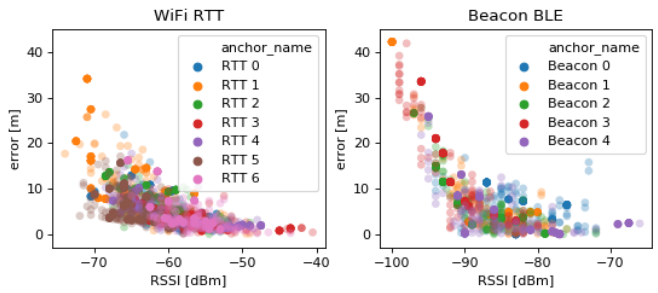


Figure 5. Accuracy of Bluetooth (left) and WiFi (right) anchors for the *zig zag* dataset. Plotted is the absolute error between real distance and measured distance on y, vs. the received signal strength indicator (RSSI) on x.

B. Online calibration

The visual system is not only used to reset the probability map from time to time, it is also exploited for online calibration of the unknown parameters of the different systems: the offset of RTT distance measurements (introduced primarily by multipath and unknown latency issues [26]), the transmit power of the Beacons, and the absolute orientation of the phone (IMU magnetometer tends to be very noisy and gyroscope measurements tend to drift). When a visual measurement is recorded, we record measurements for the following second, assuming the user is standing still during this time, and use the median of the recorded measurements to calibrate each modality. We use the thereby obtained offsets for each WiFi access point, and transmit powers for each Bluetooth beacon, to correct the obtained distance estimates in real-time.

C. Feature weights

The weights of the different features need to be chosen so that the features complement each other appropriately, and none of them dominate the final probability map. Inspecting Equations (6), (12) and (14), the main parameters to tune are the standard deviations of each modality. One can obtain estimates of standard deviations from each device experimentally. However, we found that finding one system-dependent standard deviation for each modality yielded sufficiently good results. The chosen standard deviations are shown along with the resulting probability maps for a sample datapoint in Figure 3. Note that the Beacon features get absorbed by the other two features, which is desired since we expect much lower accuracy from Beacons than from RTT or IMU.

As previously mentioned, when it comes to RSSI features, we assume that the received signal strength, expressed in dBm, follows

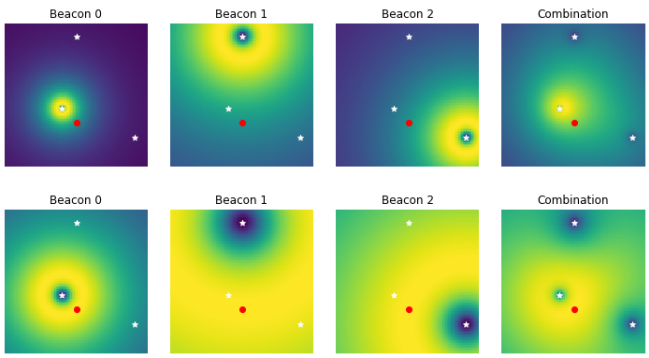
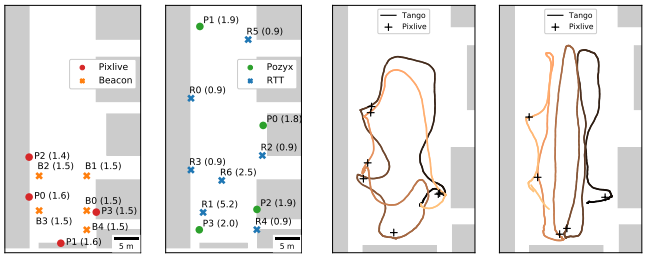


Figure 6. Lognormal feature for Bluetooth RSSI measurements without (top) and with (bottom) correction factor (11), on simulated data. Shown in white and red are anchors and the device positions, respectively. Note that the distribution flattens out for high distances (for instance for Beacon 1). With the correction, the maximum of the total distribution (right column) concentrates around the true position.

(a) (b) (c)

Figure 7. (a) Picture of the BC atrium at EPFL, the location of the experiments. (b-c) Two example images used as visual features for calibration.



(a) (b)

Figure 8. (a) Anchor layout. The height of each anchor in meters is included in parentheses. (b) Studied datasets with ground truth obtained from Tango. Sparse visual position estimates from Pixlive are also shown.

a normal distribution. This noise model choice has the convenient effect that distance measurements from close beacons are considered more accurate than measurements from far beacons. Intuitively speaking, constant variance across different signal strengths will not have the same effect on ranging quality. To give an example, if a 6 dB change amounts to doubling of the distance, the variation in received signal strength that amounts to 6 dB is not the same thing if the true distance is 1 meter or 10 meters. The simulated probability distributions plotted in Figure 6 obviate this intuition.

VI. Results

We show the effectiveness of the proposed method in two real-world experiments in a university building.

A. Environment of experiments

The area for experiments is a large, furnished hall containing glass-windowed lecture rooms, depicted in Figure 7 (a). Five visual anchors are mounted on walls and glass windows as shown in Figure 7 (b) and (c). The different RTT and Bluetooth anchors are distributed over the area as shown in Figure 8 (a). We walk two different trajectories, shown in Figure 8 (b), named *double loop* and *zig zag*, respectively, because of their characteristic shapes. The pose estimates obtained occasionally from scanning of the visual anchors(at 5 and 6 different points, respectively) are also depicted. During the experiment, there is light traffic of students coming in and out of lecture rooms.

For ground-truth data, we use the Tango visual system, however the sparsity of robust visual features induced by uniform and repetitive structures and the large glass walls posed problems for this system. Therefore, we added multiple feature-rich posters on the floor to ensure robust localization. We emphasize that these added features are only used to obtain ground-truth data and are not part of our proposed system. In addition to our proposed system, we obtain UWB-based position estimates for comparison.

While the phones acquiring Bluetooth, RTT, IMU and visual features for indoor localization are carried by the test subject, they are followed by a second subject carrying a laptop that is used only

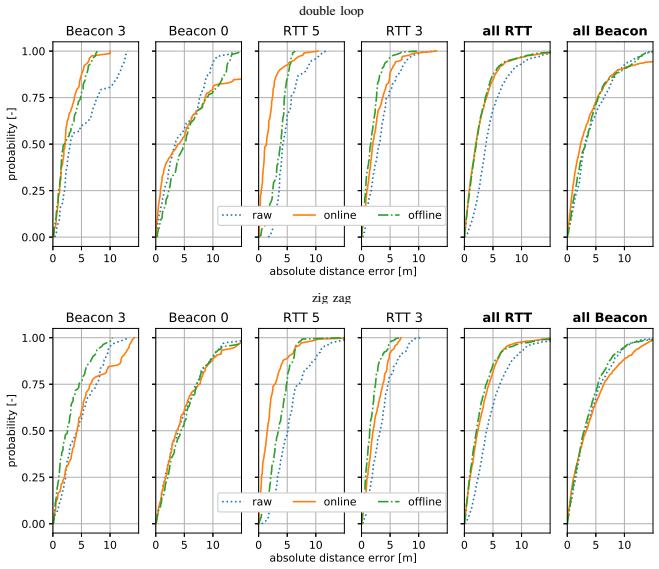


Figure 9. Cumulative distribution functions of absolute distance measurement errors with and without online/offline calibration. On average, online and offline calibration improve the distance accuracy of RTT distance measurements significantly, however online calibration can induce high errors for certain Bluetooth beacons.

for the UWB processing. Both the subject and the laptop carrier thus create challenging non-line-of-sight conditions for certain anchors.

B. Overview of used technologies

The ground truth Tango data is obtained with the augmented reality platform from Google. It is acquired from an indoor localization application running on a *Lenovo Phab 2 Pro* mobile device with a motion tracking camera and RGB-IR camera [33].

WiFi measurements are gathered from RTT-enabled *Fitlet2* access points by *Compulab* using the WiFi Indoor Positioning application [7] from a *Google Pixel* smartphone running *Android 9 Pie*. The measurements from the other systems are acquired with an *LG Nexus 5X* mobile phone. The beacons from *Kontact.io* are low-cost Bluetooth Low Energy emitting devices for proximity detection. IMU data is obtained from the smartphone's gyroscope and accelerometer. The sensor data is gathered and transmitted by a custom Android application. The visual position estimates are acquired from PixLive and obtained with a custom Android application that uses Vidinoti's SDK [32]. All the systems send their output to a *python* server which stores the acquired data.

The UWB system, used for comparison, is called *Pozyx* [14] and requires four fixed anchors and one tag mounted on an *Arduino UNO*, which is connected with a USB cable to a laptop running the acquisition server.

We fix the two phones and the UWB tag on a custom portable wooden mount, which makes sure their relative positions stay the same throughout the experiment. The Tango device is carried in the other hand so that it can be moved freely to scan the environment. The devices are thus not exactly co-located, from which we expect a small additional positioning error in the range of 5-10 centimeters.

C. Data accuracy and calibration

We first evaluate the accuracy of the obtained distance measurements, with and without online calibration. Figure 9 shows the obtained distance accuracy for the two studied datasets. For comparison, we also plot the result from offline calibration, obtained using measurements at 5 static positions (ca. 60 seconds each) before the experiments started. For each dataset, we show two example Beacon and RTT anchors, respectively, and the cumulative error for all anchors. While calibration is always beneficial for RTT anchors,

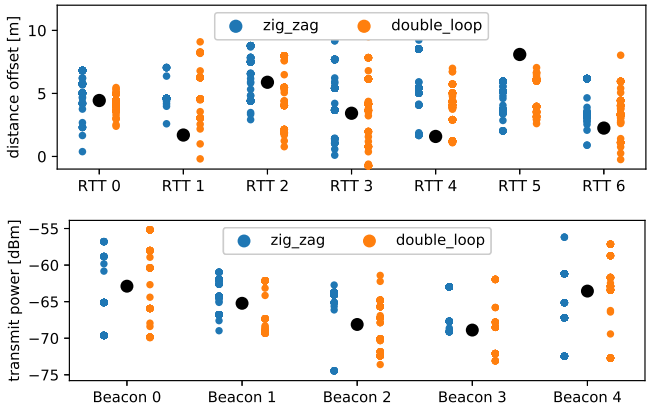


Figure 10. Distribution of calibration parameters for WiFi (top) and Bluetooth (bottom) anchors. The parameters calibrated offline are shown in black. We show the obtained parameters for each dataset separately. Parameters are recalculated whenever a visual measurement is present. The high variance in optimal parameters motivates the continuous online calibration over offline calibration.

it can lead to faulty results for the Bluetooth beacons, for instance for *Beacon 0* in the *double loop* dataset. We have found that this is due to the high variance of Bluetooth RSSI, which means that using a small window for calibration can lead to high bias. Figure 10 shows that there is indeed a high variance of the calibrated parameters for the two different datasets. However, the localization results show that faulty distances are successfully compensated for by other more accurate modalities. Both offline and online calibration are therefore valid choices, however online calibration has the advantage of not needing any additional setup time and is the preferred solution.

D. Evaluation of available modalities

For localization, we initialize a grid including connectivity information given the map of the building. A grid size of 0.5 meters is found to yield fast enough inference and satisfactory resolution. Since we are limited to one floor, we only use one layer in z-direction at the approximate height of the devices during experiments (1.2 meters).

We first evaluate the performance of the proposed system for different combinations of measurement modalities and calibration schemes. Table I shows a summary of the obtained localization

Table I
COMPARISON OF PROPOSED METHOD WITH DIFFERENT CALIBRATION SCHEMES AND MODALITIES USED. THE BEST SCORES PER ROW AND COLUMN ARE HIGHLIGHTED IN BOLD AND COLOR, RESPECTIVELY.

	double loop					
	mean error[m]			median error[m]		
calibration	**none**	**offline**	**online**	**none**	**offline**	**online**
IMU	4.55	4.55	4.55	3.94	3.94	3.94
RTT	5.85	4.06	**3.34**	4.99	3.11	**2.48**
BLE	7.08	6.65	**6.18**	5.85	5.68	**4.64**
RTT+IMU	4.07	3.21	**2.87**	3.99	3.03	**2.78**
BLE+IMU	4.36	4.21	**3.78**	3.56	3.85	**3.35**
RTT+BLE	5.77	4.06	**3.18**	4.83	3.20	**2.48**
RTT+BLE+IMU	4.28	3.27	**2.60**	4.09	2.90	**2.29**

	zig zag					
	mean error[m]			median error[m]		
calibration	**none**	**offline**	**online**	**none**	**offline**	**online**
IMU	5.32	5.32	5.32	4.99	4.99	4.99
RTT	5.06	3.36	**3.05**	5.02	3.07	**2.76**
BLE	6.05	**5.71**	7.93	5.51	**5.39**	6.40
RTT+IMU	3.70	2.99	**2.62**	3.78	3.06	**2.49**
BLE+IMU	5.08	5.04	**4.46**	3.70	4.41	**3.26**
RTT+BLE	4.90	3.24	**2.93**	4.76	2.97	**2.68**
RTT+BLE+IMU	3.60	2.93	**2.44**	3.50	2.98	**2.33**

accuracy, in terms of median and mean localization errors over the whole dataset. We denote by localization error the Euclidean distance between the position estimate and the ground truth obtained from Tango.

It is immediately apparent that adding IMU features increases the accuracy of localization significantly, even though the IMU measurements on their own would yield poor localization results. Adding Beacon measurements on average only slightly improves accuracy. This is expected since Beacon measurements are the least reliable ones. Furthermore, the Beacon feature function is relatively flat compared to the other features by design, so they only have little impact on the global probability distribution.

In terms of calibration schemes, online calibration yields the best results for almost all combinations of systems. Calibration is particularly powerful for RTT measurements, which otherwise can exhibit a high offset: it almost halves the median and mean error for both datasets.

Finally, we emphasize that it is best to combine the three available modalities, leading to the lowest median and mean errors. Considering the significant differences in measurement quality (note the difference in localization performance when using each system individually), our system thus weights all modalities correctly when combining them. Indeed, the 2D localization plots in Figure 13 show that Bluetooth and Wifi RTT measurements alone lead to jumpy estimates, while using IMU only induces high drift. The proposed method favorably combines the used modalities.

E. Comparison with other methods

We show the performance of our algorithm, which we call CRF, and other distance-based localization methods, in Figure 11. We add the recursive solution discussed in IV-B, where we backtrack the trajectory from the final position estimate (called *CRF recursive*). We compare to the two algorithms proposed by [5]: the grid-search implementation of Range Least Squares (RLS, denoted by *grid-L2*) and the Squared Range Least Squares (SRLS) solution. We also introduce a variation of RLS using the median rather than mean distance error (denoted by *grid-L1*) and a simple weighted centroid algorithm similar to [1]. In our centroid algorithm, we linearly interpolate the 3 closest anchor coordinates, using the inverse of the distance measurements as weights. For fairness, we always use the Pixlive measurements as position estimate if it is available.

Figure 11 shows 2D plots of the obtained localization using the best-performing combination from Table I in terms of mean localization error (RTT, Beacon and IMU measurements and online calibration). Thanks to the time consistency imposed by the IMU features in our implementation, the position estimate is smooth and its shape is close to the actual trajectory of the target. For the distance-only methods, the estimate is very volatile and the shape of the trajectory is hard to discern.

The cumulative distribution functions of the localization error for the different algorithms are shown in Figure 12. Compared to the UWB-based solution Pozyx, which requires designated hardware, our method yields higher localization errors. However, the position accuracy of the proposed method is the best amongst the shown smartphone-compatible solutions. In particular, high error estimates are significantly reduced in the proposed framework. The difference in performance between the four benchmark methods is quite small: for the grid-based methods and SRLS, this is explained by the relatively high distance accuracy (after calibration), meaning that the median and mean are similar, and also that RLS and SRLS are expected to behave similarly; the only difference being the squaring of distances for SRLS [5]. The reasonable performance of the centroid algorithm is explained by a relatively dense deployment of anchors.

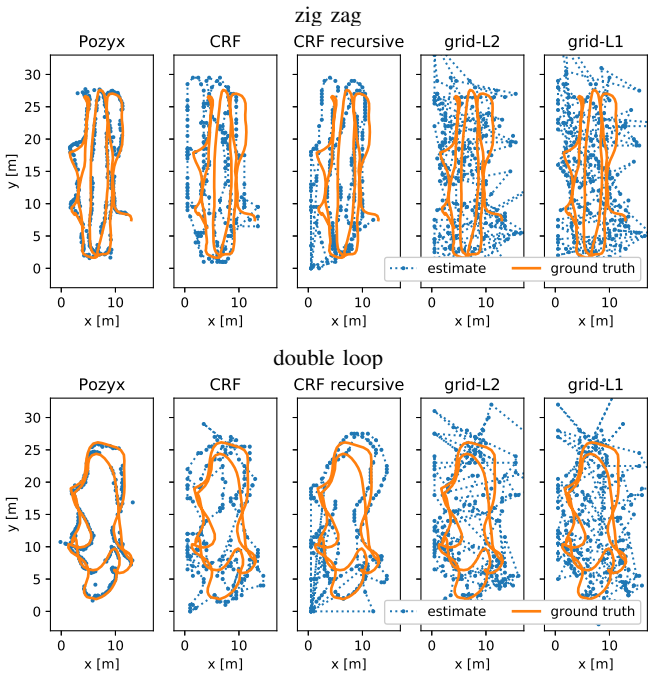


Figure 11. Visualization in 2D of the proposed localization method compared to standard and state-of-the-art methods, on the two studied datasets. Our method (CRF) yields smooth and accurate position estimates, comparable with the UWB-based system Pozyx. The recursive location inference (CRF recursive) smooths the results a posteriori and corrects for the slight drift in the *zig zag* dataset.

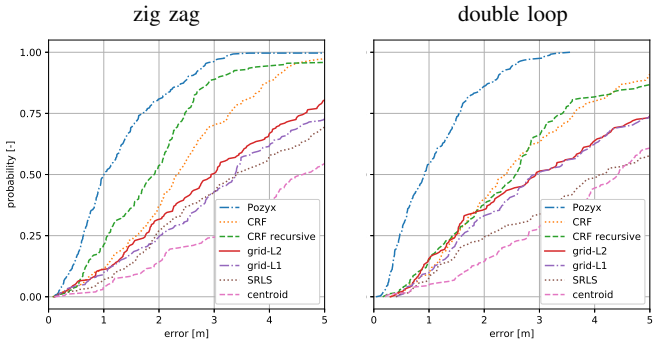


Figure 12. Cumulative distribution function of localization error. We compare the proposed localization methods with other distance-based methods on two different datasets. Over all, our method (CRF) is the best-performing method among the shown algorithms, and beaten only by the UWB-based Pozyx system, which is not smartphone compatible. The recursive application of our algorithm improves the accuracy for the *zig zag* dataset a posteriori.

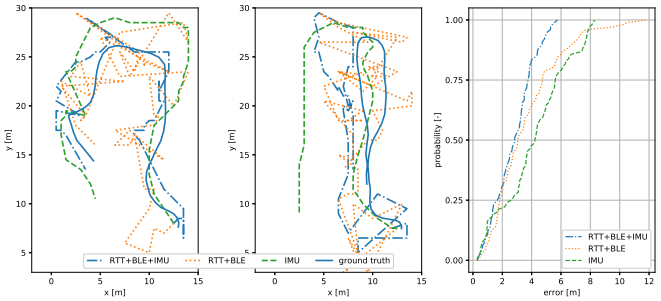


Figure 13. Comparison of localization results depending on the measurement types used, for the first 60 seconds of measurements of the *double loop* (left) and *zig zag* (middle) datasets. The total cumulative distribution function of the localization errors is shown on the right.

VII. Conclusion

We have proposed a smartphone-compatible multi-modal indoor localization system that integrates various measurement types using a probabilistic framework. Experimental results show that the system yields more accurate estimates than classical approaches, and that even when entire modalities are missing, the localization continues to be accurate. The setup can be quickly installed with no training phase required, which also makes it robust to environmental changes. Finally, the system is passive for the vast majority of the time with the user only actively scanning the environment from time to time.

In the future, we envisage that the system could be extended in a number of ways. For example, more sophisticated step detection algorithms, such as the techniques proposed in [4], could lead to improved tracking.

In addition, the sparse use of visual features could be further developed to include modern SLAM systems such as Apple's ARKit and Google's ARCore when they are active. The system would still fall back on RF signals and IMU when the smartphone is returned to the user's pocket.

Finally, more measurement modalities such as angular information from Bluetooth 5.1 could be leveraged in the existing framework, to further increase the robustness and accuracy of the localization.

References

[1] Islam Alyafawi, Simon Kiener, and Torsten Braun. Hybrid indoor localization using multiple Radio Interfaces. *17th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–7, 2016.

[2] Gilles Baechler, Frederike Duembgen, Golnoosh Elhami, Miranda Krekovic, Robin Scheibler, Adam Scholefield, and Martin Vetterli. Combining range and direction for improved localization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3484–3488, April 2018.

[3] Alfonso Bahillo, Santiago Mazuelas, Javier Prieto, Patricia Fernández, Ruben M. Lorenzo, and Evaristo J. Abril. Hybrid RSS-RTT localization scheme for wireless networks. *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7, 2010.

[4] Bertrand Beaufils, Frédéric Chazal, Marc Grelet, and Bertrand Michel. Activity recognition from stride detection: a machine learning approach based on geometric patterns and trajectory reconstruction. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2018.

[5] Amir Beck, Petre Stoica, and Jian Li. Exact and Approximate Solutions of Source Localization Problems. *IEEE Transactions on Signal Processing*, 56(5):1770–1778, 2008.

[6] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary robust independent elementary features. In *The European Conference on Computer Vision*, pages 778–792, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[7] Compulab. *WILD Minimal - WiFi RTT proof-of-concept for Android 9 Pie*, 2018 (accessed May 7, 2019).

[8] Jose a. Costa, Neal Patwari, and Alfred O. Hero. Distributed weighted-multidimensional scaling for node localization in sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 2(1):39–64, 2006.

[9] Ivan Dokmanic, Reza Parhizkar, Juri Ranieri, and Martin Vetterli. Euclidean Distance Matrices: Essential theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 32(6):12–30, 2015.

[10] Vânia Guimarães, Lourenço Castro, Susana Carneiro, Manuel Monteiro, Tiago Rocha, Marília Barandas, João Machado, Maria Vasconcelos, Hugo Gamboa, and Dirk Elias. A motion tracking solution for indoor localization using smartphones. *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2016.

[11] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[12] Mohamed Ibrahim, Hansi Liu, Minitha Jawahar, Viet Nguyen, Marco Gruteser, Richard Howard, Bo Yu, and Fan Bai. Verification : Accuracy Evaluation of WiFi Fine Time Measurements on an Open Platform. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 417–427, 2018.

[13] Mariusz Kaczmarek, Jacek Ruminski, and Adam Bujnowski. Accuracy analysis of the RSSI BLE SensorTag signal for indoor localization purposes. In *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 1413–1416. IEEE, 2016.

[14] Pozyx Labs. *Pozyx Accurate Positioning*, 2009 (accessed May 7, 2019).

[15] Xiaohui Li, Xiong Cai, Yongqiang Hei, and Ruiyang Yuan. NLOS identification and mitigation based on channel state information for indoor WiFi localisation. *IET Communications*, 11(4):531–537, 2017.

[16] Ze Li, Zengshan Tian, Mu Zhou, Zhenyuan Zhang, and Yue Jin. Awareness of Line-of-Sight Propagation for Indoor Localization Using Hopkins Statistic. *IEEE Sensors Journal*, 18(9):3864–3874, 2018.

[17] Dimitrios Lymberopoulos and Jie Liu. The Microsoft Indoor Localization Competition: Experiences and Lessons Learned. *IEEE Signal Processing Magazine*, 34(5):125–140, 2017.

[18] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer Science & Business Media, 2003.

[19] Davide Macagnano and Giuseppe Thadeu Freitas de Abreu. Algebraic Approach for Robust Localization with Heterogeneous Information. *IEEE Transactions on Wireless Communications*, 12(10):5334–5345, 2013.

[20] Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In *The European Conference of Computer Vision (ECCV)*, pages 183–196. Springer Berlin Heidelberg, 2010.

[21] Alex Mariakakis, Jeongkeun Lee, and Kyu-han Kim. SAIL: Single Access Point-Based Indoor Localization. *The 12th ACM International Conference on Mobile Systems, Applications, and Services*, pages 315–328, 2014.

[22] George Oguntala, Raed Abd-Alhameed, Stephen Jones, James Noras, Mohammad Patwary, and Jonathan Rodriguez. Indoor location identification technologies for real-time IoT-based applications: An inclusive survey. *Computer Science Review*, 30:55–79, 2018.

[23] Richard J. Radke. *Computer Vision for Visual Effects*. Cambridge University Press, New York, NY, USA, 2012.

[24] Valentin Radu and Mahesh K. Marina. HiMLoc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced WiFi fingerprinting. *2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10, 2013.

[25] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-Effort Crowdsourcing for Indoor Localization. *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, pages 293–304, 2012.

[26] Maurizio Rea, Aymen Fakhreddine, Domenico Giustiniano, and Vincent Lenders. Filtering Noisy 802.11 Time-of-Flight Ranging Measurements From Commoditized WiFi Radios. *IEEE/ACM Transactions on Networking*, 25(4):2514–2527, 2017.

[27] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *The European Conference of Computer Vision (ECCV)*, pages 430–443. Springer Berlin Heidelberg, 2006.

[28] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An Efficient Alternative to SIFT or SURF. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2564–2571, Washington, DC, USA, 2011. IEEE Computer Society.

[29] Yuanchao Shu, Cheng Bo, Guobin Shen, Chunshui Zhao, Liqun Li, and Feng Zhao. Magicol: Indoor Localization Using Pervasive Magnetic Field and Opportunistic WiFi Sensing. *IEEE Journal on Selected Areas in Communications*, 33(7):1443–1457, 2015.

[30] Nitesh B. Suryavanshi, Reedy K. Viswavardhan, and Vishnu R. Chandrika. Direction finding capability in bluetooth 5.1 standard. In *Ubiquitous Communications and Network Computing*, pages 53–65. Springer, Cham, 2019.

[31] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT press, 2005.

[32] Vidinoti. *Pixlive SDK Release 6.0*, 2017 (accessed May 26, 2019).

[33] Wera Winterhalter, Freya Fleckenstein, Bastian Steder, Luciano Spinello, and Wolfram Burgard. Accurate indoor localization for RGB-D smartphones and tablets given 2D floor plans. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[34] Zhuoling Xiao, Hongkai Wen, Andrew Markham, and Niki Trigoni. Lightweight map matching for indoor localisation using conditional random fields. *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks (IPSN)*, pages 131–142, 2014.

[35] Zhuoling Xiao, Hongkai Wen, Andrew Markham, and Niki Trigoni. Robust indoor positioning with lifelong learning. *IEEE Journal on Selected Areas in Communications*, 33(11):2287–2301, 2015.

[36] Zhuoling Xiao, Hongkai Wen, Andrew Markham, Niki Trigoni, Phil Blunsom, and Jeff Frolik. Non-Line-of-Sight Identification and Mitigation Using Received Signal Strength. *IEEE Transactions on Wireless Communications*, 14(3):1689–1702, 2015.

[37] Zheng Yang, Chenshu Wu, and Yunhao Liu. Locating in fingerprint space: Wireless Indoor Localization with Little Human Intervention. *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 269–280, 2012.