The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Sub-Meter Indoor Localization in Unmodified Environments with Inexpensive Sensors

Morgan Quigley, David Stavens, Adam Coates, and Sebastian Thrun

Abstract— **The interpretation of uncertain sensor streams for localization is usually considered in the context of a robot. Increasingly, however, portable consumer electronic devices, such as smartphones, are equipped with sensors including WiFi radios, cameras, and inertial measurement units (IMUs). Many tasks typically associated with robots, such as localization, would be valuable to perform on such devices. In this paper, we present an approach for indoor localization exclusively using the low-cost sensors typically found on smartphones. Environment modification is not needed. We rigorously evaluate our method using ground truth acquired using a laser range scanner. Our evaluation includes overall accuracy and a comparison of the contribution of individual sensors. We find experimentally that fusion of multiple sensor modalities is necessary for optimal performance and demonstrate sub-meter localization accuracy.**

I. INTRODUCTION

Precise localization is a key component of a variety of applications, such as navigation, asset tracking, and even advertising. The localization problem has attracted intense research in the mobile robotics community for decades. Robust navigation systems are now widely available and are often based around Bayes filter variants which observe the world through laser range-finders and robot odometry [1]. However, most fielded localization systems involve high-precision, expensive sensors such as laser range sensors, high-quality inertial units, or extensive infrastructure.

In this paper, we address a different problem: indoor localization using the sensor suite currently available in smartphones. Typified by the Apple iPhone and Android devices, smartphones often contain a WiFi radio, camera, accelerometer, and magnetometer, in addition to the GSM or CDMA radio used to handle voice calls. This sensor suite is quite different from the canonical laser range finder and odometer found in most robot localization tasks, and presents a different (though related) set of challenges.

The system includes two components: a mapping platform and a mobile localization platform. The mapping platform is a typical modern robot shown in Figure 1. This robot autonomously acquires maps of indoor environments and aligns them with off-the-shelf SLAM algorithms. The principal contribution of this paper is the mobile localization system, which autonomously localizes itself to the maps using only consumer-grade sensors typical of what can be found on a handheld device such as a smartphone. In particular, the sensors we use are WiFi signal-strength measurements, low-resolution camera images, and an inexpensive accelerometer.

The authors are with the Computer Science Department of Stanford University, Stanford, CA 94305, USA. {mquigley, dstavens, acoates, thrun}@cs.stanford.edu

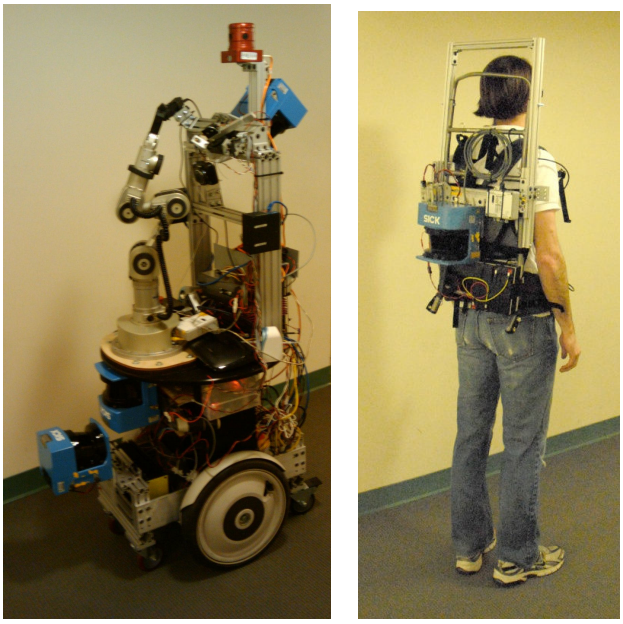

Fig. 1. To build models for low-cost localization, we first drive a robot through the environment to build a LIDAR point-cloud map using SLAM. Next, a pedestrian equipped with backpack-mounted LIDAR walks through the environment to acquire sensor models, carrying the consumer-grade device of interest. The 3-d map and sensor models can be acquired once and then used for localization on standalone consumer-grade devices.

Our method requires no additional environment instrumentation or modification beyond standard, widely-deployed WiFi infrastructure. In practice, the mapping platform is used once for each environment (e.g., a shopping center or airport). The resulting map may then be used by many roaming devices, bringing high-quality localization to these low-cost sensor platforms.

We test our implementation by evaluating its accuracy against ground truth results acquired using the backpack-mounted sensing system shown in Figure 1. We find that our method provides sub-meter precision with consumer-grade sensors and without environment modification or instrumentation. We demonstrate that WiFi is excellent for quick global convergence, but a camera performs better for precise position tracking. Sensor fusion gives the best of both. Realistic test scenarios are considered. In particular, the map and localization data are collected at different times of day on different days, after the environment was allowed to undergo typical daily changes. The system offers potential for location-aware, consumer-oriented services such as social networking, direct ad delivery, or convenient payment.

978-1-4244-6676-4/10/$25.00 ©2010 IEEE

II. RELATED WORK

The literature on localizing a robot (or other rigid sensor platform) against a map is long and rich. [1] provides a comprehensive literature review which we summarize and extend. The idea goes back at least as far as the robot Odysseus [2], which compared sensor measurements in a local grid to a global map and competed at the National Conference on Artificial Intelligence (AAAI) in 1992. A continuum of algorithms exist across a variety of sensor and map configurations. [3] uses sonar to detect coarse landmarks in maps and localize with an extended Kalman filter (EKF). Later, grid-based methods were developed. In contrast to EKFs, these methods represent the posterior as a histogram and are not constrained to Gaussian noise assumptions. Grid-based methods usually rely on landmarks, however. Grid-based localization was used successfully in sewer pipes [4], in a museum [5], and in an office environment [6]. [7] used learning to determine the best landmarks for reliable localization. Most recently, Monte Carlo Localization (MCL) [8] was developed, replacing landmarks with raw measurements and the histogram posterior with particles. In a hybrid of ideas between MCL and grid-based methods, [9] introduces MCL with features. Several papers have utilized MCL with cameras including [10], [11], [12], [13], [14]. Others have localized by direct image matching, without using a probabilistic filter or motion model [15], [16]. Localization with signal-strength mechanisms such as WiFi have been studied in the literature as well [17], [18], [19], [20], [21], including systems that bootstrap automatically without an explicit map-making step [22], [23]. Much additional work also exists that we must omit due to space, including work using Bluetooth.

There are several key differences between this work and the previous literature. First, we limit the sensor suite to those available on a typical smartphone. In contrast, much of the previous literature involves sensors that are not practical for such a device, including laser range finders, wheel encoders, and expensive inertial measurement units (IMUs). Our work uses only a consumer-grade IMU, camera, and WiFi radio. Second, while work does exist on low-cost sensors such as WiFi or cameras, these sensors are usually studied individually. We focus on *probabilistic sensor fusion*. As long as the sources of measurement uncertainty such as noise, bias, and incorrect invariance are conditionally independent probabilistically, combining multiple sensors will have a positive impact on performance. This is true even if the sensors are inexpensive. We demonstrate that while WiFi offers fast global convergence, cameras provide more precise tracking. Sensor fusion allows us to achieve the best of both, in contrast with prior work. Third, we construct an explicit ground-truth data set for comparison. To acquire ground-truth, we use a robot with a laser scanner, a high-end IMU, and wheel odometry. Running SLAM [1] on this data set, ground-truth is constructed to sub-decimeter precision. (This does not compromise our goals as the map can be built once, offline, and then used for thousands of users with inexpensive sensors, for example, in a shopping mall or airport.) This allows us to examine many useful properties experimentally, such as absolute accuracy and the accuracy of each sensor individually. Finally, many previous vision-only localization, or image-registration, work involves one-shot or *ad-hoc* methods for fusing multiple observations, or operates on topological maps, e.g., [24]. In contrast, our work uses a systematic probabilistic approach, the Bayes filter, to fuse models of the sensors and the motion of the pedestrian as more observations are incorporated into the estimate.

Fig. 2. 2-d map of the environment used in these experiments, as produced by GMapping and the robot shown in Figure 1

III. APPROACH

Our system is based around three levels of sensing and inference. The first two are used for offline map-building, and the third is used for online localization. These stages are described in detail in the following sections.

A. Robotic SLAM

The first tier of our system captures the 3-d structure of the environment. This is performed by a robotic platform equipped with three LIDAR scanners and a panoramic camera, as shown in Figure 1. To build up a 2-D map of the environment and correct the odometry of the robot, a horizontal LIDAR is used with the GMapping SLAM system, an efficient open-source implementation of grid-based FastSLAM [25].

The GMapping system was used out-of-the-box to produce the 2-d map shown in Figure 2. The robot path corresponding to this map was then used to project the vertical and diagonal LIDAR clouds into 3-d by backprojecting rays through the rectified images into the LIDAR cloud. The robotic mapping phase of our system is thus able to measure the 3-d structure and texture of the environment. However, this alone is not enough to permit localization via a smartphone sensor suite; what is needed is a precise sensor model of how the low-cost sensors behave in the environment of interest. This is handled by the next phase of our system.

B. Obtaining Training Data

Non-parametric methods are a simple way to capture the complex phenomena seen by the low-cost sensors. For example, it would be difficult to parametrically model the various radio-frequency (RF) propagation effects that occur with WiFi signal power in an indoor environment. Issues such as occlusion/shadowing from building structural elements, interference between multiple access points, directionality of

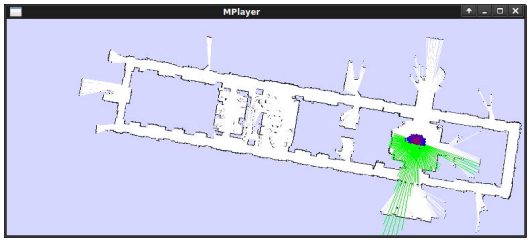

Fig. 3. A typical rendering of the particle filter used to localize the "ground truth" pedestrian using rearward LIDAR. The green LIDAR scan is rendered from the most likely particle in the filter.

the transmit and receive antennas, etc., result in a complex power distribution pattern. Similarly, the camera of a smartphone captures an enormously complex stream of data. A simple (indeed, perhaps the simplest) way to predict these complex observations is to simply acquire many observations from a large number of known positions in the environment.

Obtaining training data for these non-parametric techniques is non-trivial: the location of the observation (WiFi signal power or camera image) must be known for it to be useful to subsequent localization algorithms. A major potential application of low-cost localization is for indoor pedestrian navigation, and the pedestrian's body can have an effect on the received signal strength (e.g., the person's body is directly between the receiving and transmitting antennas). We thus created a system for accurately localizing pedestrians, and used this to obtain training data for non-parametric modeling of the spatial RF signal power.

To localize the pedestrian, we affixed a rearward-facing laser range finder to a backpack, as shown in Figure 1. We then employed a particle filter to fuse the laser observations with a crude motion model of a pedestrian. We note that this is more challenging than the canonical robot localization task, since mobile robots typically have odometry which is locally stable. Our pedestrian-localization system, in contrast, only knows if a person is walking or not; we found that low-cost MEMS accelerometers are far too noisy to simply integrate to position estimates. Instead, we used a simple sliding-window classifier on the spectrum of the acceleration vector to detect when to apply a "walking" motion model. We also found that low-cost magnetometers were not reliable in our testing environment: a steel-framed building with many computers, power cables, and other electronic equipment capable of inducing local magnetic disturbances. While our testing environment may have been particularly unfriendly, we suspect that similar local magnetic perturbations would confound attempts at relying heavily on magnetometer data in many indoor environments.

As the accelerometer and magnetometer can only give a coarse measurement of the path of a pedestrian, our LIDAR-based pedestrian particle filter relies heavily on frequent "gentle" resampling of the particle cloud. More specifically, our measurement model had a far higher uniform component than typical, and incorporated measurements from every laser scan, in order to correctly track the pedestrian through turns. A typical rendering of the particles is shown in Figure 3.

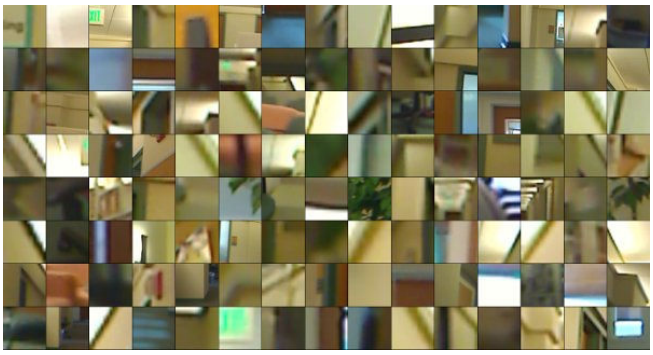

Fig. 4. Exemplar image regions corresponding to the "visual words" used during image matching. See text for details.

C. Camera Sensor Model

The literature on place recognition using visual images contains many proposed methods. For these experiments, we selected three different approaches from the recent computer vision literature: a "bag of words" method using SURF descriptors of interest points [26] [27], a "bag of words" method using HoG descriptors of a dense uniform grid [28], and a color-histogram method [24]. We further augmented the first two methods by adding a spatial pyramid [29]. We will describe these methods in turn.

In the bag of words model, we first construct a dictionary of "visual words". This is done by extracting SURF [27] descriptors from a large set of images captured by the mapping platform cameras, then quantizing using K-means clustering. The resulting 128-dimensional cluster centroids are stored with indices 1 to k. Figure 4 shows image patches whose descriptors are at the center of clusters computed by K-means. Given an image, we can compute the "bag of words" description in the following way: (i) Extract SURF descriptors from the image, (ii) Map each descriptor to the index of the nearest centroid in the dictionary, and (iii) Construct a histogram with the frequency counts for each index (i.e., the number of descriptors that were mapped to each index). Though the histogram discards all of the geometric information about the locations of the descriptors in the image, they have nevertheless been shown to function effectively as compact descriptions of the image content.

Our HoG-based method used a similar approach. However, instead of using descriptors of interest points, we sampled the image on a dense grid. As a result, the number of HoG descriptors extracted from each image was always the same. To produce a similar data compression as the SURF-based method, we chose to extract HoG descriptors from 32x32 blocks arranged on a 15x20 grid across the image. This resulted in 300 HoG descriptors per image, which was similar to the average number of SURF keypoints found in the same images using the OpenCV SURF implementation. As before, we used k-means to quantize the HoG descriptors, and built histograms of the quantized descriptors for each image.

As previously mentioned, the "vanilla" bag of words algorithm discards the spatial configuration of the descriptors in the image plane. The "spatial pyramid" approach is one proposed method to incorporate coarse spatial information, and is fully developed in [29]. Briefly, this method repeatedly subdivides the image into quadrants, and constructs histograms for each quadrant on each level. For example, the two-level spatial pyramid would have one global histogram for the whole image, and one histogram for each quadrant, for a total of five histograms. Similarly, the three-level pyramid has $1+4+16=21$ histograms. This approach has been shown to offer improved performance over the single-histogram technique.

For a radically different approach, we also implemented a color-histogram technique. This technique is conceptually much simpler: the image is first converted to hue-saturation-value (HSV) space, after which a histogram is constructed of the hue values of all pixels in the image. The conversion to HSV is done to provide some invariance to illumination changes. The resulting representation is essentially a polar histogram of the color wheel.

To use these image representations in a localization filter, we need to produce an estimate of the probability that an image representation $\mathbf{z}$ was produced from pose $\mathbf{x}$. To compute this probability using an approach analogous to that of laser range-finders, we would need to project a textured 3-d model of the world into each particle's camera frame, and compute some sort of distance function. This would be computationally difficult, even on a GPU. Instead, we compute a coarse, yet experimentally justified, approximation: we estimate $p(\mathbf{z}|\mathbf{x})$ through a nearest-neighbor lookup on the training-set images $\mathbf{y}_{\mathbf{i}_{\mathbf{img}}}$ and poses $\mathbf{y}_{\mathbf{i}_{\mathbf{pose}}}$ in histogram space, and augment a histogram distance metric with a penalty for using images that are far from the candidate pose $\mathbf{x}$.

Intuitively, if the pose $\mathbf{x}$ is in the exact position as a pose in the training set, and the corresponding image histograms are identical, $p(\mathbf{z}|\mathbf{x})$ should be very high. Furthermore, $p(\mathbf{z}|\mathbf{x})$ should fall off smoothly as the image and pose start to differ from the training image histogram $\mathbf{y}_{\mathbf{i}_{\mathbf{hist}}}$ and training image pose $\mathbf{y}_{\mathbf{i}_{\mathbf{pose}}}$, so that query images taken near (but not exactly on) the poses of the training images will still receive a significant probability. Conversely, if the query image $\mathbf{z}$ is significantly different from the map image $\mathbf{y}_{\mathbf{i}_{\mathbf{hist}}}$, or the candidate pose $\mathbf{x}$ is significantly different from the map image pose $\mathbf{y}_{\mathbf{i}_{\mathbf{pose}}}$, the probability should be very small.

We experimented with various probability distributions, and found experimentally that the heavy tails of a Laplacian distribution were better suited for this sensor than a Gaussian distribution. The parameters λ_1 and λ_2 allow for independent scaling between the histogram distance and the pose distance. We also penalized for yaw deviation, as the query image and the training image should be pointed in nearly the same direction for comparison to be meaningful. The combined model first finds the nearest neighbor, using the aforementioned weighted distance metric, and then models that distance as a zero-mean Laplacian distribution:

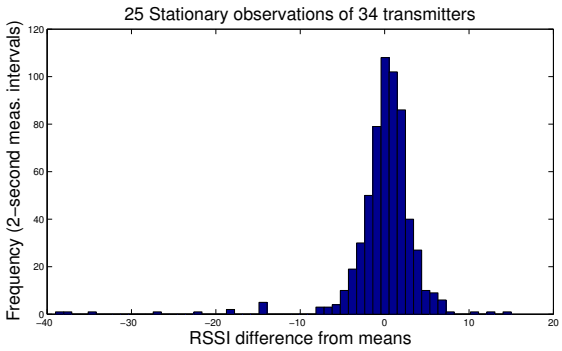

Fig. 5. Empirical justification of the Gaussian + uniform model of the WiFi power measurements. The plot shows the frequency of power measurement deviations from their respective means. This dataset was gathered while sitting stationary for 60 seconds, and includes 34 transmitters, most of which were observed 25 times.

$$p(\mathbf{z}|\mathbf{x}) \propto \mathbf{exp}\frac{-\mathbf{min_i}\lambda_1 \left\|\mathbf{z}-\mathbf{y}_{\mathbf{i}_{\mathbf{img}}}\right\|_1 + \lambda_2 \left\|\mathbf{x}-\mathbf{y}_{\mathbf{i}_{\mathbf{pose}}}\right\|_2}{\sigma} \tag{1}$$

Large changes in ambient illumination will cause low-cost cameras to have numerous artifacts, such as higher noise as the camera gain is raised in dim lighting. This, in turn, will cause a different number of interest points to be found in the image, resulting in a vertical shifts of the histogram. To provide some measure of invariance to global illumination for the SURF-based method, we normalize the image histograms before computing their distance.

D. WiFi Sensor Model

WiFi signal power measurements do not suffer from the correspondence-matching problem often associated with robotic sensors. Signal power measurements from scanning WiFi radios are returned with the transmitter's MAC address, a 48-bit number unique to the hardware device (barring pathological spoofing cases). Thus, even though the power measurement is noisy, WiFi observations can provide excellent context for global localization.

To simplify the probabilistic treatment, we assume conditional independence of the WiFi signals. This assumption is impossible to justify without access to the firmware of the WiFi radio, and we suspect that the assumption does not hold up (for example, if two WiFi radios are broadcasting on the same channel, a nearby radio may mask the presence of a more distant radio). However, we found experimentally that assuming conditional independence provides a useful likelihood function, and has the added benefit of computational simplicity.

To model the WiFi noise, we used a Gaussian distribution summed with a uniform distribution. This is empirically justified by the stationary observations shown in Figure 5, which were gathered from 34 transmitters over 60 seconds. There is a Gaussian-like bump around the expected mean, and a small number of large deviations on both sides. More

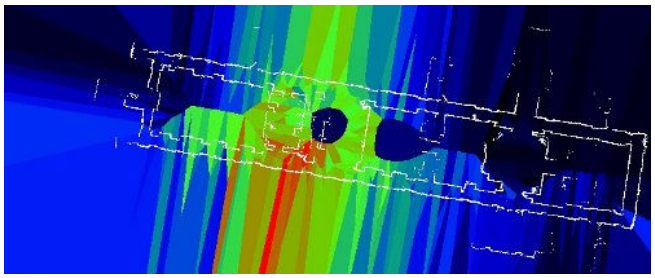

Fig. 6. Pre-computed nearest-neighbor prediction of the WiFi signal strength of a particular MAC address at any point in the environment. The walls of the environment are overlaid for clarity.

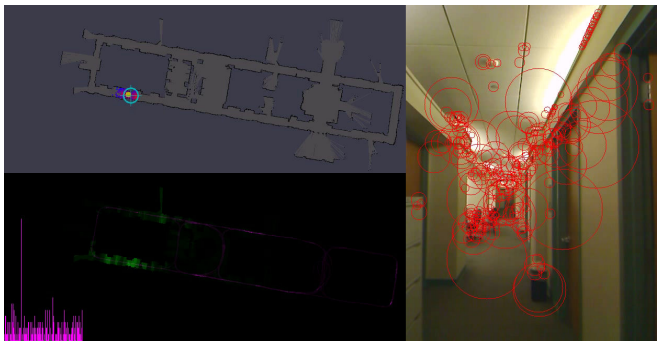

Fig. 7. Visualization of the unified vision + WiFi localization system. Upper-left shows the particle cloud, which is overshadowed by the centroid of the particle distribution (yellow) and the ground-truth position (cyan crosshairs). Right shows the current camera image, with SURF keypoints circled. Lower-left shows the joint likelihood of the WiFi observations. Extreme lower-left visualizes the histogram of the bag-of-words representation image.

formally, for a set of signal power measurements $\mathbf{z_i}$ and a robot pose $\mathbf{x}$

$$p(\mathbf{z}|\mathbf{x}) \propto \prod_i \mathbf{exp}\left(\frac{-\|\mathbf{z_i} - h_i(\mathbf{x})\|_2^2}{\sigma^2}\right) \tag{2}$$

where $h_i(\mathbf{x})$ is the predicted power measurement for transmitter i at pose x. To make this prediction, we simply employ nearest-neighbor over the training set: since each observation in the training set occurred at a known location (thanks to the laser scanner employed at training time), we build up a pre-computed map of the nearest-neighbor prediction of the WiFi signal power levels. A sample nearest-neighbor map is shown in Figure 6. We compute a nearest-neighbor map for each MAC address (transmitter) seen in the training set. With these maps, the computation of $p(\mathbf{z}|\mathbf{x})$ is linear in the number of MAC addresses in $\mathbf{z}$.

E. Localization

Once the sensor models are acquired, we incorporate them in a particle filter to introduce temporal constraints on the belief state and to fuse the models in a systematic fashion. The particle filter is Monte Carlo Localization (MCL) as described in [8]. The update step of the particle filter requires a motion model. Because magnetometers are unreliable in indoor environments such as the steel-framed building used in these experiments, we were unable to directly observe heading changes of the pedestrian. Instead, we used the motion model of the particle filter to continually hypothesize motions of the pedestrian.

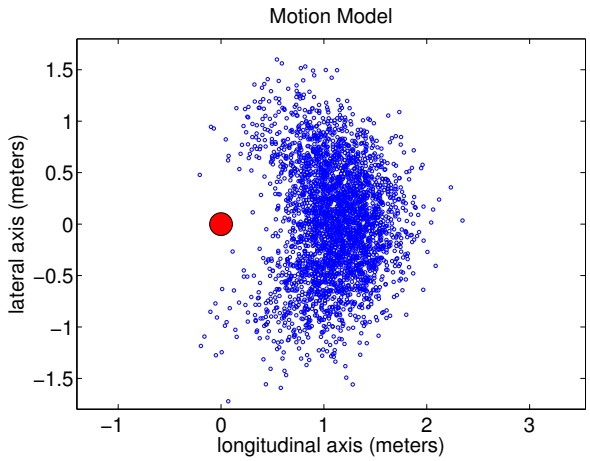

Fig. 8. Pedestrian motion model, shown after a one-second integration. Without odometry, the particle filter must generate sufficient diversity in its hypotheses to handle corners.

Our pedestrian motion model was empirically developed to match the trajectories observed by the laser-equipped ground-truth pedestrian. The motion model assumes that pedestrians usually travel in the direction they are facing, and this direction usually does not change. We model this by sampling the future heading from a Gaussian distribution $\mathcal{N}_1$ centered on the current heading. The velocity of the pedestrian is sampled from a Gaussian distribution $\mathcal{N}_2$ with a mean of 1.2 meters/second, which was empirically found using the LIDAR-based pedestrian localizer. These distributions are summed with a 2-d zero-mean Gaussians $\mathcal{N}_3$ to encourage diversity in the particle filter. More formally, to sample from the motion model,

$$\mathbf{v}' = R_\theta \begin{bmatrix} \mathcal{N}_2\left(\mu_{vel}, \sigma_1\right) \\ 0 \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \mathcal{N}_3\left(0, \sigma_2\right) + \mathbf{v}' \\ \mathcal{N}_1\left(0, \sigma_3\right) \end{bmatrix} \tag{4}$$

The parameters to this model were tuned in the LIDAR-based localization scenario, where the time between each laser scan was 27 milliseconds. To scale up to the larger intervals seen in the WiFi- and camera-based filters, particles were simply propagated through the previous equations the appropriate number of times. Running the model for one second produces the particle distribution shown in Figure 8 (dimensions in meters).

Our motion model also encodes the fact that the target cannot go through walls. As a result, when the target platform passes an intersection of corridors, particles are rapidly generated to implicitly cover the possibility that the pedestrian has turned.

Fig. 9. Images from the training set (top) differed from images in the test set (bottom) due to illumination changes and typical furniture re-arranging.

As is common practice in particle filters, to prevent premature convergence of the particles during global localization, and to handle unmodeled effects when tracking (e.g., lens flare when facing a sunbeam from a window, passers-by occluding the camera, RF anomalies, etc.), we add a uniform distribution to the measurement models described in the previous section.

IV. RESULTS

To quantify the performance of our system, we collected two data sets from the second floor of the Stanford University Computer Science Department. The data sets are approximately 13 minutes long and contain paths approximately one kilometer long. The first data set was recorded in the daytime, and the second data set was recorded in the nighttime, several days later, as shown in Figure 9. In the interim, many chairs were moved around in meeting spaces, different office doors were open (or shut), and some clutter was moved. We made no effort to normalize the environment between testing and training, other than to ensure that interior lights were turned on. However, no major renovations, redecorations, or organized clean-up occurred in the intervening days.

The first data set was used solely for training the sensor models. The second data set was used to generate localization estimates using the models learned from the first data set. The backpack shown in Figure 1 was worn while collecting both datasets, to permit quantitative analysis of localization errors of the low-cost sensors with respect to the LIDAR localization scheme, which is our best estimate of ground truth.

The data was collected on a laptop carried by the pedestrian. A handheld Logitech Webcam Pro 9000 was run at 640x480, 30 frames per second, and the raw YUV frames were recorded to disk. The internal WiFi card (Intel 4965) of a Dell m1330 laptop provided the WiFi data. Code was adapted from the Linux "iwlist" command to scan the RF environment every two seconds. Accelerometer data was provided by a handheld MicroStrain 3DM-GX2 at 100 hz. All of these sensors are comparable in performance to those in high-end smartphones; we used a laptop simply for ease of data-collection.

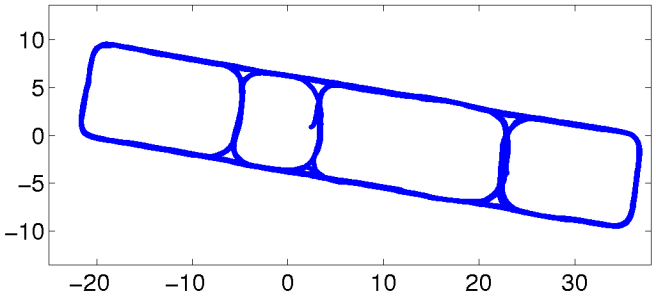

Fig. 10. The ground-truth LIDAR track of the 1-kilometer test set used for quantitative evaluation. The test set contained 62 corner turns, and a mixture of navigating tight corridors and more visually-open spaces such as cubicles. Distances are in meters.

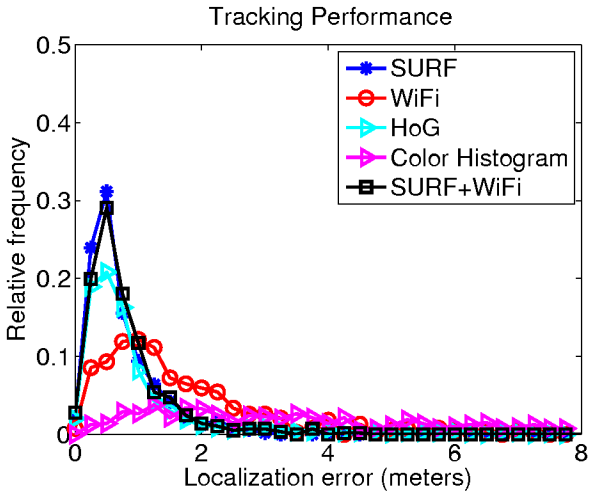

Fig. 11. Histograms of localization errors during the tracking benchmark on a continuous 1-kilometer test set. Errors are measured with respect to LIDAR ground-truth. The SURF and HoG performance is for the global (1-level) spatial pyramid. Adding WiFi to SURF slightly decreases its long-term average tracking accuracy. The color histogram performs poorly.

We performed empirical evaluation of several vision methods, WiFi by itself, and finally a combination of the empirically-best visual method and WiFi.

The first evaluation benchmark is a histogram of the localization errors observed on a 1-kilometer path through the test environment. This test dataset included 62 corners, and is shown in Figure 10.

TABLE I

QUANTITATIVE RESULTS FOR THE TRACKING TASK

Metric	WiFi	SURF	HoG	Color	SURF+WiFi
mean error (m)	1.81	**0.73**	4.31	10.90	0.78
std. dev of error (m)	0.99	**0.57**	9.15	10.65	0.64

The results of evaluating 1-level SURF and dense HoG, color-histogram, WiFi, and SURF+WiFi on the "tracking" benchmark are shown in Figure 11. The SURF method outperforms the WiFi method, and the combined SURF+WiFi system performs no better than the SURF-only system. The

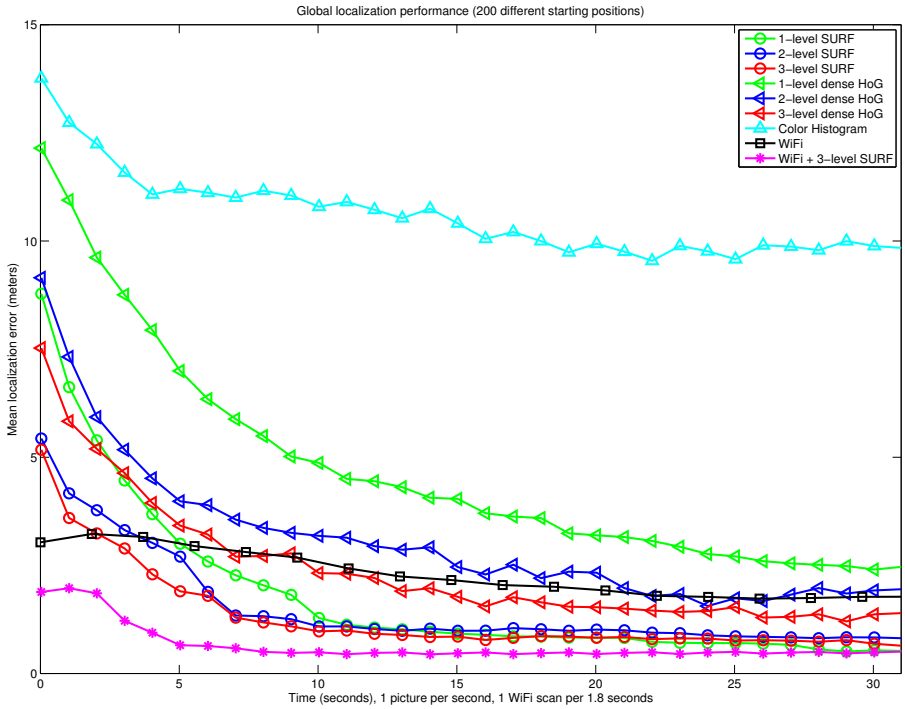

Fig. 12. Global localization performance. The localization systems were started with a uniform prior at 200 different starting points in the test set. Errors against ground-truth were averaged at each timestep to show the expected convergence properties of each system. All methods show improvement as more observations are incorporated. The combination of WiFi and the best visual algorithm (3-level spatial pyramid of SURF descriptors) produces the best performance.

dense HoG method does significantly worse, and the color histogram method performs poorly. The quantitative results are shown in Table I.

The second benchmark measures the speed of global localization by averaging the localization error as a function of the time since the localizer was started. These results were computed by starting the localization systems on the test data at 200 regularly-spaced starting points. A graphical plot is shown in Figure 12.

This benchmark reveals an interesting duality of the sensor suite: the WiFi system, thanks to having an intrinsic solution to the correspondence problem, can quickly achieve a mean error of 3-4 meters. However, due to the many sources of noise in the WiFi signal power measurement, the WiFi-only system cannot obtain meter-level performance. In contrast, the best visual methods (2- and 3-level SURF spatial pyramid) are able to obtain excellent tracking results, but take much longer to converge due to the repetitive nature of some regions of the test environment (e.g., long corridors), or inherent ambiguity of some starting positions (e.g., facing the end of a corridor).

Probabilistic fusion of the best visual method (3-level SURF spatial pyramid) and the WiFi measurements produces a system that combines the strengths of both modalities: quick global convergence to an approximate position fix, followed by precise tracking. The particle filter performs the sensor fusion automatically, using the sensor models and the motion model.

V. DISCUSSION

We have evaluated our system's performance over significant periods, showing it offers sub-meter precision in typical environments on average (Fig. 11, Table I). Our experiments also indicate that WiFi offers fast global convergence to better than 4 meters, but that computer vision offers greater tracking accuracy over long periods (Fig. 12, Table I). Combining the methods offers the best of each: fast convergence and excellent tracking precision (Fig. 12). We conclude sensor fusion is essential to achieve the highest indoor localization precision with the sensor suite of a typical smartphone.

We note in Table I that the mean accuracy of the the SURF visual method alone exceeds that of SURF and WiFi together. This is not surprising since WiFi is less accurate and thus will slightly decrease overall precision. However, as stated above, the sensor-fused system is still superior in the envisioned typical usage of pedestrian navigation, due to its much faster convergence. Future work could explore a measurement model that trusts the WiFi less after initial convergence in order to improve precision, reintegrating the WiFi if loss of global localization is detected (e.g., the kidnapped robot problem [1]). It is unclear if that would

noticeably improve practical performance in typical usage.

Any map-based localization system is prone to failure if the environment changes significantly from the time of map-making to the time of localization. However, our algorithm has several built-in methods of being robust to such changes that lead to its successful performance. Due to the probabilistic nature of our algorithm, map changes will be seen as unlikely observations. This will dilute the certainty of our localization, but will not cause localization failure. Consider, for example, the not infrequent case of a WiFi access point going offline for repair. In areas where there are many other sources of localization information (eg: rich visual features or other WiFi access points) there may be minimal degradation in the algorithm's performance. In the unlikely event that the failed access point was the only information source available, the particles will spread out and the filter will become more uncertain. (Similar to coastal navigation [30], the user can be warned if localization failure is imminent and can be directed to an area where successful localization is more likely. We do not implement that here.) Also, our best visual system uses SURF descriptors in a bag-of-words model. SURF descriptors themselves are invariant to changes in scale and rotation, which offers robustness to minor environmental changes (e.g., picture slightly rotated, door slightly opened, etc.). The bag-of-words model itself does not enforce a constraints on feature arrangement, resulting in a more robust (though less discriminative) representation of the image. Thus, again, the method is robust to minor environmental changes such as minor motion of chairs or other rigid objects.

VI. CONCLUSIONS AND FUTURE WORK

We presented a precision indoor localization system which uses only sensors comparable to those in current (2010) smartphones—several grades less expensive and less accurate than what is typically found on a research robot. Our method requires no environment instrumentation or modification. We implemented and rigorously tested our algorithm, demonstrating its effectiveness at sub-meter localization in a test environment. Our results indicate sensor fusion is essential, as WiFi is effective for fast global convergence whereas computer vision is preferred for precision tracking.

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[2] R. G. Simmons, S. Thrun, C. Athanassiou, J. Cheng, L. Chrisman, R. Goodwin, G. T. Hsu, and H. Wan, "Odysseus: An autonomous mobile robot," *AI Magazine*, 1992.

[3] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 376–382, 1991.

[4] J. Hertzberg and F. Kirchner, "Landmark-based autonomous navigation in sewerage pipes," in *Proc. of the First Euromicro Workshop on Advanced Mobile Robotics*, 1996.

[5] W. Burgard, A. B. Cremers, D. Fox, D. Haehnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial Intelligence*, vol. 114, pp. 3–55, 1999.

[6] R. G. Simmons, J. Fernandez, R. Goodwin, S. Koenig, and J. O'Sullivan, "Lessons learned from xavier," *IEEE Robotics and Automation Magazine*, vol. 7, pp. 33–39, 2000.

[7] S. Thrun, "Bayesian landmark learning for mobile robot localization," *Machine Learning*, vol. 33, 1998.

[8] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *Proc. of the International Conference on Robotics and Automation (ICRA)*, 1999.

[9] P. Jensfelt, D. Austin, O. Wijk, and M. Anderson, "Feature based condensation for mobile robot localization," in *Proc. of the International Conference on Robotics and Automation (ICRA)*, 2000, pp. 2531–2537.

[10] S. Lenser and M. Veloso, "Sensor resetting localization for poorly modelled mobile robots," in *Proc. of the International Conference on Robotics and Automation (ICRA)*, 2000.

[11] D. Schulz and D. Fox, "Bayesian color estimation for adaptive vision-based robot localization," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004.

[12] J. Wolf, W. Burgard, and H. Burkhardt, "Robust vision-based localization by combining an image retrieval system with monte carlo localization," *IEEE Transactions on Robotics and Automation*, 2005.

[13] B. Kröse, N. Vlassis, and R. Bunschoten, "Omnidirectional vision for appearance-based robot localization," in *Revised Papers from the International Workshop on Sensor Based Intelligent Robots*. Springer-Verlag, 2002, pp. 39–50.

[14] N. Vlassis, B. Terwijn, and B. Kröse, "Auxillary particle filter robot localization from high-dimensional sensor observations," in *Proc. of the International Conference on Robotics and Automation (ICRA)*, 2002.

[15] S. Se, D. Lowe, and J. Little, "Vision-based global localization and mapping for mobile robots," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 364–375, 2005.

[16] N. Yazawa, H. Uchiyama, H. Saito, M. Servieres, and G. Moreau, "Image based view localization system retrieving from a panorama database by surf," in *Proc. of the IAPR Conference on Machine Vision Applications*, 2009.

[17] M. Berna, B. Lisien, B. Sellner, G. Gordon, F. Pfenning, and S. Thrun, "A learning algorithm for localizing people based on wireless signal strength that uses labeled and unlabeled data," in *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.

[18] B. Ferris, D. Haehnel, and D. Fox, "Gaussian processes for signal strength-based location estimation," in *Proc. of Robotics: Science and Systems (RSS)*, 2006.

[19] F. Duvallet and A. Tews, "Wifi position estimation in industrial environments using gaussian processes," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[20] J. Letchner, D. Fox, and A. LaMarce, "Large-scale localization from wireless signal strength," in *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2005.

[21] A. Haeberlen, E. Flannery, A. Ladd, A. Rudys, D. Wallach, and L. Kavraki, "Practical robust localization over large-scale 802.11 wirless networks," in *Proc. of the International Conference on Mobile Computing and Networking*, 2004.

[22] H. Lim, L. Kung, J. Hou, and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *Proc. of IEEE INFOCOM*, 2006.

[23] P. Bolliger, "Redpin - adaptive, zero-configuration indoor localization through user collaboration," in *Proc. of the First ACM Int. Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, 2008.

[24] C. Zhou, Y. Wei, and T. Tan, "Mobile robot self-localization based on global visual appearance features," 2003.

[25] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *IEEE Transactions on Robotics*, 2006.

[26] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *In Workshop on Statistical Learning in Computer Vision, ECCV*, 2004, pp. 1–22.

[27] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, 2008.

[28] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005.

[29] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," 2006.

[30] N. Roy and S. Thrun, "Coastal navigation with mobile robots," in *Advances in Neural Information Processing 12 (NIPS)*, 1999.