Biologically inspired SLAM using Wi-Fi

Reference:

Berkvens Rafael, Jacobson Adam, Milford Michael, Peremans Herbert, Weyn Maarten.- *Biologically inspired SLAM using Wi-Fi*

2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), September 14-18, 2014, Chicago, Ill., USA - ISBN 978-1-4799-6933-3 - 2014, p. 1804-1811

Handle: http://hdl.handle.net/10067/1190380151162165141

Biologically Inspired SLAM Using Wi-Fi

Rafael Berkvens[1], Adam Jacobson[2], Michael Milford[2], Herbert Peremans[1], and Maarten Weyn[1]

***Abstract*— Wi-Fi is a commonly available source of localization information in urban environments but is challenging to integrate into conventional mapping architectures. Current state of the art probabilistic Wi-Fi SLAM algorithms are limited by spatial resolution and an inability to remove the accumulation of rotational error, inherent limitations of the Wi-Fi architecture. In this paper we leverage the low quality sensory requirements and coarse metric properties of RatSLAM to localize using Wi-Fi fingerprints. To further improve performance, we present a novel sensor fusion technique that integrates camera and Wi-Fi to improve localization specificity, and use compass sensor data to remove orientation drift. We evaluate the algorithms in diverse real world indoor and outdoor environments, including an office floor, university campus and a visually aliased circular building loop. The algorithms produce topologically correct maps that are superior to those produced using only a single sensor modality.**

I. Introduction

Simultaneous Localization And Mapping (SLAM) is the process of creating a map of an unknown environment, while at the same time localizing within that map [1]. A robot requires self motion information and a method of recognizing when it has revisited a particular location to perform SLAM. Self motion information is typically acquired using a proprioceptive sensor, such as wheel encoders or visual odometry, whereas place recognition is achieved through using an exteroceptive sensor such as cameras, laser range scanners, or antennas measuring radio frequency (RF) signals. Recently, researchers have presented the method of fingerprinting, seen in Fig. 1, to represent locations in the environment [2]–[7]. Current state of the art Wi-Fi SLAM algorithms have limited spatial resolution and do not address errors in rotational odometry.

In this paper, we use a biologically inspired mapping system to enable more relaxed sensor assumptions and reduced pose estimation requirements [8], [9]. The use of a biologically inspired mapping system also removes the need for initialization using external sensing modalities (GPS), global references, or prior mapping data to constrain odometry error [7]. We present a method for incorporating an exteroceptive RF sensor in the form of Wi-Fi with a model of the rodents' hippocampus by using the RatSLAM algorithm [10], enabling the implementation of a novel sensor fusion algorithm and use a compass to remove rotational error.

[1]Rafael Berkvens, Maarten Weyn, and Herbert Peremans are with University of Antwerp, Antwerp, Belgium `rafael.berkvens@uantwerpen.be`

[2]Adam Jacobson and Michael Milford are with the School of Electrical Engineering and Computer Science at the Queensland University of Technology, Brisbane, Australia `adam.jacobson@connect.qut.edu.au`

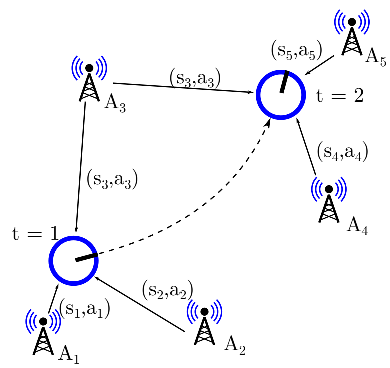


Fig. 1. Schematic overview of how Wi-Fi access points are collected in fingerprints. As the robot moves from $t = 1$ to $t = 2$, the Wi-Fi access points in range and their respective received signal strengths change, creating a measurement which enables localization.

In this paper, we make the following contributions: implementation of RatSLAM utilizing the Wi-Fi localization model by Weyn [2]; implementation of sensor fusion between Wi-Fi and camera; implementation of both Wi-Fi and compass; and results utilizing Wi-Fi in four diverse environments. The focus of this research is testing the validity of Wi-Fi as a sensor for the biologically inspired RatSLAM system by providing substantial qualitative results.

The only sensor assumption required for implementation into the RatSLAM system is that the sensor data is locally repeatable and generally globally unique. Unique environmental sensor data is typically captured using low resolution images [9], [11], however, it has been shown that it is possible to integrate alternative sensing devices, such as cameras, lasers, depth cameras, and biomimetic sonar into the RatSLAM algorithm [12], [13].

The paper continues as follows: in Section II, we briefly explain Wi-Fi localization and RatSLAM; in Section III, we give an overview of our proposed Wi-Fi SLAM algorithm and sensor fusion techniques; in Section IV we describe our our testing platforms and environment; in Section V we show the results of our approach in four different environments; and in Section VI we make our conclusion and discuss some future work.

II. Related Work

Wi-Fi localization is typically performed using fingerprinting. Fingerprinting is a form of pattern matching, where the received signal strengths (RSS) at a certain location of different transmission devices are stored as an identification for localization. In Wi-Fi fingerprinting, these RSS values can be obtained by querying the access points in range. The returned RSS values can then be linked to the device address of the corresponding access point. A fingerprint z taken at time t would be written as:

$$z_t = \left\{(s_1^t, a_1^t), (s_2^t, a_2^t), \dots, (s_i^t, a_i^t)\right\} \quad (1)$$

where s_i is the RSS value received from access point a_i, as seen in Fig. 1.

To perform localization using Wi-Fi fingerprints, they must first be measured and stored in a database, linked to the location where they were captured. Subsequently, when attempting to localize in the environment, a fingerprint of the environment is created and matched against the values in the database. In a naive implementation, the location linked to the fingerprint in the database that is most similar to the measured fingerprint can be used for localization.

In the state of the art of Wi-Fi SLAM we distinguish three other approaches than our own, which are discussed below. Ferris et al. [3] created a Wi-Fi SLAM algorithm based on Gaussian Process Latent Variable Models (GP-LVM). Huang et al. [5] improves this method by applying GraphSLAM [14]. Wi-Fi GraphSLAM argued its superiority to GP-LVM based Wi-Fi SLAM because it both scales better computationally, reducing the $O(N^3)$ operations of GP-LVM to $O(N^2)$, and is applicable in environments that are only sparsely distributed with Wi-Fi signals, where GP-LVM requires dense environments. They also relieve the hard constraint imposed by GP-LVM that similar fingerprints come from the same location. By nature of GraphSLAM, this is an offline SLAM system, which means that it can only be processed after all the measurements for an experiment are collected. Lastly, Faragher et al. [7] created a Wi-Fi SLAM algorithm based on Distributed Particle SLAM (DPSLAM) which used an approach based on gridmapping the environment with fingerprints. They used an initialization by GPS data and opportunistically used wireless data when available. This resulted in an efficiently calculated, online estimation of the robot's location.

None of the the current literature present any solutions for the limited spatial resolution of the Wi-Fi sensor modality. Furthermore, no current state of the art algorithm has attempted to address the rotational ambiguity of Wi-Fi, instead present results where paths are constantly traversed in the same direction. This rotational ambiguity removes the capability of the systems to resolve rotational errors.

We are interested in the applicability of Wi-Fi as a sensor for biologically inspired SLAM in the form of RatSLAM. As shown in Fig. 2, RatSLAM consists of three main elements: the local view templates, the pose cell network, and the experience map. The RatSLAM architecture allows the agnostic implementation of sensor data and remove odometry constraints which are present in other mapping systems [7], [12]. The local view cells store sensory snapshots of the environment, called local view templates. These local view templates are used to recognize environments which have been seen before, by comparing the input of the exteroceptive sensor with the network of templates. When the new input matches with an existing local view template, energy is injected in the pose cell network which facilitates loop closure. The pose cell network filters the local view template matches based on the odometry signal and the repetition of matches, to prevent the occurrence of false positive loop closures. The experience map creates a geometrically consistent, topological map from the matches in the local view templates and the information from the pose cell network. For more details, see [11].

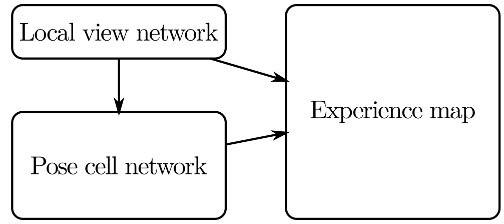


Fig. 2. RatSLAM schematic overview. The local view cells encode snapshots of the environment, the pose cells represent the robots' current internal pose belief, and the experience map is a topological map based on the combination of the pose and local view cells [11].

There have been many attempts to improve the place recognition for topological SLAM systems using fusion between multiple sensor modalities [15], [16]. These systems attempt to fuse feature based localization techniques to produce a richer sensory snapshot of the environment. Attempts have also been made to fuse and evaluate sensor reliability agnostically of their sensor or data type [12]. None of these methods have explicitly presented a way to reliably integrate the localization information present within Wi-Fi data with other sensor modalities.

III. Approach

In this section, we describe our novel Wi-Fi RatSLAM algorithm. First, we will explain how we utilized the compass sensor. Subsequently, we discuss the measurement model used. We also detail the developed sensor fusion algorithm, the fusion between camera and Wi-Fi using RatSLAM.

A. Compass

Wi-Fi localization can identify a location as novel or familiar, but it is incapable of reliably determining the orientation of the robot at that location. Sensors like cameras have the potential to calculate the relative change in orientation. This is not possible with Wi-Fi, limiting operation of Wi-Fi SLAM systems.

The relative orientation is used to help the RatSLAM algorithm remove errors associated with poor rotational odometry:

$$\Delta\theta(j) = \theta_j - \theta_t \quad (2)$$

where θ_j is the heading direction of the measurement z_j stored in the database with which the current measurement z_t matched. This relative orientation is used when creating nodes in the experience map. The value of $\Delta\theta(j)$ is essential when relaxing the pose graph on loop closure.

B. Wi-Fi

The Wi-Fi localization method described above is similar to the appearance based input matching of the RatSLAM algorithm. The key difference between the classical input for RatSLAM and Wi-Fi fingerprints is that the fingerprints must be compared by access point, some of which will only be present at certain locations in the environment. Additionally, Wi-Fi is a signal with typically more noise than classical RatSLAM inputs.

Another difference between the classical RatSLAM sensors and our new Wi-Fi sensor is the sampling rate. Video cameras, laser range scanners, and other sensors typically scan in the order of 10 Hz or more. While RatSLAM does not inherently require such high frequencies, our driver only reaches a sample rate of about 1 Hz, which causes map sparseness. This behavior is consistent with other state of the art algorithms [5], [7].

Each new fingerprint is matched to the existing database, and only if the fingerprint is found to be novel, it will be stored in the local view network and linked to a location in the pose cell network. The order in which access points are received can differ from measurement to measurement, as the timing of replies from access points is not always the same and some access points dynamically switch channels. Additionally, not every access point is seen in every scan taken at the same location. When an access point is found in the latest measured fingerprint and in the fingerprint j in the database, the weight for that access point is combined with the other weights as follows:

$$w_{hit}(j) = \prod_{x=0}^{H-1} e^{-\frac{(s_x - s_x^j)^2}{2\sigma^2}} \tag{3}$$

where H is the number of such matching access points.

Additional information can be attained from any access points which are missing or extra when comparing the current and previous scans. If the fingerprints are very similar except for an extra access point with a high RSS value, it is unlikely that they are taken at the same location. If the extra access point has a very low RSS value, it carries little information, as these access points are less likely to be seen in every scan. This idea leads to the incorporation of a weighting scheme for extra and missing access points:

$$w_{extra}(j) = \prod_{x=0}^{E-1} e^{-\frac{(P_{extra}(s_x))^2}{2\sigma^2}} \tag{4}$$

$$w_{miss}(j) = \prod_{x=0}^{M-1} e^{-\frac{(P_{miss}(s_x^j))^2}{2\sigma^2}} \tag{5}$$

where the penalties P_{extra} and P_{miss} are distributed as indicated by Weyn [2]. The number of extra access points in the latest measured fingerprint is E. M is the number of access points that are missing from the latest fingerprint when compared to the fingerprint j in the database. In (3), (4), and (5), σ^2 is equal to 20 dBm^2, based on the findings by Chiou et al. [17] to accommodate for sampling errors and antenna direction. The kernel choice is discussed in [2].

The total weight is then:

$$w(j) = \sqrt[H]{w_{hit}(j)} \cdot w_{extra}(j) \cdot w_{miss}(j) \cdot \tfrac{H}{H+E+M} \tag{6}$$

where the last term is intended to give matching access points a higher weight. This weight is used both as similarity measure and for determining if a fingerprint is novel.

C. Camera Template Matching

To reduce computation requirements, camera images are down-sampled to a resolution of 12×9 and stored as a single line vector. It has previously been shown that low resolution images as low as 1 pixel holds sufficient unique and locally repeatable data to enable localization [18]. The camera templates are evaluated using a kernel matching technique to determine the similarity between the current camera view and all previously learned camera templates:

$$c(j) = \prod_{x=0}^{s-1} e^{\frac{(p_x - p_x^j)^2}{2\sigma^2}} \tag{7}$$

where p_x and p_x^j is respectively the value of element x in the current camera template and the jth camera template. The length of the camera template is denoted by s and σ is the camera kernel width.

D. Wi-Fi Camera Sensor Fusion

Here, we present a method for fusing camera and Wi-Fi information. Sensor information is evaluated using kernel matching techniques for both sensor modalities producing a similarity score between samples for the camera and the Wi-Fi sensors allowing the generation of a place recognition signal. To account for the different sampling rates of both sensors, the Wi-Fi sensor is evaluated at the same rate as the camera information, utilizing the most current available Wi-Fi snapshot.

Camera and Wi-Fi information is then evaluated independently, the best template match b is the template with the highest similarity score determined using:

$$b_{camera} = \arg\max_{0 \le j < n} c(j) \tag{8}$$

$$b_{wifi} = \arg\max_{0 \le j < n} w(j) \tag{9}$$

where n is the number of templates. Using explicit sensor knowledge allows us to design a complementary sensor fusion system, which utilizes different sensors modalities when appropriate. Wi-Fi data is a sensor modality with a broad place recognition signal, whereas the camera sensor modality produces extremely accurate place recognition signal whilst within the same perceptual environment. Here we utilize the camera place recognition signal while it is available and supplement it with Wi-Fi location data once the camera is no longer useful. The algorithm evaluates each sensor's utilization based on whether the current template is

(a) The Pioneer 3DX with laptop mounted on top, used for the office experimentation.

(b) The laptop and web camera used for the large scale campus experimentation.

Fig. 3. The two sensor setups used for the experiments in this paper.

a template match to a previously stored template. A template is determined to be a match, m, for each sensor using:

$$m_{wifi} = \begin{cases} 1 & \text{if } w(b) \geq t_{wifi} \text{ and } |n - b_{wifi}| \geq k \\ 0 & \text{if } else \end{cases} \tag{10}$$

$$m_{camera} = \begin{cases} 1 & \text{if } c(b) \geq t_{camera} \text{ and } |n - b_{camera}| \geq k \\ 0 & \text{if } else \end{cases} \tag{11}$$

where t_{wifi} and t_{camera} are the similarity score thresholds for the Wi-Fi and camera sensors respectively. The recency threshold, k, stops false template matches to samples within the last k frames. For each sensor, $m = 1$, indicates that the template match b is a familiar template and is currently stored within the template database. If $m = 0$, the current template is a novel template; only when template matches from both sensors are deemed to be novel, a new template is added to the database. Depending on the available template matches from the sensing modalities, the best global template match b_{out} is input to the RatSLAM system in place of the local view cells, where the best global template match is defined by:

$$b_{out} = \begin{cases} b_{camera} & \text{if } m_{camera} = 1 \\ b_{wifi} & \text{if } m_{camera} = 0 \text{ and } m_{wifi} = 1 \\ n & \text{if } else \end{cases} \tag{12}$$

This algorithm allows the discrimination between sensing modalities as they become available.

IV. Experimental Setup

Here, we present the experimental setup, including details of the data acquisition platforms and testing environments.

A. Data Acquisition Platforms

Two testing platforms were utilized to capture the experiments' data. The office, orientation, and circular building experiments were performed using a Pioneer 3DX mobile robot with a consumer grade laptop mounted on top of it, see Fig. 3a. An additional front facing web camera was placed on the robot, to enable comparison to the visual RatSLAM algorithm. A Shimmer 9DOF Kinematics IMU was placed on the robot, of which the digital compass was used in the circular building experiment. It is elevated by a cardboard box to minimize the effect of the inference created by the robot's metal parts. The robot drove at a fixed speed of approximately 0.25 m/s and its wheel encoders were used as the odometry signal.

For the large scale campus experiment, the data acquisition platform consisted of a standard consumer grade laptop with a web camera mounted onboard, see Fig. 3b. The odometry signal was collected using a constant velocity model combined with low resolution visual odometry for rotation.

Both laptops collected the Wi-Fi fingerprints using a custom driver for the Robot Operating System (ROS) [19], based on the Wireless Extensions for Linux [20].

B. Testing Environments

The office experiment consisted of driving the robot three times up and down an office corridor. The physical run is drawn in Fig. 4a. It starts from location A and goes to location B, where the robot turns and goes back along the same path, repeating this three times.

The path traversed for the large scale campus experiment is shown in Fig. 4b. The path is approximately one kilometer long, moving through buildings and outside areas, ensuring large amount of perceptual change through day and night cycles. The path is traversed once during the day and once at night.

The circular building experiment consists of driving the robot around in a university campus with a circular layout, shown in Fig. 4c. The robot drives two and a half times in one direction, after which it turns, and drives the same trajectory for two and a half times in the opposite direction. The traveling of the opposite direction is done explicitly, since other research seldom explicitly explores this possibility. This run is drawn in Fig. 4c. The total path is approximately 750 m long.

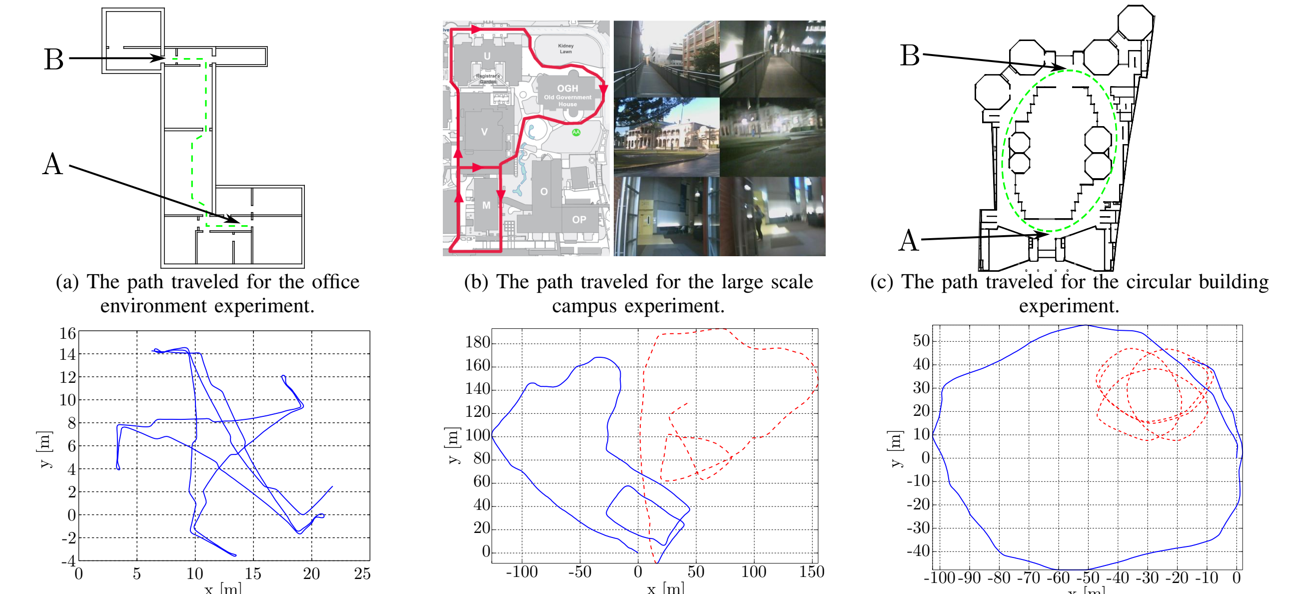
The parameters used for the office, large scale campus, and circular building experiments can be found in Table I. A detailed explanation on how to find these parameters can be read in [21]. These parameters are hand tuned, but we are looking into automating this process [22].

V. Results

In this section, we present results from four experiments, showing experience maps and template graphs for the office, large scale campus environment and the circular building environment.

A. Office experiment

The Wi-Fi template matching result is shown in Fig. 5a. The vertical axis shows the template and experience ID numbers and the horizontal axis shows the time in seconds. A template is a unique sensory snapshot captured by the robot.



(a) The path traveled for the office environment experiment.

(b) The path traveled for the large scale campus experiment.

(c) The path traveled for the circular building experiment.

(d) The traveled path for the office experiment according to the robot's odometry alone.

(e) The traveled path for the large scale campus experiment according to the visual odometry alone.

(f) The traveled path for the circular building experiment according to the robot's odometry alone.

Fig. 4. Overview of experiment setups.

TABLE I
PARAMETER LIST USED FOR THE RATSLAM EXPERIMENTS. MORE DETAILS ON THE PARAMETERS CAN BE FOUND IN [21].

Office Experiment

Parameter	Wi-Fi value	Description
t_{wifi}	0.6	Sensor recognition parameter
k	20	Recency threshold to prevent false positive matches
`pc_cell_x_size`	1	Size of the pose cell in the RatSLAM system
`pc_vt_inject_energy`	0.03	Pose cell energy injection per template match
σ	4.47	Kernel width

Large Scale Campus Experiment

Parameter	Wi-Fi value	Description
t_{wifi}	0.25	Wi-Fi sensor recognition threshold
t_{camera}	0.96	Camera sensor recognition threshold
k	4s	Recency threshold for camera only experiment
k	40s	Recency threshold for Wi-Fi only experiment
k	40s	Recency threshold for sensor fusion experiment
`pc_cell_x_size`	1	Size of the pose cell in the RatSLAM system
`pc_vt_inject_energy`	0.2	Pose cell energy injection per template match
σ_{wifi}	4.47	Kernel width for Wi-Fi
σ_{camera}	2.0	Kernel width for camera

Circular Building Experiment

Parameter	Wi-Fi value	Description
t_{wifi}	0.2	Wi-Fi sensor recognition threshold
k	40s	Recency threshold for Wi-Fi only experiment
`pc_cell_x_size`	1	Size of the pose cell in the RatSLAM system
`pc_vt_inject_energy`	0.1	Pose cell energy injection per template match
σ_{wifi}	4.47	Kernel width for Wi-Fi

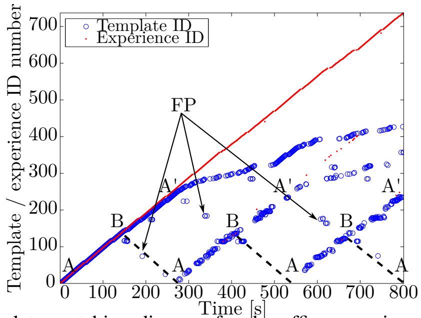
A template match is returned when the current sensory data is sufficiently similar to stored template, where each circle represents when a particular template was created or matched. Each dot represents when an experience was created or linked. An experience is a node within the experience map created by the pose cells based on sensory and pose data. The experience matches are created after filtering is performed by the pose cells which has the ability to ignore sensory template matches based on pose information and prior belief. It can be seen in Fig. 5a that there are three diagonal lines going up, indicating the three runs of the experiment. For clarity, the locations A and B are indicated on the template matching graph. While the lines show a correct transition from location A to location B, they continue in an unknown environment to location A', which is in fact location A. The dashed lines show where the robot returns to A.

In this situation, the correct Wi-Fi template matches that are found along the dashed lines or their extension, indicated by FP, are false positive matches to the algorithm. They match to the same location, but a different orientation, and when using Wi-Fi alone there is no way to reliably determine the orientation of the robot.

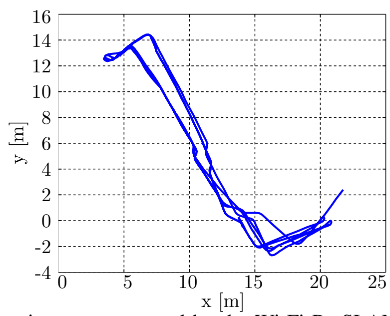
The experience map created by Wi-Fi RatSLAM for the office experiment is shown in Fig. 5b. A clear improvement compared to the raw odometry, shown in Fig. 4d is visible, in addition to a resemblance of the office environment and trajectory shown in Fig. 4a. Note that two paths, one for going up the run, one for going down are created, while in fact these are the same locations.

B. Large scale campus experiment

To provide a baseline performance to compare against, Fig. 4e shows the map produced by a raw odometry signal without the aid of a SLAM system for the large scale campus experiment. The solid blue path was traversed during the day, the dashed red path was traversed during the night. It can be seen that in this map, there is a large amount of odometric drift attributed to the visual odometry system. Using this odometry signal with RatSLAM utilizing camera template matching does not provide much improvement to the topological map, as seen in Fig 6e. There are few loop closures occurring in the map, and this loop closure is contained to matching within the same perceptual conditions. This is evident in the template graph, seen in Fig. 6b, where the template matches in the day region only match to the day



(a) The template matching diagram for the office experiment. The labels A and B are added for correspondence with Fig. 4a.



(b) The experience map created by the Wi-Fi RatSLAM algorithm.

Fig. 5. The RatSLAM template graph and experience map for the office experiment.

region and templates during the night traverse only match to other night templates. This is a fairly standard response for a camera operating in day/night conditions. However, it can be seen in the template graph produced utilizing the Wi-Fi implementation of RatSLAM, seen in Fig. 6a, that there is a massive increase in template matches, not only through times of similar perceptual conditions but across the transition from day to night.

The results produced utilizing the sensor fusion, see Fig. 6c and 6f, clearly show that where the camera template matches are available, there is an improvement in recognition in the template graphs and there is also an increase in reliable template matches through those regions. The resulting map from the sensor fusion system is slightly distorted due to odometric error and a number of missed loop closures within the lower section of the map caused by the RatSLAM system. However, it can be seen that there is no false loop closure within the map.

C. *Orientation experiment*

For this experiment, we placed our robot in four different reference orientations and created a fingerprint for that state. Then, we drove the robot 2 m away from that reference starting position, taking 20 fingerprints at 1 Hz. We repeated that trajectory every 10°. The measurements were matched using a kernel width of 12 dBm^2, a width that attempts to increase the orientation sensitivity, only taking sampling error into account according to Chiou et al. [17]. To establish a relative orientation estimation, we should have an indication of orientation when matching two fingerprints.

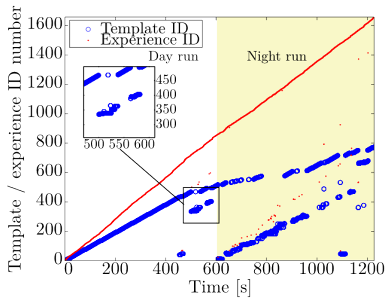
To generalize in time, we took the mean of three of these experiments. The mean variation of these experiments was 0.0130. The variation of the match quality is visualized in Fig. 7. The match quality is about the same for any orientation. Thus we conclude that for our setup, Wi-Fi fingerprints are independent from the orientation they are measured in, since there is no consequent indication of orientation when matching two fingerprints.

D. *Circular building experiment*

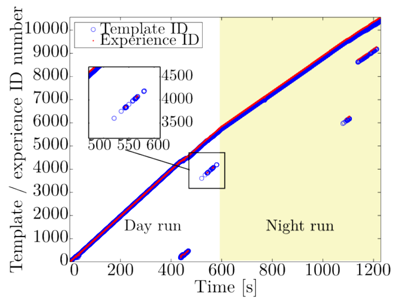
The map produced by the raw odometry signal is shown to compare the SLAM result against, see Fig. 4f. The solid blue path is traversed in one direction, the dashed red path is traversed in the opposite direction. Results for using only Wi-Fi or camera are shown in Fig. 8a, 8b, 8d, and 8e. RatSLAM using only Wi-Fi as exteroceptive sensor fails because of the previously described orientation problem, rotational error grows until the map is totally corrupted. The Wi-Fi template matches are visualized in Fig. 8a. The trend of the two and a half repetition is clearly visible, after which the robot turns, around the 1500 second mark, and the pattern repeats itself in reverse order. This pattern is also visible in the experience matches. Do note the matches indicated by label F in Fig. 8a, which are due to the unreliable orientation dependency of the Wi-Fi signal, creating an experience graph similar to the camera graph, see below. The useful matches are those that match to the experience created when going in the other direction, indicated by label G in Fig. 8a. However, these matches cannot determine in what direction a location is encountered, corrupting both directions of the experience map.

The camera template graph (Fig. 8b) demonstrates the behavior of a sensor which is capable of creating localization matches in a single orientation. The template matches are only made when traversing in the same direction, when retracing the path in the opposite direction, localization cannot be performed. The pattern in the camera template matches before the robot turns around the 1500th second is repeated but not in reverse order, as with the Wi-Fi sensor. In other words, to the camera it would seem as if an entirely new run was performed after the robot turned. This is also visible in Fig. 8e, where the experience map is shown that was created using camera RatSLAM. It can be seen that there are two different regions in the map, corresponding with the first and second half of the experiment. These are caused by the accumulated wheel encoder error and lack of loop closure from the camera. This in turn is due to the orientation dependent nature of the camera.

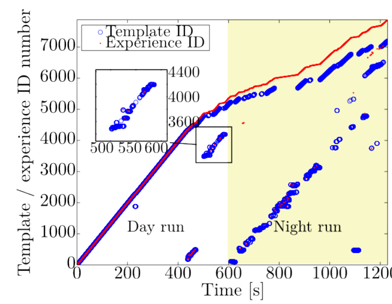
The template matches of using both Wi-Fi and compass, shown in Fig. 8c, are identical to the view template matches of using Wi-Fi alone. A location should be able to be relocated independent of the orientation it is found in. In Fig. 8d the RatSLAM experience map result is shown when using Wi-Fi and the magnetic compass, which clearly resembles the physical shape of the trajectory. Also note the removal of any experience match to experiences recorded after the 1500th second, when the robot turned. This performance is due to the capability of the system to remove rotational error



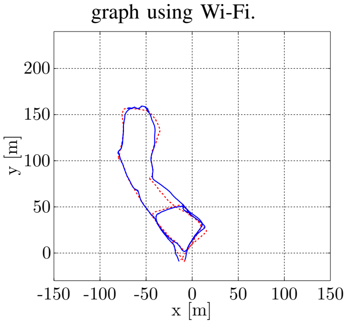
(a) The RatSLAM template and experience graph using Wi-Fi.



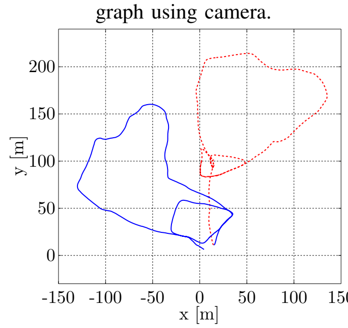
(b) The RatSLAM template and experience graph using camera.



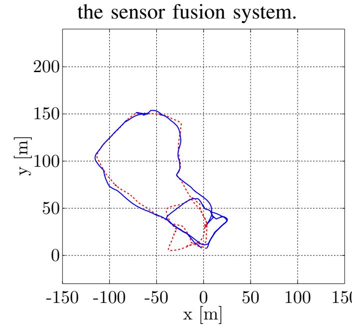
(c) The RatSLAM template matching using the sensor fusion system.



(d) The RatSLAM experience map using Wi-Fi.



(e) The RatSLAM experience map using camera.



(f) The RatSLAM experience map using the sensor fusion system.

Fig. 6. The RatSLAM template graphs and experience maps for Wi-Fi, camera, and the sensor fusion system on the large scale campus environment. The day and night run are indicated by background shading on the template graphs. On the experience maps, the day run is shown as a blue solid line while the night run is a red dotted line.

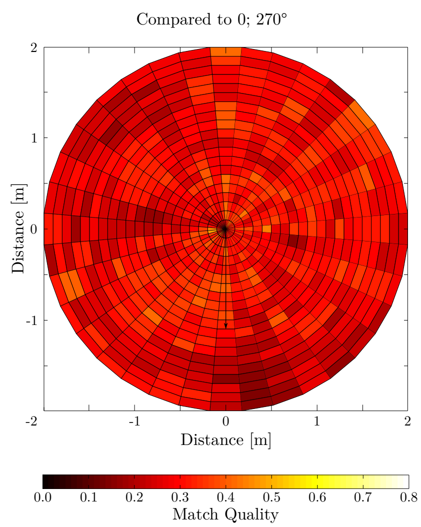


Fig. 7. Local view template similarity. The total weight $w(j)$, see (6), drawn for a reference point at $0; 270°$. An arrow indicates the reference orientation, which is also where we would expect higher match quality if our sensor was orientation sensitive.

from the erroneous wheel encoder data and accurately perform loop closure between the two regions of the experiment.

VI. Conclusion and Future Work

We have shown that Wi-Fi is a valid and useful sensor to use in the RatSLAM algorithm. The sensor has been used across three distinctly different environments producing minimal false positives and enabling the creation of topologically correct maps. Sensor fusion between Wi-Fi and camera have allowed improvements in the spatial resolution of the Wi-Fi sensor. We have also eliminated the problem of orientation independent fingerprints by using a digital compass and shown that it gives superior results.

Sensor fusion allows the minimization of the flaws from each sensor. Combining sensor modalities enables processing at a higher frame rate (through sampling the Wi-Fi readings at the camera frequency), and allows the system to dynamically utilize available sensing modalities.

We would like to extend this work to opportunistically include more radio frequency sensors, such as Bluetooth, RFID, or DASH7. Additionally, we are in the process of implementing other Wi-Fi SLAM techniques to create an objective comparison. Future work will endeavor to fuse multiple sensory modalities into a multi-scaled biologically inspired localization framework to enable improvements in localization performance, allow sensory redundancy and enable graceful degradation in localization accuracy.

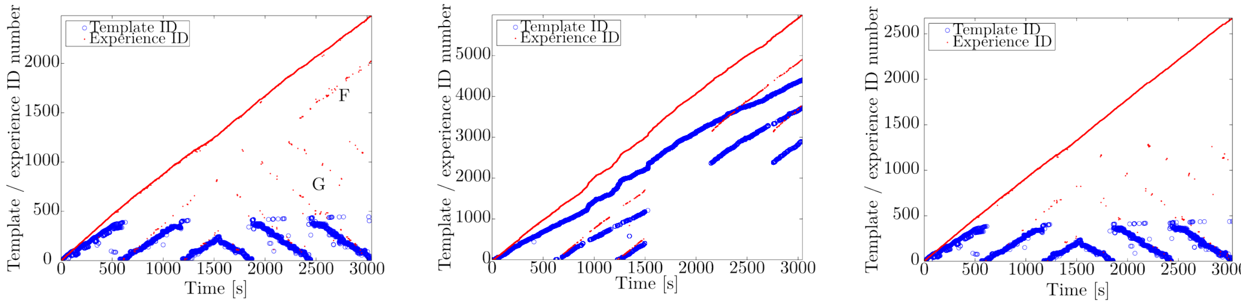
References

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge: The MIT Press, 2006.

[2] M. Weyn, "Opportunistic Seamless Localization," Ph.D. dissertation, University of Antwerp, 2011.

[3] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," *Proc. 20th Int. Jt. Conf. Artif. Intell.*, pp. 2480–2485, 2007.

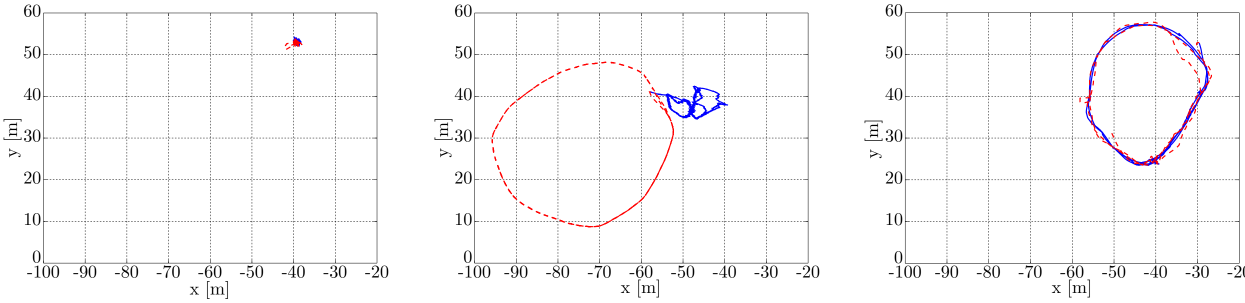
[4] J.-S. Gutmann, G. Brisson, E. Eade, P. Fong, and M. Munich, "Vector Field SLAM," in *Robot. Autom. (ICRA), 2010 IEEE Int. Conf.* Anchorage, Alaska: IEEE, May 2010, pp. 236–242.



(a) The RatSLAM template and experience graph using Wi-Fi.

(b) The RatSLAM template and experience graph using camera.

(c) The RatSLAM template and experience graph using Wi-Fi and compass.



(d) The RatSLAM experience map using Wi-Fi.

(e) The RatSLAM experience map using camera.

(f) The RatSLAM experience map using Wi-Fi and compass.

Fig. 8. The RatSLAM template graphs and experience maps for Wi-Fi RatSLAM for the circular building experiment. The first half of the experiment is shown as a blue solid line in the experience maps. The second half of the experiment is shown as a red dashed line in the experience maps.

[5] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Aggarwal, "Efficient, generalized indoor WiFi GraphSLAM," in *Robot. Autom. (ICRA), 2011 IEEE Int. Conf.* Shanghai, China: IEEE, May 2011, pp. 1038–1043.

[6] L. Bruno and P. Robertson, "WiSLAM: Improving FootSLAM with WiFi," in *Indoor Position. Indoor Navig. (IPIN), 2011 Int. Conf.* Guimaraes, Portugal: IEEE, 2011, pp. 1–10.

[7] R. M. Faragher, C. Sarno, and M. Newman, "Opportunistic radio SLAM for indoor navigation using smartphone sensors," in *Position Locat. Navig. Symp. (PLANS), 2012 IEEE/ION*. Myrtle Beach, SC: IEEE, Apr. 2012, pp. 120–128.

[8] B. Tian, V. A. Shim, M. Yuan, C. Srinivasan, H. Tang, and H. Li, "RGB-D based cognitive map building and navigation," in *Intell. Robot. Syst. (IROS), 2013 IEEE/RSJ Int. Conf.*, Nov. 2013, pp. 1562–1567.

[9] M. Milford and G. Wyeth, "Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1038–1053, Oct. 2008.

[10] ——, "Persistent navigation and mapping using a biologically inspired SLAM system," *Int. J. Rob. Res.*, vol. 29, no. 9, pp. 1131–1153, 2010.

[11] G. Wyeth and M. Milford, "Spatial cognition for robots," *IEEE Robot. Autom. Mag.*, vol. 16, no. 3, pp. 24–32, Sept. 2009.

[12] M. Milford and A. Jacobson, "Brain-inspired Sensor Fusion for Navigating Robots," in *Robot. Autom. (ICRA), 2013 IEEE Int. Conf.* Karlsruhe, Germany: IEEE, 2013, pp. 2891–2898.

[13] J. Steckel and H. Peremans, "BatSLAM: Simultaneous localization and mapping using biomimetic sonar." *PLoS One*, vol. 8, no. 1, p. e54076, Jan. 2013.

[14] S. Thrun and M. Montemerlo, "The Graph SLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures," *Int. J. Robot. Res.*, vol. 25, no. 5-6, pp. 403–429, 2006.

[15] J. Collier, S. Se, and V. Kotamraju, "Multi-sensor Appearance-Based Place Recognition," in *Comput. Robot Vis. (CRV), 2013 Int. Conf.* IEEE, May 2013, pp. 128–135.

[16] W. Maddern and S. Vidas, "Towards robust night and day place recognition using visible and thermal imaging," *RSS 2012 Beyond laser Vis. Altern. Sens. Tech. Robot. Percept.*, 2012.

[17] Y.-S. Chiou, C.-L. Wang, S.-C. Yeh, and M.-Y. Su, "Design of an adaptive positioning system based on WiFi radio signals," *Comput. Commun.*, vol. 32, no. 7-10, pp. 1245–1254, May 2009.

[18] M. Milford, "Vision-based place recognition: how low can you go?" *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 766–789, 2013.

[19] M. Quigley, K. Conley, B. Gerkey, T. Foote, J. Leibs, R. Wheeler, and A. Ng, "ROS: an open-source Robot Operating System," in *ICRA Work. open source Softw.*, 2009.

[20] J. Tourrilhes, "Wireless Tools for Linux," 2009. [Online]. Available: http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html

[21] D. Ball, S. Heath, J. Wiles, G. Wyeth, P. Corke, and M. Milford, "OpenRatSLAM: an open source brain-based SLAM system," *Auton. Robots*, pp. 1–28, Feb. 2013.

[22] A. Jacobson, Z. Chen, and M. Milford, "Autonomous Multisensor Calibration and Closed-loop Fusion for SLAM," *J. F. Robot.*, Feb. 2014.