

# FreMEn: Frequency Map Enhancement for Long-Term Mobile Robot Autonomy in Changing Environments

Tomáš Krajník, Jaime P. Fentanes, João M. Santos, and Tom Duckett

**Abstract**—We present a new approach to long-term mobile robot mapping in dynamic indoor environments. Unlike traditional world models that are tailored to represent static scenes, our approach explicitly models environmental dynamics. We assume that some of the hidden processes that influence the dynamic environment states are periodic and model the uncertainty of the estimated state variables by their frequency spectra. The spectral model can represent arbitrary timescales of environment dynamics with low memory requirements. Transformation of the spectral model to the time domain allows for the prediction of the future environment states, which improves the robot's long-term performance in changing environments. Experiments performed over time periods of months to years demonstrate that the approach can efficiently represent large numbers of observations and reliably predict future environment states. The experiments indicate that the model's predictive capabilities improve mobile robot localization and navigation in changing environments.

**Index Terms**—Localization, long-term autonomy, mapping.

## I. INTRODUCTION

**A**DVANCES in the field of mobile robotics are gradually enabling long-term deployment of autonomous robots in human environments. As these environments change over time, the robots have to deal with the fact that their world knowledge is incomplete and uncertain. Although probabilistic mapping methods [1] have demonstrated the ability to represent incomplete knowledge about the environment, they generally assume that the corresponding uncertainty is caused by inherent sensor noise rather than by natural processes that cause the environment to change over time. Thus, traditional mapping methods treat measurements of dynamic environment states as outliers [2]. This undermines the ability of the mapping methods to reflect

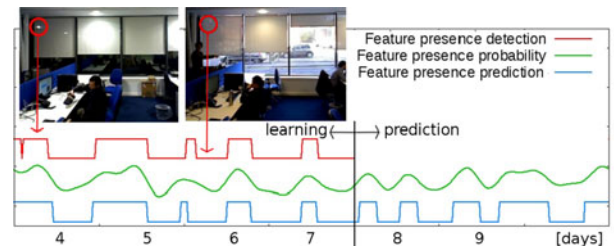


Fig. 1. Frequency-enhanced model of a single image feature visibility. The observations of image feature visibility (red) are processed by the FreMEn method that extracts the time-dependent probability of the feature being visible (green). This allows us to reconstruct and predict the feature's visibility for a given time (blue).

the environment dynamics and provide support for long-term mobile robot autonomy. Recent works have demonstrated that exploiting the outlying measurements allows us to characterize some environment changes, which improves robot localization in changing environments [3]–[6].

In our approach, we assume that some of the mid- to long-term processes that exhibit themselves through environment changes are periodic. These processes can be both natural, e.g., seasonal foliage changes, or artificial, e.g., human activities characterized by regular routines. Regardless of the primary cause of these processes, we hypothesize that the regularity of the environment changes can be exploited by robots to build more robust representations of their surroundings. We propose to represent the probability of the elementary environment states by combination of harmonic functions whose amplitudes and periodicities relate to the influences and frequencies of the hidden processes that cause the environment variations. This allows for efficient representation of the spatiotemporal dynamics as well as prediction of future environment states. To obtain the parameters of the harmonic functions, we propose to treat the long-term observations of the environment states as signals, which can be analyzed in the frequency domain.

An advantage of our approach is its universal applicability—it can introduce dynamics to any stationary environment model that represents the world as a set of independent components. Introduction of the dynamics is achieved simply by representing the uncertainty of the elementary states as probabilistic functions of time instead of constants that are updated only when the given state is observed by a robot. The approach, which was originally introduced in [7], was successfully applied to

Manuscript received June 5, 2016; revised December 10, 2016; accepted January 9, 2017. Date of publication March 14, 2017; date of current version August 7, 2017. This paper was recommended for publication by Associate Editor M. Kaess and Editor C. Torras upon evaluation of the reviewers' comments. This work was supported by the EU ICT Project 600623 "STRANDS" and the Czech Science Foundation under Project 17-27006Y.

T. Krajník is with the Lincoln Centre for Autonomous Systems, University of Lincoln, Lincoln LN6 7TS, U.K., and also with Faculty of Electrical Engineering, Czech Technical University, Prague 16636, Czechia (e-mail: tkrajnik@lincoln.ac.uk).

J. P. Fentanes, J. M. Santos, and T. Duckett are with the Lincoln Centre for Autonomous Systems, University of Lincoln, Lincoln LN6 7TS, U.K. (e-mail: jpulidofentanes@lincoln.ac.uk; jsantos@lincoln.ac.uk; tduckett@lincoln.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TRO.2017.2665664

landmark maps to improve localization [4] and to topological maps to improve the robotic search [8] and navigation [46]. The application of the method to occupancy grids not only reduces memory requirements [9], but also enables lifelong spatiotemporal exploration [10], [47] of changing environments. In this paper, we summarize and extend the previous results by a thorough examination of the method's ability to efficiently represent environment changes over long time periods, predict the future environment states, and use these predictions to improve the robustness of robot localization and navigation. While the main aim of our method is to deal with periodic changes, we also show that its combination with a persistence model allows us to learn and deal with nonperiodic dynamics as well.

## II. RELATED WORK

While the mapping of stationary environments has been widely studied [11] and generating large-scale stationary environment models has been in the spotlight of robotics research for a long time, mapping methods that explicitly model the environment dynamics gained importance only after robots attained the ability of autonomously operating for longer time periods [44].

The first approaches to address the problem of dynamic environments were object-centric. These methods identify moving objects and remove them from the environment representations [12], [13] or use them as moving landmarks [14], [15] for self-localization. But not all dynamic objects actually move at the moment of mapping, which means that their identification requires long-term observations. To tackle this issue, Ambrus *et al.* [16] proposed to process several 3-D point clouds of the same environment obtained over a period of several weeks to identify and separate the movable objects and refine the static environment structure at the same time. While object-centric representations can handle some problems of dynamic mapping, they still assume that most of the environment is static, which makes them unsuitable for scenarios where the environment varies significantly.

Considering this aspect, other authors propose approaches that assume the map to never be complete and perform mapping in a continuous manner, adding new features to the map every time the robot observes its environment. In these approaches, managing map size is crucial [17]–[20].

Alternatively, some authors propose systems that learn a fixed set of possible states for the dynamic objects, e.g., corresponding to open and closed doors [21], [22], which can limit the map size, but this approach is limited in the real scenarios, where the number of states is unpredictable.

Other approaches do not attempt to explicitly identify movable objects, but rely on less abstract environment representations. For example, Biber and Duckett [17] and Arbuckle *et al.* [23] represent the environment dynamics by multiple temporal models with different timescales where the best map for localization is chosen by its consistency with current readings. Dayoub *et al.* [24] and Rosen *et al.* [25] each present a feature persistence system based on temporal stability in sparse visual maps that can identify environmental features which are more likely to be stable. Yguel *et al.* [26] propose to model occupancy

grid maps in the wavelet space in order to optimize the amount of information that has to be processed for path planning.

Churchill and Newman [3] propose to integrate similar observations at the same spatial locations into “experiences” which are then associated with a given place. They show that associating each location with multiple “experiences” improves autonomous vehicle localization. Tipaldi *et al.* [5] represent the states of the environment components (cells of an occupancy grid) with a hidden Markov model and show that their representation improves localization robustness. Kucner *et al.* [27] learn conditional probabilities of neighboring cells in an occupancy grid to model typical motion patterns in dynamic environments. Another method can learn appearance changes based on a cross-seasonal dataset and use the learned model to predict the environment appearance [6] showing that state prediction can be useful for long-term place recognition in changing environments. Finally, Krajník *et al.* [7] represent the environment dynamics in the spectral domain and apply this approach to image features to improve the localization [4] and to occupancy grids to reduce memory requirements [9].

While most of the aforementioned methods are aimed specifically at the problem of lifelong localization, our approach was shown to be applicable in other scenarios as well [28]. In this paper, we extend the results and experimental analysis presented in [4], [7], and [9]. The efficiency of spatiotemporal representation, which was only briefly mentioned in [9], is now thoroughly investigated on a FreMen 4D (3D+time) occupancy grid, which represents almost 2 million observations of a small office over 112 days. Compared to the work presented in [4], which provides only a coarse evaluation compared to a naïve localization method, the experiments in this paper demonstrate how the localization robustness depends on the number of predicted visual landmarks, compare its performance to the experience-based approach [3], and present additional evaluation on outdoor datasets. This paper also demonstrates that integration of the method in the ROS navigation stack improves both the accuracy of robot localization and the efficiency of navigation.

## III. SPECTRAL REPRESENTATION FOR SPATIOTEMPORAL ENVIRONMENT MODELS

Many environment models used in mobile robotics consist of independent components that can be in two distinct states. For example, the cells of an occupancy grid are occupied or free, edges of a topological map are traversable or not, doors are open or closed, rooms are vacant or occupied, landmarks are visible or occluded, etc. The states of the real world cannot be observed directly, but through sensors that are affected by noise. Thus, the state of each world model component is uncertain, which is typically represented by the probability of a particular component being in a given state, e.g., the uncertainty of occupancy of the  $j$ th cell is typically represented by  $p_j = P(s_j = \text{occupied})$ . This allows us to counter the effect of noisy measurements by employing statistical methods, such as Bayesian filtering [1]. However, the mathematical foundations described in [1] assume a static world, i.e., the probabilities of

the world components are assumed to be constant. While this still allows us to update the environment model if a change has been observed for long enough, the old states are simply “forgotten” over time and the system does not learn from the change observed.

We propose to represent the uncertainty of the environment states not as probabilities  $p_j$ , but as probabilistic functions of time  $p_j(t)$ . Assuming that the variations of the environment are caused by a number of unknown processes, some of which exhibit periodic patterns, the  $p_j(t)$  can be represented by a combination of harmonic functions that relate to these periodic processes. To identify the parameters of these harmonic functions, we propose to use spectral analysis methods, namely the Fourier transform [29].

#### A. Fourier Transform

The Fourier Transform is a well-established mathematical tool widely used in the field of statistical signal processing. In a typical case, it transforms a function of time  $f(t)$  into a function of frequency  $F(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-j\omega t} dt$ . The function  $F(\omega)$  is commonly referred to as the frequency spectrum of  $f(t)$ . The Fourier transform is invertible, and therefore, one can recover the function  $f(t)$  from its spectrum  $F(\omega)$ , i.e.,  $f(t) = \mathcal{F}^{-1}(F(\omega))$ . If one wants to analyze or alter the periodic properties of a process characterized by a function  $f(t)$ , it is reasonable to calculate its spectrum  $F(\omega)$ , perform the analysis or alteration in the frequency domain, and then transform the altered spectrum  $F'(\omega)$  back to the temporal domain. Such a process is referred to as spectral analysis.

Typically,  $F(\omega)$  is a complex-valued function, whose absolute values and arguments correspond to the amplitudes and phase shifts of the frequency components  $\omega$ . Given that  $f(t)$  is a real periodic discrete function, the spectrum  $F(\omega)$  can be represented by a finite set of complex numbers.

#### B. Proposed Representation

Although the approach can be applied to most state-of-the-art representations, we will explain it with an occupancy grid. To keep this explanation simple, we assume that the occupancy of the individual cells is independent of each other and explain the approach on a single cell. So let us assume that at a given time  $t$ , a single cell of an occupancy grid is either *occupied* or *free*. Let us represent the state as a binary function of time  $s(t) \in \{0, 1\}$ , where  $s(t) = 0$  corresponds to the cell being *free* at time  $t$  and vice versa.

The main idea behind the proposed model is to treat the values of the function  $s(t)$  as real numbers and calculate the Frequency spectrum of the sequence  $s(t)$  by means of a (Discrete) Fourier transform as

$$S(\omega) = \mathcal{F}(s(t)). \quad (1)$$

The resulting frequency spectrum  $S(\omega)$  is a discrete complex function whose absolute values  $|S(\omega)|$  correspond to the influences of periodic processes on  $s(t)$ . In other words, each local maximum of  $|S(\omega)|$  indicates that the function  $s(t)$  might be influenced by a hidden process whose period is  $T = 2\pi/\omega$ . Since

we do not want to represent the state  $s(t)$  directly, but as a combination of  $l$  periodic processes, we select the  $l$  most prominent (i.e., of highest absolute value) coefficients of the spectrum  $\mathcal{S}(\omega)$  and store them along with their frequencies  $\omega_i$  in a set  $\mathcal{P}(\omega)$ . The coefficients stored in the set  $\mathcal{P}$  are then used to recover a function  $p(t)$  by means of the inverse Fourier transform

$$p(t) = \varsigma(\mathcal{F}^{-1}(\mathcal{P}(\omega))) \quad (2)$$

where  $\varsigma$  denotes a function that ensures that  $p(t) \in [0, 1]$ . For our purposes, we choose a simple saturation function  $\varsigma(x) = \min(\max(x, 0), 1)$ , which achieved better results than other normalization schemes in our experiments.

Now, let us assume that

$$\begin{aligned} P(s(t) = 1) &= p(t) \\ P(s(t) = 0) &= 1 - p(t). \end{aligned} \quad (3)$$

The  $\varsigma$  function ensures that both  $1 - p(t)$  and  $p(t)$  are always positive, i.e.,

$$P(s(t)) \geq 0 \quad (4)$$

for all possible states  $s(t)$ . The cell is always either *free* or *occupied*, i.e., the state  $s(t)$  is always either 0 or 1, meaning that

$$P(\{s(t) = 0\} \cup \{s(t) = 1\}) = 1. \quad (5)$$

Finally, the sum of all  $P(s(t))$  for all  $s(t) \in \{0, 1\}$  is

$$P(\{s(t) = 1\}) + P(\{s(t) = 0\}) = 1 - p(t) + p(t) = 1. \quad (6)$$

Since  $P(s(t))$  satisfies (4)–(6), which are Kolmogorov’s axioms, we can assume that  $P(s(t))$  is a probability. Thus, the function  $p(t)$  recovered from the frequency spectrum of  $s(t)$  by (2) represents the probability that the cell is *occupied* at time  $t$ .

By thresholding the probability  $p(t)$ , we can calculate an estimate  $s'(t)$  of the original state  $s(t)$ . However, the original observation of  $s(t)$  can differ from the probabilistic estimate  $s'(t)$ . In the case that the given application has to preserve all past observations correctly, the differences between  $s'(t)$  and  $s(t)$  are stored in an outlier set  $\mathcal{O}$ .

Thus, our model of the state consists of two finite sets  $\mathcal{P}$  and  $\mathcal{O}$ . The set  $\mathcal{P}$  consists of  $l$  triples  $abs(P_i)$ ,  $arg(P_i)$ , and  $\omega_i$ , which describe the amplitudes, phase shifts, and frequencies of the model spectrum. Each such triple is related to the importance, time offset, and periodicity of one particular periodic process influencing the state  $s(t)$ . We will refer to the number of modeled processes  $l$  (i.e., to the number of triples in  $\mathcal{P}$ ) as the “order” of the spectral model. The set  $\mathcal{O}$  represents a set of  $k$  time intervals, during which the state  $s(t)$  did not match the state  $s'(t)$  calculated from  $p(t)$ . To achieve low memory requirements, the set  $\mathcal{O}$  is  $\Delta$ -encoded, i.e., it is implemented as a sequence of values, indicating the starts and ends of time intervals when the predicted and observed state did not match, i.e.,  $s'(t) \neq s(t)$ . Thus, each such interval is represented by its limits  $[t_{2k}, t_{2k+1})$ .

Fig. 2 provides a graphic representation of the model building process and a commented video is available at [30]. The process starts with the measured state  $s(t)$  (red line, left box), which is transformed into the frequency domain  $S(\omega)$  (right top, red).



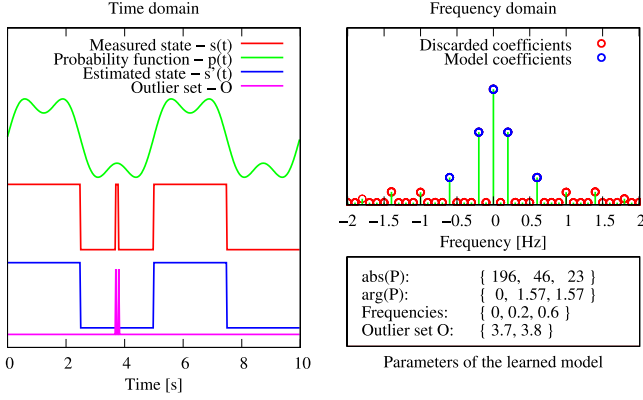


Fig. 2. Example of the measured state and its spectral model. The left part shows the time series of the measured state  $s(t)$ , probability estimate  $p(t)$ , predicted state  $s'(t)$ , and outlier set  $\mathcal{O}$ . The upper right part shows the absolute values of the frequency spectrum of  $s(t)$  and indicates the spectral coefficients, which are included in the model, i.e., in the set  $\mathcal{P}$ . The spectrum is symmetric and the spectral coefficient with frequency 0 corresponds to mean probability of  $s(t) = 1$ . Thus, the model encodes two periodic nodes—its order  $l$  is 2.

The most relevant spectral components  $\mathcal{P}(\omega)$  (right top, green) are then selected and transformed back to the time domain as  $p(t)$  (green line, left box). The probability  $p(t)$  is then thresholded to obtain  $s'(t)$  (left, blue line) and the difference is stored in the outlier set  $\mathcal{O}$  (left box, violet line).

To be able to build, maintain, and use this representation, we define four operations: reconstruction of the measured state  $s(t)$ , addition of a new measurement, model update, and prediction of the future state with a given confidence level.

1) *Reconstruction of the Measured State:* The aforementioned representation allows us to retrieve the past cell state  $s(t)$  as

$$s(t) = (\mathcal{F}^{-1}(\mathcal{P}(\omega)) \geq 0.5) \oplus (t \in \mathcal{O}) \quad (7)$$

where  $\oplus$  is an XOR operation. The idea behind this equation is to reconstruct the probability  $p(t)$  from the spectrum  $\mathcal{P}$ , set  $s'(t)$  to 1 if  $p(t)$  exceeds 0.5 and finally apply the XOR operator to negate  $s'(t)$  if  $t$  belongs to the set of outliers  $\mathcal{O}$ .

2) *Addition of a New Measurement:* Whenever a real state  $s^m(t)$  is measured, we calculate  $s(t)$  by means of (7) and if it differs from  $s^m(t)$ , the current time  $t$  is added to the representation of the set  $\mathcal{O}$ :

$$s^m(t) \neq ((\mathcal{F}^{-1}(\mathcal{P}(\omega)) \geq 0.5) \oplus (t \in \mathcal{O})) \rightarrow \mathcal{O} = \mathcal{O} \cup t. \quad (8)$$

Since (8) takes into account the current contents of the outlier set  $\mathcal{O}$ , the time  $t$  is added to  $\mathcal{O}$  only when  $s'(t)$  starts and stops matching  $s(t)$ , which results in  $\Delta$ -encoding of the set  $\mathcal{O}$ . Nevertheless,  $p(t)$  does not predict  $s(t)$  with perfect accuracy and the set  $\mathcal{O}$  is likely to grow as measurements are added. After some time, the outlier set  $\mathcal{O}$  itself might contain information about dynamics that were previously unobserved and is thus not included in the set  $\mathcal{P}$ . To take into account the new information, our method offers an efficient way to update the entire spectral model.

3) *Model Update:* To update the spectral model, we reconstruct  $s(t)$  including the newly added measurements by (7) and calculate its spectrum  $\mathcal{S}(\omega)$ . Again, we select the  $l$  coefficients with highest absolute values of the spectrum  $\mathcal{S}(\omega)$ , store them in  $\mathcal{P}(\omega)$  and reconstruct the outlier set  $\mathcal{O}$  using (8). In a typical situation, the updated spectrum  $\mathcal{P}$  would reflect  $s(t)$  more accurately, causing reduction of the set  $\mathcal{O}$ . The spectral model order  $l$  can be changed prior to the update step without any loss of information. Thus, we can change the model order and recalculate it whenever required. In our experiments, model update was typically performed on a daily basis as discussed in Section IV-B.

4) *Estimation and Prediction of Future States:* Note, that (7) can calculate  $s(t)$  for any time  $t$  and that the threshold value of 0.5 can be set arbitrarily. In fact, a threshold  $c$  such that  $p(t) \geq c$  represents a confidence level of the grid cell being *occupied* at time  $t$ . Therefore, we can use (7) for future prediction of  $s(t)$  with a given confidence level  $c$ . In the case of prediction, the outlier set  $\mathcal{O}$  is not included in the calculation and the predicted state might not match the real future state, so we denote the prediction as  $s'(t, c)$ . To simplify notation, we also define  $s'(t)$  as  $s'(t, 0.5)$ . Therefore,  $s'(t, c)$  and  $s'(t)$  can be calculated as

$$s'(t, c) = \mathcal{F}^{-1}(\mathcal{P}(\omega)) \geq c. \quad (9)$$

An example of the second-order spectral model which represents a quasi-periodic function is provided in Fig. 2.

5) *Estimation and Prediction for a Single Time Instant:* In many cases (such as in the scenarios described in Sections VII and VIII), one does not need to recover or predict environment states over a long time interval, but for a single time instant. Here, it is impractical to use (9) or (2), because these use the inverse Fast Fourier transform, which generates an entire sequence of probabilities. Instead, one can exploit the sparsity of the spectral model  $\mathcal{P}(\omega)$  and calculate  $p(t)$  simply as

$$p(t) = \alpha_0 + \sum_{j=1}^n \alpha_j \cos(\omega_j t + \varphi_j) \quad (10)$$

where  $\omega_j$ ,  $\varphi_j$ , and  $\alpha_i$  represent the frequencies, time shifts, and amplitudes of the spectral components stored in the set  $\mathcal{P}(\omega)$ . The parameter  $\alpha_0$ , which corresponds to  $\omega_0 = 0$  is the mean of  $s(t)$ .

### C. Nonuniform Sampling Scheme

Typically, the Fourier transform is applied not to continuous functions, but to discrete sequences of data measured on a regular basis. The assumption of equally spaced samples  $s(t)$  allows us to employ the fast Fourier transform (FFT) algorithm, which calculates the frequency spectrum  $\mathcal{S}(\omega)$  in a very efficient manner.

However, the FFT-based model update requires recovery of the entire sequence of the observed states, which becomes computationally expensive over time. Additionally, the FFT relies on the assumption that the observations of the environment states can be performed frequently and on a regular basis, which is hard to satisfy even in laboratory settings. The requirement of regular observations also means that the robot's activity has to

be separated into a learning phase, when it frequently visits individual locations to build its dynamic environment model, and a deployment phase when it uses its model to perform the tasks requested. This division means that while the robot can create a dynamic model which is more suitable for long-term operation, it cannot be updated and thus cannot adapt to variations that were not present during the learning phase. Thus, the predictive capability of the method will become less and less reliable over time, which will negatively affect the efficiency of robot operation in long-term scenarios. To allow the robot to cope with the changing dynamics, we introduce a generalized method that can build and update the spatiotemporal model from sporadic irregular observations in an incremental manner.

This version of the method maintains a sparse frequency spectrum, which is a set  $\mathcal{C}$  of complex numbers  $\gamma_k$  for each modeled state. These correspond to the set  $\Omega$  of modeled periodicities  $\omega_k$  that might be present in the environment. Each time a state  $s(t)$  is observed at time  $t$ , the aforementioned representation is updated as

$$\begin{aligned}\gamma_0 &\leftarrow \frac{1}{n+1} (n\gamma_0 + s(t)) \\ \gamma_k &\leftarrow \frac{1}{n+1} (n\gamma_k + (s(t) - \gamma_0)e^{-j\omega_k t}) \quad \forall \omega_k \in \Omega \\ n &\leftarrow n+1\end{aligned}\quad (11)$$

where  $n$  represents the number of observations. The proposed update step is analogous to incremental averaging—the absolute values of  $|\gamma_k|$  correspond to the average influence of a periodic process (with a frequency of  $\omega_k$ ) on the values of  $s(t)$ . To perform predictions, we select the  $l$  components with the highest absolute value of  $\gamma_k$  from the set  $\mathcal{C}$ , store them in the set  $\mathcal{P}(\omega)$ , calculate  $\alpha_j = |\gamma_j|$ ,  $\varphi_j = \arg(\gamma_j)$  and predict  $p(t)$  using (10).

The choice of set  $\Omega$ , which determines the periods of the potential cyclic processes, depends on the memory size that can be allocated for the model and the longest period that is going to be modeled. In the indoor navigation experiment described in Section VIII,  $\Omega$  consisted of 168 components covering periodicities from one week to 1 h. In the outdoor case Section VII-B,  $\Omega$  consisted of 1000 components covering periods from one year to 8 h. The discussion about the optimal choice of set  $\Omega$  along with other details of the nonuniform sampling scheme is provided in [10]. In the case of uniform sampling, the spectrums generated by (11) and FFT are identical. However, while the set of modeled periodicities of the FFT-based method scales naturally with the duration of the data collection, the set of periods  $\Omega$  captured by the nonuniform scheme is fixed.

#### D. Modeling Persistence

The aforementioned representation is primarily aimed at modeling the environment changes from a long-term perspective. Thus, the predictions of future states are based on the observed periods of the changes in the past. While this is useful for long-term forecasts, prediction of near future states should take into account not only the states' periodicity, but also their persistence. For example, if a given visual feature was observed 10 s ago, it is quite likely that it will be still observable even

though it is not usual to observe it at this time of day or week. To enable the deployment of the proposed method on continuously operating mobile robots, the ability to perform short-term predictions is also important. Thus, we extended the FreMEN representation with a persistence model, which acts as short-term memory that represents the expectation that the given state did not change since the last observation if the observation was performed recently. This is achieved by extending the update scheme of (11) by

$$\begin{aligned}\tau^{-1} &\leftarrow \frac{1}{n+1} \left( n\tau^{-1} + \frac{|s(t) - s(t_l)|}{t - t_l} \right) \\ s(t_l) &\leftarrow s(t) \\ t_l &\leftarrow t\end{aligned}\quad (12)$$

where  $s(t_l)$  represents the last observation at time  $t_l$  and  $\tau$  represents the modeled state persistence, i.e., the mean time between the state's changes. To predict the value of state  $s(t)$  for a future time  $t$ , we calculate

$$p'(t) = s(t_l)e^{\frac{t_l-t}{\tau}} + p(t) \left( 1 - e^{\frac{t_l-t}{\tau}} \right) \quad (13)$$

where  $p(t)$  is calculated by means of (10). Note that for the predictions which closely follow the last observation, i.e.,  $|t - t_l| \ll \tau$ , the expression  $e^{\frac{t_l-t}{\tau}}$  is close to 1, which means that the expected occupancy would be the same as the one recently observed. Using (13) to predict the more distant future, i.e.,  $|t - t_l| \gg \tau$ , causes the expression  $e^{\frac{t_l-t}{\tau}}$  to be close to 0, which suppresses the effect of the latest observation on  $p'(t)$  and emphasizes  $p(t)$ , which represents the behavior of the predicted state from a long-term perspective. The experiments presented in Section VIII show that the addition of the persistence model to the FreMEN representation allows us to deal with nonperiodic changes as well.

#### IV. PERFORMANCE EVALUATION

In the rest of this paper, we examine the tractability of using our approach, the Frequency Map Enhancement (FreMEN), as a core component of spatiotemporal models for mobile robotics. In particular, we investigate the following questions:

- 1) How many parameters of the spectrum typically have to be stored to represent and predict the environment state?
- 2) How efficiently can it represent long-term observations?
- 3) What is the accuracy of its predictions?
- 4) How can the approach benefit long-term autonomy of mobile robots?

To answer these questions, we analyzed several types of environment models gathered by a mobile robot which was continuously operating for several months in a human-populated indoor environment. To quantitatively evaluate the performance of the FreMEN, we use four different criteria relevant to mobile robot mapping. The prediction and estimation errors  $\epsilon_p$  and  $\epsilon_e$  relate to the faithfulness of the FreMEN, i.e., its ability to correctly estimate and predict the environment states for a given time period. The compression ratio relates to the memory efficiency of the FreMEN representation, i.e., the memory needed to represent

the long-term observations of the environment. The update time relates to the computational complexity of the FreMEN.

#### A. Prediction and Estimation Error

Knowing the coefficients  $P_i(\omega_i)$  of the spectrum  $\mathcal{P}$  allows us to calculate an estimate  $s'(t)$  of the original state  $s(t)$  by (9). A natural concern is the accuracy of reconstruction of  $s'(t)$ , because it will affect the prediction capabilities of the spectral model and the size of the outlier set  $\mathcal{O}$ . One can expect that increasing the spectral model order (i.e., including more coefficients in  $\mathcal{P}$ ) would enable more precise reconstruction of  $s'(t)$  from the spectral model  $\mathcal{P}$  alone. However, as the number of parameters grows, the model becomes more adjusted to the specific time series of the observations  $s(t)$ , which decreases its ability to predict the environment state in the future.

To evaluate the quality of the spectral model, we define the estimation error  $\epsilon(t_a, t_b)$  as the ratio of the correctly estimated signal  $s'(t)$  on a given time interval  $t \in [t_a, t_b]$  to the length of the entire interval

$$\epsilon(t_a, t_b) = \frac{1}{t_b - t_a} \int_{t_a}^{t_b} |s'(t) - s(t)| dt. \quad (14)$$

The estimation error can be also calculated from the intersection of the intervals in the outlier set  $\mathcal{O}$  and  $(t_a, t_b)$  as

$$\epsilon(t_a, t_b) = \frac{|(t_b, t_a) \cap \mathcal{O}|}{|(t_b, t_a)|}. \quad (15)$$

Since the outlier set  $\mathcal{O}$  is  $\Delta$ -encoded, the calculation of (15) can be performed very efficiently.

Typically, the error would be calculated for the entire series of observations, i.e., from time 0 to the time of the latest observation  $\tau$ . We call this error the estimation error of the spectral model and denote it as  $\epsilon_e = \epsilon(0, \tau)$ .

Suppose that the sequence  $s(t)$  includes observations made from 0 until  $\tau$  and that the spectral model  $\mathcal{P}(\omega)$  had been calculated using only observations made between 0 and  $\tau'$ , where  $\tau' < \tau$ . Then, calculation of  $s'(t)$  for  $t \in (\tau', \tau]$  by (9) is actually a prediction. Thus, the estimation error  $\epsilon(\tau', \tau)$  relates to the ability of the spectral model to predict future states from past observations. We denote the error  $\epsilon(\tau', \tau)$  as the prediction error  $\epsilon_p$ . Note that the aforementioned situation happens every time the model is updated: the value of  $\tau'$  corresponds to the time of the last update, while the outlier set already contains observations that have been obtained after  $\tau'$ . Since calculation of  $\epsilon_e$  and  $\epsilon_p$  by (15) is computationally efficient, the proposed algorithm can use it to decide whether a model update is needed as well as the optimal order of the spectral model. This can be employed to adapt the model order based on the observed dynamics rather than using a fixed model order.

Although the calculation of both errors is similar, they represent different properties of the FreMEN model. The estimation error  $\epsilon_e$  relates to the ability of the spectral model to recover past observations and  $\epsilon_p$  represents the ability to predict future states. While  $\epsilon_e$  decreases with the model order, the dependence of  $\epsilon_p$  on the model order is more complex.

Note also that (14) relates only to the reconstruction of the states  $s(t)$  from the spectral model  $\mathcal{P}$  before the outlier set is taken into account. The application of the outlier set  $\mathcal{O}$  allows us to recover the sequence  $s(t)$  in an exact way.

#### B. Choosing the Model Order

As mentioned before, the dependence of the prediction error  $\epsilon_p$  on the model order  $l$  is not straightforward. Choosing too low a value  $l$  causes overgeneralization, while choosing too high a value of  $l$  causes overfitting of the FreMEN model. To select the proper value of the model order  $l$ , we evaluate the model's predictive capability for different values of  $l$ , choose the order  $l'$  with the lowest prediction error  $\epsilon_p$  and then perform the model update with the value  $l'$ . In a typical scenario of robot deployment in our project [31], updates of the FreMEN models are performed at midnight every day when the robot replenishes its batteries at its charging station. Before updating, the performance of the FreMEN models with different orders  $l$  is evaluated by comparing their predictions to the observations gathered since the last update (i.e., since midnight the previous day). Then, the models are updated using the order which achieved the lowest prediction error. A typical value for an optimal model order  $l'$  is 2 or 3 and the typical time to establish the optimal order and update the spatiotemporal models used in our robot deployments is less than a minute.

#### C. Compression Ratio

The compression ratio indicates the efficiency of the model in representing the spatiotemporal dynamics of the environment. Rather than evaluating the compression ratio from a theoretical point of view, we adopt a more practical approach and base our calculations on the actual size of the file that contains the spectral model. Assuming that a file of size  $z$ [bits] contains a FreMEN model of an environment with  $n$  states and  $m$  observations, and that a traditional model would use one bit per observation, the compression ratio is simply

$$r = \frac{mn}{z}. \quad (16)$$

In some scenarios, maintaining an entire outlier set  $\mathcal{O}$  might be infeasible due to memory constraints, and the past observations  $s(t)$  are represented solely by the set  $\mathcal{P}$ . While this results in lossy compression with quality corresponding to the estimation error  $\epsilon_e$ , the memory size of this reduced representation is independent of the number of measurements and is determined by the number of modeled states  $n$  and the model order  $l$ , which can be selected *a priori*.

#### D. Update Time

The computational complexity of the proposed method is given by the complexity of the fast Fourier transform algorithm, which is  $O(m \log m)$ , where  $m$  is the length of the processed sequence. This indicates that the time  $t$  needed to build, update, or reconstruct a spectral model with  $n$  states and  $m$  observations by (1) and (7) is  $t \sim m n \log(m)$ . Thus, the update time of the

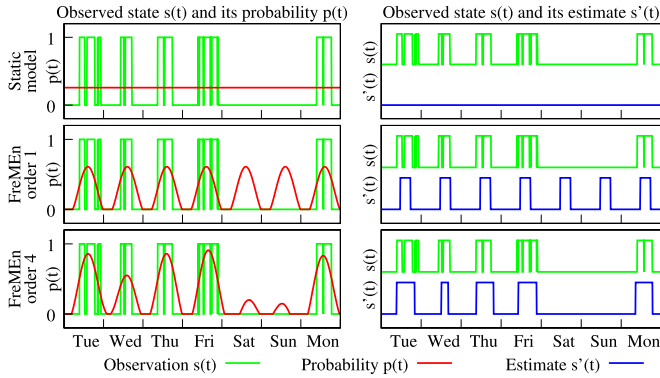


Fig. 3. Week-long state model of a single cell of an occupancy grid. While the traditional static model simply assumes that the probability of the cell occupancy is  $\sim 25\%$ , the frequency-enhanced model captures the cell's dynamics. Note the model improvement as more spectral components are included.

FFT-based model increases with the number of past observations.

However, the computational complexity of the incremental calculation scheme performed by (11) depends only on the number of new observations  $m'$ , the number of independent states  $n$ , and the number of maintained spectral components  $k$ , and therefore, does not depend on the number of past observations. On the other hand, it requires to maintain a larger number of spectral components and is less memory efficient than the FFT-based model.

Since we are concerned with the practical applicability of our approach rather than with theoretical bounds of computational complexity, we measured the real time required to calculate and update the spectral models in our evaluations.

## V. SINGLE-STATE DYNAMIC MODEL

To experimentally verify the feasibility of the proposed approach, we first gathered a week-long dataset containing a single state.

This dataset was gathered by a RGB-D camera monitoring a small university office from a fixed location. Its range measurements were used to establish the occupancy of a single  $20 \times 20 \times 20$  cm cell located in the middle of the room entrance. This cell was occupied when the door was closed and when people passed through the door, otherwise it was free. The office had an “open door” policy, i.e., the door remained open whenever the office was occupied. Therefore, the measured state  $s(t)$  corresponded strongly to the presence of people inside the office. Every time someone went through the door, the monitored cell was briefly occupied and the room was considered empty, which introduced noise on the measured state  $s(t)$ . The measurements were taken continuously for one week (July 23–29, 2013) at a rate of 30 Hz, so the state observation consists of 18 million values. For the purpose of this evaluation, we subsampled the values by 15, which means that the state  $s(t)$  is measured twice per second, so that  $s(t)$  consists of more than a million values. After this week, two additional single-day datasets (July 31 and August 5 2013) were gathered.

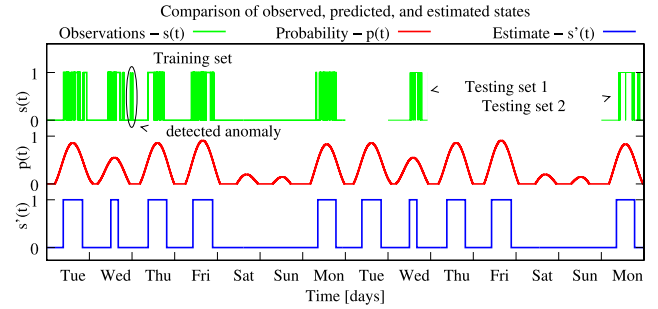


Fig. 4. Comparison of state observations, established probabilistic model, and predicted values.

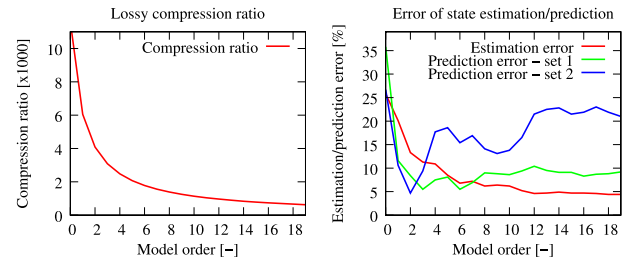


Fig. 5. Influence of the number of spectral components (model order  $l$ ) on the model's compression ratio and errors of estimation and prediction.

To evaluate the proposed method's capability to represent the temporal dynamics of the observed state, we built several spectral models of the training dataset. Fig. 3 shows that the spectral model captures the state dynamics, which results in a more faithful representation of the given state. The first row of Fig. 3 shows that the traditional probabilistic model would simply assume that the door is open with 25% probability. Modeling the state with FreMen of order 1, i.e., considering only one periodic process results in a model that suggests that the door is likely to be open during the afternoon rather than at night—see the second row of Fig. 3. Adding three other spectral components results in a model that captures the weekly periodicities as well—the probability of the door being open (see the last row of Fig. 3) during weekends is lower than during the working days. This result suggests that the method's ability to model the dynamics of the measured state increases with the number of model parameters included.

The dependence of the estimation and prediction errors on the number of components of the spectral model is shown in Fig. 5. To estimate the dependence of the model estimation and prediction errors on the number of model parameters, we built a spectral model of the one-week-long training dataset. The accuracy of estimation  $\epsilon_e$  was calculated as the difference between the original and reconstructed signal by (14). Moreover, we calculated the accuracies of prediction  $\epsilon_{p1}$  and  $\epsilon_{p2}$  for two days of the following week, see Fig. 4.

The results in Fig. 4 indicate that the static model (i.e., FreMen order 0) achieves an estimation error of 25%–35%. The results also indicate that while the estimation error  $\epsilon_e$  decreases monotonically with the model order, the prediction error is not



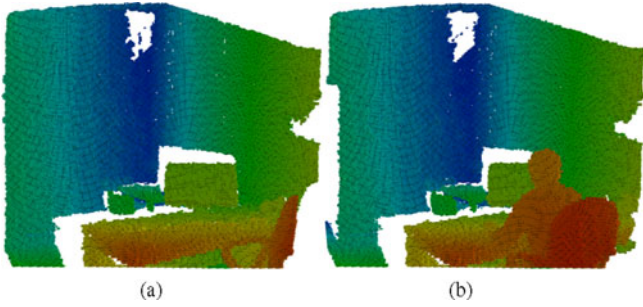


Fig. 6. Fine-grained 3-D occupancy grids of the “Office” dataset. (a) Empty office 3-D grid. (b) Occupied office 3-D grid.

necessarily monotonic. Rather, the local minima of the prediction errors suggest that for the purpose of predictions, one should use spectral models of orders around 2 or 3 to prevent overfitting. The overfitting effect is more prominent with the second testing set, which might be caused by the longer prediction horizon.

The test indicates that the spectral model can represent millions of measurements with only a few complex numbers. Fig. 5 shows that the spectral model without the outlier set  $\mathcal{O}$  achieves compression ratios in the order of 1 : 1000 while losing less than 5% of information. The size of the  $\Delta$ -encoded outlier set was about 360 values representing 180 time intervals where the spectral model did not match the measured sequence, which corresponds to a lossless compression ratio of  $\sim 1:100$ . The time needed to build the spectral representation on an i7 processor was 3.7 ms, which illustrates the efficiency of the chosen Fast Fourier transform implementation [32].

Our method can be also used to detect anomalies, i.e., situations where a local state of the world deviates significantly from the spectral world model of the robot. Since our model can predict the state  $s(t)$  with a given confidence value by (9), we can assume that a measurement  $s^m(t)$  is anomalous with confidence level  $c$  if  $s^m(t) < s(t, c)$  or if  $s^m(t) > s(t, 1 - c)$ . Fig. 4 shows that FreMen-based anomaly detection with confidence level 99% correctly detected a situation when the room was accessed by an unexpected visitor shortly after midnight.

## VI. LARGE SPATIOTEMPORAL REPRESENTATION

To evaluate the ability of the proposed method to represent the long-term dynamics of three-dimensional environments, we collected two million occupancy grids of a University office over the course of 112 days. Similarly to the previous experiment, the dataset was collected by a stationary RGB-D camera that captured and stored a depth image every 5 s. These range measurements were integrated into a FreMen occupancy grid [9], where the occupancy of each cell was modeled by the proposed method. Fine-grained occupancy grids captured by the RGB-D camera are shown in Fig. 6 (for the purpose of visualization, the resolution of the grids shown is higher than those in the dataset). Each day, the spectral model of the entire grid was updated and the resulting representations were saved in separate files. To evaluate the efficiency of the resulting 4-D representations, we measured the compression ratios, estimation

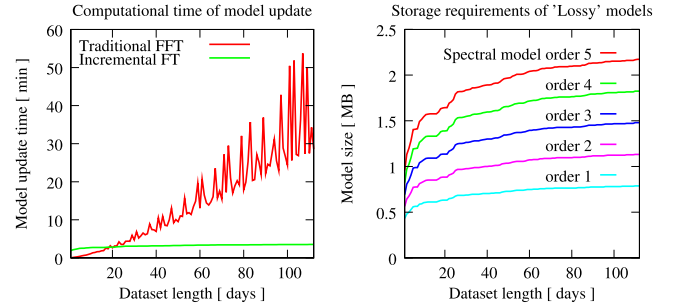


Fig. 7. Computational and memory requirements of the FreMen spatiotemporal occupancy grids.

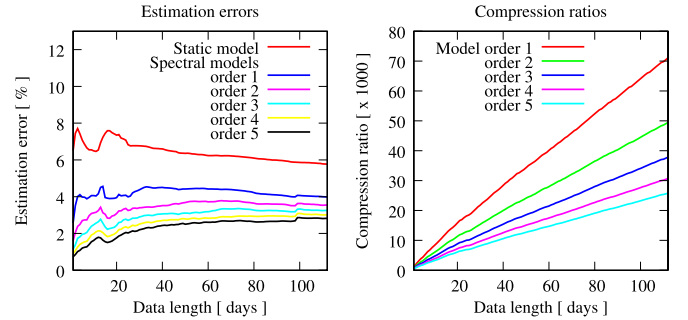


Fig. 8. Estimation errors and compression ratios of the FreMen spatiotemporal occupancy grids.

precisions, and times needed to calculate the update. The compression ratios were calculated simply by comparing the size of the saved files to the theoretical size of a traditional model by (16), where the number of modeled states  $n$ , i.e., the number of cells in the grid was  $\sim 213\,000$  and 17 200 observations per day were considered. This means that storing all the observed states would require  $\sim 500$  MB per day and a naïve representation of the entire dataset would require around 50 GB of storage space. The estimation error of the entire model was calculated as an average of estimation errors of the individual cells that changed at least once—calculating the average estimation error for all cells would result in small numbers, because most of the cells represent space that is always empty. Finally, the update time was obtained by the direct measurement of the time needed to update the spectral models of all the grid cells. These experiments were performed on an i7-4500U processor with 16 GB of RAM.

Five types of spectral models were calculated. The first, “lossless” model maintains not only the spectral representation, but also an outlier set  $\mathcal{O}$  of each cell, and can recover all the measurements accurately. The other, “lossy—order 1–5” models did not use the outlier set and maintained 1 to 5 spectral components of the dynamic cells. The dependencies of the sizes of the “lossy” models on the length of the dataset represented are shown in Fig. 7. One can see that after some initial growth, the storage requirements of the models stabilize at the order of megabytes. The growth of the “lossy” models is caused by the fact that longer data collection means that more cells change



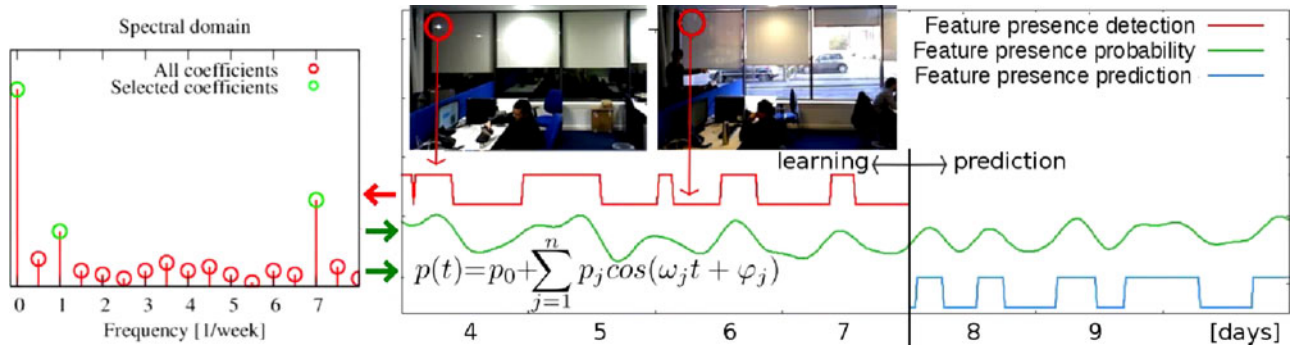


Fig. 9. Frequency-enhanced feature map [4] for visual localization: The observations of image feature visibility (center, red) are transferred to the spectral domain (left). The most prominent components of the model (left, green) constitute an analytic expression (center, bottom) that represents the probability of the feature being visible at a given time (green). This is used to predict the feature visibility at a time when the robot performs self-localization (blue).

their states at least once, which causes the method to extend their temporal models.

Given that the naïve representation of the dataset grows by 500 MB per day, the compression rates of the “lossy” models actually grow in time (see Fig. 8) and are in orders of 10 000. The “lossless” representation grows linearly with time at a rate of 2 MB per day achieving compression rates of 1:250. Fig. 7 also shows that the time needed to update the model, which represents  $4 \times 10^{11}$  cell observations is reasonably short—creation of a 16-week-long spatiotemporal model takes less than 1 h. Using the nonuniform incremental Fourier Transform results in an update time that exhibits a similar trend to the “lossy” model sizes. This is caused by the fact that the number of cells for which the transform has to be calculated increases over time, i.e., the same effect that causes the growth of the “lossy” models. Finally, the estimation errors of the spatiotemporal models with different orders are presented in Fig. 8, which shows that as the model includes more spectral components, its estimation error and compression rates drop. Compared to the “Static” model, which fails to correctly estimate approximately 6% of the states, the “lossy” FreMen estimates fail in 3% to 4% cases. This means that using the FreMen method reduces the amount of incorrectly estimated states by 30%–50%. Using the lossless method results in faithful (0% error) state reconstruction at the expense of a lower (1:250) compression rate.

## VII. FREMEN FOR MOBILE ROBOT LOCALIZATION

The results of the previous experiments demonstrate that through explicit modeling of the environment dynamics, our method can efficiently represent the evolution of indoor environments over time. Moreover, we have shown that the method can predict future environment states. In this experiment, we evaluate the usefulness of these predictions for mobile robot localization in indoor and outdoor environments. The considered scenario is vision-based localization. Given a topological map, where each node is associated with a set of image features visible at that particular location, the robot has to decide on its current location based on its camera image. The difficulty is that the appearance of the locations (i.e., visibility of the image features) varies over time. This problem has been tackled by

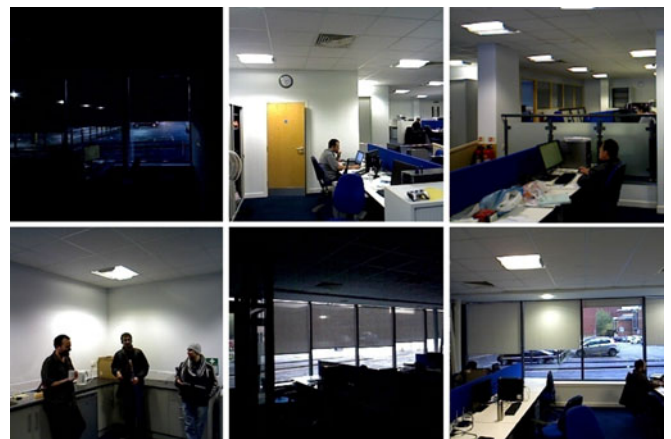


Fig. 10. Example images of the indoor training dataset. Shows the appearance of six monitored locations on November 2013.

attempting to identify the most stable [24] or most useful [33] features, or by remembering several appearance models for the same location [3]. Other approaches [6], [25] attempted to infer the environment appearance for the particular time(s) by modeling the persistence [25] or systematic appearance change of visual features [6]. In this experiment, we predict the visibility of the individual image features at a given time by FreMen, see Fig. 9 and video at <http://fremen.uk>.

### A. Indoor Localization

The environment considered is a large open-plan office of the Lincoln Centre for Autonomous Systems, where an autonomous robot captured RGB-D images of eight designated areas every 10 min. During a week-long data collection session in November 2013, the robot visited each of the eight locations 144 times per day, collecting a training dataset that contains more than 8000 images. To document the appearance change over one year, we provide images from the three testing datasets in Fig. 11. The three testing datasets were collected one week (November 2013), three months (February 2014), and one year (December 2014) after the training dataset collection. Each of these datasets was gathered for 24 h and contains over 1000 images. Representative examples of the images of the training



Fig. 11. Example images of the indoor testing datasets. Shows the evolution of one of the monitored places over the course of one year. (a) November 2013. (b) February 2014. (c) December 2014.

dataset are shown in Fig. 10. The gathered images were processed by the BRIEF algorithm [34], which was evaluated as one of the best performing image feature extractors in outdoor scenarios of long-term localization [35], [36], and our tests confirmed its good performance in indoor scenarios as well. The features of the training dataset belonging to the same locations were matched and, thus, we obtained their visibility over time, which was then processed by our method. To choose the order of the FreMEN models, we adopted the scheme described in Section IV-B, i.e., to select the correct order  $l$ , the FreMEN models were trained initially on the first six days of the training data and their predictive capability was evaluated on the last training day. Next, the models were trained using the entire 7-day-long dataset. Thus, we obtained a dynamic appearance-based model of each topological location that can predict which features are likely to be visible at a particular time, see Figs. 1 and 9.

To test if these predictions actually improve robot localization, the following procedure was performed for each of the  $\sim 3000$  images in the testing datasets. First, the method established the time  $t_c$  when the testing image was captured. Then, the dynamic map created during training was used to calculate the probability of each feature's visibility at time  $t_c$ . Next, the  $n$  most likely visible features at each location were selected, which resulted in eight sets denoted as  $\mathcal{F}_i$ , each containing  $n$  image features. Finally, the features of the testing image were extracted and matched to the sets  $\mathcal{F}_i$ . If the set with the highest number of matches corresponded to the real location of the robot, localization was considered successful, otherwise it was considered a failure.

To compare the proposed algorithm with other localization methods, we implemented a simple version of the experience-based approach developed by the Churchill and Newman [3]. During training, this method attempts to determine the robot location based on the camera input, and if it fails, the current appearance (aka experience) is added to the set of “experiences” that are associated with the given location. Thus, each location is associated with several experiences which are matched to the currently perceived sensory data. While the method introduces a certain computational overhead caused by the fact that there are more experiences than actual locations, this overhead is compensated by the method's robustness to significant appearance changes. This computational overhead was reduced in [37] by inferring the most probable appearances that the robot will experience around a given location. Since we use a slightly different

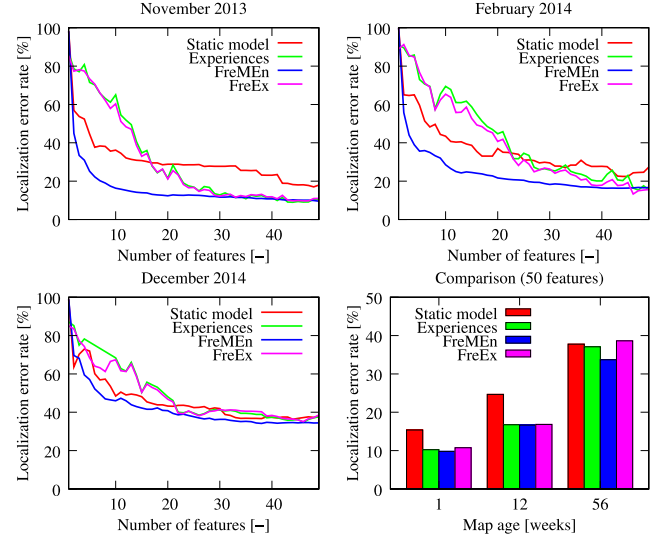


Fig. 12. Localization error rates for different indoor testing datasets, methods, and feature numbers. The first three graphs show the dependence of the error on the number of features used for localization. The fourth graph compares the localization errors of the different methods and datasets assuming that the number of features used is 50.

setup and scenario than the one considered in [3], we had to introduce a slightly different version of the experience-based localization. In our case, an experience consists of the robot position and image coordinates and descriptors of the detected visual features, and we did not use the optimizations introduced in [37].

We also attempted to reduce the aforementioned computational overhead by combining the experience-based approach with FreMEN—the FreMEN was used to calculate the probability of a given experience for a given time, so we could use only the relevant experiences for localization. In the following evaluation, this frequency-enhanced experience method will be coined as “FreEx.” Processing of our training dataset by the experience-based method generated over 170 different experiences tied to eight different locations.

The dependence of the average localization error for each indoor testing dataset on the number of features  $n$  used for localization is shown in Fig. 12. The results indicate that the localization robustness of the FreMEN is only marginally better compared to the experience-based method and they both outperform the “static” approach that relies on the most stable image features. However, while the FreMEN approach improves the robustness by predicting the appearance of the eight locations, the experience-based method requires that the current camera image is matched to all of the 170 experiences, which is computationally more expensive. This is partially mitigated by the FreEx approach, which typically localizes the robot based on 100 experiences, which are selected from the 170 learned ones based on the current time.

While the results show that explicit representation of environment change improves the localization robustness, the improvement diminishes with map age. Since we can observe the same effect for the FreMEN and experience-based methods, the effect is probably not caused by change in the environment dynamics.



Fig. 13. Seasonal variations at location I of the Michigan dataset. (a) Winter 2012. (b) Summer 2012.

Rather, the environment is subject to unexpected and cumulative changes, which affect its appearance in a way that is not possible to predict by the approaches evaluated. This issue severely affected the FreEx approach, which failed to correctly predict the relevant experiences to be used for visual localization. The effect of map decay could possibly be mitigated by active re-observation of locations that were not visited for a long time, e.g., by means of lifelong exploration [10]. This problem also leads to fascinating questions that regard forgetting of obsolete observations and adaptation of the forgetting speed to the rate of environmental change, although these questions are beyond the scope of the work presented here.

### B. Outdoor Localization

To evaluate the performance of the FreMEN for visual localization in outdoor environments, we performed the same comparison on two datasets, which capture the seasonal changes of ten different locations in two semiurban environments.

The images of the first five locations were obtained from the North Campus long-term vision and lidar dataset (NCLT) which was collected at University of Michigan to support research on image features for dynamic lighting conditions [38]. The original NCLT dataset [39] was gathered during 27 data-collection sessions performed over 15 months and includes LIDAR, GPS, and odometry data. For our evaluation, we selected five different locations from the NCLT dataset and created the training dataset from 12 images captured at each location at a different time. To create the testing dataset, we randomly selected three images per location from the set of images not used for training. Unlike the two aforementioned datasets, the Michigan set was not gathered on a regular basis and thus, we used the nonuniform version of FreMEN introduced in Section III-C.

The second set of outdoor images was obtained from the Stromovka dataset [40] that was collected in one of Prague's arboretums to support research on long-term teach-and-repeat navigation [41]. The Stromovka dataset contains images that were captured by a mobile robot every month from September 2009 until the end of 2010, and three additional image sets that were collected during 2011 and 2012. Compared to the Michigan dataset, the Stromovka one spans a longer time period and contains more foliage and fewer buildings. Moreover, seasonal weather variations in Prague are more extreme than in Ann Arbor, see Figs. 13 and 14. Thus, the appearance variations of the Stromovka dataset images are greater than the Michigan ones.

To perform the evaluation, we trained both methods using the datasets gathered during the first 12 months. Then, we



Fig. 14. Seasonal variations at location I of the Stromovka dataset. (a) Winter 2010. (b) Summer 2010.

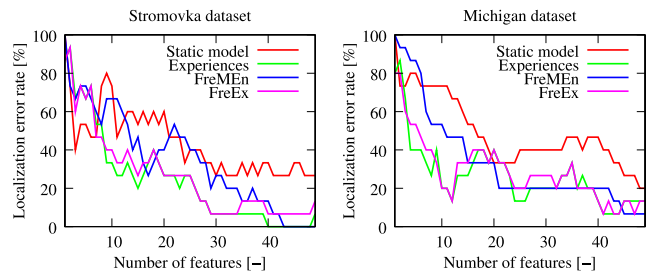


Fig. 15. Localization error rates for the Stromovka and Michigan outdoor datasets. Shows the dependence of the error rates on the methods and feature numbers used.

calculated the localization error rates on the testing sets, which were collected during the following months and years. The dependence of the localization error for both outdoor datasets on the number of features  $n$  is shown in Fig. 15. Similarly to the indoor case, the localization error rates of the FreMEN and experience-based methods were much lower compared to the “static” method, which neglects the appearance change and takes into account only the most stable features. However, the FreMEN localization was computationally more efficient, because it had to match the current camera image to five predicted maps, while the experience-based approach used 15 and 21 different experiences in the Stromovka and Michigan cases, respectively. For the case of outdoor datasets, we did not have enough data to properly estimate the best-performing model order  $l$ , so we set  $l$  to a conservative value of 1.

The aforementioned localization experiments were performed with a relatively low number of image features per image, because the number of locations to distinguish is low. In such cases, extracting a large number of image features will cause the evaluated methods to exhibit a similar performance. To demonstrate the advantages of our approach while utilizing the full power of the feature extractors available would require long-term data collection in much larger environments.

### C. Predictive Capability

To evaluate the predictive capability of the FreMEN approach, we calculated the average probability that a predicted feature will actually be visible in the testing images and compared this with a static approach. First, we calculated the ten most stable features across the training sets and calculated how often these are matched to the features extracted from the testing images of the same location. This corresponds to the Static



TABLE I  
PROBABILITY OF FEATURE REOBSERVATION [%]

Method	Indoor			Outdoor	
	Nov'13	Feb'13	Dec'14	Strom	Mich
Static	39.5	25.7	24.3	38.1	30.8
FreME	55.2	31.2	26.8	47.5	40.8

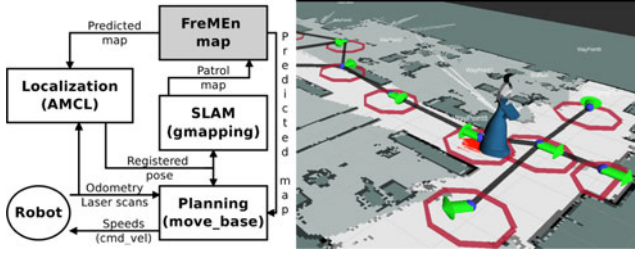


Fig. 16. Navigation system overview. Proposed navigation stack on the left and predicted and observed 2-D grids on the right.

method described in the previous sections. Then, we repeated the procedure with the ten features, which were predicted by FreME to be most likely visible at the given time. The results, summarized in Table I indicate that the image features predicted by the FreME method for a particular time are more likely to be visible compared to the features that were most frequently reobserved in the training sets.

### VIII. FREMEN FOR MOBILE ROBOT NAVIGATION

The experiments presented previously were conducted in an offline manner on prerecorded data. To use FreME online as an integral component of a long-running autonomous system, we developed a FreME occupancy grid which was integrated in the ROS navigation stack [42]. This spatiotemporal grid uses the nonuniform version of FreME with the recency model proposed in Section III-D. During autonomous navigation, our robots build temporally local maps and integrate them into the global spatiotemporal grid. Through reobservation of the same spatial locations, the spatiotemporal grid obtains information about long-term environment dynamics and gains the ability to predict the future environment states. This predictive ability enables the generation of time-specific 2-D maps which can be used by the robot's localization and planning modules. The integration of this predictive spatiotemporal model in the system and a visualization of the map building process are shown in Fig. 16. In this scenario, we evaluated the impact of the proposed spatiotemporal representation on localization accuracy and efficiency of path planning. To do this, we deployed a mobile robot for several days at the Lincoln Centre for Autonomous Systems, having it regularly patrolling the office in a predetermined path several times per hour, using the proposed modification of the ROS navigation stack. The patrolled area contained a 1.5-m-wide corridor. On its sides, there are storage cupboards that are used by research staff and closed at the end of their working day. When a cupboard door is left open, the corridor

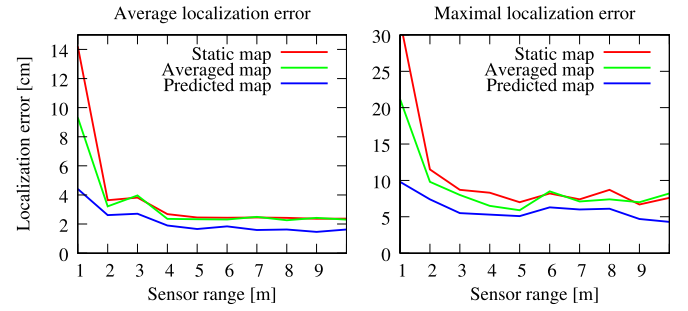


Fig. 17. Localization error for different ranges of the laser scanner and different types of the maps. Predicting a map for a particular time improves localization accuracy, although the improvement is only marginal for long-range sensors.

TABLE II  
NAVIGATION STATISTICS

Environment	Static	Changing	
		Average	Predicted
Map	Static		
Average speed	$[\frac{m}{s}]$	0.21	0.15
Recovery events	[-]	21	12

appears to be wider and its center may be perceived as displaced to one side.

To evaluate the accuracy of robot self-localization, we installed an independent localization infrastructure over the monitored corridor [45]. To estimate the impact of the environment change and sensor range on the localization precision, we processed laser, odometry, and ground truth data from 20 different passes of the robot and trimmed the laser data at different lengths. We then performed standard ROS-based AMCL localization on the “static,” “averaged,” and “predicted” 2-D maps and compared the robot positions to the ground truth obtained by the independent localization infrastructure.

The results shown in Fig. 17 indicate that use of the time-specific, predicted maps improves the localization precision in a significant way if the range of the laser rangefinder is lower than the overall map size. However, a small difference in localization precision can have a significant impact on the efficiency of the robot navigation and quality of the constructed maps.

To evaluate the navigation efficiency, we processed navigation statistics of 180 different patrol runs. The data from each patrol run contains the robot's average speed and the number of events where normal navigation behavior failed and the robot had to perform custom recovery behaviors in order to proceed with its patrol. The gathered navigation statistics were divided into three groups of 60 patrols each. The first group contained patrols where the system was using a static map when no environment changes were happening. The second group contained patrols where the robot was using an “averaged” map, which slowly adapts to the observed change. The third group contained patrols where the robot was using a “predicted,” time-specific map that took into account not only the periodicity, but also the persistence of the observed changes.

Table II indicates that in a static environment, the robot could navigate efficiently even when using a static map, but as soon as the environment began to change, the navigation efficiency was affected in a negative way. However, the negative effect of the changes was slightly lowered through the use of the proposed dynamic map, which represents the environment changes in an explicit way.

## IX. CONCLUSION

We have presented a novel approach for spatiotemporal environment modeling in the context of mobile robotics. The approach is based on an assumption that from mid- to long-term perspectives, the environment is influenced by various processes, some of these being periodical. We hypothesize that certain regularities in the environment dynamics can be represented by the periodicity, amplitude, and time shift of these underlying processes, and propose to identify these parameters through spectral analysis based on the Fourier transform.

Knowledge of these processes allows us to represent the elementary states of the environment models by probabilistic functions of time, which enables efficient representation of arbitrary timescales, anomaly detection, and prediction of future states. To evaluate the performance of the proposed method in real long-term scenarios, we applied it to data gathered by mobile robots over extended time periods of months and years.

The results indicate that the proposed method can represent arbitrary timescales with constant (and low) memory requirements, achieving compression rates between  $10^3$  and  $10^5$  while predicting the future states with error rates of less than 10%. We have also demonstrated that our method's prediction of the environment appearance improved vision-based localization in changing environments. Moreover, we demonstrated that integrating the method in the ROS navigation stack improves the efficiency of robot navigation.

In the future, we would like to extend the approach so that it can take into account sensor noise and represent not only binary, but also higher dimensional states, such as object positions. While the method itself does not exceed the performance of other approaches for persistent localization in changing environments, such as [3], [5], [6], [43], its simplicity enables its application to other scenarios related to long-term autonomy and life-long learning. To provide an overview of the method's applications and to allow its use by other researchers, we have released the method's source code, examples of use, and datasets at <http://fremen.uk>.

## ACKNOWLEDGMENT

The authors would like to thank M. Kulich and P. Urcola for remarks regarding probability theory.

## REFERENCES

- [1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. Cambridge, MA, USA: MIT Press, 2005.
- [2] D. Austin, L. Fletcher, and A. Zelinsky, "Mobile robotics in the long term-exploring the fourth dimension," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 2, 2001, pp. 613–618.
- [3] W. S. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *Int. J. Robot. Res.*, vol. 32, pp. 1645–1661, 2013.
- [4] T. Krajník, J. P. Fentanes, O. M. Mozos, T. Duckett, J. Ekekrantz, and M. Hanheide, "Long-term topological localization for service robots in dynamic environments using spectral maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 4537–4542.
- [5] G. D. Tipaldi, D. Meyer-Delius, and W. Burgard, "Lifelong localization in changing environments," *Int. J. Robot. Res.*, vol. 32, pp. 1662–1678, 2013.
- [6] P. Neubert, N. Sünderhauf, and P. Protzel, "Superpixel-based appearance change prediction for long-term navigation across seasons," *Robot. Auton. Syst.*, vol. 69, pp. 15–27, 2014.
- [7] T. Krajník, J. P. Fentanes, G. Cielniak, C. Dondrup, and T. Duckett, "Spectral analysis for long-term robotic mapping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 3706–3711.
- [8] T. Krajník, M. Kulich, L. Mudrová, R. Ambrus, and T. Duckett, "Where's waldo at time t? using spatio-temporal models for mobile robot search," in *Proc. Int. Conf. Robot. Autom.*, 2015, pp. 2140–2146.
- [9] T. Krajník, J. Santos, B. Seemann, and T. Duckett, "Fractomap: An efficient spatio-temporal environment representation," in *Advances in Autonomous Robotics Systems*. New York, NY, USA: Springer, 2014, pp. 281–282.
- [10] J. M. Santos, T. Krajník, J. Pulido Fentanes, and T. Duckett, "Lifelong information-driven exploration to complete and refine 4D spatio-temporal maps," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 684–691, Jul. 2016.
- [11] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millennium*, San Mateo, CA, USA: Morgan Kaufmann, pp. 1–35, 2002.
- [12] D. Hähnel, D. Schulz, and W. Burgard, "Mobile robot mapping in populated environments," *Adv. Robot.*, vol. 17, pp. 579–597, 2003.
- [13] D. Wolf and G. Sukhatme, "Mobile robot simultaneous localization and mapping in dynamic environments," *Auton. Robots*, vol. 19, pp. 53–65, 2005.
- [14] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *Int. J. Robot. Res.*, vol. 26, no. 9, pp. 889–916, 2007.
- [15] D. Migliore, R. Rigamonti, D. Marzorati, M. Matteucci, and D. G. Sorrenti, "Use a single camera for simultaneous localization and mapping with mobile object tracking in dynamic environments," in *Proc. ICRA Workshop Safe Navigation Dynamic Environ.*, 2009.
- [16] R. Ambrus, N. Bore, J. Folkesson, and P. Jensfelt, "Meta-rooms: Building and maintaining long term spatial models in a dynamic world," in *Proc. Int. Conf. Intell. Robots Syst.*, 2014, pp. 1854–1861.
- [17] P. Biber and T. Duckett, "Dynamic maps for long-term operation of mobile service robots," in *Proc. Robot., Sci. Syst.*, 2005, pp. 17–24.
- [18] M. Milford and G. Wyeth, "Persistent navigation and mapping using a biologically inspired SLAM system," *Int. J. Robot. Res.*, vol. 29, no. 9, pp. 1131–1153, 2010.
- [19] K. Konolige and J. Bowman, "Towards lifelong visual maps," in *Proc. Int. Conf. Intell. Robots Syst.*, 2009, pp. 1156–1163.
- [20] S. Hochdorfer and C. Schlegel, "Towards a robust visual SLAM approach," in *Proc. Int. Conf. Adv. Robot.*, 2009, pp. 1–6.
- [21] C. Stachniss and W. Burgard, "Mobile robot mapping and localization in non-static environments," in *Proc. 20th Nat. Conf. Artif. Intell.*, 2005, pp. 1324–1329.
- [22] N. Mitsou and C. Tzafestas, "Temporal occupancy grid for mobile robot dynamic environment mapping," in *Proc. Mediterranean Conf. Control Autom.*, 2007, pp. 1–8.
- [23] D. Arbuckle, A. Howard, and M. Mataric, "Temporal occupancy grids: A method for classifying the spatio-temporal properties of the environment," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, 2002, pp. 409–414.
- [24] F. Dayoub, G. Cielniak, and T. Duckett, "Long-term experiments with an adaptive spherical view representation for navigation in changing environments," *J. Robot. Auton. Syst.*, vol. 59, pp. 285–295, 2011.
- [25] D. M. Rosen, J. Mason, and J. J. Leonard, "Towards lifelong feature-based mapping in semi-static environments," in *Proc. Int. Conf. Robot. Autom.*, 2016, pp. 1063–1070.
- [26] M. Yguel, O. Aycard, and C. Laugier, "Wavelet occupancy grids: A method for compact map building," in *Field and Service Robotics*. New York, NY, USA: Springer, 2006, pp. 219–230.
- [27] T. Kucner, J. Saarinen, M. Magnusson, and A. J. Lilienthal, "Conditional transition maps: Learning motion patterns in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 3–8, 2013, pp. 1–6.
- [28] T. Krajník, J. Pulido Fentanes, J. Machado Santos, and T. Duckett, "Frequency map enhancement: Introducing dynamics into static environment models," in *Proc. ICRA 2016 Workshop AI Long-Term Autonomy*, 2016.

- [29] R. N. Bracewell and R. Bracewell, *The Fourier Transform and Its Applications*. New York, NY, USA: McGraw-Hill, 1986, vol. 31999.
- [30] T. Krajník, J. P. Fentanes, O. M. Mozos, T. Duckett, J. Ekekrantz, and M. Hanheide, "Long-term mobile robot localization in dynamic environments using spectral maps," in *Proc. AAAI Conf. Artif. Intell. (Video Session)*, 2015. [Online]. Available: [http://www.aaaivideos.org/2015/03\\_spectral\\_map\\_localization/](http://www.aaaivideos.org/2015/03_spectral_map_localization/)
- [31] N. Hawes *et al.*, "The STRANDS project: Long-term autonomy in everyday environments," *IEEE Robot. Autom. Mag.*, 2017, to be published.
- [32] "The FFTW C library," 2016. [Online]. Available: <http://www.fftw.org/>
- [33] P. Mühlfellner, M. Bürki, M. Bosse, W. Derendarz, R. Philippsen, and P. Furgale, "Summary maps for lifelong visual localization," *J. Field Robot.*, vol. 33, pp. 561–590, 2015. [Online]. Available: <http://dx.doi.org/10.1002/rob.21595>
- [34] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.
- [35] T. Krajník, P. Cristóforis, M. Nitsche, K. Kusumam, and T. Duckett, "Image features and seasons revisited," in *Proc. IEEE Eur. Conf. Mobile Robots*, 2015, pp. 1–7.
- [36] T. Krajník, P. Cristóforis, K. Kusumam, P. Neubert, and T. Duckett, "Image features for visual teach-and-repeat navigation in changing environments," *Robot. Auton. Syst.*, vol. 88, pp. 127–141, 2016.
- [37] C. Linegar, W. Churchill, and P. Newman, "Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 90–97.
- [38] N. Carlevaris-Bianco and R. M. Eustice, "Learning visual feature descriptors for dynamic lighting conditions," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2769–2776.
- [39] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *Int. J. Robot. Res.*, vol. 35, pp. 1023–1035, 2015.
- [40] "Stromovka dataset," 2017. [Online]. Available: <http://mobilerobotics.eu/datasets/stromovka>
- [41] T. Krajník *et al.*, "Simple, yet stable bearing-only navigation," *J. Field Robot.*, vol. 27, pp. 511–533, Sep./Oct. 2010.
- [42] T. Krajník, J. P. Fentanes, M. Hanheide, and T. Duckett, "Persistent localization and life-long mapping in changing environments using the frequency map enhancement," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4558–4563.
- [43] D. Mishkin, M. Perdoch, and J. Matas, "Place recognition with WxBS retrieval," in *Proc. CVPR Workshop Visual Place Recog. Changing Environ.*, 2015.
- [44] C. Cadena *et al.*, "Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [45] T. Krajník, M. Nitsche, J. Faigl, T. Duckett, M. Mejail, and L. Přeučil, "External localization system for mobile robotics," in *Proc. Int. Conf. Advanced Robot.*, 2013. doi: 10.1109/ICAR.2013.6766520.
- [46] J. P. Fentanes, B. Lacerda, T. Krajník, N. Hawes and M. Hanheide, "Now or later? Predicting and maximising success of navigation actions from long-term experience," *Int. Conf. Robot. Autom.*, May 2015, pp. 1112–1117. doi: 10.1109/ICRA.2015.7139315.
- [47] M. J. Santos, T. Krajník, and T. Duckett, "Spatio-temporal exploration strategies for long-term autonomy of mobile robots," *Robot. Auton. Syst.*, Elsevier, 2016.



**Tomáš Krajník** received the Ph.D. degree in artificial intelligence and biocybernetics from Czech Technical University, Prague, Czech Republic, in 2012.

He is currently a Research Fellow at the Lincoln Center of Autonomous Systems, Lincoln, School of Computer Science, University of Lincoln, U.K. His research interests include life-long autonomous navigation, spatio-temporal modelling, and aerial robots.



**Jaime Pulido Fentanes** received the Ph.D. degree in industrial engineering and automation from the University of Valladolid, Valladolid, Spain.

He is currently a Research Fellow at the Lincoln Centre for Autonomous Systems, School of Computer Science, University of Lincoln, Lincoln, U.K. where he is involved with multiple projects including the EU FP7 project STRANDS, which aims to enable a robot to achieve robust and intelligent behavior over long periods of time. His research interests include mobile robotics, mapping and navigation, and robot exploration.



**João Machado Santos** received the M.Sc. degree in electrical engineering and computers, with specialization in automation, from the Faculty of Sciences and Technology, University of Coimbra, Coimbra, Portugal, in 2013. He is currently working toward the Doctoral degree at the Lincoln Centre for Autonomous Systems, School of Computer Science, University of Lincoln, Lincoln, U.K.

He is currently involved with the project STRANDS within the FP7 framework, which aims to enable a robot to achieve robust and intelligent behavior over long periods of time.

His research interests include mobile robotics, mapping, localization, and exploration.



**Tom Duckett** received the Ph.D. degree in the AI Group at the University of Manchester, U.K.

He is currently a Professor of computer science at the University of Lincoln, Lincoln, U.K., where he also leads the Lincoln Centre for Autonomous Systems. He worked previously at the Centre for Applied Autonomous Sensor Systems, Örebro University, Örebro, Sweden, where he led the Learning Systems Laboratory. Prior to becoming an academic, he worked for several years as a programmer, developing and supporting software solutions for the fresh food industry. His research interests include autonomous robots, artificial intelligence, and machine perception, with applications including agri-food and assistive technologies.