Semantic Place Understanding for Human–Robot Coexistence—Toward Intelligent Workplaces

Stefano Rosa, Andrea Patanè, Chris Xiaoxuan Lu, and Niki Trigoni

***Abstract*—Recent introductions of robots to everyday scenarios have revealed unprecedented opportunities for collaboration and social interaction between robots and people. However, to date, such interactions are hampered by a significant challenge: having a semantic understanding of their environment. Even simple requirements, such as "a robot should always be in the kitchen when a person is there," are difficult to implement without prior training. In this paper, we advocate that robot–people coexistence can be leveraged to enhance the semantic understanding of the shared environment and improve situation awareness. We propose a probabilistic framework that combines human activity sensor data generated by smart wearables with low-level localization data generated by robots. Based on this low-level information and leveraging colocation events between a user and a robot, it can reason about the two types of semantic information: first, semantic maps, i.e., the utility of each room and, second, space usage semantics, i.e., tracking humans and robots through rooms of different utilities. The proposed system relies on two-way sharing of information between the robot and the user. In the first phase, user activities indicative of room utility are inferred from wearable devices and shared with the robot, enabling it to gradually build a semantic map of the environment. In the second phase, via colocation events, the robot teaches the user device to recognize the type of room where they are colocated. Over time, robot and user become increasingly independent and capable of semantic scene understanding.**

***Index Terms*—Activity recognition, human–computer interaction, intelligent robots.**

I. Introduction

HIGH-LEVEL semantic understanding of the environment is still an open problem for complex cyber physical systems involving robots and people. We envision that in the next five years, such systems will become ubiquitous: robots' presence will continue to grow in workplaces, and low-cost robots will increasingly assist humans in domestic environments. The use of wearable sensors in manufacturing has been investigated, with a particular focus on augmented reality and dedicated assistance [2], [16]. Existing robotic and wearable sensor systems, however, still lack maturity in terms of how they perceive the environment.

For example, robots typically perceive space in terms of low-level metric, topological or feature maps. Recent work has motivated the need for a high-level understanding of the environment (e.g., semantic, affordances or high-level geometry) in order to enable emerging robotics applications [6]. To date, vision-based techniques for semantic mapping are well studied, but they are labor intensive as they require careful training and/or fine-tuning. Our vision instead is that semantic information can be automatically acquired by robots over time as a result of coexistence with users.

Similarly, wearable devices held by humans, e.g., smartphones or smartwatches, require tedious training (e.g., Wi-Fi fingerprinting) and/or bespoke sensor infrastructure (e.g., UWB/Bluetooth) to localize themselves within a room, and even then, they lack semantic understanding of the utility of the room. Again, we advocate that this capability should be acquired spontaneously by human-held devices as a result of them interacting with robots.

To this end, we propose a system that enables robots and wearable devices to have a semantic understanding of their environment via colocation and interaction with each other. We believe that this is a key to a variety of applications from issuing simple commands to robots such as "Go to the kitchen," to tasks of collaborative nature like "The robot should go to the kitchen when the user (her smartphone) is there." In an industrial scenario, room-level localization of users could enable real-time dynamic context-aware reasoning [4], in particular, in the framework of Industry 4.0, in which the use of arrays of sensors on the shop-floor could be replaced by a few mobile sensors, carried by autonomous mobile service robots and by users.

The first intuition behind our approach is that user activities provide informative hints about the utility of each room. For example, a bedroom can be easily identified if people often sleep in that room. However, the association between room types and activities is not always unique. For instance, a user may eat in the dining room most of the time, but may occasionally opt to do so in the living room. The problem that arises is *how to reliably infer semantic labels for different rooms of the space given two incomplete and noisy sources, i.e., robots' perception of space and users' activity context*.

Once we address the problem of semantic mapping, it paves the way for inferring the sequence of room types that human devices traverse. A robot, who is now aware of semantic room labels, can *teach* human mobile devices how to recognize them

Manuscript received February 8, 2018; revised June 28, 2018; accepted September 18, 2018. Date of publication October 30, 2018; date of current version March 13, 2019. This work was supported in part by the EPSRC Program under Grant GoW EP/M019918/1 and EU's Horizon 2020 Marie Skłodowska-Curie under Grant No. 722022. This paper was recommended by Associate Editor Z. Wang. *(Corresponding author: Stefano Rosa.)*

The authors are with the Department of Computer Science, University of Oxford, Oxford 0X13QD, U.K. (e-mail: stefano.rosa@cs.ox.ac.uk; andrea.patane@cs.ox.ac.uk; xiaoxuan.lu@cs.ox.ac.uk; niki.trigoni@cs.ox.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/THMS.2018.2875079

based on their own signals. Specifically, we show how a robot can help mobile devices to tune the parameters of the *hidden Markov model* (HMM) that they use for localization.

To summarize, semantic mapping and semantic localization are two faces of the same coin; we address both by leveraging opportunistic *colocation* events between robots and human-held devices. Through the diverse lenses of robots and wearable devices, we show that they can both develop a semantic understanding of their space.

In particular, the contributions of this paper are as follows.

1) A method for inferring semantic labels (room types) for different rooms by exploiting user activities and opportunistic colocation events.
2) A method for exploiting the inferred semantic map and colocations in order to train the parameters of an HMM for user localization.
3) We propose a bidirectional recurrent neural network (RNN) with approximate variational inference for classification of complex daily activities from a smartwatch.
4) We validate the results in two work environments coinhabited by robots and humans wearing smartwatches.

The remainder of this paper is structured as follows. Section II provides an overview of related work. Section III presents the architecture of our system. Section IV describes the semantic representation of the map and the mapping procedure. Section V discusses the training of the HMM for user localization. Section VI evaluates the proposed approaches in different scenarios and Section VII presents our conclusions and directions for future work.

II. Related Work

A. Daily Activity Recognition

Activity recognition, and in particular wearable activity recognition, is an important problem that has drawn significant attention from the research community in the last ten years. Although different sensor modalities have been studied, we focus on the most related work that uses inertial sensor data (acceleration and gyroscope) for activity recognition. We first discuss recent work on classifying activities, and then discuss how activity information has been used within simultaneous localization and mapping (SLAM) frameworks.

Ranjan and Whitehouse [20] used inertial data from a wrist-mounted device to detect activities performed on household objects. Ramos *et al.* [19] proposed to combine smartwatches and smartphones for activity recognition and evaluate different features. A *deep belief network* composed by stacked *restricted Boltzmann machines* is used in [5] for detecting activities based on spectrograms of acceleration data. A hybrid of deep learning and hidden Markov models (DL-HMM) is also presented for sequential activity recognition. An alternative dense approach of labeling each sensor sample in a sequence, as opposed to labeling a whole window of data, is explored using fully convolutional networks in [29]. The above-mentioned papers focus entirely on improving activity recognition; in this paper, we propose a novel approach to activity recognition, based on variational long short-term memory (LSTMs), that gives us estimates of classification uncertainty. This is a distinct advantage as it enables us to integrate the activity classification model into a purely probabilistic model, wherein uncertainty about activity translates to uncertainty about semantic room labels in a principled manner.

We are now in a position to overview how activity classification has been explored within SLAM frameworks. Hardegger *et al.* [11] proposed a three-dimensional (3-D) SLAM algorithm for users wearing wearable sensors, by including detected activities as landmarks in a particle filter SLAM approach. In [10], the approach is extended into a unified Bayesian framework for semantic SLAM with the goal of adding robustness to errors in activity recognition. However, in both approaches, the user carries a multitude of inertial sensors (wrist-mounted, hip-mounted, and foot-mounted inertial measurement units (IMUs)) and does not exploit interactions with robots. Moreover, while the approach is shown to work on some medium-length trajectories, particle filter based SLAM methods are known to suffer from the forgetting problem over longer trajectories (due to the nature of resampling, the best trajectory could be discarded over time). In [14], a method is proposed for tagging maps with objects. The object's position is inferred by detecting user activities and location, but the detected activities are not used in the map estimation and there is no information exchange between the robot and the user. To our knowledge, this is the first paper that infers both user activity and its uncertainty from noisy wearable sensors, and feeds this information to colocated robots, which then learn semantic maps of the environment.

B. Semantic Mapping

Semantic mapping is the problem of associating high-level semantic attributes to low-level geometric features. Both perception and suitable map representations are active areas of research, but to date most work in the robotics community has been devoted to camera sensors [6]. Pronobis and Jensfelt [18] presented a conceptual model for semantic map representation, with different levels of abstraction, from sensor data to concepts, such as rooms, with associated properties, such as shape, appearance, and detected objects. The layered structure of the spatial knowledge is used for reasoning at the semantic level, starting from laser range finders and camera sensors. A number of works have focused on assigning semantic concepts to high-level map features, such as planar surfaces [21]. Pillai and Leonard [17] segmented known objects in the map based on semantic labels. Recently, Xiang and Fox [27] proposed a novel RNN architecture for semantic labeling on RGB-D videos. Semantic information is integrated with dense 3-D SLAM techniques, such as KinectFusion, in order to obtain a 3-D semantic map of the environment. The most closely related work on semantic mapping is the recent work on inferring room labels [22] using visual place categorization. A convolutional neural network is trained on the SUN Scene Understanding dataset, and addresses the closed-set limitation by training a set of one-versus-all classifiers for recognizing new semantic classes.

The above-mentioned techniques rely on training data that associate visual sensor data to higher level semantic labels. Such learning tends to be very sensitive to the environment and incurs a significant manual fine-tuning effort in each environment. For example, the appearance of a kitchen may vary significantly across different work and home environments. In our work, we avoid environment-specific training; we rely on activity inference that transfers well between different environments, and exploit robot–person interaction to gradually learn room types from user activities over time. The only other work that exploited robot–person interaction is presented in [14], but only to perform activity and associated object recognition in a more reliable manner by combining the camera sensor of the robot with the inertial sensors of the user.

C. User Localization

Indoor localization techniques have gained significant maturity offering both infrastructure-based (e.g., UWB [3], acoustic [23], and Bluetooth low energy (BLE) beacons [32]) and infrastructure-less (e.g., Wi-Fi [15], [24], geomagnetic [25], and inertial [28]) solutions. In general, infrastructure-based methods require the deployment and maintenance of bespoke localization hardware, which greatly limits their application. On the other hand, infrastructure-less methods exploit ambient signals in the environment and are less costly. However, these methods typically require offline training in the form of learning signal maps of Wi-Fi or geomagnetic signals. The user's positions can then be localized by matching the online collected signals with the surveyed signal map. Even after significant training effort, location estimation can still be inaccurate in the online phase due to the environmental dynamics and pose variations of users.

Unlike previous work on learning physical signal maps, the adopted semantics are abstract and tightly related to user activities. In our context, the aim is to infer semantic paths, e.g., the user went from the conference room to the kitchen and back to his office. Previous work [13] on combining user activities with Wi-Fi and acoustic data to localize users at room level in domestic environments required a labor-intensive training phase for building the Wi-Fi map. Instead, we move away from location-based training efforts, and rely on lifelong learning from human–robot interactions. The idea is to progressively build confidence on the semantics of different rooms and make wearable devices increasingly aware of their environment.

III. System Architecture

This section provides a high-level overview of our system. We start by describing its actors and their sensing capabilities, and then proceed to overview the two main components of the system.

A. Actors and Sensing Capabilities

The proposed system includes two types of actors: a mobile assistive robot and a user holding a wearable device, e.g., a smartwatch. No other infrastructure is necessary.

1) Mobile Robot: We assume that the mobile robot is equipped with proprioceptive sensors, such as wheel encoders or an inertial sensor and an exteroceptive distance sensor such as a laser range finder, sonars, or infrared sensors. Those sensors are required in order for the robot to create a map of a previously unseen environment and localize therein, as well as perform basic navigation in it. We do not rely on camera sensors, since cameras are often forbidden in workplaces for privacy reasons, and would also pose privacy issues in home environments.

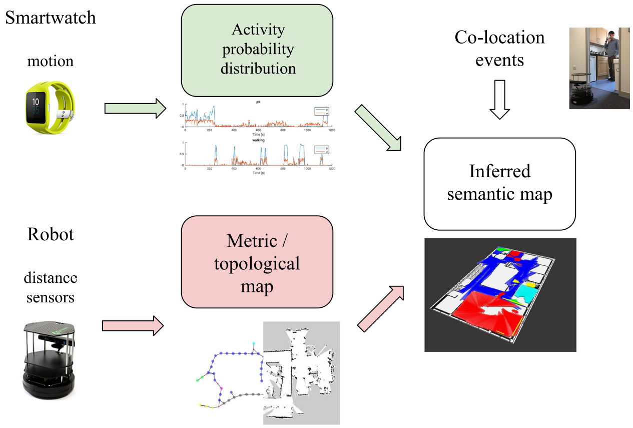


Fig. 1. Architecture of the semantic mapping subsystem.

2) User: We make the assumption that the user is carrying a smart device, e.g., a smartwatch on his right arm if right-handed or on left arm if left-handed. Smartwatches are a sensible choice for detecting human activities from inertial data, and are not intrusive compared to other sensors. Smartphones can be used to infer low-level activities, such as walking, resting, climbing stairs, etc., but are not useful for detecting a richer set of daily activities, such as washing hands. It should be noted, however, that smartwatches still present some limitations when having to distinguish between activities that present similar motions, e.g., washing hands and washing dishes. In this paper, we model such an uncertainty and take it into account in building semantic maps and localizing users within them.

B. System Components

Our system consists of two main subsystems, one responsible for building the semantic map of the environment, and one for localizing users with wearables within the semantic map. These two subsystems are discussed ahead in more detail.

1) Semantic Mapping: Fig. 1 provides an overview of the first subsystem, designed to infer the semantic labels of map cells. In this phase, we assume that the robot has already built a grid map representation of the environment using an existing SLAM algorithm, such as *gmapping*. The robot is also able to localize in the map using its sensors and a suitable localization algorithm, such as *amcl*. Moreover, the robot is able to navigate the environment by planning trajectories and avoiding obstacles. Such aspects of robot functionality are already mature and accessible to researchers and practitioners in mobile robotics.

The user is wearing a smartwatch, which is acquiring inertial measurements (accelerations and angular velocities). Based on these measurements, we infer probability distributions of activities using bidirectional long short-term memory (BLSTM) neural networks. Whenever the robot happens to be colocated

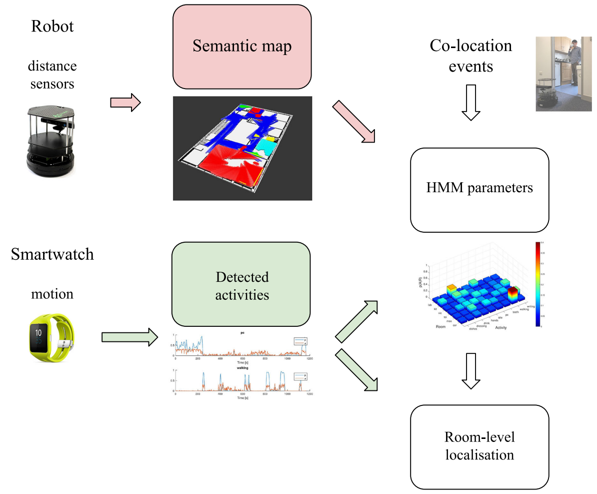


Fig. 2. Architecture of the user localization subsystem.

with the user in the same room, the robot detects the human figure with its sensors and registers the colocation event.

The semantic map subsystem takes as input motion data from the user, metric/topological maps inferred from the robot, and colocation events detected by the robot, and combines them to infer semantic labels for each grid cell of the robot map. Details are further discussed in Section IV.

2) Semantic Localization: Having obtained a semantic map of the environment through the previous process, our system includes a second component for localizing users within the semantic map, as shown in Fig. 2. Our aim is to infer trajectories that are not sequences of time–xy-floor coordinates, but sequences of time-room label tuples.

In order to obtain such semantic paths reliably, we combine the semantic map learnt from the previous phase with user activity distributions and colocation events between robot and user. Fusing the above information in a probabilistic framework, we are able to train the parameters of an HMM, which we then apply to infer the user's semantic paths. It is worth noting that colocation events are only used for training the HMM; they are not used at the inference stage. This means that the system can learn to track the user through rooms independently of whether the robot happens to be there. Further details on this part of the system are provided in Section V.

IV. Semantic Mapping

In this section, we describe the first phase of our approach, in which the robot is able to create a semantic map on top of the metric map of the environment by accumulating information on user activities over time during robot–user colocation events. We first introduce BLSTM neural networks and describe the proposed activity classification network architecture. Then, we describe the semantic mapping creation process.

A. Activity Recognition

BLSTM RNNs have recently shown promising results when applied to the problem of human activity recognition (HAR) [9], [30]. Inspired by these works, we started off by training a BLSTM network that uses raw acceleration and gyroscope data as input. However, the disadvantage of this method is that it does not offer a Bayesian probabilistic interpretation of the quality of classification results. In order to estimate the uncertainty surrounding our classification results, we applied for the first time the approach of *variational* LSTMs [7] to the problem of activity recognition. In what follows, we first introduce the reader to pure and BLSTMs, and then explain the benefits of the variational approach.

Traditional RNNs are a type of neural network where the layers operate not only on the input data but also on the delayed versions of the hidden layers and/or output. Therefore, the network has an internal state which it can use as a "memory" to keep track of past inputs and its corresponding decisions. Traditional RNNs, however, suffer from the problem of forgetting, as they are unable to learn long-term trends in the input data. This is known as the *vanishing gradient problem*. In [12], LSTM networks were introduced as a modified version of RNNs in order to address the vanishing point problem. Through the inclusion of gating cells that allow the network to selectively store and forget past memories, the input gate $\mathbf{g}^i$ controls how the input enters into the contents of the memory cell for the current time step. The forget gate $\mathbf{g}^f$ determines when the memory cell should be emptied by producing a control signal in the range 0–1 which clears the memory cell as needed. The output gate $\mathbf{g}^o$ determines whether the contents of the memory cell should be used at the current time step. $\mathbf{g}^c$ is the cell state vector

$$\begin{aligned}
\mathbf{g}^i &= \sigma(\mathbf{W}^i * \mathbf{h}_{t-1} + \mathbf{I}^i * \mathbf{x}_t)\\
\mathbf{g}^f &= \sigma(\mathbf{W}^f * \mathbf{h}_{t-1} + \mathbf{I}^f * \mathbf{x}_t)\\
\mathbf{g}^o &= \sigma(\mathbf{W}^o * \mathbf{h}_{t-1} + \mathbf{I}^o * \mathbf{x}_t)\\
\mathbf{g}^c &= \tanh(\mathbf{W}^c * \mathbf{h}_{t-1} + \mathbf{I}^c * \mathbf{x}_t)\\
\mathbf{m}_t &= \mathbf{g}^f \odot \mathbf{m}_{t-1} + \mathbf{g}^u \odot \mathbf{g}^c\\
\mathbf{h}_t &= \tanh(\mathbf{g}^o \odot \mathbf{m}_{t-1})
\end{aligned} \tag{1}$$

where $\mathbf{W}^u, \mathbf{W}^f, \mathbf{W}^o$, and $\mathbf{W}^c$ are weight matrices and $\mathbf{I}^u, \mathbf{I}^f, \mathbf{I}^o$, and $\mathbf{I}^c$ are projection matrices. σ is the logistic sigmoid function. $\mathbf{m}_t$ is the internal state of the cell and $\mathbf{h}_t$ is the hidden vector.

LSTMs have been showed to be able to learn temporal behavior and have been extensively used in many applications. Hence, they seem a natural choice for detection of complex activities from sequences of data that present a temporal component.

BLSTMs [8] are a variant of LSTMs composed by one forward LSTM and one backward LSTM running in reverse on the data and with their features concatenated at the output layer. This enables information from both past and future to come together. BLSTMs have been found to perform better when dealing with small datasets.

A limit of RNNs is their tendency to overfit. Dropout can help to a certain extent, but it has been shown to fail when applied to recurrent layers. Gal and Ghahramani [7] suggested the use of dropout in LSTMSs for an approximate Bayesian inference. In the proposed variant, dropout is also used in the recurrent connections, and the same dropout masks are repeated at each time step for inputs, outputs, and recurrent layers.

Variational LSTMs have been shown to outperform the classic variant, while at the same time offering a useful Bayesian representation of the output, giving an estimate of the output uncertainty. However, to our knowledge, they have not yet been explored in the context of HAR.

In the variational variant, (1) becomes

$$\begin{aligned}
\mathbf{g}^i &= \sigma(\mathbf{W}^i * (\mathbf{h}_{t-1} \odot \mathbf{z}_h) + \mathbf{I}^i * (\mathbf{x}_t \odot \mathbf{z}_x)) \\
\mathbf{g}^f &= \sigma(\mathbf{W}^f * (\mathbf{h}_{t-1} \odot \mathbf{z}_h) + \mathbf{I}^f * (\mathbf{x}_t \odot \mathbf{z}_x)) \\
\mathbf{g}^o &= \sigma(\mathbf{W}^o * (\mathbf{h}_{t-1} \odot \mathbf{z}_h) + \mathbf{I}^o * (\mathbf{x}_t \odot \mathbf{z}_x)) \\
\mathbf{g}^c &= \tanh(\mathbf{W}^c * (\mathbf{h}_{t-1} \odot \mathbf{z}_h) + \mathbf{I}^c * (\mathbf{x}_t \odot \mathbf{z}_x)) \qquad (2)
\end{aligned}$$

where $\mathbf{z}_x$ and $\mathbf{z}_h$ are random binary masks that remain constant at each step.

The other difference from standard LSTMs is that at prediction time the dropout remains active. Each prediction is repeated n times, in our case 50 times, and it is possible to compute the mean class prediction and the associated variance over the set of n samples, obtaining a prediction vector HAR, where each element i denotes the probability $p[i]$ of activity i and the uncertainty $\sigma[i]$ around it

$$\text{HAR}[i] = (\text{HAR\_}p[i], \text{HAR\_}\sigma[i]).$$

The ability to have an estimation of the uncertainty associated with the detection is crucial when including this information in a probabilistic framework.

B. Semantic Map Inference

1) Topological Mapping: As in [18], at the lower level, a SLAM algorithm creates a grid map of the environment using the robot sensors. Using a template-based door detector [18] on laser distance data, the robot is able to group together multiple cells into individual rooms. We use the concept of *room* in a broad sense to denote both regular rooms and corridors. The aim of semantic mapping is to assign semantic categorical labels (e.g., kitchen, bathroom, corridor, etc.) to each cell in the grid.

2) Detecting Colocation Events: Once the robot builds a grid map of the environment, it starts roaming through it and records any colocation events with users. In this section, we explain how to robustly detect colocation events and identify the user with whom the robot is colocated. For detecting humans, we use fusion of distance data from the laser range finder on board of the robot; we use an open-source code of an existing detector that learns to recognize human legs [26].

However, in our application, we must ensure that the detected person is effectively the user wearing the smartwatch. To this end, we placed one BLE beacon onboard of the robot and measured the received signal strength (RSS) at the smartwatch. On detecting a beacon, the smartwatch sends to the robot the user identifier along with that user's HAR (activity distribution) vector. Note that other methods based on RSS beyond blue tooth low energy (BTLE) could be used for identifying users, for example, Wi-Fi typically available on smartwatches.

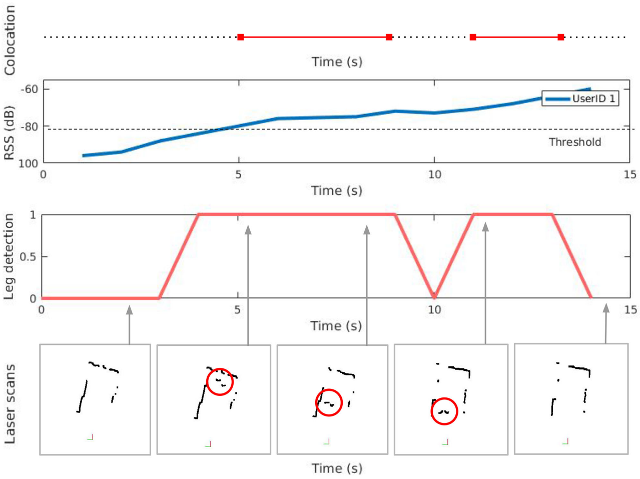


Fig. 3. Colocation detection. Bottom: Laser scans as seen by the robot, with the detected leg pattern highlighted in red; third row: output of leg detector (the output is 1 if any person is detected, 0 otherwise); second row: RSS from user ID 1 (the RSS threshold is shown by the dashed horizontal line); top row: detected colocation events.

When the robot detects a user and receives probabilistic activity data from that user, it triggers a colocation event. Fig. 3 shows an example of the colocation detection while the user is approaching the robot.

3) Semantic Map Updates: Each cell c in the robot's grid map is assigned a vector smap$[c]$ indicating the probability that cell c belongs to a room of a particular type. We use the abbreviation "smap" to refer to the semantic map, for example,

$$\text{smap}[c] = \begin{bmatrix} 0.20 \longrightarrow & \text{office} \\ 0.40 \longrightarrow & \text{kitchen} \\ 0.15 \longrightarrow & \text{bathroom} \\ 0.35 \longrightarrow & \text{bedroom} \\ \dots & \end{bmatrix}.$$

Let smap$[c]^r$ be the element of smap$[c]$ that corresponds to a certain room type r, for example, smap$[c]^{r=\text{kitchen}}$ is the current estimate of the probability that cell c is in the kitchen. At bootstrap, smap$[c]$ is uniformly distributed over all room types.

On detecting a colocation event, the robot highlights a number of cells that are within its view, with the intention of updating their semantic map probabilities. Fig. 4 shows the cells that are within the sensing range of the robot when it detects a person nearby. Note that if a robot is situated in a room and looks in the direction of the door, it ignores those cells that are beyond the door frame.

The probabilities of selected cells having different room types are then updated as follows:

$$\text{smap}[c]^r := \text{smap}[c]^r \times \sum_{a \in \text{Activities}} p(r|a) \times \text{HAR\_}p[a] \qquad (3)$$

where $p(r|a)$ is the probability of being in a room given activity a, and HAR_$p[a]$ is the probability that the user is actually performing that activity. In practice, this is implemented as a sum of logs of the prior and conditional probabilities, instead of a product of probabilities [22].

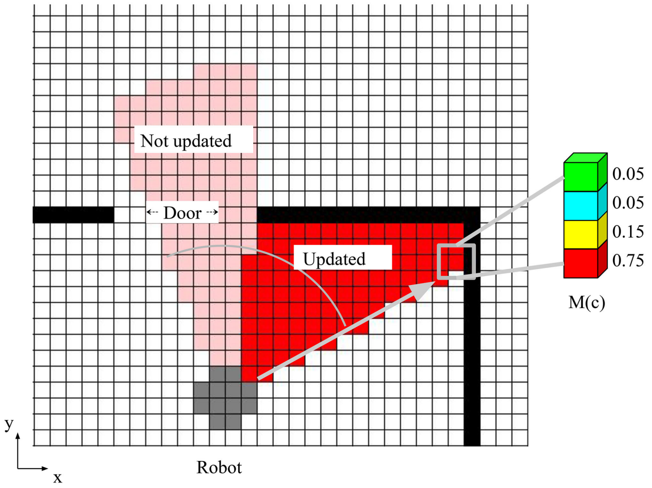


Fig. 4. Grid mapping update. Each cell is represented as a vector of room-type probabilities (shown in different colors) and is updated based on laser observations using a ray-tracing procedure along each laser measurement.

Probabilities $p(r|a)$ are drawn from the *Concept net* open-source knowledge graph [1], which gives a list of all possible activities associated with each room type, with a weight that represents the strength of the relationship between room and activity. We can exploit these weights, after normalization, in order to obtain usable priors.

The semantic map is updated after each robot–user colocation event. It can further be refined by taking into account that cells belonging to the same room should be of the same type. By averaging out the smap values of all cells in the same room, we obtain a probabilistic semantic label for each room.

V. User Localization

In this section, we propose a simple graphical model for room-level localization based on HMMs. The model is based on the joint probability distribution between user location and activity. The states of the model represent semantic room types and the transitions represent the transition probability between different room types, e.g., from kitchen to bathroom.

The model alternates between two phases, depending on the predicted activity, namely a *walking phase*, and a *stationary activity phase*. If a series of walking activities are detected, the model estimates the length of the walking phase in seconds (this is possible since activities are detected at a constant rate) and treats it as a single walking event, representing a transition between two nodes.

Otherwise, if another activity is detected, the model updates the probability distribution of each node according to emission probabilities, as in a classical HMM.

Let $\mathbf{p^t} = (p_1^t, \ldots, p_n^t)$ be the probability vector for the current location at time t, where n is total number of rooms. Each time a new activity is detected, the vector $\mathbf{p^t}$ is updated using one of the two rules discussed as follows.

A. Walking Phase Update

We model the walking phase via a random variable w, which contains information about the currently performed walking activity. Examples of possible interpretations for w are walking time, number of steps, walking distance, or even a part of a trajectory. In this paper, we consider the simple case that w represents the walking time between two stationary user activities.

Assuming w is a continuous random variable, we have

$$p_i^t = \sum_{j=1}^{m} p(r^{t-1} = r_j) \int p(r^t = r_i | w, r^{t-1} = r_j) p(w) dw \tag{4}$$

where we have assumed that W and r^{t-1} are statistically independent. The integral in the above-mentioned formula marginalizes over the uncertainty on the walking random variable w, whereas the sum marginalizes over the uncertainty of the location at the previous step r^{t-1}.

The term $p\left(r^t = r_i | w, r^{t-1} = r_j\right)$ represents the likelihood of the transition from room type r_j to room type r_i via a walking event w.

This formulation accounts for the uncertainty on the estimation of the walking times between rooms. For simplicity, we can evaluate the walking time w without uncertainty by estimating the duration of multiple contiguous walking activities. This results in the simpler formula

$$p_i^t = \sum_{j=1}^{m} p\left(r^{t-1} = r_j\right) \mathcal{N}\left(w; \mu_{ij}, \sigma_{ij}\right) \tag{5}$$

where μ_{ij} and σ_{ij} are the mean and standard deviation of the time required to walk from a room type r_i to a room type r_j, respectively.

In summary, the walking activity events are concatenated into a single walking event that acts as a control input in the HMM and impacts the transition probability between different room types.

B. Stationary Activity Phase Update

In the stationary activity phase, state probabilities are only updated using emission probabilities. The emission probability for a given room type represents the probability of observing an activity a given room type r. The state probabilities are then updated as follows:

$$p_i^t = p_i^{t-1} \sum_{j=1}^{m} p(a^t = a_j | r^t = r_i) p(a^t = a_j). \tag{6}$$

The factor $p\left(a^t = a_j\right)$ is the probability of the user performing activity a_j at time step t. It is the result of the activity prediction represented as HAR_$p(a_j)$ in Section IV-A.

Empirically, we found increased localization accuracy by tweaking the above-mentioned formula into

$$p_i^t = p_i^{t-1} \sum_{j=1}^{m} p(a^t = a_j | r^t = r_i) p(a^t = a_j)(1 - \sigma_j^t) \tag{7}$$

where the factor $\left(1 - \sigma_j^t\right)$ penalizes the effect of activity predictions that show a high standard deviation. By setting σ_j^t to HAR_$\sigma[a]$, the model is able to embed the uncertainty estimation from the variational BLSTM (see Section IV-A).

C. Training Phase

Note that the conditional probability $p(a^t = a_j | r^t = r_i)$ is learnt automatically before it is used within the HMM for localization. This occurs during the colocation events between the robot and the user. Whenever they are both in room r_i, the activity recognition module returns a vector HAR as discussed in Section IV-A. HAR vectors corresponding to the same room are averaged out in order to learn the conditional probability of activity given room.

VI. Experimental Results

We implemented our neural network using the Keras library and Tensorflow as the optimization back end. The semantic mapping system is implemented using the *robot operating system* (ROS). The source code as well as the user activity dataset used in the experiments will be available online.

A. User Activities

1) Data Collection Protocol: For training our network, we gathered inertial data from a set of 20 users [of ages between 24 and 60 (with $\mu = 31$)]. Users were given a smartwatch (Sony Smartwatch 3) to be worn on their right hand if right-handed or on the left if left-handed. We defined a list of complex daily activities typical of domestic environments. Each subject was asked to perform the activities, one by one, based on his/her own interpretation and style. In order to sufficiently sample the continuous movement of nontransient actions, each subject was asked to perform each activity continuously for 60 s or more. We define the following list of ten activities.

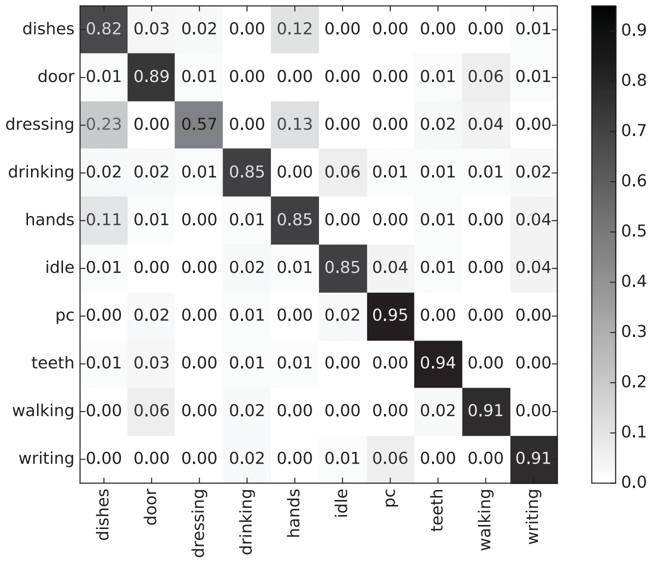
1) washing dishes;
2) opening door;
3) dressing up;
4) drinking/eating;
5) washing hands;
6) idling;
7) using a PC/laptop;
8) brushing teeth;
9) walking;
10) writing.

Two are simple activities (walking and idling), whereas the rest are complex activities that are typically performed very differently by different people and in different environments. In total, 3 h and 10 min of data were collected.

2) Training: We train our network architecture using standard backpropagation and the ADAM optimizer. For activity recognition, the input of the network is a sequence of three-axial acceleration data and three-axial angular velocity data of fixed length. Since the sensors present different sampling rates (the accelerometer samples acceleration at ∼100 Hz, while the gyroscope samples at a lower ∼30 Hz), we oversample the gyroscope data in order to match that of the accelerometer, using piecewise cubic spline interpolation.

TABLE I
Optimal Values for the Neural Network Hyperparameters

Parameter	Value
BLSTM layers #	2
Neurons layer 1	50
Neurons layer 2	200
$p_{W,1}$	0.8
$p_{U,1}$	0.05
$p_{W,2}$	0.05
$p_{U,2}$	0.05
p_{do}	0.05
batch size	64
learning rate	0.001



	dishes	door	dressing	drinking	hands	idle	pc	teeth	walking	writing
dishes	0.82	0.03	0.02	0.00	0.12	0.00	0.00	0.00	0.00	0.01
door	0.01	0.89	0.01	0.00	0.00	0.00	0.00	0.01	0.06	0.01
dressing	0.23	0.00	0.57	0.00	0.13	0.00	0.00	0.02	0.04	0.00
drinking	0.02	0.02	0.01	0.85	0.00	0.06	0.01	0.01	0.01	0.02
hands	0.11	0.01	0.00	0.01	0.85	0.00	0.00	0.01	0.00	0.04
idle	0.01	0.00	0.00	0.02	0.01	0.85	0.04	0.01	0.00	0.04
pc	0.00	0.02	0.00	0.01	0.00	0.02	0.95	0.00	0.00	0.00
teeth	0.01	0.03	0.00	0.01	0.01	0.00	0.00	0.94	0.00	0.00
walking	0.00	0.06	0.00	0.02	0.00	0.00	0.00	0.02	0.91	0.00
writing	0.00	0.00	0.00	0.02	0.00	0.01	0.06	0.00	0.00	0.91

Fig. 5. Confusion matrix for the variational BLSTM over ten classes.

We experimentally found that a window size of 3 s offers the best results for complex activity classification in most cases. This is due to the fact that these activities are composed by a series of movements that span over a longer time window, compared to classic activities, such as walking, running, biking, etc. We divide the data into windows of 3 s with an overlap of 50%. The data are subsampled to a frequency of 50 Hz and a median filter is applied on the raw data in order to smooth outlier measurements.

The optimal hyperparameters for the network were found using the Hyperas python package with tree-structured Parzen estimator (TPE) optimization and are reported in Table I. p_W, p_U, and p_{do} represent the dropout ratios for the $\mathbf{W}$ weight matrices, for the $\mathbf{U}$ weight matrices, and for the drop-out layer, respectively. Note that the batch size is dependent on the hardware setup.

Fig. 5 shows the classification results. The network achieves an accuracy of 87.5% on the test set. In order to validate the choice the proposed architecture, we also compared with three baselines: a nonvariational LSTM, a nonvariational BLSTM (i.e., dropout was disabled at prediction time), both using the same hyperparameters, and another nonrecurrent deep method

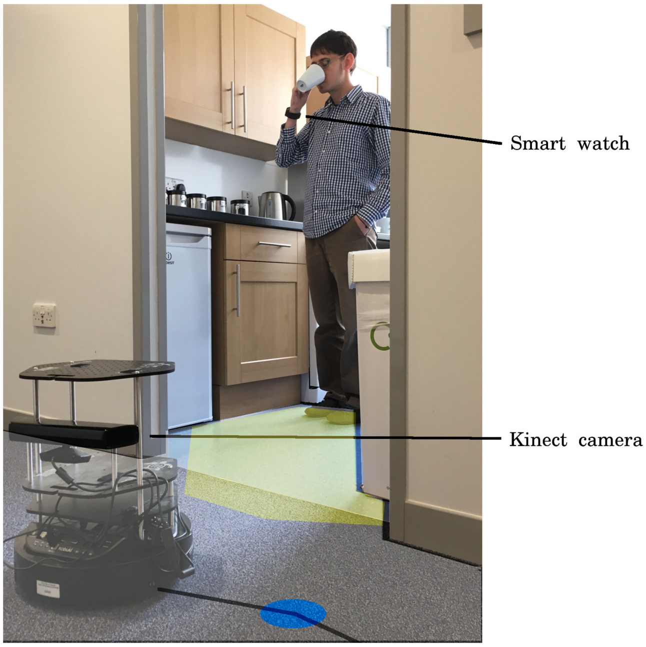


Fig. 6. Setup for the experimental tests during a colocation event, while the user is performing an activity. The user is wearing a smartwatch; the robot is using a Kinect camera for simulating a laser range finder.

[5]. The LSTM and the BLSTM achieved an accuracy of 78% and 82%, respectively. Alsheikh *et al.* [5] achieved 82.5% accuracy.

B. Semantic Mapping

We test the semantic mapping in both an office-like environment and a domestic environment. In our experiments, users are equipped with a smartwatch, connected via Wi-Fi to the robot. The robot is a Turtlebot 2 equipped with a Microsoft Kinect camera. The robot is using the Kinect to simulate a laser range finder to localize in the map, to detect doors using a simple template matching algorithm available in ROS and to detect the user using a simple classifier for leg detection based on laser scans. As mentioned before, the camera is not used due to privacy concerns. The first scenario is an office-like environment, composed by a series of rooms and a corridor. We had access to the planimetry of the floor in the form of CAD files, but the robot could build a map beforehand by performing SLAM. In our experiment, we are interested in mapping five rooms (lab, conference room, kitchen, office, and bathroom). There is a sixth multipurpose room in the center, but it is not included in the experiment since it is not represented by any particular set of activities. The setup for the experiment is shown in Fig. 6. For the second scenario, a grid map was built autonomously by the robot beforehand using the *gmapping* ROS package.

The experiment lasted for a total of 30 min per user, with the robot and the user moving in the environment, entering various rooms and triggering colocation episodes, and the cumulative result is shown in Fig. 7. It should be noted that in our experiments, the robot was wandering autonomously from room to room in a randomized manner.

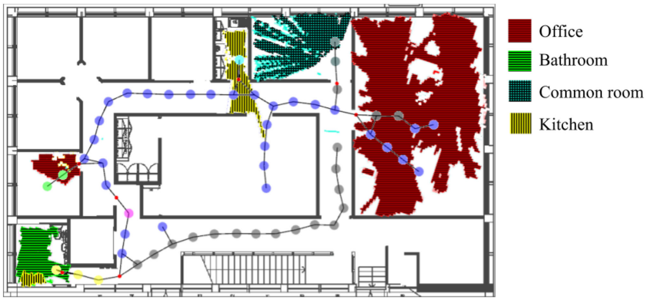


Fig. 7. Resulting semantic map for the first scenario (activities only). The estimated topological and semantic maps are superimposed a CAD map. Each color of the map corresponds to a different room type (blue = corridor; red = lab; light red = office; yellow = kitchen; green = bathroom; and cyan = common room). The topological map is represented by colored circles (each color represents a different room and red dots represents detected doors).

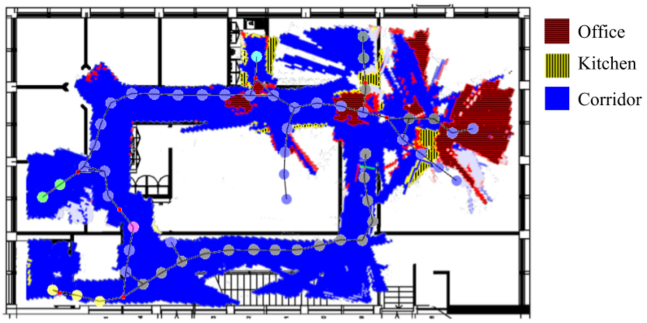
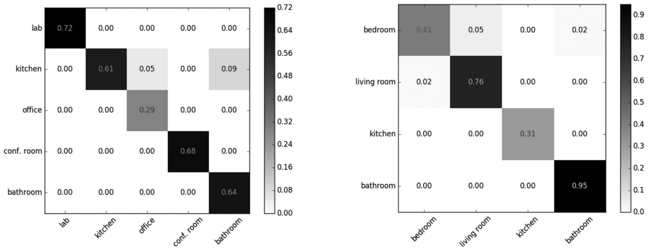


Fig. 8. Resulting semantic map for the first scenario with the approach proposed in [22].



(a) (b)

Fig. 9. Confusion matrix for the semantic maps in the two scenarios. Each row represents one room; each column represent a semantic label; we report the percentage of cells in each room that are classified with a particular semantic label. (a) First scenario. (b) Second scenario.

Some issues are visible in the resulting semantic maps. For instance, one door leading from the kitchen to the corridor was not correctly detected at first. This is due to the difficulty to tune the parameters of the door detector for different types of openings. This led to the part of the corridor nearby the kitchen to be labeled as corridor. The resulting semantic map is somewhat sparse in certain areas since there were few colocation episodes. Over a longer period of time, we can expect the map to become more complete. On the other hand, the probabilistic mapping procedure was able to cope with misclassified activities among the users, by smoothly updating the map probabilities over time.

Fig. 9 reports the ratio of map cells identified as a particular room type for the five rooms in the first scenario. The values are

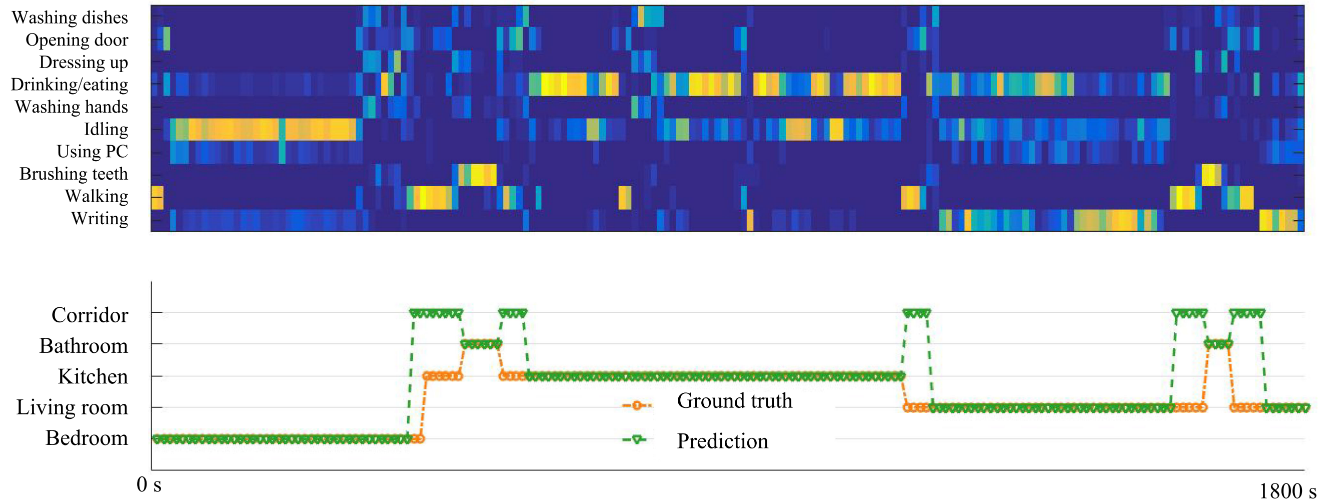


Fig. 10. Trace of activities and room location aligned in time. The top image shows the estimated activity probabilities; the bottom image shows the predicted location as well as the ground truth.

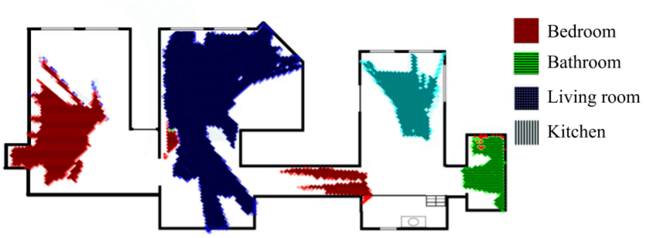


Fig. 11. Resulting semantic map for place classification in a domestic environment. Here, red = bedroom; blue = living room; cyan = kitchen; and green = bathroom. The map constructed by the robot is used here.

computed as the ratio between the cells classified as a particular room type and the total number of cells in each room. Note that the final mapped area is dependent on the presence of furniture or obstacles and on the trajectory of the robot. The values in Fig. 8(a) reflect the fact that only partial areas of each room have been mapped. For instance, as the office was occupied by a desk and several chairs, the robot could not reach the whole room.

In order to provide a baseline for semantic mapping, we also show the result of semantic mapping using the visual place classification approach from [22]. The result is shown in Fig. 8. In [22], a convolutional neural network based on AlexNet was pretrained on the *Places205* dataset [31] for place classification. The network takes RGB images in input from the Microsoft Kinect camera mounted on the robot. We only use the subset of the 205 place labels, which are relevant to the testing environment (office, kitchen, conference room, and corridor). It can be seen how the *corridor* class, absent from our method, is correctly classified by the network given in [22] at the cost of a large number of false positives. No fine-tuning of the network was done.

Fig. 11 shows the results of one run in a household composed by four rooms (bedroom, living room, kitchen, and bathroom), while Fig. 8(b) reports the ratio of map cells identified as a particular room type for the four rooms in the scenario. The experimental results show consistently accurate classifications.

C. User Localization

In this experiment, we show how we can combine the semantic map obtained in the first phase and successive colocation events in order to learn the parameters of a simple graphical model for user localization at room level, independently from the robot. We perform these experimental tests in the same two scenarios of the previous experiment. Inertial data were collected from a test set of five users. We show the localization results and compare them with the ground-truth location, which is obtained by placing BLE beacons in each room of interest in both scenarios.

The system first learns the correlation between room locations and activities in the form of emission probabilities for the different activities given room types. This is done over a series of colocation events over time. Since the robot has access to the semantic maps from the previous experiment, it is able to learn the emission probabilities over time. The relation between the activities and the six semantic rooms considered is plotted in Fig. 12. We expect that the activities performed in rooms which are of the same type to be similar (e.g., lab and office), so in this experiment we combine the two room types. Notice that the *opening door* activity is not considered, as it is not related to a specific room, but to the transition between rooms.

We use Laplace smoothing on the estimated transition probabilities. As the classes are somewhat unbalanced (e.g., people tend to spend most of the working day in the lab), the classification accuracy for each specific class is weighted by the number of samples in the class. The room-to-room distances used to estimate transition probabilities are obtained from the topological map built by the robot on top of the metric map.

We provide statistical results for the localization module in Table II for both environments and averaged over a test set of five users, and we compare our graphical model with a baseline

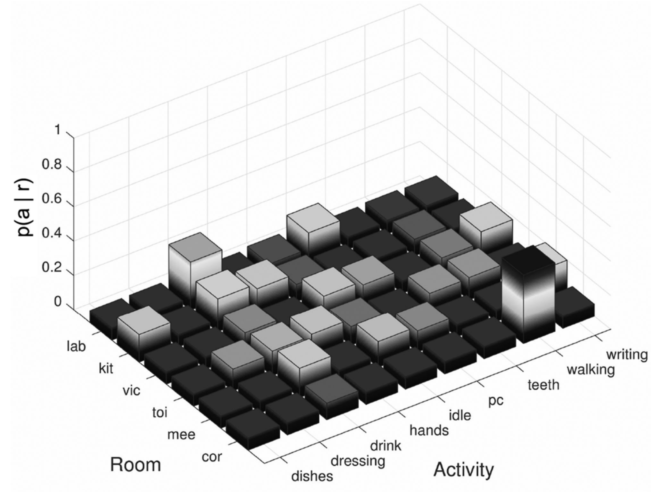


Fig. 12. Learned emission probabilities for activities performed in each room (office-like environment).

TABLE II
PREDICTION ACCURACY OF USER LOCALIZATION FOR BOTH SCENARIOS FOR THE PROPOSED MODEL AND A BASELINE HMM IMPLEMENTATION

		Precision	Recall	f_1 score
Baseline HMM	Office-like	0.8	0.71	0.75
	Domestic	0.8	0.75	0.77
Proposed model	Office-like	0.81	0.91	0.86
	Domestic	0.87	0.95	0.91

approach consisting of a trivial HMM implementation, where the transition and emission probabilities are the same as the proposed graphical model. It should be noted that for the office-like scenario we used 3 s windows, while for the domestic scenario a window of 5 s gave the best results. In Fig. 10, we show the detected activities along with the predicted room locations for one user in the second scenario, for a duration of 30 min. The results show how the proposed model can outperform a classical HMM in our particular task.

VII. CONCLUSION

This paper presented a framework that integrates assistive robots, which will be present in workplaces and households of the future, and consumer wearable devices, for sharing information between robots and users that benefit each other. In our scenario, a robot and the user coexist in a workplace or household. The robot creates a map using any sensor that can provide distance measurements, then it is able to navigate the environment using standard navigation algorithms. The user wears a smartwatch that continuously acquires inertial data. Whenever the robot and the user meet, user activities are used to build additional semantic layers on top of the map, representing room-type probability. We propose the use of a variational BLSTM network for recognizing complex spatio-temporal activities from raw data that keeps the whole framework probabilistic. Once a semantic map is available, raw data from the user's wearable device can be used to detect room types. Over time, we train a room-based graphical model for room-level localization for the user even in the absence of the robot. In the model, nodes represent room types and transitions represent transitions between room types. This enables the robot to know the type of room the user is in at any time for executing high-level tasks. Future work could be devoted to integrating a pedestrian dead reckoning algorithm into the localization module. Another interesting extension would be to investigate active exploration strategies for the robot in order to maximize the chance of colocation events. Finally, semantic user localization could provide real-time context information to context-aware reasoning systems for supporting users without the need to instrument the environment, relying instead on mobile autonomous robots and wearable sensors.

REFERENCES

[1] Conceptnet.io. Website. 2018. [Online]. Available: http://conceptnet.io/

[2] M. Aehnelt and B. Urban, "Follow-me: Smartwatch assistance on the shop floor," in *HCI in Business*, F. F.-H. Nah, Ed. Cham, Switzerland: Springer, pp. 279–287.

[3] A. Alarifi *et al.*, "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors*, vol. 16, no. 5, May 2016, Art. no. 707.

[4] K. Alexopoulos, S. Makris, V. Xanthakis, K. Sipsas, and G. Chryssolouris, "A concept for context-aware computing in manufacturing: The white goods case," *Int. J. Comput. Integr. Manuf.*, vol. 29, no. 8, pp. 839–849, 2016.

[5] M. A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, and H.-P. Tan, "Deep activity recognition models with triaxial accelerometers," in *Proc. Workshop Artif. Intell. Appl. Assistive Technol. Smart Environ.*, 2016.

[6] C. Cadena *et al.*, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[7] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1019–1027.

[8] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional LSTM networks for improved phoneme classification and recognition," in *Proc. Int. Conf. Artif. Neural Netw.*, Springer, 2005, pp. 799–804.

[9] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, AAAI Press, 2016, pp. 1533–1540.

[10] M. Hardegger, D. Roggen, A. Calatroni, and G. Tröster, "S-smart: A unified Bayesian framework for simultaneous semantic mapping, activity recognition, and tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 7, no. 3, pp. 1–28, 2016.

[11] M. Hardegger, D. Roggen, and G. Tröster, "3d ActionSLAM: Wearable person tracking in multi-floor environments," *Pers. Ubiquitous Comput.*, vol. 19, no. 1, pp. 123–141, 2015.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[13] S. Lee, Y. Kim, D. Ahn, R. Ha, K. Lee, and H. Cha, "Non-obstructive room-level locating system in home environments using activity fingerprints from smartwatch," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Osaka, Japan, Sep. 7–11, 2015, pp. 939–950.

[14] G. Li, C. Zhu, J. Du, Q. Cheng, W. Sheng, and H. Chen, "Robot semantic mapping through wearable sensor-based human activity recognition," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 5228–5233.

[15] X. Lu, H. Wen, H. Zou, H. Jiang, L. Xie, and N. Trigoni, "Robust occupancy inference with commodity WiFi," in *Proc. IEEE 12th Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2016, pp. 1–8.

[16] S. Makris, P. Karagiannis, S. Koukas, and A.-S. Matthaiakis, "Augmented reality system for operator support in human–robot collaborative assembly," *CIRP Ann.-Manuf. Technol.*, vol. 65, no. 1, pp. 61–64, 2016.

[17] S. Pillai and J. Leonard, "Monocular SLAM supported object recognition," in *Proc. Robot., Sci. Syst.*, Rome, Italy, Jul. 2015.

[18] A. Pronobis and P. Jensfelt, “Large-scale semantic mapping and reasoning with heterogeneous modalities,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 3515–3522.

[19] F. B. A. Ramos, A. Lorayne, A. A. M. Costa, R. R. de Sousa, H. O. de Almeida, and A. Perkusich, “Combining smartphone and smartwatch sensor data in activity recognition approaches: An experimental evaluation,” in *Proc. Int. Conf. Softw. Eng. Knowl. Eng.*, 2016, pp. 267–272.

[20] J. Ranjan and K. Whitehouse, “Towards recognizing person-object interactions using a single wrist wearable device,” in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Heidelberg, Germany, Sep. 12–16, 2016, pp. 722–731.

[21] R. F. Salas-Moreno, B. Glocken, P. H. J. Kelly, and A. J. Davison, “Dense planar SLAM,” in *Proc. IEEE Int. Symp. Mixed Augmented Reality*, Sep. 2014, pp. 157–164.

[22] N. Sunderhauf *et al.*, “Place categorization and semantic mapping on a mobile robot,” in *Proc. IEEE Int. Conf. Robot. Automat.*, Stockholm, Sweden, May 2016, pp. 5729–5736.

[23] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik, “Indoor localization without infrastructure using the acoustic background spectrum,” in *Proc. 9th Int. Conf. Mobile Syst., Appl., Services*, ACM, 2011, pp. 155–168.

[24] D. Vasisht, S. Kumar, and D. Katabi, “Decimeter-level localization with a single Wi-Fi access point,” in *Proc. 13th USENIX Conf. Netw. Syst. Des. Implementation*, Berkeley, CA, USA, 2016, pp. 165–178.

[25] S. Wang, H. Wen, R. Clark, and N. Trigoni, “Keyframe based large-scale indoor localisation using geomagnetic field and motion pattern,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 1910–1917.

[26] J. Xavier, M. Pacheco, D. Castro, A. E. B. Ruano, and U. Nunes, “Fast line, arc/circle and leg detection from laser scan data in a player driver,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2005, pp. 3930–3935.

[27] Y. Xiang and D. Fox, “DA-RNN: Semantic mapping with data associated recurrent neural networks,” in *Proc. Robot., Sci. Syst.*, 2017.

[28] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, “Robust indoor positioning with lifelong learning,” *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2287–2301, Nov. 2015.

[29] R. Yao, G. Lin, Q. Shi, and D. C. Ranasinghe, “Efficient dense labelling of human activity sequences from wearables using fully convolutional networks,” *Pattern Recognit.*, vol. 78, pp. 252–266, 2018.

[30] Y. Zhao, R. Yang, G. Chevalier, and M. Gong, “Deep residual Bidir-LSTM for human activity recognition using wearable sensors,” 2017, arXiv:1708.08989.

[31] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning deep features for scene recognition using places database,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 487–495.

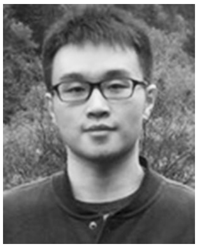
[32] Y. Zhuang, J. Yang, Y. Li, L. Qi, and N. El-Sheimy, “Smartphone-based indoor localization with Bluetooth low energy beacons,” *Sensors*, vol. 16, no. 5, p. 596, 2016.

Stefano Rosa received the Ph.D. degree in mechatronics engineering from Politecnico di Torino, Torino, Italy, in 2014.

He is currently a Research Fellow with the Department of Computer Science, University of Oxford, Oxford, U.K. His current research interests include cross-modality learning for long-term navigation, human–robot interaction, and intuitive physics understanding.

Andrea Patanè received the Bachelor’s and Master’s degrees in mathematics from the University of Catania, Catania, Italy, in 2014 and 2016, respectively. He is currently working toward the D.Phil. degree with the Autonomous Intelligent Machines and Systems Centre for Doctoral Training, University of Oxford, Oxford, U.K.

He is involved with the AffecTech ITN as an early stage Researcher.



Chris Xiaoxuan Lu received the M.Eng. degree from Nanyang Technology University, Singapore, in 2015. He is currently working toward the Ph.D. degree with the Department of Computer Science, University of Oxford, Oxford, U.K.

His research interests include ubiquitous and mobile computing, with a focus on enabling ambient intelligence for Internet of Things via cross-modality inference.

Niki Trigoni received the D.Phil. degree in computer science from the University of Cambridge, Cambridge, U.K., in 2001.

She is currently a Professor with the Department of Computer Science, Oxford University, Oxford, U.K., and a Fellow with Kellogg College, Oxford, U.K. She was a Postdoctoral Researcher with Cornell University (2002–2004) and a Lecturer with Birkbeck College (2004–2007). At Oxford University, she is currently the Director of the EPSRC Centre for Doctoral Training on Autonomous Intelligent Machines and Systems, a program that combines machine learning, robotics, sensor systems, and verification/control. She also leads the Cyber Physical Systems Group, which is focusing on intelligent and autonomous sensor systems with applications in positioning, healthcare, environmental monitoring, and smart cities. The group’s research ranges from novel sensor modalities and low-level signal processing to high-level inference and learning.