An indoor localization system for telehomecare applications

Augusto Luis Ballardini, Lorenzo Ferretti, Simone Fontana, Axel Furlan and Domenico Giorgio Sorrenti

***Abstract*—In this paper we present a novel probabilistic technique, based on the Bayes filter, able to estimate the user location, even with unreliable sensor data coming only from fixed sensors in the monitored environment. Our approach has been extensively tested in a home-like environment, as well as in a real home, and achieves very good results. We present results on two datasets, representative of real life conditions, collected during the testing phase. We detect the patient location with sub-room accuracy, an improvement over the state of the art for localization using only environmental sensors. The main drawback is that it is only suitable for applications where a single person is present in the environment, like as with other approaches that do not use any mobile device. For this reason we introduced the "telehomecare" term, therefore differentiating from generic telemedicine applications, where many people can be in the same environment at the same time.**

***Index Terms*—indoor localization, motion sensor, telehomecare, wireless sensor network, smart home.**

I. Background and Objectives

The population in today's western countries is rapidly growing older. Kinsella and Phillips [1] estimated that by 2030 23.5% of the European population will be above 65 years old, and that by 2050 only 22% of the world elderly people will live in what are nowadays called developed countries. In this context, being able to provide high quality, yet cheap, heath-care and assistance is gaining importance. A common approach to achieve this objective is the use of remote-monitoring of patients through pervasive networks of sensors. A pervasive network can provide several benefits to elderly people: from constant monitoring of vital parameters, to emergency communications and memory enhancement. These capabilities can help to early diagnose illness or, in some situations, can even be life-saving. Consider e.g., fall detection or heart-beat monitoring applications, where the monitoring system can give critical help by automatically calling medical assistance [2]–[4]. Remote assistance has many advantages over both retirement homes and "human" home assistance. First of all it is much cheaper [5], [6]; this is an important factor, given the increase of the aged population [7]–[9], and the declining amount of resources available for these people in many countries. Moreover, elderly people often prefer to remain in their home [10]. Self-sufficient people that still require some level of assistance or monitoring will greatly benefit from being remotely assisted at their homes. Within a remote monitoring system, the localization component plays a key role as the location of the person is used in several ways, e.g., to analyze behavioral patterns so that certain afflictions may be diagnosed early [11], to generate an alert, e.g., when a person remains still for too much time, or to provide a better interaction with the user, for example displaying visual contents and playing audio reminders in the room where the person is. In the last decade, various approaches to indoor localization have been developed. These can be roughly divided into two main categories: those using a device worn or carried by a person and those that rely only ¡on environmental sensors. The majority of the approaches falling in the first class use some type of wireless device carried by the person, in conjunction with a static infrastructure. A number of techniques have been developed along these lines: RFID-based localization, cell of origin, time of arrival, received signal strength indicator (RSSI), time difference of arrival and others. The interested reader can refert to [12] for an extensive review. These techniques can often achieve meter accuracy and, sometimes, even sub-meter accuracy, but they have one main drawback: they require a device carried by the person. This represents a major disadvantage, since a home-care system should be as minimally intrusive as possible. Ideally a person should be monitored without being affected by the monitoring system. Moreover, the eventuality of forgetting to wear the device or not wearing it on purpose for short periods, such as when getting up during the night or when showering, is definitely not negligible and would void the safety function of the telehomecare system. Systems of this class tend to work less than optimally in real-world scenarios as their accuracy may decline greatly depending on other electro-magnetic fields and on the presence of objects affecting the electro-magnetic field [13].

There are various contributions in the literature that approach the person localization problem while avoiding wearing devices. Noury *et al.* [14], [15] used motion sensors and magnetic contact switches placed on the doors, in conjunction with a rule-based system, to track the position of a single person in an apartment. Lee *et al.* [16] described a system that uses raw data from motion sensors, without any elaboration or filtering. A similar work has be developed by Ogawa *et al.* [17] that describe a system using motion sensors in conjunction with flame and CO_2 detectors and magnetic switches. A similar approach also has been developed by Han *et al.* [18], with the addition of an Autoregressive Hidden Markov Model to better model the occupancy pattern.

De Miguel-Bilbao *et al.* [19] performed a comparative analysis of three indoor localization platforms, one using passive infrared sensors (the most common type of motion sensor), one using UWB (Ultra WideBand) localization (a wireless-based indoor localization technology that requires the person to carry a device), and one based on RFID (Radio-Frequency Identification), showing that the first two have the same level of accuracy, while the third performs slightly worse. Hauptmann *et al*. [20] proposed a solution that uses cameras to track people in a nursing-home. However, it is well-known that people usually do not like to live constantly supervised by cameras [21]; also our experience demonstrates this opposition to cameras, as we had to spend time to persuade people that our devices were just PIRs (Passive InfraRed motion sensors), that do not include any video-gathering device.

Djuriç *et al*. [22] developed an approach to indoor localization, using PIR sensors, that uses a particle filter to improve the accuracy of the system. However, this kind of approaches uses ceiling mounted sensors and needs to know the exact footprint of the field of view of the PIRs in order to work properly. This is a huge disadvantage, because there is lot of uncertainty on both the field of view and the range of PIR sensors, and hypothesizing to handle them at installation time makes the approach unrealistic. Moreover, furniture and walls may block the field of view of a sensor, thus making calculating its exact footprint even more difficult. Hence, this kind of techniques is hardly usable in a real home environment (the authors provide only simulated results) and the installation would be very complex and error prone, since the exact sensor positions, w.r.t. some global reference frame, have to be measured. A similar approach, that uses a mixture of Gaussian based tracking algorithm instead of a particle filter, has also been developed by La Scala *et al*. [23].

Another common approach consists in using an array of PIR sensors, whose fields of view overlap. With this configuration, the system is able to determine a more accurate localization by evaluating when motion is detected in the intersection of several fields of view [24], [25]. However, the large uncertainty on the real shape of the field of view of the PIR sensors is an obstacle to an effective utilization of this family of techniques too. Indeed, the exact shape of the field of view has to be known in order for this systems to work properly. An attempt to mitigate this problem has been made by Kim *et al*. [26], that used a Bayesian classifier along with the analog output of a single PIR sensor, in order to better estimate when a person is inside the field of view, when on the boundaries and when outside. The problem of correctly estimating the real field of view of PIR sensors has been reported by other authors too, [27], and it is one of the challenges that led us to develop the proposed approach. Yokoishi *et al*., in order to tackle the unreliability of PIR sensors in a real-life scenario, determined the occupancy of a single room, for energy saving goals, using a particle filter and multiple motion sensors, [28]. While similar to our approach, this approach was limited to a single room and thus lacks sub-room accuracy and occupancy determination in a whole apartment, along with easy map construction and sensor placement.

Álvarez-García *et al*. [29] compiled, for the EvALL competition, a list of five criteria for the evaluation of indoor localization systems in assisted living environments. This competition is divided into three phases, each with different objectives and difficulties, so that the same criteria have to be calculated differently in different phases. During the first phase the systems are required to localize a person in various areas of interest, a very similar scenario to the one described in this paper. The criteria are:

- Accuracy
- Availability
- Installation complexity
- User acceptance
- Interoperability

It is our opinion that having common criteria to evaluate many different assisted living localization systems is very important. So, regarding our proposal, we will provide quantitative and/or qualitative values for each listed criterion.

A common approach to indoor localization for assisted-living is to divide the home into areas of interest, usually coinciding with the rooms. We tried to make improvements over this approach using smaller areas of interest, to achieve a finer grained localization. This gain is very useful in a behaviour monitoring software, given the wide range of activities that can be performed in some rooms. By localizing the patient in different areas of a room, we are able to distinguish, for example, when a person is sitting on the sofa, from simply reporting that she/he is in the dining room. The main improvement of our proposal is related to handling the sensors' inherent noisiness. Previous state-of-the-art works do not account for this important factor. Some approaches simply use the sensor output "as-is" [16], [17], while others try to minimize the impact of incoherences, such as when a person is detected in a zone non contiguous to the one occupied previously, or when she/he is detected in more zones simultaneously [14]. Instead, our solution has been purposely designed to deal with sensors providing realistic data, i.e. noisy data. To accomplish this task we use a Bayes filter, a filtering technique commonly used for solving the problem of estimating the state of a system from noisy observations [30]. In this way we are able to deal with both false negative and false positive sensor readings. Thus, we do not need an explicit definition of the rules anymore (see e.g. [15]), but we implicitly deal with the noise using the filtering approach. This approach requires, on one hand, a probabilistic model of the sensors, so to take into account their uncertainty; on the other hand a motion model that probabilistically describes how a person moves from an area to another. We can cope with the vast range of possible incoherences caused by sensor noise, without the need to model each possibility individually. Instead, using a rule based system, this would be necessary.

II. The localization system

The goal of the proposed work is to localize a person within her/his own home. The home has been divided into multiple areas, sometimes corresponding to single rooms, sometimes to a specific part of a room (such as the sofa area in a dining room). To perform the localization we use a set of passive motion sensors, which have low energy requirements and are really unobtrusive. These are the same kind of sensors typically used in burglar alarm systems. In contrast with the majority of the approaches found in the literature, our system does not require the user to wear a device. However, this important advantage led us to design our solution for environments where only a single person is present (a very common requirement among systems that do not use any mobile device [14]–[17]). Nevertheless we do not see this as limiting the real-world usefulness of our work, because the system is able to smoothly recover from localization failures, even from those caused by the presence of another person in the home. Furthermore, telehomecare systems are often aimed at people living alone most of the time, i.e., living in conditions such that telemonitoring looses its relevance when also other people (e.g., a doctor, relatives, etc.) move through the home.

A. Hardware infrastructure

To track the position of a person inside a home, we used a network of ZigBee devices, [31], equipped with Passive InfraRed motion sensors (PIR). A PIR is essentially a motion sensor that detects changes inside its field of view, but only in the infrared spectrum. The human body emits infrared radiations and therefore activates the sensor. It has to be noticed that there might be other sources of infrared radiations inside a home, for example a heater or the sun coming through a window. However, many modern PIR sensors are designed in order to detect only infrared rays corresponding to the wavelengths emitted by the human body, so to reduce, though of course not to eliminate, the detection of other disturbing sources. This kind of sensors, while being cheap and very easy to use, have proved to be relatively unreliable. Specifically, the range and the field of view vary greatly among different testing conditions and among different units and may be very different from what is specified on the datasheet. For this reason, in a realistic non-controlled environment, an approach that is robust against noise and against uncertainty on the area covered by a sensor is essential . This was one of the main goals of our work and the main innovation w.r.t. other approaches presented in Section I.

ZigBee is a network protocol aimed at low-power wireless sensor networks, so it is a perfect fit for our purposes. Our network infrastructure relies on three types of devices: *end devices*, *routers* and a *coordinator*. The *end devices* are the nodes actually used to sense the environment. They are equipped with a PIR and other environmental sensors to measure pressure, temperature, humidity and the light intensity in a particular area of the home. These other sensors provide additional data to the remote monitoring software, but are not used for the localization task, so they won't be described further in this paper. The *routers* are in charge of delivering to the *coordinator* the messages coming from the subset of *end devices* assigned to them, but they do not have any sensing capability. Lastly, the *coordinator* is connected to a personal computer to which it delivers each message coming from the wireless network. We opted for a *tree* topology rather than for a *star* one, since the former is more energy efficient: the *end devices* do not need to be awake all the time, but are allowed to fall back to the sleep mode immediately after transmitting the sensed information. To further reduce energy consumption, we developed the firmware on the *end devices* in order to send the output of the PIR sensor only once per second, rather than asynchronously whenever some motion is detected. While a person is passing through the field of view of a sensor, it may detect her/his motion several times, so, without our policy, this may result in multiple transmissions. Given the high energy cost of a wireless transmission and the fact that humans move at a relatively slow pace, a message every second proved to be enough with our approach, while providing a significant reduction in power consumption. So, the message coming from an *end device* specifies whether the PIR has detected any motion in its field of view within the last second.

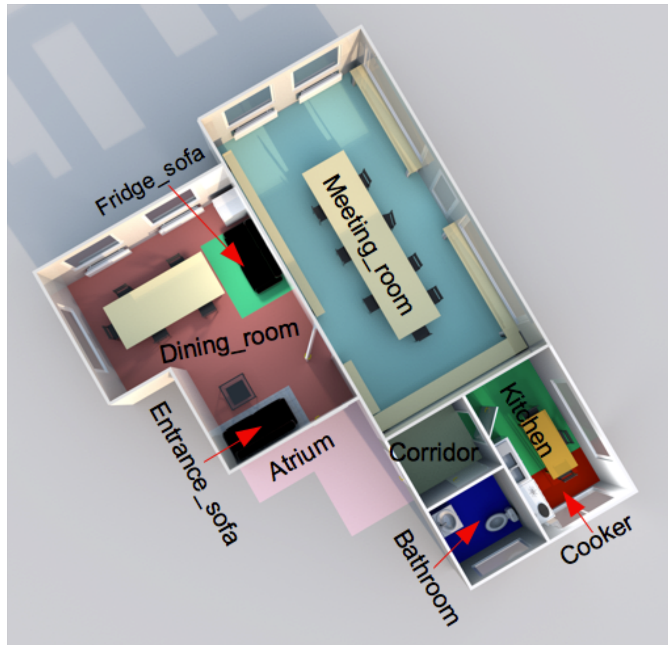


Fig. 1. The map of our test environment. Different areas are highlighted with different colours.

The *end devices* may be installed in several ways. The easiest way is to install them so that a sensor's field of view corresponds roughly to a macro area of the localization algorithm. However, this is not always possible. For example, considering the map in Fig. 1, it would not be possible to cover just the *Entrance_sofa* area with a single sensor. In order to accomplish this, we would need a sensor with a very short range and/or field of view, or we should have fixed one at the ceiling, just above the sofa, which is quite an uncomfortable installation option. Using sensors with different ranges and/or fields of view on different sensing nodes was a solution that we wanted to avoid, because it greatly reduces the flexibility of the system: a node would not be reusable in different parts of the home and it would have to be replaced with an identical one in case it broke. Fixing the nodes at the ceiling, instead, has been considered not suitable for both safety and practical installation reasons. We solved the issue using two conveniently placed sensors, as will be illustrated in Section III. This is just an instance of the cases where the field of view of some sensors covers multiple areas. Of course, situations where some areas are covered by more than one sensor and where some areas are only partially covered are also common. Our proposal has been designed in order to cope with all these real-life situations.

procedure BAYESFILTER($\{p_{k,t-1}\}, u_t, z_t$)
  for all k **do**
    $\overline{p}_{k,t} \leftarrow \sum_i p(X_t = x_k | u_t, X_{t-1} = x_i) p_{i,t-1}$
    $p_{k,t} \leftarrow \eta p(z_t | X_t = x_k) \overline{p}_{k,t}$
  end for
  return $\{p_{k,t}\}$
end procedure

Fig. 2. The Bayes Filter algorithm [30]

B. The algorithm

The localization problem may be formulated as the problem of estimating the state of a dynamic system, given an uncertain estimate of the previous state and noisy measurements. The state of the system is represented by the area where the person is, while the measurements are the outputs of the sensors. A naïve approach could use the raw outputs of the sensors to deduce where the person is: when a sensor detects some movement, the person is believed to be in its field of view. However, this approach has many drawbacks. One of the most important is related to the fact that real-world sensors are noisy. Most of the time a sensor detects motion only when a person is really traversing its field of view, but sensors may fail, as they may detect some motion when actually there is none (false positive) or fail to detect movement in their field of view (false negative). These problems are particularly relevant when the person is near the boundary of a sensor's field of view or very distant from the sensor. Another problem of the naïve approach is that it does not take into account the fact that a person cannot suddenly move ("jump") from a room to another, non-adjacent, room. In order to model this physical constraint, we leverage a concept normally used in state estimation: the state transition equation. In our case the unknown state is the location of the person, and the state transition equation represents how a person can move from a location to another; it is therefore called *motion model*.

To deal with these problems we used a probabilistic approach, called Bayes Filter. Instead of stating where the person is, this approach estimates the probability distribution of the presence of the person, over the whole state space. For each area, the probability that the person can be found in that area is provided. This approach has the advantage of dealing with noise, of using both a motion model and the sensors' measurement model, and of being able to smoothly recover from a failure. The Bayes filter is depicted in Fig. 2, where $p_{k,t}$ is the probability of the state k at time t, $\{p_{k,t}\}$ is the probability distribution on the state space and u_t is the control vector, representing an external variable affecting the state transition. In our case there is no external control that intervenes on the person, as she/he deliberates autonomously on her/his motion. Lastly z_t is the measurement, and $p(z_t|X_t = x_k)$ is the sensor model. Two main steps compose the algorithm: prediction and update. The prediction step leverages the motion model to calculate the probability that a person is located in each particular area, a priori of the sensor measurements. This is done given the probability distribution at the previous time step. $p(X_t = x_k|u_t, X_{t-1} = x_i)$ is calculated as shown in Equation (1), where *prob_stay*, *prob_move* and *prob_jump* are parameters of the algorithm and $N(area_k)$ is the set of $area_k$ neighboring areas.

$$p(X_t = x_k|u_t, X_{t-1} = x_i) = \\ = \begin{cases} prob\_stay & \text{if } area_i = area_k \\ prob\_move & \text{if } area_i \in N(area_k) \\ prob\_jump & \text{if } otherwise \end{cases} \quad (1)$$

Equation (1) assigns a probability to the event of the person remaining in the same area, a (typically smaller) probability to the event of the person moving to another adjacent area, and a (small, but not zero) probability to the event of the person jumping to a non-adjacent area. These probabilities are given at configuration time and reflect the following intuition: the chances that a person disappears from an area and reappears in a non-adjacent one are mainly related to a pose estimation error at the previous time steps; thus, the *prob_jump* probability should be very low, but not zero, to allow to recover from a failure at a previous time, as illustrated below. Secondly, theoretically the *prob_stay* and *prob_move* probabilities could be equal; however, it is important to notice that the PIR sensors detect motion and not proximity. As a consequence, if the user stops within the field of view of a sensor (absence of new motion data), the belief about her/his position should not change significantly: it should be localized in the last position, while it appears to be reasonable to have an increase of the uncertainty of this outcome, which means to take into consideration that the user might have moved to a nearby area. The fact that the probability of a person jumping to a non-adjacent area is not zero may seem unreasonable, but this is a key issue of the algorithm. This probability should not be zero because sometimes the localization algorithm may fail; by assigning a quite small probability to the event of the user jumping from an area to a distant one, our approach can recover from a failure, given an adequate evidence, i.e., several measurements reporting that the person is in another area. In our opinion, this is one of the main advantages of using a probabilistic approach over a rule-based one.

For example, referring to the map in Fig. 1, let us assume that the person goes from the atrium to the cooker area, passing through the corridor and the kitchen, but, for some reason, the sensor in the corridor does not detect any movement. A simple rule-based system will never localize correctly the person, until she/he returns to the corridor. Our probabilistic approach, given that the sensor in the cooker correctly detects the movement, will eventually correctly localize the person. This scenario, in a real, non-controlled world, is more probable than it may seem. It is often hard to provide 100% sensor coverage of a particular area, so false negatives may happen. Another situation where our formulation is useful is represented by the following example: the person is in the dining room and moves to the *fridge_sofa* area, but passes near the door connecting to the meeting room. As already mentioned, the borders of the sensor's field of view are not so neat, so it is pretty likely that in the described situation the sensor in the meeting room detects some movement. A simple rule-based system will first localize the person in the meeting room and it will not re-

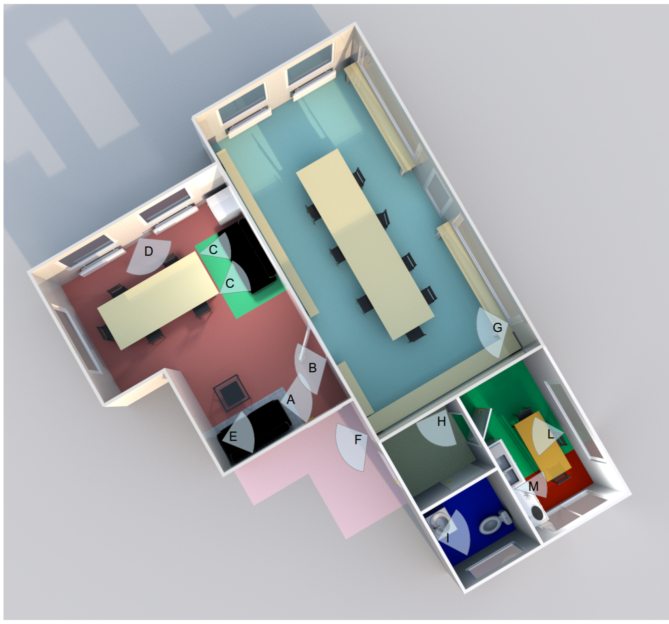


Fig. 3. Sensor placement and their field of view in our test environment

localize her/him in the *fridge_sofa* area, because these areas are non adjacent. Instead, our probabilistic approach correctly deals with false positives and will eventually localize the person in the correct area. Of course, the frequency of this kind of problems may be consistently minimized with a better sensor displacement, however this is not always possible in a real home and motion sensors are really unreliable at the borders of their field of view.

The second step of the Bayes Filter algorithm is the update step. This fundamental step weights the prediction with the likelihood of the measurement $p(z_t|X_t = x_k)$ that could be expected considering the predicted value of the state. In our implementation the measurement is a list containing the identifiers of the sensors that detected some movement in the last time interval. The likelihood of the measurement, given an area (i.e., considering the person being in that area), can be calculated in many different ways, depending on the displacement of the sensors in the area. In the easiest case, a sensor covers just a single area, so we assign a high likelihood to a measurement vector containing the identifier of that sensor, a lower likelihood to a measurement vector containing the identifiers of the sensors in adjacent areas (because the real field of view is uncertain) and an even lower probability otherwise. However, in practice, we can have various sensor configurations: multiple sensors may cover a single area, a sensor may cover multiple areas or various combinations of these. As it would be very hard to formulate an expression that covers the whole set of real world cases, our algorithm lets the person in charge of configuring the localization system specify a list of boolean expressions with associated likelihood. The algorithm evaluates these expressions in sequence and eventually falls back to a default (low) likelihood. For example, for the map in Fig. 1 and the sensor placement in Fig. 3, for the dining room area we used the expression in Equation (2), where G, E, C, B and A are the identifiers of the motion sensors and *high_prob*, *low_prob* and *lowest_prob* are user-specified parameters.

$$p(z_t|X_t = DiningRoom) = \\ = \begin{cases} lowest\_prob & \text{if } G \wedge \neg B \\ high\_prob & \text{if } A \\ low\_prob & \text{if } E \vee C \\ high\_prob & \text{if } D \vee B \\ lowest\_prob & \text{if } otherwise \end{cases} \tag{2}$$

The identifiers of the motion sensors in Equation (2) are boolean variables whose values at a certain time are given by Equation (3), where S is the set of all the sensors' identifiers and $M(t)$ is the set of the identifiers of the sensors that detected movement at time t.

$$\forall s \in S, s = True \iff s \in M(t) \tag{3}$$

The person in charge of the configuration of the localization system, after the installation of the network of sensors, will compile the list of boolean expressions for each location area, which will represent the specificities of that installation.

The algorithm in Fig. 2 requires the a priori distribution of the belief about the area where the person is, from the previous time step. The algorithm is run every time a new measurement vector is received. For this reason it needs an initial distribution, a priori of any measurement. Many different types of initial distribution can be used, however we assign a high probability to a human-provided initial state and a really low one to all the other states. Alternatively, a uniform distribution can be used, which results very useful when there is no knowledge about the initial position. It is worth noticing that, even in this second case, the proposed approach converges to the correct user localization after few iterations, thanks to its effective probabilistic formulation.

The algorithm described so far has a drawback: when an empty measurement vector is received for several consecutive seconds (for example because the personis not moving), the uncertainty starts spreading over the state space, eventually bringing the system back to a state of little knowledge about the person's position (nearly uniform distribution over the state space). This is particularly significant while the person is sleeping, because she/he will stay still enough to result undetected by the motion sensors. For this reason we used a prediction model slightly different with respect to the base one described in Equation (1): when a non-empty measurement vector is received, Equation (1) is used, otherwise Equation (4) is applied instead.

$$p(X_t = x_k|u_t, X_{t-1} = x_i) = \\ = \begin{cases} prob\_stay & \text{if } area_i = area_k \\ prob\_jump & \text{if } otherwise \end{cases} \tag{4}$$

In this case, we assign a high probability to the event of the person staying in the same area as the previous iteration and a very low one to all the others. This formulation reflects the fact that, if an empty measurement vector is received, most likely the person is stationary in a given area.

The output of the algorithm in Fig. 2 is a probability distribution over the state space. However, in the field of remote assistance, usually a unique location of the person is required. The simplest way to obtain this information from a distribution is to pick the state with the highest probability. This approach works quite well, although it is usually better to pick the most likely state only when its probability is above a certain threshold or when the difference w.r.t. the second most likely state is above another threshold. These selection approaches make it clear that there are limits under which we are not able to say with enough confidence where the person is. Despite this may seem a drawback, a localization system reporting that it does not know where a person is located is much better, and correct, than a system reporting a wrong location. This would not be possible without a probabilistic approach.

C. Technical details

The software implementing our proposal has been written using the Python programming language and consists of components that use the publish/subscribe paradigm to exchange messages. Python has been chosen because of its suitability to rapid prototyping and, most important, for its cross-platform nature. Performance was not an issue, given the low computational resources required by the algorithm.

In our system there are two main components: the localization and the data producer component. The latter interacts with the first via messages, using the publish/subscribe paradigm provided by the RabbitMQ framework [32]. This means that the data producer puts messages on a topic and that any component listening to that topic may view them. This provides a high level of decoupling among components and it is particularly useful because we can either use, as data producer, a software that interacts with the real sensor network, one that simply reads a dataset, or even one that simulates realistic data, without changing a single line of code or the configuration of the parameters of the localization component.

Each area of the monitored environment has its specific measurement function, i.e., set of boolean expressions, depending on the particular sensor configuration in that area. Our software provides a straightforward way to specify these functions, via a configuration file, without the need to re-write the component programming code. It uses an XML file to describe a high level map of the environment: each area has a name, an initial value for the probability of being the area where the person is, a set of neighbouring areas (i.e. areas that share a boundary with it) and a set of expressions with an associated likelihood (Fig. 4). These expressions are written in prefix notation so they are easy to write, even for non programmers; moreover they are very easy to parse and unambiguous, even without parentheses (although parentheses may be used to improve human readability). The likelihood values expressed in natural language (high, low, lowest) are automatically translated into numbers, using user-specified parameters. In our opinion, the fact that these expressions are

```
<zone name ="Dining_room">
  <prob >0.85</prob>
  <neighbor name="Fridge_sofa" />
  <neighbor name="Entrance_sofa" />
  <neighbor name="Meeting_room" />
  <sensor l_value="lowest">(and G (not B)
      )</sensor>
  <sensor l_value="high">A</sensor>
  <sensor l_value="low">(or E C)</sensor>
  <sensor l_value="high">D</sensor>
  <sensor l_value="high">B</sensor>
</zone>
<zone name ="Entrance_sofa">
  <prob >0.05</prob>
  <neighbor name="Dining_room" />
  <sensor l_value="high">(and E (not A))
      </sensor>
  <sensor l_value="low">D</sensor>
</zone>
```

Fig. 4. Part of the configuration file for our test environment

easy to write even for non programmers is very important, because, in real applications the system needs to be customized according to the deployment environment. A technique that does not require programming skills is thus necessary.

III. Results

The proposed approach and implementation software has been extensively tested in a real-life situation at a patient home. We recorded data during the normal daily activity of an elderly woman for two half days. Thus we had the opportunity to prove that our approach works well even in a real non-controlled situation. A sequence of screenshots of our localization system running during these testing sessions is depicted in Fig. 5. Notice how, when no motion is detected, the confidence on the patient position decreases (Fig. 5e and 5f), while it increases quickly when some motion is detected again (Fig. 5g).

Moreover, we evaluated the system also in an ad-hoc setup environment located at the 3rd floor of the Computer Science Department of Università degli Studi di Milano - Bicocca. We have chosen this area because it is very similar to a real home environment: it has a bathroom, a kitchen and a dining room with two sofas. It also sports a more office-like meeting room, half of which has been used as our "control room" while the other half was actually used in the experimental activities and is covered by sensor G. A map, including the displacement and the fields of view of the sensors is depicted in Fig. 3, while the different areas of interest are depicted in Fig. 1.

For both safety and practical reasons we were not able to place the sensors on the ceiling, although this would be the ideal location for the motion sensors. However, our experiments show that the proposed approach is able to achieve high accuracy even with non-ideal sensor displacement. In order to achieve sub-room accuracy in the *entrance_sofa* area, we

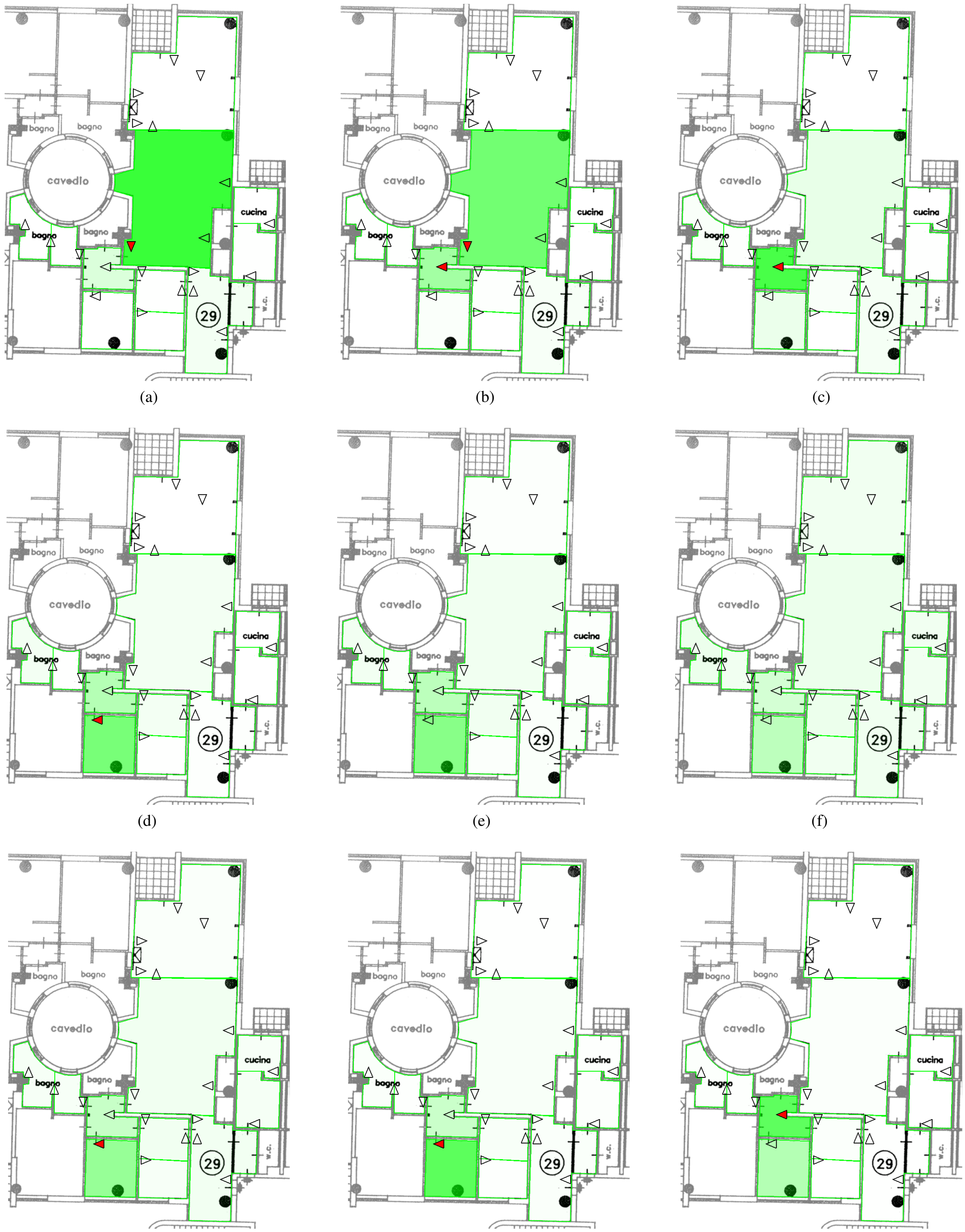


Fig. 5. A sequence of screenshots of our localization system during a testing session at a patient home. The areas where the patient is most likely located are darker. The triangles represents where the motion sensors were placed. Filled triangles are those representing sensors that detected some motion. When only a single sensor detects motion, the corresponding area assumes a high likelihood (Fig. 5a). It then decreases during transitions (Fig. 5b) and raises again when the person is entirely in a single area (Fig. 5c). When no motion is detected, the confidence on the patient position decreases (Fig. 5e and 5f), while it increases quickly when some motion is detected again (Fig. 5g).

Fig. 6. The "shield" we placed on the side of some sensors to reduce their field of view.

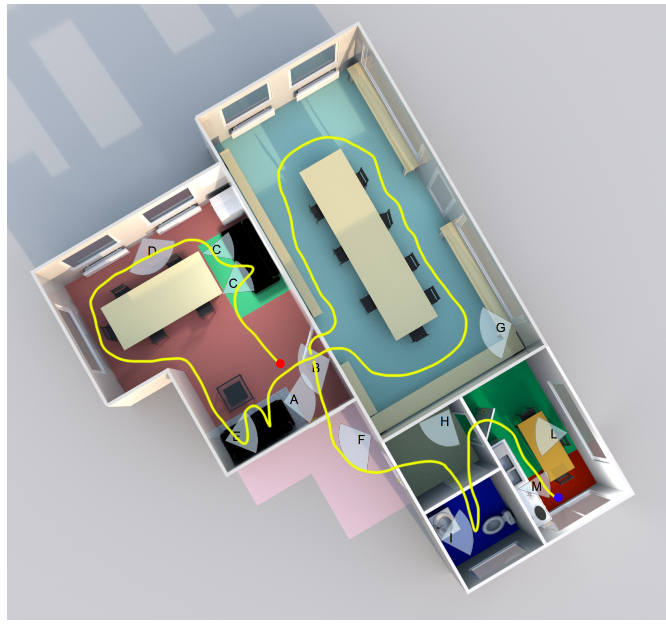


Fig. 7. The path followed during the data collection sessions. The red point is the starting position, the blue point the ending position.

decided to place two sensors: one on the left side of the sofa (sensor E), covering both the sofa and the entrance of the dining room, and one covering only the entrance of the dining room (sensor A). In this way we were able to supply our system with enough information to discriminate the motion in the area by exploiting when the sofa sensor was active while the entrance sensor was not. This escamotage was necessary because there was no way to place a single sensor able to discriminate on its own the motion in the *entrance_sofa* area. Nevertheless, this real-life problem gave us the opportunity to show the flexibility of our approach.

The motion sensors we used have a horizontal field of view of about 100°, but, to cover some areas, we needed a sensor with a narrower field of view. This problem could be solved using different sensors for different areas, however a much simpler and flexible solution is to limit the field of view of the sensor, e.g., by positioning a small shield (Fig. 6) on the side of the sensor. Other field of view limiting solutions could be devised, and easily used in our system, by acting on the boolean expressions representing the likelihood of each sensor reading on each state value.

A. The public dataset

Even though we looked for publicly available datasets for evaluating the performance of our proposal, we could not find anything suitable. Therefore, in order to report the performance of our algorithm we collected two sequences of situations of a person walking in the test environment. The path followed during the data collection is depicted in Fig. 7, starting from the entrance of the meeting room and ending in the kitchen. Along with data coming from the motion sensors, we collected a ground truth, in order to allow a quantitative evaluation of the accuracy of our approach. This ground truth provides information about when the person is moving from an area to another, as well as the name of the destination area.

The datasets are available to the community for further research and comparisons. For each sequence there are two files, both with .pkl extension: one containing the sensor data and one for the corresponding ground truth. The first is composed of a sequence of tuples saved with the *Pickle Python* module, [33]. The data from these files need to be loaded with the appropriate functions; the first element of each tuple is the timestamp of the measurement, the second is a list of sensors that detected some motion at that time. The ground truth file has a similar format, but the second element of the tuple is a string stating the area in which the person is actually entering.

During the data gathering sessions we had two major problems which, in our opinion, represent real-life situations: the sensor in the *fridge_sofa* area detected motion very frequently even when no person was there and the sensor in the atrium detected motion also when this was taking place in the dining room. These problems made the datasets very noisy. We did not suppress this noisy data in order to check one of the main strength of our approach: its ability to work even with realistically noisy sensor data. Along the lines presented in Section II-B, during our experiments we set the values of *prob_stay*, *prob_move* and *prob_jump* to 0.65, 0.34 and 0.01, respectively. These values have been obtained empirically and, at the moment, no automatic way to estimate them exists. Anyway the algorithm proved to be very robust against variations of these parameters, as long as the general rules described in Section II-B are respected. Moreover, we used these parameters' values also in an environment different from our testing area (the patient's home described in Section III), without the need to fine-tune them to the specific situation. On the two datasets, our algorithm performed the localization of the person with an error rate of 5% and 9%, respectively. The error rate has been calculated using Equation (5), where n_{err} indicates how many times the software returned a wrong localization and n_{it} is the total number of localizations returned. A localization is considered to be wrong if the person is in another area w.r.t. the one reported by the localization system. This is a common criterion among similar systems and has already been used in other works (see [19], [29]).

$$Error\ rate = \frac{n_{err}}{n_{it}} \times 100 \quad (5)$$

These error values may seem high at a first glance, however we need to consider that almost every error occurred while the person was moving from an area to another (see Fig. 8); it is quite hard to say with high accuracy where the boundaries of the areas are. If we remove the errors occurring during such area transitions, the error rate drops to 3.03% for the first dataset and 3.53% for the second. For the EvALL competition [29], a system is required to recognize when a person enters in a set of predefined areas of interest and stays there for 5 seconds; in that scenario the errors occurring during transitions would be irrelevant, so our system would perform very well.

Regarding the evaluation criteria described by Álvarez-García *et al*. [29], the accuracy has already been discussed. The availability was very high, in fact the software always returned a localization, as expected. The installation complexity is reasonably low, considering on one side that the algorithm is robust against the noise coming from various sources and against sensor misplacement and uncertainty on the sensors' field of view (as we have shown with our experimental activity); on the other side because the installation of the sensors is realistically feasible, e.g., not requiring mandatory mounting on the ceiling, providing reasonably simple to compile configuration files, even for non-trained personnel. Regarding the user acceptance, our system requires only the installation of fixed sensors, so the person can move freely, without carrying any device (this was one of the main goals of our research). The interoperability of our software with other systems is high. Since it is written in Python, it can be used on any operating system and it returns the localization both into a database and via message passing, using the RabbitMQ messaging system. Thus the information produced is easily available to other software modules. Moreover, while we used motion sensors only, the measurement function can be easily extended to take into account also other kind of sensors, such as CO_2 detectors or magnetic switches on the doors. While we purposely avoided using ZigBee and other wireless based techniques for localization because of the issues described in Section I, the information coming from these kind of systems can, nevertheless, be incorporated into our formulation of the measurement function, so to obtain a more robust localization. Of course this would imply, for the assisted person, the need to wear a device. Basically our approach, besides providing indoor localization using motion sensors, can be also seen as a generic framework where data coming from multiple and heterogeneous sensors can be used to provide a more accurate and robust localization.

Quantitative comparisons with other approaches of the same kind are, unfortunately, not possible, since there is no other public dataset available and since quantitative results are often not given. For this reason we decided to make our dataset public, so that it can be a common benchmark for future developments. It is available on our website at http://www.ira.disco.unimib.it/research/funded-research-projects/hcim/indoor-localization-dataset/

IV. Conclusions

In this paper we propose a novel approach to tackle the problem of indoor localization in assisted living environments. Our approach is based on a probabilistic filtering technique and is able to localize a single person within a home divided in macro-zones, using only fixed, easy-to-install motion sensors. Compared with other approaches, our system is able to achieve localization with sub-room accuracy without using any mobile device carried by the user. Because of this, we had to limit the field of application to environments where only a single person is present, however this is a common limit among approaches that use only environmental sensors. We used only passive infrared motion sensors, although the measurement function of the filter may be easily expanded to incorporate other kind of information. The performed experiments show that the system is robust against sensor noise and misplacement, a feature that makes it very easy to install in any home-like environment and suitable for real-world applications.

References

[1] K. G. Kinsella and D. R. Phillips, *Global aging: The challenge of success*. Population Reference Bureau Washington, DC, USA, 2005, vol. 60, no. 1.

[2] A. Lymberis and D. de Rossi, "Myheart: fighting cardiovascular disease by preventive lifestyle and early diagnosis," *Wearable eHealth Systems for Personalised Health Management: State of the Art and Future Challenges*, vol. 108, p. 36, 2004.

[3] P. Rubel, J. Fayn, G. Nollo, D. Assanelli, B. Li, L. Restier, S. Adami, S. Arod, H. Atoui, M. Ohlsson *et al.*, "Toward personal ehealth in cardiology. results from the epi-medics telemedicine project," *Journal of Electrocardiology*, vol. 38, no. 4, pp. 100–106, 2005.

[4] P. Rubel, F. Gouaux, J. Fayn, D. Assanelli, A. Cuce, L. Edenbrandt, and C. Malossi, "Towards intelligent and mobile systems for early detection and interpretation of cardiological syndromes," in *Computers in Cardiology 2001*. IEEE, 2001, pp. 193–196.

[5] H. C. Noel, D. C. Vogel, J. J. Erdos, D. Cornwall, and F. Levin, "Home telehealth reduces healthcare costs," *Telemedicine Journal & e-Health*, vol. 10, no. 2, pp. 170–183, 2004.

[6] S. M. Finkelstein, S. M. Speedie, and S. Potthoff, "Home telehealth improves clinical outcomes at lower cost for home healthcare," *Telemedicine Journal & e-Health*, vol. 12, no. 2, pp. 128–136, 2006.

[7] L. Graham, M. Zimmerman, D. Vassallo, V. Patterson, P. Swinfen, R. Swinfen, and R. Wootton, "Telemedicine—the way ahead for medicine in the developing world," *Tropical doctor*, vol. 33, no. 1, pp. 36–38, 2003.

[8] D. Vassallo, P. Swinfen, R. Swinfen, and R. Wootton, "Experience with system in three developing countries," *J. Telemed. Telecare*, vol. 7, no. 1, pp. 56–58, 2001.

[9] R. Wootton, "Telemedicine support for the developing world," *Journal of Telemedicine and Telecare*, vol. 14, no. 3, pp. 109–114, 2008.

[10] Y. M. Chae, J. Heon Lee, S. Hee Ho, H. Ja Kim, K. Hong Jun, and J. Uk Won, "Patient satisfaction with telemedicine in home health services for the elderly," *International journal of medical informatics*, vol. 61, no. 2, pp. 167–173, 2001.

[11] S. Bonhomme, E. Campo, D. Estève, and J. Guennec, "Prosafe-extended, a telemedicine platform to contribute to medical diagnosis," *journal of telemedicine and telecare*, vol. 14, no. 3, pp. 116–119, 2008.

[12] S. A. Mitilineos, D. M. Kyriazanos, O. E. Segou, J. N. Goufas, and S. C. Thomopoulos, "Indoor localisation with wireless sensor networks," *Progress In Electromagnetics Research*, vol. 109, pp. 441–474, 2010.

[13] S. Wagner, M. Handte, M. Zuniga, and P. J. Marrón, "Enhancing the performance of indoor localization using multiple steady tags," *Pervasive and Mobile Computing*, vol. 9, no. 3, pp. 392–405, 2013.

[14] N. Noury, G. Virone, and T. Creuzet, "The health integrated smart home information system (his 2): rules based system for the localization of a human," in *Microtechnologies in Medicine & Biology 2nd Annual International IEEE-EMB Special Topic Conference on*. IEEE, 2002, pp. 318–321.

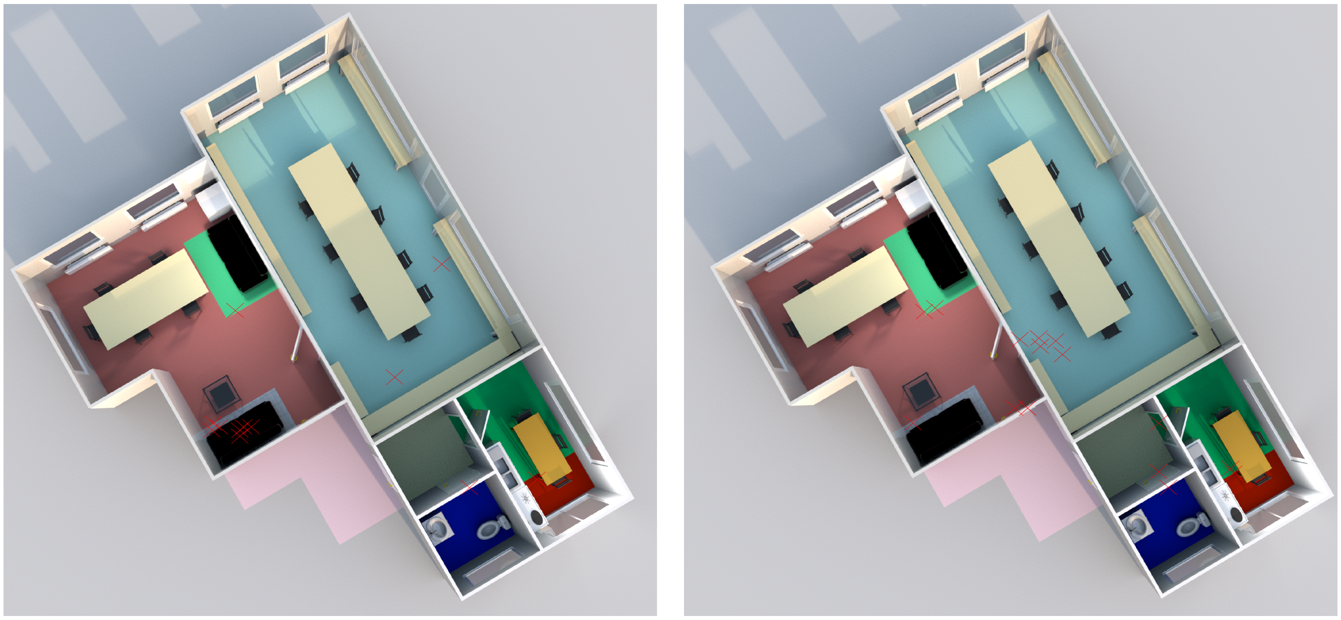


Fig. 8. The map of our test environment annotated with localization errors for the two datasets. Each cross indicates where an error occurred.

[15] N. Noury, T. Hervé, V. Rialle, G. Virone, E. Mercier, G. Morey, A. Moro, and T. Porcheron, "Monitoring behavior in home using a smart fall sensor and position sensors," in *Microtechnologies in Medicine and Biology, 1st Annual International, Conference On. 2000*. IEEE, 2000, pp. 607–610.

[16] S.-W. Lee, Y.-J. Kim, G.-S. Lee, B.-O. Cho, and N.-H. Lee, "A remote behavioral monitoring system for elders living alone," in *2007 International Conference on Control, Automation and Systems*, 2007, pp. 2725–2730.

[17] M. Ogawa, R. Suzuki, S. Otake, T. Izutsu, T. Iwaya, and T. Togawa, "Long-term remote behavioral monitoring of the elderly using sensors installed in domestic houses," in *Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint*, vol. 3. IEEE, 2002, pp. 1853–1854.

[18] Z. Han, R. Gao, and Z. Fan, "Occupancy and indoor environment quality sensing for smart buildings," in *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, May 2012, pp. 882–887.

[19] S. de Miguel-Bilbao, J. Roldán, J. García, F. López, P. García-Sagredo, and V. Ramos, "Comparative analysis of indoor location technologies for monitoring of elderly," *simulation*, vol. 1368, p. 08.

[20] A. G. Hauptmann, J. Gao, R. Yan, Y. Qi, J. Yang, and H. D. Wactlar, "Automated analysis of nursing home observations," *IEEE Pervasive Computing*, vol. 3, no. 2, pp. 15–21, 2004.

[21] R. Steele, A. Lo, C. Secombe, and Y. K. Wong, "Elderly persons' perception and acceptance of using wireless sensor networks to assist healthcare," *International journal of medical informatics*, vol. 78, no. 12, pp. 788–801, 2009.

[22] P. Djurić, M. Vemula, and M. Bugallo, "Target Tracking by Particle Filtering in Binary Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2229–2238, Jun. 2008.

[23] B. La Scala, M. Morelande, and C. Savage, "Robust Target Tracking with Unreliable Binary Proximity Sensors," in *2006 IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings*, vol. 4, May 2006, pp. IV–IV.

[24] G. Monaci and A. Pandharipande, "Indoor user zoning and tracking in passive infrared sensing systems," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, Aug. 2012, pp. 1089–1093.

[25] K. N. Ha, K. C. Lee, and S. Lee, "Development of PIR sensor based indoor location detection system for smart home," in *SICE-ICASE, 2006. International Joint Conference*, Oct. 2006, pp. 2162–2167.

[26] H. H. Kim, K. N. Ha, S. Lee, and K. C. Lee, "Resident Location-Recognition Algorithm Using a Bayesian Classifier in the PIR Sensor-Based Indoor Location-Aware System," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 39, no. 2, pp. 240–245, Mar. 2009.

[27] A. O'Brien, K. McDaid, J. Loane, J. Doyle, and B. O'Mullane, "Visualisation of movement of older adults within their homes based on PIR sensor data," in *2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*, May 2012, pp. 252–259.

[28] T. Yokoishi, J. Mitsugi, O. Nakamura, and J. Murai, "Room occupancy determination with particle filtering of networked pyroelectric infrared (PIR) sensor data," in *Sensors, 2012 IEEE*. IEEE, 2012, pp. 1–4. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6411114

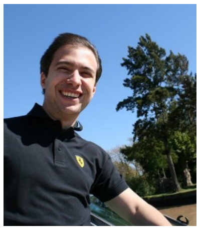
[29] J. A. Álvarez-García, P. Barsocchi, S. Chessa, and D. Salvi, "Evaluation of localization and activity recognition systems for ambient assisted living: The experience of the 2012 evaal competition," *Journal of Ambient Intelligence and Smart Environments*, vol. 5, no. 1, pp. 119–132, 2013.

[30] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.

[31] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and zigbee standards," *Computer communications*, vol. 30, no. 7, pp. 1655–1695, 2007.

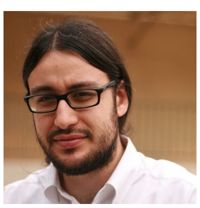
[32] "RabbitMQ — Messaging that just works," http://www.rabbitmq.com/, accessed: 20140-10-10.

[33] "pickle — Python object serialization," https://docs.python.org/2/library/pickle.html, accessed: 20140-10-10.



Augusto Luis Ballardini Augusto Luis Ballardini was born in Buenos Aires, Argentina in 1984. He is a PhD student in the Informatics and Robotics for Automation Lab, University of Milano - Bicocca, Italy. His current research focuses on autonomous vehicles localization, leveraging heterogeneous data sources. He is currently a visiting student in the University of Alcalá, Alcalá de Henares, Spain. Augusto earned his BS and MS in Computer Science from the University of Milano - Bicocca. His interests include localization algorithms, laser scanning technologies and computer vision for autonomous driving vehicles.

Lorenzo Ferretti Lorenzo Ferretti was born in Milan, Italy in 1989. He received the Bachelor Degree in Computer Science from the University of Milano - Bicocca, Italy in 2014. After the Bachelor he has collaborated with the IRALab at the University of Milano - Bicocca as research assistant. He is currently completing the Master of Science in Informatics at USI (Università della Svizzera Italiana) in Switzerland.



Simone Fontana Simone Fontana was born in Rome, Italy in 1987. He is a PhD student in Computer Science at the Informatics and Robotics for Automation Laboratory of University of Milano - Bicocca, Italy. Altought he worked in the field of telehomecare, his current research topic is multi-robot mapping.

Axel Furlan Axel Furlan was born in Trieste, Italy in 1983. He received the Bachelor Degree in Computer Science from the Univeristy of Trieste, Italy, in 2005, and the Master Degree and Ph.D. in Computer Science from the University of Milano - Bicocca, Italy, in 2009 and 2014, respectively. During his Ph.D. he has been a visiting scholar at the University of Michigan and at the Stanford University. From 2014 to the mid 2015 he has been a Postdoc at the IRALab at the University of Milano - Bicocca. He is currently workin as Data Fusion Engineer at the ADAS-CTO Group of the Magneti Marelli Company.

Domenico Giorgio Sorrenti Domenico G. Sorrenti was born in Milano Italy in 1962, obtained a maturità classica diploma (high school, with literature and philosophy orientation) at Liceo G. Carducci, Milano, in July 1981. He studied Electronic Engineering at Politecnico di Milano, and obtained the Laurea (Master) in February 1989 with a thesis on a camera - laser 3D vision system for mobile robots. He then enrolled in the Ph.D. programme in Computer and Control Engineering at Politecnico di Milano and obtained the Ph.D. degree in December 1992 with a thesis ”Delaunay triangulation for modeling the environment of autonomous robots”. Since 1995 he has been assistant professor in computer science with Università di Milano, in 1999 moved to Università degli Studi di Milano - Bicocca where, from 2005, he is associate professor in computer engineering. He currently teaches courses on Computer Architecture, Machine Perception, and Robotics for the laurea (Bachelor), laurea magistrale (Master) and the Ph.D. programmes in Computer Science. His main research interests are in robotic perception for autonomous vehicles. His research has been funded by EEC, National agencies (mainly MIUR), Lombardy region, and private companies. He is a member of the IEEE and of IAPR.