Person Tracking in Urban Scenarios by Robots Cooperating with Ubiquitous Sensors

Luis Merino Jesús Capitán Aníbal Ollero

Abstract— **The introduction of robots in urban environments opens a wide range of new potential applications for service robotics. One of these applications is people guidance. To accomplish this task, the robot needs information about the position of the person. Sensors embedded in the urban environment can complement the perception of the robot in this case. The paper shows how the combination of the robot sensorial information with that from a camera network and with a wireless sensor network, is very useful to cope with tracking failures by being more robust under occlusion, clutter and lighting changes. The paper summarizes the main characteristics of the algorithms for tracking with the fixed surveillance cameras and cameras on board robotic systems. It also presents results on position tracking by using the strength of the radio signal from the nodes of Wireless Sensor Network (WSN). The estimates from all these sources are then combined using a decentralized data fusion algorithm to provide an increase in performance. This scheme is scalable, can cope with communication latencies and degrades smoothly with communication failures. We present results of the system, operating in real time, in a large outdoor environment, including 22 non-overlapping cameras, 30 wireless sensor nodes and one mobile robot.**

I. INTRODUCTION

Many major cities in Europe are looking for means of reducing the traffic in certain areas, in order to mitigate air and noise pollution, traffic jams and, in general, to improve the quality of life. It is intended to develop automatic systems to perform, in free car areas, services such as person guiding, people and objects transportation, surveillance, etc. The EU Project called URUS (**U**biquitous Networking **R**obotics in **U**rban **S**ettings) [1] considered a team of mobile robots, a set of static cameras and a Wireless Sensor Network (WSN) for these tasks.

In particular, the application of person guidance requires the ability to determine and track the position of the person to be guided. This application requires the collaboration of different systems, as, in many cases, a single autonomous entity (i.e. a robot or a static surveillance camera) is not able to acquire all the information required because of the characteristic of the task or the harmful conditions (i.e. loss of visibility). The set of fixed cameras can obtain global views of the scene. However, as they are static, they cannot deal with non-covered zones, shadows can affect the system and so forth. Robots carry local cameras and can move to adequate places, reacting to the changing conditions. However, their field of view is limited and they can lose the person they are tracking. Wireless devices can also help to localize the people, by estimating their positions measuring the signal strength from different static receivers. However, the resolution obtained is usually low, and depends on the density of anchored receivers. In this paper we show how the information from the above different systems can be fused to improve the performance. In order to cope with scalability, a decentralized data fusion algorithm is employed. In this algorithm only local estimation and local communication are used.

Then, this paper focuses on the perception system of URUS and the experimental results obtained (see [1] for a more general description of the project). Next section will review related work. After an overview of the full system in Section II, the paper will present the individual input sensor algorithms. Thus, Sections III and IV summarize the process to extract information from a set of fixed cameras and from cameras on board robots. Section V explains the use of the signal strength from wireless sensors for tracking. Finally, the results of the tracking from all sensors are used to infer the position of the person in a global coordinate system through a data fusion process. This system is described in Section VI. The paper ends showing results obtained during the experiments of the URUS project, in an urban scenario involving 22 fixed cameras, a WSN of 30 wireless Mica2 nodes, and a mobile robot.

A. Related Work

There has been an increasing amount of research on person tracking in the literature. Most works describe a single system or algorithm for person tracking using vision, laser range-finders or other sensors.

Thus, there has been many attempts to track people and other moving objects using networks of fixed cameras. The early tracking algorithms [2], [3] require both camera calibration and overlapping fields of view to compute the handover of objects of interest between cameras. Others [4] can work with non-overlapping cameras but still require calibration. More recent works [5], [6] do not require *a priori* calibration to be explicitly stated; instead they use the observed motion over time to establish reappearance periods between cameras.

This work is partially supported by the FROG Project (ICT-288235) and the URUS project (IST-2006-045062) funded by the European Commission, and the project RURBAN, funded by the Andalusian Government (P09-TIC-5121). Jesús Capitán is also funded by Fundação para a Ciência e a Tecnologia (ISR/IST pluriannual funding) through the PIDDAC Program funds and by project CMU-PT/SIA/0023/2009 (Carnegie Mellon-Portugal Program).

Luis Merino is with Pablo de Olavide University, Seville, Spain lmercab@upo.es.

Jesús Capitán is with Instituto Superior Tecnico, Lisbon, Portugal jescap@isr.ist.utl.pt.

Aníbal Ollero are with University of Seville, Seville, Spain [aollero]@cartuja.us.es.

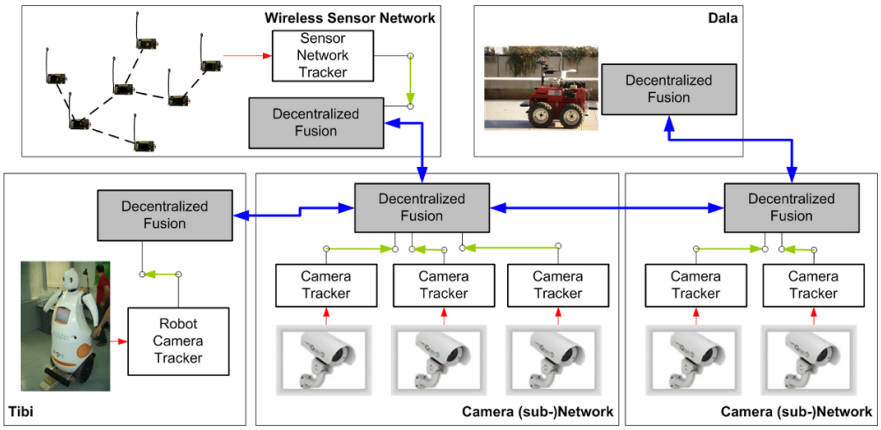


Fig. 1: A block description of the URUS perception system. The different subsystems are integrated in a decentralized manner through a set of decentralized data fusion nodes. Locally, each system can process and integrate its data in a central way (like the WSN) or in a distributed way (like the camera network). Some systems could obtain information from the rest of the network even in the case they do not have local sensors.

Tracking from mobile platforms like robots in outdoor scenarios is a hard problem affected by clutter, illumination changes in the case of vision approaches, occlusions, etc. Most of the techniques combine people detection and people tracking modules for the task. The people detection module tries to obtain person hypotheses analyzing the sensor data, and is usually computationally demanding. Many classification techniques are used for this task, like boosting [7], SVM [8], etc. The tracking module is usually a feature tracking algorithm applied to the initial hypothesis given by the detection module, which can run at higher pace than the detection algorithm, like CamShift [9]. In most cases, both modules support each other, so when the tracker is lost new hypotheses from the detector can be used. More complex combinations, including what is called cognitive feedback are also considered [10].

There is also work devoted to the tracking of mobile nodes by using radio signals, which is the problem of estimating the position of a mobile node from the signal received by a set of static devices whose positions are known. A tutorial on the main issues and approaches for the problem is presented in [11]. Many algorithms use, beside signal strength, additional information to obtain range estimates or even direction of arrival estimates. For instance, [12] considers the use of particle filters for tracking a mobile node using Time of Arrival, Difference of Time of Arrival and power measurements, presenting results in simulation. The work [13] uses the Doppler shift of interference signals to estimate the velocity and position of mobile nodes. These approaches require the precise synchronization of the emission of signals. In our approach, only signal strength is used, through a calibrated model for radio propagation. There are approaches in which this model is learnt; [14] presents an approach in which Gaussian Processes are used as non-parametric models for the errors in indoor signal propagation.

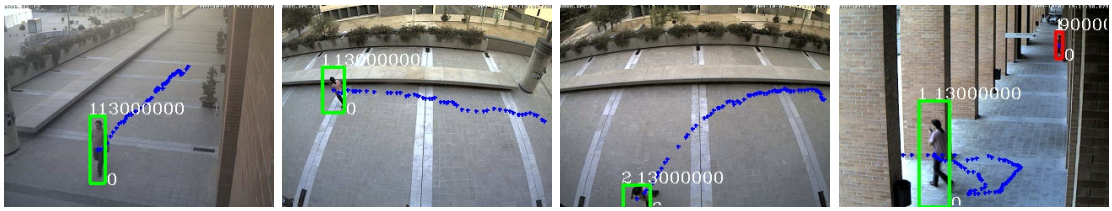
The key issue in the paper is to show how the combination of the local information obtained by the robot with the information received from ubiquitous sensors in the environment can improve greatly the results. Moreover, a decentralized data fusion approach is employed, producing an scalable solution with respect to the number of subsystems.

II. URUS System Overview

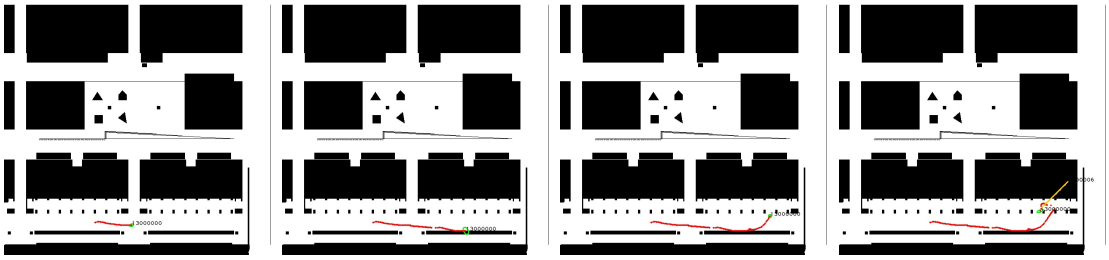
The URUS system consists of a team of mobile robots, equipped with cameras, laser range-finders and other sensors for localization, navigation and perception; a fixed camera network of more than 20 cameras for environment perception; and a WSN of 30 Mica2 nodes that uses the signal strength of the received messages from a mobile device to determine the position of a person carrying it.

Figure 1 shows a simplified version of the perception system used in URUS. The system consists of a set of fusion nodes which implement a decentralized data fusion algorithm. Each node only employs local information (data from local sensors; for instance, a subset of cameras, or the sensors on board the robot) to obtain a local estimation of the variables of interest (in this case, the position of the person being tracked). Then, these nodes share their local estimations among themselves if they are within communication range. The nodes only use local communications and data, and then the system is scalable. Also, each node can accumulate information from its local sensors, so temporal communication failures can be tolerated without losing information.

Notice that the way in which a particular fusion node processes its local data can have a distributed or even centralized implementation itself. For the camera network, each fusion node considers information from a small subset of cameras, which are processed in a distributed way (as it will be described in Section III), with a separate tracker obtaining estimations from each camera. For the case of the WSN, messages from all the network are processed in a gateway to localize the mobile node using the signal strength (Section V). Moreover, each robot locally processes its data (on-board cameras). Then, the local estimations of



(a)



(b)

Fig. 2: (a) Tracks on the image plane of 4 different cameras. The identity is correctly handed over the cameras using the weak cues described in Section III. (b) Estimated position of the person on the experimental site.

the different elements are fused in a decentralized way using the algorithm presented in Section VI.

Thus, the system is easily scalable: for instance, a new set of cameras could be included by adding a new fusion node in charge of these cameras (and maybe a new server to process the information of these cameras); even robots without local sensors (receiving information from the rest of the nodes) could be added to the system.

III. Fixed Camera Tracking

The network of fixed cameras covers a wide area of the experiment site and therefore, in most cases, they initiate the person guidance; they are able to track objects of interest both on and across different cameras without explicit calibration periods.

Within this paper, the fixed camera tracking algorithm of Gilbert and Bowden [15], [16] is used. A local tracker processes the data from each camera. Background modelling and subtraction is used to identify foreground objects, and Kalman filtering is used to provide temporal correspondence between detected objects.

Very interestingly, the trackers are able to learn inter-camera relationships for inter-camera object handling, even without camera calibration or overlapping. By using weak cues, the system is able to incrementally build probability distributions on the possibility that a person leaving one camera enters a different camera some time interval after. This information, combined with color histograms, is used for inter-camera tracking. More details can be found in [15].

Although not used by the system to estimate the inter-camera relationships, the cameras are homography-calibrated, so it is possible to obtain a 2D estimation on the position of the people tracked in the map of the URUS scenario. Figure 2 shows an example, in which a person is tracked using 4 different cameras with little or no overlap at all (and its identity maintained) using the techniques described in [15]. The figure shows the estimated position of the person using only information from the camera network.

IV. Robot Camera Tracking

The robots carry on-board cameras that are used for person guiding. This cameras can be used to obtain local estimations on the position of the person to be guided. The algorithms employed for this are based on a combination of state-of-the-art algorithms for person detection and tracking.

The person detection algorithm applied to the image is the one in [17]. This detection module is launched when the robot is requested to guide a person and it is close to the location where the person is waiting. Once the person is detected, it is tracked by using a tracking algorithm which is based on the CamShift technique [9]. While the algorithm is able to handle temporal occlusions, the tracking system is not enough to maintain the track on the person continuously due to changes in illumination, the changing field of view of the camera due to the robot motion, or even the person going out of the field of view. Therefore, the results from the tracking and the detection applications are combined, so that the robot employs the person detector whenever the tracker is lost to recover the track. The algorithm determines that the person is lost employing some heuristics, like the track going out to the limits of the image or size restrictions on the blob. As a result, the robots can obtain estimations of the pose of the person on the image plane.

Some improvements can be applied to the features in order to cope with illuminations changes [18]. However, in general, these algorithms are not robust enough to be able to guide one person through the whole scenario. Furthermore, they can track the wrong people sometimes. Moreover, from information from one camera alone it is not possible to estimate the full 3D position of the person. Next sections will

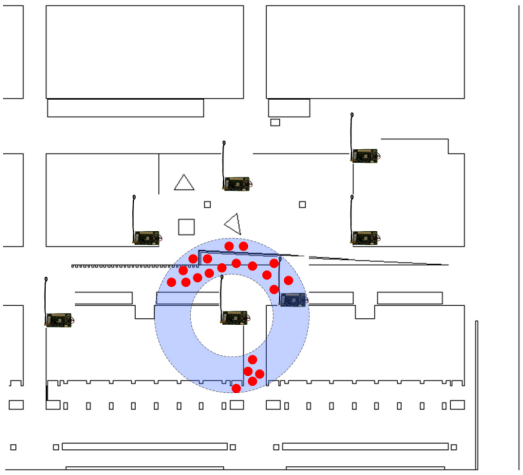


Fig. 3: Particles (red) are used to represent person hypotheses. The signal received by a set of static nodes can be used to infer the position of the node. The filter is initiated when the first message is received by sampling uniformly from a spherical annulus around the receiver. Map information is also taken into account (only free spaces within the annulus are considered).

show how the combination of the local camera information and the information from the other subsystems (camera network and WSN) can overcome these problems.

V. Wireless Sensor Network Tracking

A network of wireless Mica2 sensor nodes is also considered. The signal strength received by the set of static nodes (Received Signal Strength Indicator, RSSI) can be used to infer the position of a person carrying one of the nodes (the emitter). The algorithm to estimate and track the node position is based on particle filtering. In the particle filter, the current belief about the position of the mobile node is defined by a set of particles $\{\mathbf{x}_t^{(i)}\}$, which represent hypotheses about the current position of the person that carries the node (see Figure 3).

In each iteration of the filter, kinematic models of the motion of the person and map information are used to predict the future position of the particles. The likelihood of these particles is updated any time new messages are received from the static network. The technique is summarized in Algorithm 1, where $\mathbf{z}_t^j$ is the measurement provided by each static node j, consisting of its position $\mathbf{x}^j$ and the strength $RSSI_t^j$ of the received signal from the mobile node. Next subsections further describe the main steps in this algorithm.

A. Prior, prediction and importance functions

The filter is initialized with the first message received from the mobile node, considering an uniform distribution on a spherical annulus around the receiver. The map of the scenario is taken into account when sampling from this prior (see Figure 3), considering that the person is not inside any building.

Algorithm 1 $\{\mathbf{x}_t^{(i)}, \omega_t^{(i)}; i = 1, \ldots, L\} \leftarrow$ Particle_filter($\{\mathbf{x}_{t-1}^{(i)} \omega_{t-1}^{(i)}; i = 1, \ldots, L\}, \mathbf{z}_t^j = \{\mathbf{x}^j, RSSI_t^j\}$)

1: **for** $i = 1$ to L **do**
2: $\quad \mathbf{x}_t^{(i)} \leftarrow$ sample_kinematic_model $(\mathbf{x}_{t-1}^{(i)})$
3: **end for**
4: **if** Message from network $\mathbf{z}_t^j$ **then**
5: $\quad$ **for** $i = 1$ to L **do**
6: $\quad\quad$ Compute $d_t^{(i)} = \|\mathbf{x}_t^{(i)} - \mathbf{x}^j\|$
7: $\quad\quad$ Determine $\mu(d_t^{(i)})$ and $\sigma(d_t^{(i)})$
8: $\quad\quad$ Update weight $\omega_t^{(i)} = p(RSSI_t^j|\mathbf{x}_t^{(i)})\omega_{t-1}^{(i)}$ with $p(RSSI_t^j|\mathbf{x}_t^{(i)}) = \mathcal{N}(\mu(d_t^{(i)}), \sigma(d_t^{(i)}))$
9: $\quad$ **end for**
10: **end if**
11: Normalize weights $\{\omega_t^{(i)}\}, i = 1, \ldots, L$
12: Compute $N_{eff} = \dfrac{1}{\sum_{i=1}^{L} (\omega_t^{(i)})^2}$
13: **if** $N_{eff} < N_{th}$ **then**
14: $\quad$ Resample with replacement L particles from $\{\mathbf{x}_t^{(i)}, \omega_t^{(i)}; i = 1, \ldots, L\}$, according to the weights $\omega_t^{(i)}$
15: **end if**

Each time step, the position of the particles are predicted from their previous position (Line 2 of Algorithm 1). The prediction function uses a Brownian motion model [19]. This model is combined with map information to discard unfeasible motions (like going through walls); particles arriving at occupied places are rejected and substituted by new sampled particles. Other prediction models could be used as well.

B. The likelihood function

The likelihood function $p(RSSI_t|\mathbf{x}_t)$ plays a very important role in the estimation process, since each time a message is received this likelihood is used to update the particles weights (Lines 5 to 9). The likelihood models the correlation that exists between the distance that separate two nodes and the $RSSI$ value, although this correlation decreases with the distance between the two nodes, transmitter and receiver [20]. This is mainly caused by radio-frequency effects such as radio reflection, multi-path or antenna polarization.

The model used here considers that the conditional density $p(RSSI_t^j|\mathbf{x}_t)$ can be approximated as a Gaussian distribution for a given distance $d_t^j = \|\mathbf{x}_t - \mathbf{x}^j\|$ between the mobile node and static node j, as follows:

$$RSSI_t^j = \mu(d_t^j) + \mathcal{N}(0, \sigma(d_t^j)) \quad (1)$$

where the functions $\mu(d_t^j)$ and $\sigma(d_t^j)$ are non-linear functions of the distance (which itself is a non-linear function of the state). These functions are estimated during a calibration procedure (the form of the functions and the calibration procedure are described in [20]).

C. Filter evolution

Although Section VII will show additional results, Figure 4 presents the evolution of the particles for a particular

tracking experiment performed at the experimental site. 500 particles are employed, and the algorithm runs at more than 1 Hz. When the filter converges to a Gaussian distribution, the estimated mean and covariance can be fed to the decentralized fusion system that will be explained in the next section.

VI. Decentralized Data Fusion for People Guidance

Using the trackers described above, the camera network, the robots and the WSN will be able to obtain local estimations of the position of the people on the image plane or in a 3D coordinate system. That information provided by each tracker, characterized as Gaussian distributions (mean and covariance matrix), can be fused in order to obtain a more accurate estimation of the 3D position of the person.

As commented in Section II, the idea is to implement a decentralized fusion approach, in which each node only employs local information (data only from local sensors, for instance, a camera subnet, or the sensors on board the robot), and then *shares* its estimation with other nodes (see Figure 1). Thus, scalability and robustness are improved and bandwidth requirements alleviated. This fusion algorithm is based on an Information Filter and is described in [21], [22]. Here, the main concepts are summarized.

A. Delayed-State Information Filter

The Information Filter (IF), which corresponds to the dual implementation of the Kalman Filter (KF), is a suitable approach for decentralized state estimation. Whereas the KF represents a Gaussian distribution on the state $\mathbf{x}_t$ using its first $\boldsymbol{\mu}_t$ and second $\boldsymbol{\Sigma}_t$ order moments, the IF employs the so-called *canonical representation*. The fundamental elements are the *information vector* $\boldsymbol{\xi}_t = \boldsymbol{\Sigma}_t^{-1}\boldsymbol{\mu}_t$ and the *information matrix* $\boldsymbol{\Omega}_t = \boldsymbol{\Sigma}_t^{-1}$. Prediction and updating equations for the (standard) IF can also be derived from the standard KF [21]. In the case of non-linear prediction or measurement models, first order linearisation leads to the Extended Information Filter (EIF).

A Delayed-State Information Filter maintains not just the last state, but a belief over the full trajectory of the state up to the current time step t, denoted by $\boldsymbol{\Omega}^t$ and $\boldsymbol{\xi}^t$.

B. Decentralized Information Filter

The main interest of the IF is that it can be easily decentralized. In a decentralized approach, each urban robot or ubiquitous entity represents a node i within the network, which employs only its local data $\mathbf{z}_t^i$ to obtain a local estimation of the person trajectory (given by $\boldsymbol{\xi}^{i,t}$ and $\boldsymbol{\Omega}^{i,t}$) and then *shares* its belief with its neighbours. Therefore, each node i will run a Delayed-State EIF using only its local information, and will fuse locally the received information $\boldsymbol{\xi}^{j,t}$ and $\boldsymbol{\Omega}^{j,t}$ from another node j in order to improve the local perception of the world. Ideally, the decentralized fusion rule should produce the same result locally as that obtained by a central node employing a centralized filter. In [21] the authors propose the next fusion rule:

$$\boldsymbol{\Omega}^{i,t} \leftarrow \boldsymbol{\Omega}^{i,t} + \boldsymbol{\Omega}^{j,t} - \boldsymbol{\Omega}^{ij,t} \quad (2)$$

$$\boldsymbol{\xi}^{i,t} \leftarrow \boldsymbol{\xi}^{i,t} + \boldsymbol{\xi}^{j,t} - \boldsymbol{\xi}^{ij,t} \quad (3)$$

The above equations mean that each node should sum up the information received from other nodes. The additional terms $\boldsymbol{\Omega}^{ij,t}$ and $\boldsymbol{\xi}^{ij,t}$ represent the common information between the nodes. This common information is due to previous communications between nodes, and should be removed to avoid double-counting of information (known as rumour propagation [23]). As long as a tree-shaped logical topology in the perception system (no cycles or duplicated paths of information) is assumed, this common information can be maintained by a separated EIF so-called channel filter [24].

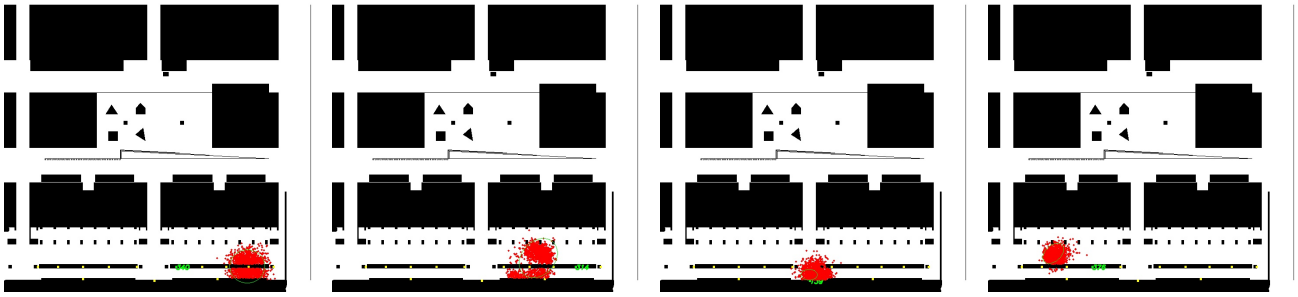
It is important to remark that, using these fusion equations *and considering trajectories (delayed states)*, the local filter can obtain an estimation that is equal to that obtained by a centralized system [21] (provided that enough time has passed to allow the information to flow through the different network nodes). Another advantage of using delayed states is that the belief states can be fused asynchronously without missing information. Each sensor can accumulate evidence, and send it whenever it is possible. Also, asequent and delayed measurements can be incorporated in the filter.

However, as the state grows over time, the size of the message needed to communicate its belief also does. For the normal operation of the system, only the state trajectory over a time interval is needed, so these belief trajectories can be bounded by marginalizing out old states. Note that the trajectories should be longer than the maximum expected delay in the network in order not to miss any measurements information.

Finally, when no assumptions about the network topology can be made (e.g. due to the existence of mobile objects, possible losses of communication links, etc), another option to remove the common information is to employ a conservative fusion rule, which ensures that the system does not become overconfident even in presence of duplicated information at the cost of losing optimality in the fusion. For the case of the IF, there is an analytic solution for this, given by the Covariance Intersection algorithm of [25].

C. Data association

Each fusion node of the system should be able to associate its local observations with the current tracks. In the case of the camera network, this is done by combining the inter-camera information and geometric information. As commented in Section III, the system is able to handle inter-camera tracking without calibration, using as weak cues reappearance probabilities and color information. Therefore, the system uses this information for data association. As this scheme may fail, the non-associated observations are also passed through a data association procedure based on the Mahalanobis distance, using the estimated global person position obtained using the homographies.



(a) (b) (c) (d)

Fig. 4: A sequence of the 500 particles employed in the filter for this experiment. Red points represent the particles. Yellow points represent the static nodes, being the green one the emitter at each frame. A person carrying the mobile node travels from right to left in the corridor at the bottom.

The data association in the case of the WSN node is straightforward, as the messages from the WSN are tagged with an ID.

The image tracker in the case of the robots maintains the identity of the tracked people while they are on the image plane. The Mahalanobis distance is also used to associate new measurements with previous tracks. Moreover, the decentralized nodes should be able to associate the received tracks with the local tracks. For this track-to-track fusion, the Mahalanobis distance is used again.

VII. Experimental Results

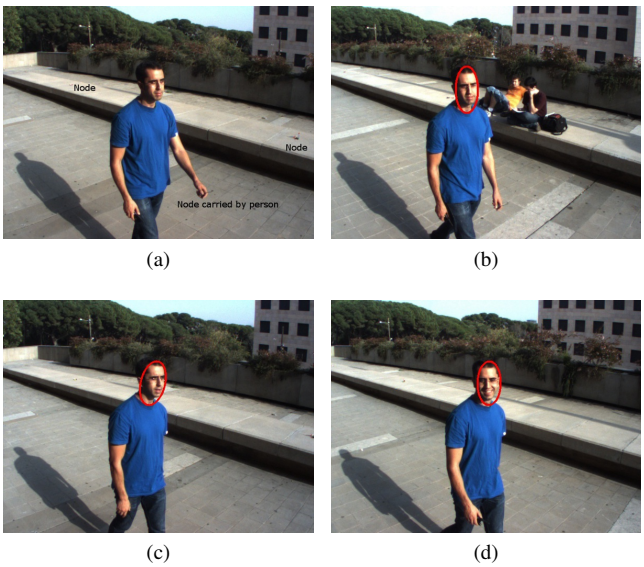
The techniques described above were tested during the experimental sessions of the URUS EU Project. These experiments were carried out at the Barcelona Robot Lab, which is an outdoor urban experimental robotics site located at the UPC (Universidad Politécnica de Cataluña) campus Nord. In order to build the system of Section II, 22 fixed color video cameras were installed and connected through a Gigabit Ethernet connection to a computer rack, as well as wireless sensor nodes for localization purposes and 9 WLAN antennas with complete area coverage.

URUS proposed people guidance as one of the possible applications for the above urban scenario. First, by means of a mobile phone, a person calls for a robot in order to receive the service. Then, the closest available robot with this functionality approaches and identifies the person, and guides him/her to the requested final destination. In all this process, the decentralized data fusion between the ubiquitous sensors is essential in order to help the robot with the guidance task.

A. Robot and WSN

In order to illustrate the benefits from the data fusion process, a first setup is presented here. This setup considers information from one camera on board the robot Romeo (4-wheel vehicle, see Figure 7) and the WSN (30 nodes). The objective was to track the position of a person cooperatively while the robot was guiding.

In this case, just two nodes of the decentralized fusion scheme were used: one on board the robot and one for the WSN. These nodes locally integrated information from a monocular camera (see Figure 5) and from the signal strength-based estimations (Section V, see Figure 5a), respectively.

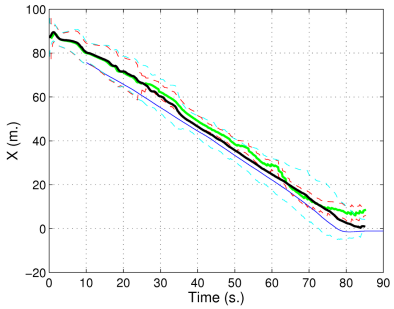


(a) (b) (c) (d)

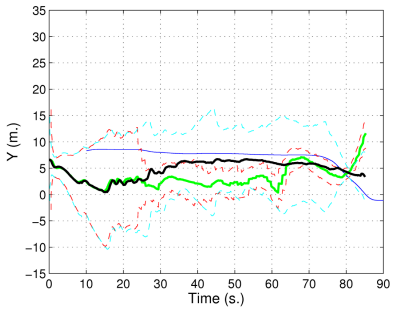
Fig. 5: (a) The person was carrying a Mica2 node during the experiment. (b,c,d) The robot was able to obtain local observations on the image plane of the face of the person.

Figure 6 shows the X and Y estimations obtained by the robot alone and when the robot combines its information with the one provided by the WSN. In this case, as ground truth we have the trajectory of the robot measured by its navigation software. The person is following behind the robot (see Figure 7) (which in this trajectory means that the X coordinates of the person are larger than that of the robot) and some meters beside the robot (a lower Y coordinate).

It can be seen how the introduction of the WSN reduces the uncertainty; as we have a monocular camera, the uncertainty on the person position is quite big in both axes when the robot is alone. In this case, the initial position of the



(a)



(b)

Fig. 6: Tracking using one on-board camera and the WSN. Black: robot alone. Green: robot and WSN. Dashed lines are the sigma intervals and the blue solid line represents the robot trajectory.

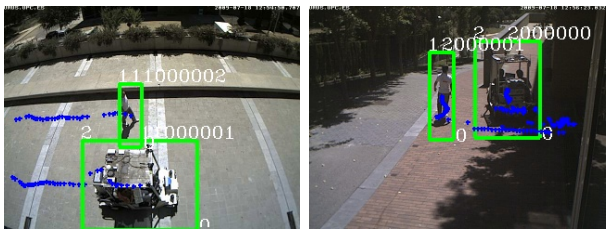


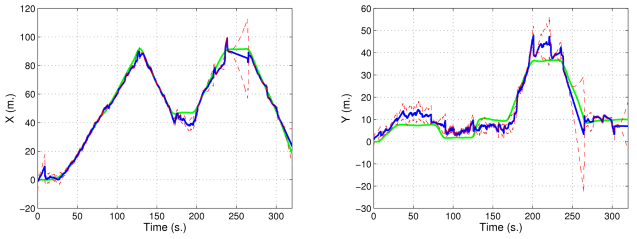
Fig. 7: Tracks obtained by the camera network.

person is computed assuming a known height of the face. In the second case, the 3D estimation of the WSN is used to initiate the filter.

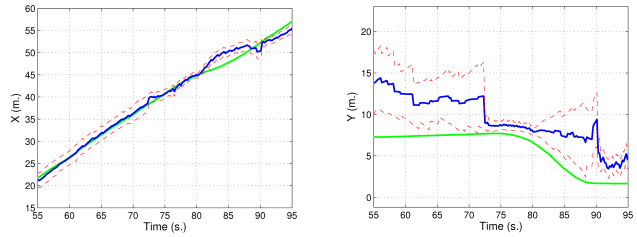
B. Robot, WSN and camera network

In this setup an experiment on a larger area is shown. This time one robot, the WSN and 7 fixed cameras were used. Again, there was a person following the robot whose position had to be estimated. The setup of the perception system was one decentralized fusion node on the robot, one for the data from the WSN and 2 fusion nodes for the fixed cameras, one integrating measurements from 3 camera trackers and the another from 4 cameras.

Figure 7 shows some examples of the tracks obtained by the camera network. Along the trajectory there were gaps in the camera coverage. Moreover, the The robot lost at times the object is following due to the changes in illumination, etc.



(a)



(b)

Fig. 8: Estimated position of the person (blue) compared to the position of the robot (green). Dashed lines represent the standard deviation of the estimation. (a) Complete trajectory. (b) An interval of the trajectory. The person was following the robot with the same X coordinate up to time 80 seconds. Then the robot changed orientation. The person was separated from the robot around 3-4 meters.

Figure 8a shows the estimated position of the person with the full system running. The total length of the experiment was around 350 meters and 5 minutes. The person was usually besides the robot (which means that the X or Y coordinates are the same). The system was able to maintain the estimation of the person position for the full trajectory. There was WSN coverage between 0 and 150 seconds, approximately. Figure 8b shows an interval of the trajectory. In this part, only WSN and robot information were available. Although the WSN measurements have low accuracy, they allow the system to bound the error from the robot monocular camera. At time 75 approximately, the person entered under coverage of the camera network, which led to a big reduction in uncertainty.

During all the above experiments, the communication between the fusion nodes on board the robot and the fusion nodes related to the camera network and the WSN was done using WiFi and 3G. A software running on the robot was able to measure the quality of the WiFi links, and to switch to 3G whenever this quality dropped below a certain threshold. The switching between communication networks created from time to time communication breakdowns of several seconds. Moreover, although 3G had a more stable coverage in the scenario, it had also lower bandwidth and higher latencies than WiFi. In order to tackle these problems, it was crucial the use of a decentralized system with delayed states, as in the meantime, the local nodes were accumulating information. When the communication links were recovered, the nodes exchanged their estimations. Moreover, as delayed

states were considered, this delayed information (and also information delayed due to the latencies) could be fused in a correct way, and no information was lost.

VIII. Conclusions

In urban scenarios, the cooperation between mobile robots and ubiquitous sensors can provide solutions to problems in which single, even if powerful, systems can fail. Very complex algorithms employing just one source of information are usually unable to cope with all the potential situations in these scenarios, affected by changes in illumination, clutter, and in which a wide area must be covered. The combination of complementary systems can be useful for this problem.

This paper has presented a system that aims to use multiple sensors to accurately track people within a guidance application. The system uses extensively data fusion procedures to incorporate all the information available. Scalability is an issue in these systems, and thus decentralized algorithms are required. The system presented is a mixture between distributed or centralized subsystems that are linked through a decentralized data fusion scheme. The addition of new robots or sub-nets of cameras does not affect the rest of the perception system in terms of storage, as only local communication and local processing is used. The algorithms are real-time and have been tested in the urban scenario proposed by the URUS Project, consisting of a camera network with 22 cameras, a WSN with 30 nodes and mobile robots.

Future developments include the integration of active sensing behaviors in the system. The WSN can be actively controlled to save energy, activating those nodes more useful for tracking. Next steps also include closing the loop, and developing more complex robot navigation algorithms for social people guiding by robots. This will be the focus of the FROG European project: besides positioning information, information like human commitment will be extracted and used to develop robot motions that are socially acceptable .

IX. Acknowledgments

The authors would like to thank all the partners in the URUS project for their support during the different experiments.

References

[1] A. Sanfeliu, J. Andrade-Cetto, M. Barbosa, R. Bowden, J. Capitan, A. Corominas, A. Gilbert, J. Illingworth, L. Merino, J. Mirats, P. Moreno, A. Ollero, J. Sequeira, and M. Spaan, "Decentralized Sensor Fusion for Ubiquitous Networking Robotics in Urban Areas," *Sensors*, vol. 10, pp. 2274–2314, 2010.

[2] T. Chang, S. Gong, and E. Ong, "Tracking Multiple People under Occlusion using Multiple Cameras," *In Proc. of BMVA British Machine Vision Conference (BMVC'00)*, pp. 566–575, 2000.

[3] V. Morariu and O. Camps, "Modeling Correspondences for Multi-Camera Tracking using Nonlinear Manifold Learning and Target Dynamics," *In Proc. of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. I, pp. 545–552, 2006.

[4] T. Huang and S. Russell, "Object Identification in a Bayesian Context," *In Proc. of International Joint Conference on Artificial Intelligence (IJCAI-97)*, pp. 1276–1283, 1997.

[5] P. KaewTrakulPong and R. Bowden, "A Real-time Adaptive Visual Surveillance System for Tracking Low Resolution Colour Targets in Dynamically Changing Scenes," *In Journal of Image and Vision Computing*, vol. 21, no. 10, pp. 913–929, 2003.

[6] T. Ellis, D. Makris, and J. Black, "Learning a Multi-Camera Topology," *In Proc. of Joint IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pp. 165–171, 2003.

[7] O. M. Mozos, R. Kurazume, and T. Hasegawa, "Multi-part people detection using 2D range data," *International Journal of Social Robotics*, 2010.

[8] L. E. Navarro-Serment, C. Mertz, and M. Hebert, "Pedestrian detection and tracking using three-dimensional ladar data," in *Proc. of The 7th Int. Conf. on Field and Service Robotics*, July 2009.

[9] G. Bradski, "Computer Vision Face Tracking as a Component of Perceptual User Interface," *In Proc. of Workshop on Applications of Computer Vision*, pp. 214–219, 1998.

[10] A. Ess, B. Leibe, K. Schindler, and L. V. Gool., "A Mobile Vision System for Robust Multi-Person Tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[11] F. Gustafsson and F. Gunnarsson, "Mobile Positioning using Wireless Networks," *IEEE Signal Processing Magazine*, pp. 41–53, 2005.

[12] P. Norlund, F. Gustafsson, and F. Gunnarsson, "Particle Filters for Positioning in Wireless Networks," in *Proceedings of EUSIPCO*, 2002.

[13] B. Kusý, A. Ledeczi, and X. Koutsoukos, "Tracking mobile nodes using RF Doppler shifts," in *Proceedings of SenSys*, 2007, pp. 29–42.

[14] G. Hollinger, J. Djugash, and S. Singh, "Tracking a moving target in cluttered environments with ranging radios," in *IEEE International Conference on Robotics and Automation*, May 2008.

[15] A. Gilbert and R. Bowden, "Incremental, Scalable Tracking of Objects Inter Camera," *In Computer Vision and Image Understanding (CVIU)*, vol. 3, pp. 43–58, 2008.

[16] A. Gilbert, J. Capitán, R. Bowden, and L. Merino, "Accurate Fusion of Robot, Camera and Wireless Sensors for Surveillance Applications," in *In Proc. Ninth IEEE International Workshop on Visual Surveillance (ICCV09)*, Kyoto, Japan, 2009.

[17] P. Viola and M. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, pp. 137–154, 2004.

[18] M. Villamizar, J. Scandaliaris, A. Sanfeliu, and J. Andrade-Cetto, "Combining color invariant gradient detector with HOG descriptors for robust image detection in scenes under cast shadows," in *Proceedings of the International Conference on Robotics and Automation, ICRA*, 2009.

[19] L. D. Stone, T. L. Corwin, and C. A. Barlow, *Bayesian Multiple Target Tracking*. Norwood, MA, USA: Artech House, Inc., 1999.

[20] F. Caballero, L. Merino, P. Gil, I. Maza, and A. Ollero, "A probabilistic framework for entire wsn localization using a mobile robot," *Journal of Robotics and Autonomous Systems*, vol. 56, no. 10, pp. 798–806, 2008.

[21] J. Capitán, L. Merino, F. Caballero, and A. Ollero, "Delayed-State Information Filter for Cooperative Decentralized Tracking," in *Proceedings of the International Conference on Robotics and Automation, ICRA*, 2009.

[22] ——, "Decentralized Delayed-State Information Filter (DDSIF): A new approach for cooperative decentralized tracking," *Robotics and Autonomous Systems*, vol. 59, no. 6, pp. 376 – 388, 2011.

[23] E. Nettleton, H. Durrant-Whyte, and S. Sukkarieh, "A robust architecture for decentralised data fusion," in *Proc. of the International Conference on Advanced Robotics (ICAR)*, 2003.

[24] F. Bourgault and H. Durrant-Whyte, "Communication in general decentralized filters and the coordinated search strategy," in *Proc. of The 7th Int. Conf. on Information Fusion*, 2004, pp. 723–730.

[25] S. Julier and J. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proceedings of the American Control Conference*, vol. 4, Jun. 1997, pp. 2369–2373.