

Choosing an RSS Device-Free Localization Algorithm for Ambient Assisted Living

Pietro Cassarà*, Francesco Potorti*, Paolo Barsocchi*, Michele Girolami*[§],

*ISTI institute of CNR, via Moruzzi 1, 56124 Pisa, Italy

Email: {cassara,potorti,paolo.barsocchi,girolami}@isti.cnr.it

[§]Department of Computer Science Largo B. Pontecorvo 3, 56127 Pisa, Italy

Abstract—Device-free localization algorithms attract, among others, the attention of researchers working in the Ambient Assisted Living (AAL) scenarios, where the target user might not be able or willing to wear any devices. We concentrate on systems that exploit the Received Signal Strength indicator coming from wireless devices whose position is known, called anchors. In this paper we select and test the main device-free localization solutions and experimentally compare their performance using a smaller number of anchors than commonly found in the literature. We illustrate the procedure used to validate our comparing procedure and we give suggestions on usability in the application scenarios typical of AAL.

To the best of our knowledge, this is the first direct comparison between different device-free algorithms using the same input data for all of them, and the first one that compares their performance with a varying number of anchors. Thanks to the characteristics of our comparison procedure, we can make suggestions about the more appropriate algorithms to use for different kinds of applications.

I. INTRODUCTION

Indoor localization systems have attracted the attention of research and industry in the last decade. In contrast with outdoor environments, where GPS is a convenient solution for most uses, the indoor arena has no clear winner yet, and research is showing more and more clearly that only a mix of different methods can provide enough performance and ease of use in most scenarios.

In this paper we concentrate on *device-free* localization systems based on RSS (Received Signal Strength) measurements. Such systems allow localization of a target (a person) who is not collaborating in any way with the system, in contrast with localization systems that require the user to carry an active device sending or receiving signals – such as a smartphone or a probe – or a passive device such as an RFID tag.

Such systems are commonly mentioned in the literature because of their ability to identify the location of one or more unknowing people with high accuracy, even through walls [1]. Typical applications are military, law enforcing, or even useful in emergency and rescue situation, where an environment which is difficult or temporarily impossible to access should be monitored for the presence and movement

of people inside. These applications require the installation of several dozens wireless transmitters around and inside the area of interest, all of which should be able to hear each other's transmitted signals and allow for multi-target detection and tracking with high precision in real-time.

Analogous applications, but with lower time and space accuracy requirements, include monitoring of spaces for the presence of people or big animals, like intruder detection and similar situations.

Other applications fall under the umbrella of Ambient Intelligence, especially the Ambient Assisted Living (AAL) scenarios [2], where the target user might not be able to wear any kind of hardware, either because of disabilities or simply because it is too complex. The location can be domestic or related to assistive environments. In these scenarios we can assume that the presence of small independent devices communicating via wireless protocols is going to be more and more ubiquitous [3], which means that device-free localization can be obtained “parasitically”, i.e. without the need for installation of devices specifically dedicated to it, but using already deployed devices which are appropriately configured via software. The real-world emergence of Internet-of-Things installations brings this scenario closer than ever: this is the scenario in which we are most interested. Note that in this scenario we envision that the number of wireless devices available in a small environment (few rooms) should be expected to be smaller than the 30-60 devices commonly seen in the literature, at least in the foreseeable future.

RSS-based device-free systems work by estimating the RSS among each pair of fixed-position wireless devices (anchors), so creating a sort of detailed RSS photograph as a function of time that depends on the position of the user into the environment. The way the RSS information is used to infer the position of the target user is a matter of research, and current solutions are still in their youth. The algorithms follow two main approaches. The first one is based on estimating the position using classification-based methods on specific features of the RSS data. [4], [5]; a time-consuming calibration phase is required to obtain the training data. The second approach, known as Radio Tomographic Imaging (RTI) [1], evaluates the interference image created by the presence of the target; calibration can be automated when necessary.

In this work we describe our experience in using three

This work was supported by the project Energia da fonti rinnovabili e ICT per la sostenibilità energetica; sottoprogetto Smart Building” of the DIITET Department of CNR (Italy) and by the POR CRO FSE 2007-2013 research program. 978-1-4673-8402-5/15/\$31.00 2015 IEEE

of the most relevant localization algorithms available in the literature. The three algorithms are compared by considering requirements coming from application scenarios typical of the AAL domain, with specific attention to the number of anchors used. In particular, we considered two algorithms based on the RTI method and one algorithm based on classification-estimation. The proposed comparison is performed in an indoor environment composed by two communicating rooms, and is based on a single set of experimental measurements to which the three algorithms are applied. This setting allows not only to compare the error performance of the algorithms selected, but also to assess the resilience of the selected systems to a reduction in the number of anchors deployed in the environment, which as mentioned is of particular interest in the AAL domain. As far as we know, this is the first analysis that compares different algorithms using the same set of experimental measures, and the first one that makes a comparison using a variable number of anchors. We anticipate that the analysis presented in this work will help to define which is the best device-free localization system as a function of the application scenarios.

In the following sections, we first illustrate the measurement campaign we performed, by describing the localization area and the sensor application used to gather the RSS measurements. Then, we describe the selected localization algorithms, the evaluation metrics taken into account and, finally, a preliminary discussion about the results obtained when comparing the three algorithms.

II. THE MEASUREMENT CAMPAIGN

We execute the measurement campaign in two rooms hosted within our research institute, we refer to them as the localization area, for a total area of 75 m² (see figure 1(a)). Some obstacles are present within the localization area, such as a desk placed in the right lower corner of the first room, and a square formed of wooden walls in the upper center part of the second room, so the net usable area is 59 m². The floor is made of square tiles of edge 60 cm. We stick on the floor a number of marks in order to draw the path that the target follows during the experiments. The target moves along the path shown in figure 1(b) at a regular speed of one step per second, with the help of a metronome, and stays still for 5 s on the points indicated with a circle.

The anchors are the nodes of a Wireless Sensor Network (WSN) deployed in the localization area. They gather the RSS values during the target's walks. Anchors are IRIS Motes from Crossbow [6], based on the RF transceiver AT86RF230 at 2.4 GHz, compliant with IEEE 802.15.4 standard, most of which are hung on the room walls. Three measurements are done, using 16, 20 and 24 anchors. In figure 1(b) the group of 16 anchors is shown with red squares. The four black squares are added for the 20 anchors measurement, and finally the four diamonds are added for the 24 anchors measurement. An additional anchor is a sink node connected to a Linux-based laptop via USB, located on the desk, which collects all the RSS value information from the network.

The RSS values are gathered with a TinyOS application running on every node of the WSN. The protocol, inspired by the SpinQueue algorithm [7], is used to schedule node transmissions, in order to prevent packet collisions and maintain a high data collection rate. When an anchor is transmitting, all other anchors receive the packet and measure the received signal strength of the received packet. These RSS measurements are transmitted to a base station along with the node's unique ID. The base station collects all RSS measurements and forwards the data to a laptop for storage and later processing.

Each anchor is tagged with a unique index, $i = 1 \dots n$, the protocol consists of rounds in which each anchor broadcasts a message. At startup, the sink node broadcasts a starting message and then just listens to the network, receiving and logging the messages coming from the network. After the starting message, the anchor with index 1 broadcasts the first packet. All other anchors receive the packet, perform the RSS measurement and store its value on the payload of the packet they will send next. After the first anchor has transmitted, the next one ($i = 2$) broadcasts its packet containing all the stored RSS values, and so on until all anchors have transmitted. The next round is then transmitted on a different IEEE802.15.4 communication channel: four different channels are used in a round-robin fashion. After four rounds, the whole procedure is restarted.

In the real implementation, this basic sketch of the algorithm is augmented with error checking and auto-correction features based on a series of timeouts which are necessary to cope with the occasional packet loss and possible consecutive packet losses which may cause lost synchronization in the token-passing procedure. This is necessary because the token-passing is completely distributed apart from the sending of the start packet at the beginning. Using the above described hardware, we are able to send about 60 packets per second.

III. DEVICE-FREE LOCALIZATION ALGORITHMS

Device-free localization algorithms based on the RSS rely on two main approaches for the target localization, namely classification and Radio Tomographic Approaches. The former is based on the classification of features linked to the target position through the measures of RSS [4], [5], the latter evaluates the interfering image produced by the user presence [1]. In the following these two different approaches are briefly described, in particular the classifier (CLAS) [5] as well as the RTI approach that comes in two flavors, namely the shadow-based radio tomographic imaging (SRTI) [1] and the variance-based one (VRTI) [8].

A. Classification-Based Method

In [5], authors propose a localization algorithm based on the learning by example (LBE) strategy to localize and track a target. The localization problem is addressed only by considering the available RSS values at the nodes of the wireless sensor network deployed in the localization environment.

Each anchor a_j , $j = 1 \dots N$ is a transceiver located at a known position (x_{a_j}, y_{a_j}) , $j = 1 \dots N$. Under the assumption

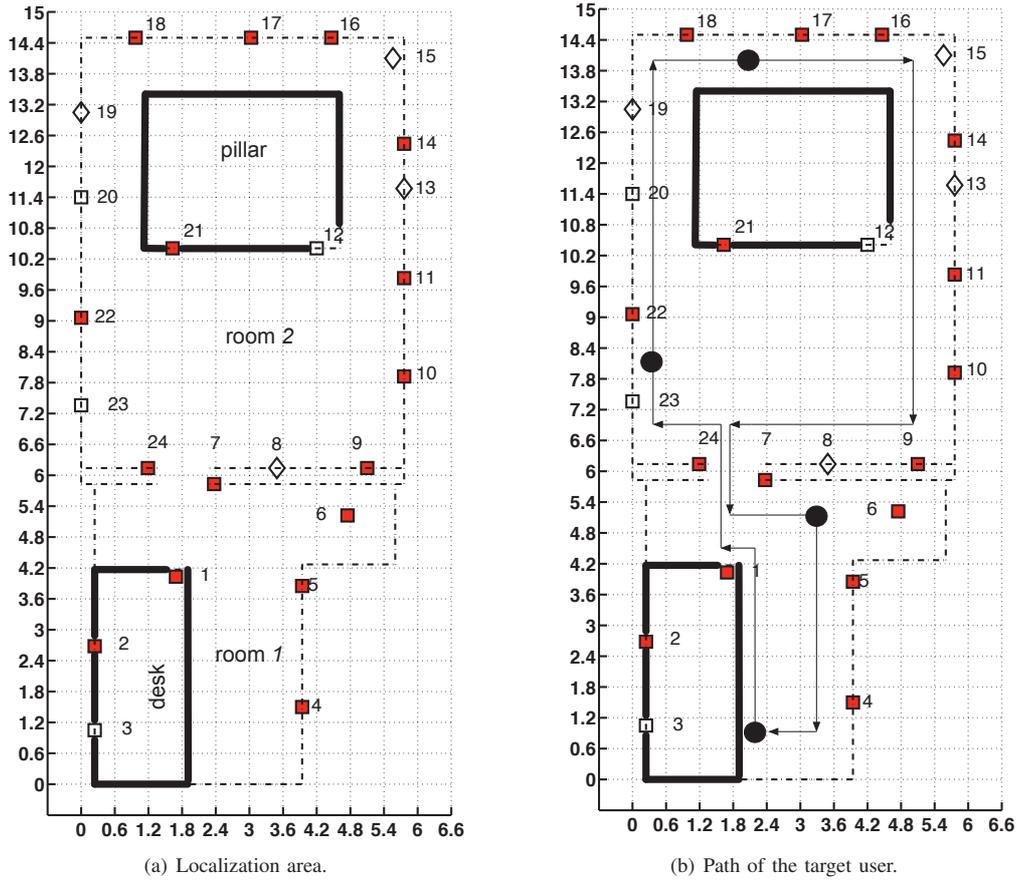


Fig. 1. Localisation area and path of the target user. The pillar has wooden walls, the dotted lines are the walls of the area, composed by two communicating rooms.

that each node communicates with all the remaining $N - 1$ nodes, a total amount of $L = N(N - 1)$ wireless links are available. The measured value of signal strength s_j^i on the link $l = (a_i, a_j)$ depends on the interactions among the electromagnetic signal radiated by the i^{th} source, the localization scenario, and the targets to be localized. The calibration phase of the RSS values is performed without the target in the localization area, in order to distinguish the impact of the user movements on the RSS values from the impact of the surrounding environment. During the localization phase the RSS values are filtered taking into account the RSS values \hat{s}_j^i collected during the calibration phase. The contribution of the surrounding environment is filtered out from the RSS measures s_j^i obtained when the target is in the area, by defining a *differential measure* of the RSS values as

$$\Gamma_{ij} = \frac{s_j^i - \hat{s}_j^i}{\hat{s}_j^i}, \quad i = 1 \dots N, \quad j = 1 \dots N - 1.$$

The differential measure is acquired for all the WSN links $\Gamma = \{\Gamma_{ij}, i = 1 \dots N, j = 1 \dots N - 1\}$. Starting from the differential measurements Γ , the addressed problem is about the definition of the probability that the target lies in a given position $\mathbf{x} = [x, y]$ of the localization area.

To evaluate the probability that the user lies in a given position, a classification technique based on Support Vector Machine (SVM) is adopted. SVM ([9], [10]) is a binary linear classifier, meaning that it assumes linear separability of two classes of data and attempts to find a hyperplane in the feature space separating the data points of the two classes. The optimum separation is achieved by the hyperplane that maximizes its distance from the marginal data points on each side (the support vectors), that is the maximum-margin hyperplane. Thanks to the use of kernel functions, used to nonlinearly map the feature space into a high-dimensional space, computation of the hyperplane can be made using quadratic programming, a computationally efficient optimization technique. SVM requires a learning phase, that is a preliminary system calibration procedure where a set of RSS measurements at a grid of points in the environment are collected. Points chosen for learning correspond to possible locations of the user that should be localized. For each point a tuple of RSS values is produced, and is stored in a learning database. Once the learning phase is completed, the system enters the localization procedure, when every time a new RSS tuple is produced, the SVM uses information learned from the tuples stored in the database to classify it, that is to find the

most likely position of the user.

An SVM method needs R training configurations Δ ,

$$\Delta = \{(\Gamma, \mathbf{x}_m, v_m)_r, r = 1 \dots R\}$$

given by the set of differential measurements Γ , a random position \mathbf{x}_m with the associated state

$$v_m = \begin{cases} 1 & \text{if the target is in } \mathbf{x}_m \\ -1 & \text{otherwise.} \end{cases}$$

During the training phase, the training set is used to find a suitable decision function Φ by means of an SVM strategy [9], [10]. Assuming that the localization area is a lattice with C squared cells, the authors define the decision function for the given cell c by

$$\Phi(\Gamma, v_c) = \sum_{p=1}^C \sum_{r=1}^R \left\{ \alpha_c^r v_c^r \Theta(\Gamma^{(r)}, \Gamma^{(p)}, p, c) \right\} + \frac{\sum_{p=1}^C \sum_{r=1}^{R_{sv}} \left\{ v_c^r - \sum_{p=1}^C \sum_{r=1}^R \left\{ \alpha_c^r \Theta(\Gamma^{(r)}, \Gamma^{(p)}, p, c) \right\} \right\}}{R_{sv}}$$

where $\Theta(\cdot)$ is the kernel function adopted for the problem addressed, the α values are the Lagrange multipliers of the optimization problem associated with the SVM problem, and R_{sv} is the support vector, i.e. the set of training data where the Lagrangian multipliers for the cell c are not equal to zero. Reference [11] provides an in-depth analysis of the problem. Through the decision function, the classification problem can be defined as a binary classification problem.

Note that the sign of the decision function can be replaced by the posterior probability $Pr\{\mathbf{v} = 1|\Gamma\}$ [12] to construct a location-probability map of the monitored area. The posterior probability gives information about the degree of membership of test data to a particular class, even if $sign[\Phi(\Gamma)]$ does not correctly classify the input pattern. This behavior is mainly due to the generalization capabilities of the SVM approach that, in presence of highly non-separable data, constructs the best separating hyperplane even if the optimal solution to the optimization problem [10] does not exist. In this way, the input test data could belong to the wrong half-plane identified by the decision function. However, taking into consideration the posterior probability it is still possible to compute the distance of that example to each class.

The mapping between the state information and the posterior probability can be provided by

$$Pr\{v_c = 1|\Gamma\} = \frac{1}{1 + exp\{\gamma\Phi(\Gamma, v_c) + \delta\}}$$

where γ and δ are obtained by resolving the optimization problem of a cost function of the training data set, as shown in [11].

Finally the estimated target position is obtained as

$$\hat{x} = \frac{\sum_{c=1}^C x Pr\{v_c = 1|\Gamma\}}{\sum_{c=1}^C Pr\{v_c = 1|\Gamma\}}$$

$$\hat{y} = \frac{\sum_{c=1}^C y Pr\{v_c = 1|\Gamma\}}{\sum_{c=1}^C Pr\{v_c = 1|\Gamma\}}$$

B. Radio Tomography Method

In [1] and [13] the authors discuss the application of tomography to a wireless sensor network, calling this method Radio Tomographic Imaging (RTI). The idea behind the RTI method is that the target modifies the RSS field in a way that depends on his locations; RTI approaches, therefore, exploit the RSS measurements observed along the peer-to-peer links to obtain an image reconstruction of the object position.

In [1] the authors describe how to localize a target through the estimation of the mean RSS (SRTI), while in [8] they improve the algorithm performance by exploiting the RSS variance (VRTI) experienced during target movements. Both algorithms are here evaluated exploiting RSS measurements performed on multiple channels (i.e. frequencies) as described in [13].

The network area is conventionally divided into pixels, so the movement of the user is discretised on the pixel set.

The problem is to find a mapping that links the measured RSS per link to the RSS per pixel. The authors adopt a very simple linear model

$$\mathbf{s} = W \mathbf{s}_{px} + \mathbf{n} \quad (1)$$

where $\mathbf{s}_{px} \in \mathbb{R}^P$ is the RTI over the pixel set, so s_{px_i} is the value of RSS for the i -th pixel, $\mathbf{s} \in \mathbb{R}^L$ is the vector of the measured value of RSS over the set of links, $\mathbf{n} \in \mathbb{R}^L$ is the noise of the measures, and finally $W \in \mathbb{R}^{L \times P}$ is the mapping matrix whose entries are the weights that link the pixel values to the link values of RSS.

The weights of the mapping matrix W can be calculated assuming that the power of the received signal is proportional to the inverse of the squared distance covered by the signal, and that the target crossing a link (a_i, a_j) influences a set of pixels. Precisely, the authors assume that the set of influenced pixels fall within the area limited by an ellipse. Hence, for the weights of the matrix W the following equation is applied:

$$w_{ij} = \frac{1}{\sqrt{LoS}} \begin{cases} \phi & \text{if } d_{ij}^1 + d_{ij}^2 < LoS + \lambda \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where LoS is the distance of the line of sight between two nodes, d_{ij}^1 and d_{ij}^2 are the distances from the center of pixel j to the two node locations for link i , and λ is a tunable parameter describing the width of the ellipse. The parameter λ is typically set in the range from 0.1 to 0.6 m. The ellipse is

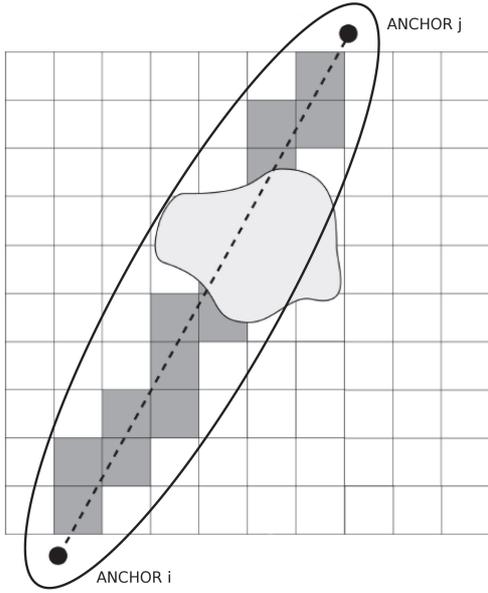


Fig. 2. Pixels affected by RSS changes due to an intervening body along the line of sight between transmitting and receiving node.

primarily used to simplify the process of determining which pixels fall along the LoS (line of sight) path, as shown in figure III-B, where gray pixels represent a person's body. Finally, the parameter ϕ is a scaling factor used to normalize the RTI, whose typical values are between 1 and 100 dB².

The model estimation of the values of the function of RSSs per pixel in the equation (1) provides a mathematical framework to relate the target's movement in space to the RSS values per link. The model is an ill-posed inverse problem, that is, it is highly sensitive to measurement and modeling noise. The solution \mathbf{s}_{px} can be calculated by the least-squares approach, but the solution can not be unique, hence a regularization method [14] must be applied to obtain the solution. The method proposed by the authors propose is Tikhonov's regularization. Using Tikhonov's method the least-squares problem is solved by

$$\hat{\mathbf{s}}_{px} = \arg \min_{\mathbf{s}_{px}} \frac{1}{2} \|\mathbf{W}\mathbf{s}_{px} - \mathbf{s}\|^2 + \alpha \|\mathbf{Q}\mathbf{s}_{px}\|^2$$

where \mathbf{Q} is the Tikhonov's matrix that produces the solution with the desired properties, and α is a tunable regularization parameter. To calculate the parameter α many algorithms have been developed [15]. The least-squares problem solution is

$$\hat{\mathbf{s}}_{px} = (\mathbf{W}'\mathbf{W} + \alpha\mathbf{Q}'\mathbf{Q})^{-1}\mathbf{W}'\mathbf{s}.$$

As stated above, the matrix \mathbf{Q} captures some features of the measured acquired. Taking into account the covariance matrix \mathbf{C} as well as the variance σ_N^2 of the noise process linked to the measures, the solution of equation (III-B) can be calculated as

$$\hat{\mathbf{s}}_{px} = (\mathbf{W}'\mathbf{W} + \sigma_N^2\mathbf{C}^{-1})^{-1}\mathbf{W}'\mathbf{s}. \quad (3)$$

The entries c_{ij} of the covariance matrix \mathbf{C} can be calculated assuming that the spatial attenuation of the field decays exponentially [13]

$$c_{ij} = \sigma^2 \exp\left(-\frac{d_{ij}}{\delta_c}\right)$$

where d_{ij} is the distance between centers of pixels i and j , σ^2 is the variance of pixel attenuation, and δ_c is a correlation parameter that can be used to determine the desired amount of smoothness in the image. The target's coordinates $\mathbf{x} = [x \ y]$ are the coordinates of the maximum value in the vector $\hat{\mathbf{s}}_{px}$ calculated by the regularization method.

As stated above the authors propose two versions of the RTI-based algorithm, the first one evaluates RSS differences from the base situation, that is the RSS shadowing due to the target, and the second one evaluate the RSS variance. The solution showed in the equation (3) works in both cases, in fact, the only difference between two methods is how the entries of the vector \mathbf{s} are evaluated. In the following we provide the descriptions of the ss entries calculation.

When using the Shadow-based method, for every link l , and every channel $c \in Ch$, average values of measured RSS values $\{\overline{RSS}_{lc}\}$ is computed when no target moves within the localization area, this time is named calibration period. The method uses the \overline{RSS}_{lc} values as a measure of the fade level following this rule: if $\overline{RSS}_{lc_1} < \overline{RSS}_{lc_2}$ then the link l is in a deeper fade in channel c_1 than in c_2 . For each link l the \overline{RSS}_{lc} values are sorted for all channels of the set C . Finally, the method creates a set \mathcal{C} of size m containing the indices of the m highest channels by fade level, and then evaluates shadowing, that is the \mathbf{s} entries as the mean values of the differences between the current measured RSS vales and the $\{\overline{RSS}_{lc}\}$, over the m highest channels by fade level.

For the Variance-based method the entries are the variances of the measured RSS values, and no calibration phase is needed.

In [8], the authors describe an RTI-based tracking algorithm developed by filtering the estimated target positions through the Kalman filter defined as

$$\begin{aligned} \hat{\mathbf{V}} &= \mathbf{V} + \sigma_m^2 \mathbf{I}_2; \\ \mathbf{G} &= \hat{\mathbf{V}}(\hat{\mathbf{V}} + \sigma_n^2 \mathbf{I}_2)^{-1} \\ \hat{\mathbf{x}} &= \hat{\mathbf{x}} + \mathbf{G}(\mathbf{x} - \hat{\mathbf{x}}) \\ \mathbf{V} &= (\mathbf{I}_2 - \mathbf{G})\hat{\mathbf{V}} \end{aligned}$$

where \mathbf{I}_2 is the 2×2 identity matrix, σ_m^2 is the variance of the targets motion process, indicating how fast the object is capable of moving. Larger values enable the filter to track faster moving objects. The authors also take into account σ_n^2 that is the variance of the measurement noise. Larger values will cause the filter to trust more the statistical predictions over the instantaneous measurements. The vector $\hat{\mathbf{x}}$ contains the Kalman estimated coordinates x and y . \mathbf{x} is a two-element vector containing the instantaneous measurement of the target coordinates through the RTI method. $\hat{\mathbf{V}}$ is the a priori error covariance matrix and \mathbf{V} is the posterior error

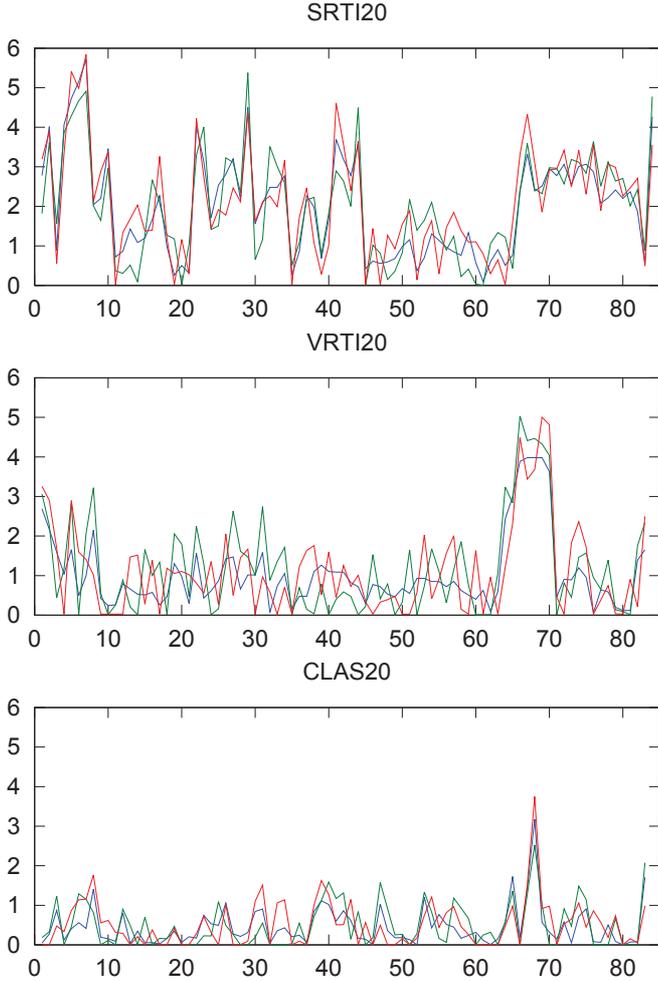


Fig. 3. Error in metres for three algorithms over the 85 positions.

covariance matrix and finally G the Kalman Gain. The authors provide some values for these parameters through a set of measurements performed during their experiments.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

We made a total of three *measurements*, with 24, 20 and 16 anchors hanging from the walls, as depicted in figure 1(a). Each measurement consisted of 85 *positions* repeated 3 times, for a total of about 250 positions on each of which the *positioning error* was computed in each of the three measurements. While the total amount of samples is small in a statistical sense, in the following we are going to explain why we think our results are credible.

Starting from the three measurements, we define a total of nine *situations*, three for each of the three *algorithms* we have used, that is the classifier (CLAS), the shadow-based radio tomographic imaging (SRTI) and the variance-based one (VRTI). In this way, we can make a comparison of the relative accuracy performance of the three algorithms when the number of anchors change.

Each measurement was repeated three times, both for obtaining a higher number of error samples and making statistics more reliable, and for being able to make a direct comparison between the *repetitions* along all the 85 positions of the path. Each graph of Figure 3 shows a line for each of the three repetitions: the repetitions exhibit clear indications of statistical noise and a strong spatial correlation between repetitions. We consider this spatial correlation a good indication of the robustness of the experimental procedure. Additionally, we can observe spatial correlation also among the different algorithms, suggesting that in the specific anchor configuration used in this measurement some positions points provide low-quality information. The fact that this translates in poor performance for all the three algorithms is an additional indication of the robustness of the experimental procedure.

The three measurements and the three repetitions were done in a completely independent way, at time distances of some hours to some days.

A. Overall comparison of the three algorithms

In Figure 4 we show the error distributions for the nine situations. For an easier comparison, the distributions are overlaid with four *scalar statistics*, namely the root of mean square (RMS) error and the 50th, 75th and 90th percentiles of errors.

From a practical standpoint, the 75th percentile is the most robust estimator, and indeed it is the one that is used in the indoor localization track of the EvAAL competition [16]. The 90th percentile is useful to get an idea of the length of the tail of the error distribution, while the median and the RMS error are useful for comparison with statistics found in the literature.

It is apparent from Figure 4 how CLAS outperforms the tomographic-based methods in terms of pure error performance. This was to be expected, given that CLAS is classification-based, and it exploits prior knowledge that is not available to the other algorithms. This knowledge comes at a significant cost both in terms of installation and of flexibility. More specifically, the training phase requires non-negligible time to be performed by a person following a rigid protocol, and a new training is required for any non-trivial change in the environment, such as a rearrangement of furniture. This is not something that is reasonably done by the end user, and probably rules out the usage of this system from some typical scenarios. For installation in private homes (common in AAL scenarios), the intervention of a technician would be required every time that furniture is reallocated. Even in places where technicians are promptly available, the required intervention rate could turn out to be prohibitive in dynamic environments, such as hospitals or industrial locations, where furniture can be expected to be routinely moved around. The time required for CLAS training by a qualified technician is about 5 s per cell, with squared cells of 0.6 by 0.6 meters, for an overall $8 \text{ s}/\text{m}^2$ plus overhead for setup and testing.

SRTI has the worst error performance. On the other hand, it needs a calibration phase which is much quicker than the training needed by CLAS and which can be completely

Error distribution (PDF) in the nine situations

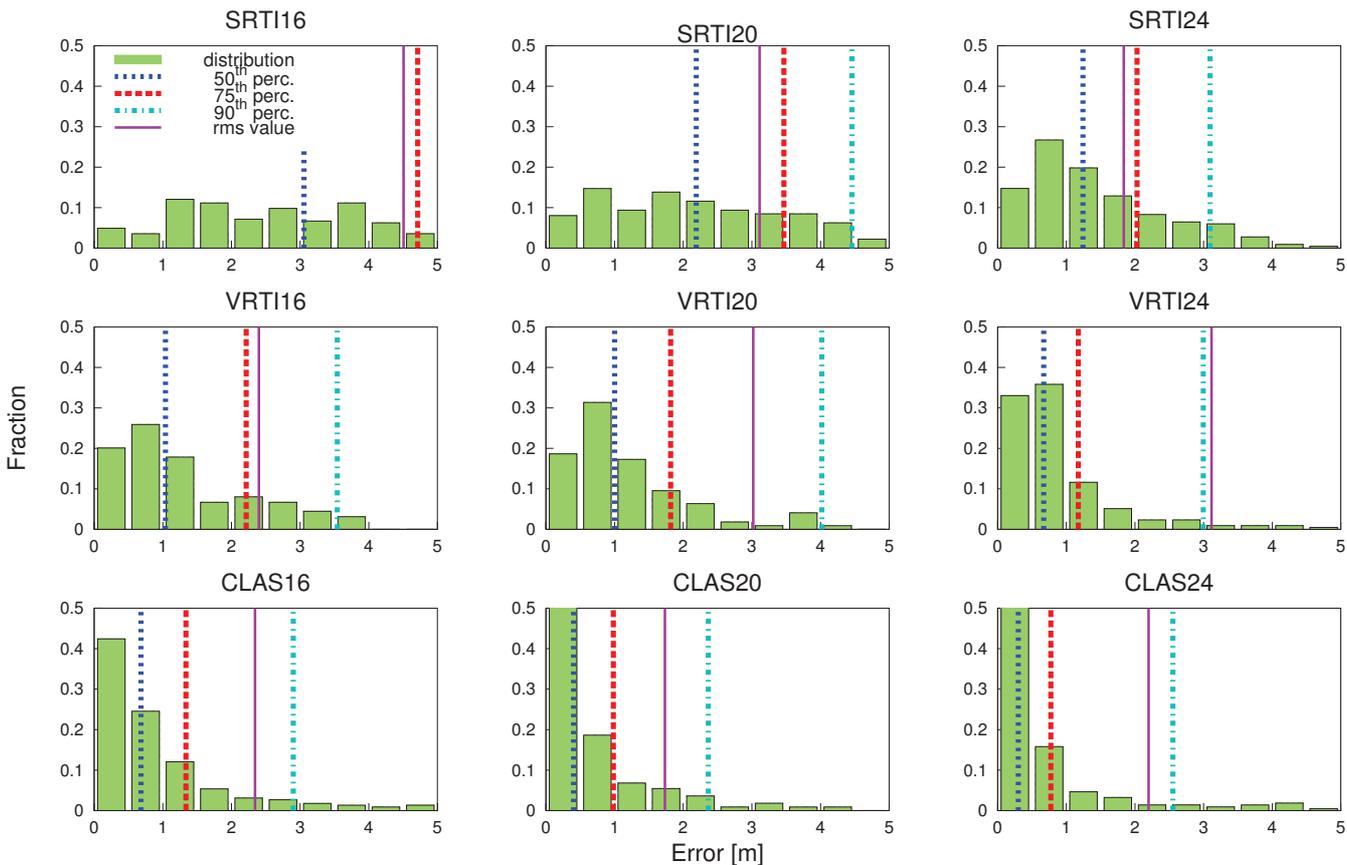


Fig. 4. Distribution of errors in the nine situations with superimposed statistics of RMS error and 50th, 75th and 90th error percentiles.

automated, requiring no human intervention. In practice, when the system senses no motion in the environment for some time, it can autonomously initiate a recalibration procedure that lasts for a few minutes, and is automatically canceled if any movement is detected in the meantime.

VRTI is midway. It does not need any training phase, because it is not sensitive to unmoving obstacles, and as such it is insensitive to changes in the environment. This strength is also its main weakness: being based on signal strength variations, it performs badly when the target does not move. Note, however, that here we are measuring the performance of the *raw* system: by adding a tracking layer with memory, such as a Kalman filter or a particle filter, we can expect performance to improve significantly with non-moving targets.

Table I summarizes the main characteristics of the three algorithms; *accuracy* is the 75th percentile of error in the 24-anchors measurement, which we consider the most appropriate metrics for an experimental localization algorithm.

The CLAS algorithm gives a higher accuracy than the others, however it requires an expensive calibration procedure making it not ideal for the application scenarios typical of the AAL domain. When changes in the localization area are sporadic, the CLAS algorithm can be adopted. However,

in these cases, when installing additional anchor devices is tolerable, other device-free technologies (such as infrared, ultra-wideband, cameras etc...) are good candidates, especially because they require few anchors and can provide high accuracy.

On the other hand, SRTI and VRTI algorithms fit best in many scenarios where the targets move often and where the localization application does not demand high accuracy.

For all the algorithms, the accuracy of the estimation increases with the number of anchors, but it is still in the usable range with as few as 16 anchors. This observation is an interesting result because, to the best of our knowledge, up to now the literature lacks studies on the influence of the placement and number of anchors on the accuracy of the device-free methods based on RSS which are the focus of this paper.

V. CONCLUSION AND FUTURE WORK

We describe and apply a reliable measurement procedure that can be used to compare the performance of different device-free RSS-based indoor positioning algorithms. The procedure uses the same experimental measurements for all three algorithms, making the comparison reliable. We show that the

TABLE I
FUNCTIONAL COMPARISON OF THE THREE ALGORITHMS.

Performance indicator	SRTI	VRTI	CLAS
Accuracy	2 m	1.2 m	0.8 m
Configuration	automated	unneded	expensive
Still target	detected	undetected	detected
Flexibility	yes	yes	no
Reference scenario	AAL, industrial, intrusion, hospital	same as SRTI	specific applications

measurements are consistent across measurement repetitions, and the results are consistent across the algorithm results.

We compare three device-free RSS-based algorithms, which we show to have relative strengths and weaknesses, so that no single winner can be indicated for the generic case. However, given the application scenario, and therefore the requirements, this study gives an indication of the best device-free localization. Table I summarizes these observations.

The algorithms evaluated in this work appear robust with respect to variation of the number of anchors deployed, more specifically we observe a gracefully decrease of their performance as the number of anchors reduces. The performance is still in the usability range with as few as 16 anchors deployed around a 75 m² indoor area consisting of two communication rooms.

The above results encourage studies on a scenario where the wireless transmissions generated by many small devices installed in an smart environment are parasitically exploited to provide information about movement of people in the area. The information so computed has time and space resolution that are strongly dependent on the number of devices, their placement and their rate of packet generation. Future work should concentrate on trying to estimate this dependence.

REFERENCES

- [1] J. Wilson and N. Patwari, "Radio Tomographic Imaging with Wireless Networks," *IEEE Transaction on Mobile Computing*, vol. 9, no. 5, pp. 621–632, May 2010.
- [2] G. Van Den Broek, F. Cavallo, and C. Wehrmann, *AALIANCE Ambient Assisted Living Roadmap*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2010.
- [3] A. Dohr, R. Modre-Opsrian, M. Drobits, D. Hayn, and G. Schreier, "The Internet of Things for Ambient Assisted Living, Schreier, in *Information Technology: New Generations (ITNG), 2010 Seventh International Conference on*, April 2010, pp. 804–809.
- [4] C. Morelli, M. Nicolini, V. Rampa, and U. Spagnolini, "Hidden Markov Models for Radio Localization in Mixed LOS/NLOS Conditions," *IEEE Transaction on Signal Processing*, vol. 5, no. 4, pp. 1525–1542, April 2007.
- [5] F. Viani, M. Martinelli, L. Ioriatti, M. Benedetti, and A. Massa, "Passive real-time localization through wireless sensor networks," in *Proc. IEEE Intl. Conf. IGARSS*, Cape Town, South Africa, July 2009, pp. 718–721.
- [6] Crossbow and Technology, "IRIS Datasheet," <http://bullseye.xbow.com:81/Products/productdetails.aspx?sid=264>, 2013.
- [7] J. Wilson and N. Patwari, "Spin: A token ring protocol for RSS collection," <http://span.ece.utah.edu/spin>.
- [8] —, "See-through walls: Motion tracking using variance-based radio tomography networks," *Mobile Computing, IEEE Transactions on*, vol. 10, no. 5, pp. 612–621, May 2011.
- [9] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*. Cambridge University Press, 2007.
- [10] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [11] A. Massa, A. Boni and M. Donelli, "A Classification Approach Based on SVM for Electromagnetic Subsurface Sensing," *IEEE Tran. On Geoscience and Remote Sensing*, vol. 43, no. 9, pp. 2084–2093, September 2005.
- [12] J. Platt, *Probabilistic outputs for support vector machines and comparison to regularized likelihood methods*. Cambridge, MA: Advances in large margin Classifiers, MIT Press, 1999.
- [13] K. Ossi, M. Bocca, and N. Patwari, "Enhancing the accuracy of radio tomographic imaging using channel diversity," in *IEEE Int. Conf. MASS*, Las Vegas, Nv, October 2012, pp. 1–9.
- [14] J. Wilson, N. Patwari, and F. G. Vasquez, "Regularization methods for radio tomographic imaging," in *Proc. Virginia Tech Wireless Symposium*, Virginia, USA, June 2009, pp. 1–9.
- [15] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*. SPRINGER, 2004.
- [16] P. Barsocchi, S. Chessa, F. Furfari, and F. Potortì, "Evaluating AAL solutions through competitive benchmarking: the localization competition," *IEEE Pervasive Computing Magazine*, vol. 12, no. 4, pp. 72–79, Oct.–Dec. 2013.