ACE: An Accurate and Efficient Multi-Entity Device-Free WLAN Localization System

Ibrahim Sabek, *Student Member, IEEE*, Moustafa Youssef, *Senior Member, IEEE*, and Athanasios V. Vasilakos, *Senior Member, IEEE*

Abstract—Device-free (DF) localization in WLANs has been introduced as a value-added service that allows tracking of indoor entities that do not carry any devices. Previous work in DF WLAN localization focused on the tracking of a single entity due to the intractability of the multi-entity tracking problem whose complexity grows exponentially with the number of humans being tracked. In this paper, we introduce *ACE*: a system that uses a probabilistic energy-minimization framework that combines a conditional random field with a Markov model to capture the temporal and spatial relations between the entities' poses. A novel cross-calibration technique is introduced to reduce the calibration overhead of multiple entities to linear, regardless of the number of humans being tracked. We design an efficient energy-minimization function that can be mapped to a binary graph-cut problem whose solution has a linear complexity on average and a third order polynomial in the worst case. We further employ clustering on the estimated location candidates to reduce outliers and obtain more accurate tracking in the continuous space. Experimental evaluation in two typical testbeds, with a side-by-side comparison with the state-of-the-art, shows that *ACE* can achieve a multi-entity tracking accuracy of less than 1.3 m. This corresponds to at least 11.8 percent, and up to 33 percent, enhancement in median distance error over the state-of-the-art DF localization systems. In addition, *ACE* can estimate the number of entities correctly to within one difference error for 100 percent of the time. This highlights that *ACE* achieves its goals of having an accurate and efficient multi-entity indoors localization.

Index Terms—Binary graph-cut, conditional random fields, device-free localization, energy minimization, Markov models, multi-entity tracking

1 Introduction

Device-free (DF) localization [40] is a concept that allows the detection and tracking of entities that do not carry any devices nor participate actively in the localization process. DF localization has a number of applications including intrusion detection, border protection, smart homes, and traffic estimation.

Different approaches have been proposed for addressing the DF detection and tracking problem that can be categorized into two main groups: Those that require special hardware and those that leverage the already installed wireless infrastructure. Radar-based systems, e.g., [8], [19], [38], computer vision systems, e.g., [14], [20], and radio tomographic imaging (RTI), e.g., [32], provide accurate DF detection and tracking. However, all require the installment of special hardware to track the DF entity. On the other hand, systems that leverage the currently installed wireless networks, e.g., WLAN [13], [21], [28], [37], [40], provide a software only solution for DF localization and have the advantage of scalability in terms of cost and coverage area.

WLAN DF localization systems are based on the concept [40] that the presence of an entity in an RF environment affects the signal strength, which can be used to detect, track, and identify the entities. Fig. 1 shows the architecture of a typical WLAN DF localization system. The system consists of signal transmitters (e.g., standard APs); signal receivers or monitoring points (MPs), such as any WiFi enabled device (e.g., laptops and APs themselves); and an application server that collects the received signal strength (RSS) for the different streams (where a stream is a single (AP, MP) pair) readings and processes them to detect events.

To track entities, and due to the complex relation between RSS and distance in indoor environment, a fingerprint has been traditionally used to capture the RSS behavior at different locations in the area of interest. To construct the fingerprint, a human stands at different locations in the area of interest and her effect on the RSS of the different streams is recorded at the MPs. For the multiple entities case, it requires trying all the combinations of entities over all calibration locations, which grows *exponentially* with the number of fingerprint locations.

In this paper, we introduce *ACE* as a system for the accurate and efficient detection and tracking of *multiple DF entities* in a WLAN environment. *ACE* is based on a probabilistic energy-minimization framework that combines a conditional random field (CRF) with a Markov model: Given a RSS vector from all the streams in the area of interest, the problem of estimating the most probable active user locations is mapped to an energy-minimization problem

- *I. Sabek is with the Department of Computer and Systems Engineering, Alexandria University, Alexandria, Egypt. E-mail: ibrahim.sabek@alexu.edu.eg.*
- *M. Youssef is with the Wireless Research Center, Egypt-Japan University of Science and Technology and Alexandria University, Alexandria, Egypt. E-mail: moustafa.youssef@ejust.edu.eg.*
- *A. Vasilakos is with the Department of Computer Engineering, University of Western Macedonia, PO Box 65251, Psychiko, Attica 15410, Greece. E-mail: vasilako@ath.forthnet.gr.*

Manuscript received 5 Sept. 2013; revised 19 Feb. 2014; accepted 10 Apr. 2014. Date of publication 24 Apr. 2014; date of current version 23 Dec. 2014. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2014.2320265

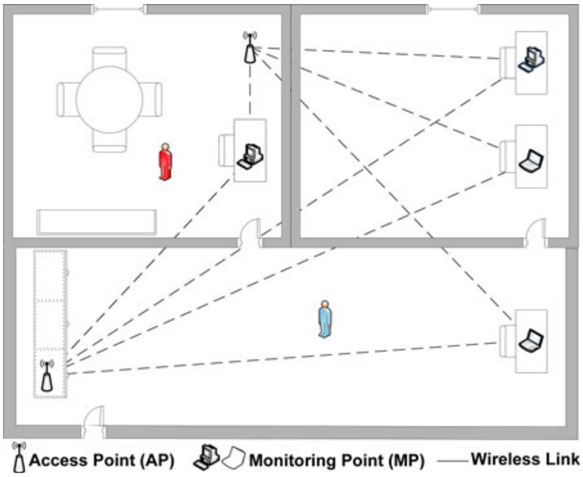


Fig. 1. Typical architecture of a DF WLAN localization system.

whose potential function is designed to preserve *smooth* and *consistent* labels for active locations relative to their neighbors and their movement history. In addition, we show that the designed energy function is *regular* in the sense that it can be mapped to a binary graph-cut problem whose solution has a linear complexity on average and a cubic polynomial in the worst case. *ACE* also introduces a novel cross-calibration technique to reduce the calibration overhead of multiple entities to *linear* in the number of locations, as compared to *exponential* for the current state-of-the-art. This also helps in increasing the system accuracy.

Since a human can affect more than one location in the area of interest, we further employ clustering on the estimated location candidates as a means for reducing outliers and obtaining more accurate tracking in the continuous space. Each detected cluster represents a human whose location in the center of mass of fingerprint locations inside the cluster. Experimental evaluation in two typical testbeds, with a side-by-side comparison with the state-of-the-art, shows that *ACE* can achieve a tracking accuracy of less than 1.3 m. This corresponds to at least 11.8 percent, and up to 33 percent, enhancement in median distance error over the state-of-the-art DF localization systems in the two testbeds, while enabling the tracking of multiple entities. In addition, *ACE* can estimate the number of entities with 100 percent accuracy to within one difference error. This accuracy advantage is obtained without scarifying computational efficiency.

In summary, the contribution of this paper is four-fold:

1) We formulate the *multi-entity* DfP problem as an energy-minimization framework that preserves both spatial and temporal smoothness and consistency (Section 2).
2) We show how to map the problem to a binary graph-cut problem and obtain its solution efficiently; and present the details of our novel cross-calibration technique that reduces the calibration complexity to linear in the number of locations, rather than exponential as in the current state-of-the-art (Section 3).
3) We present clustering techniques for reducing noise and enhancing accuracy (Section 4).

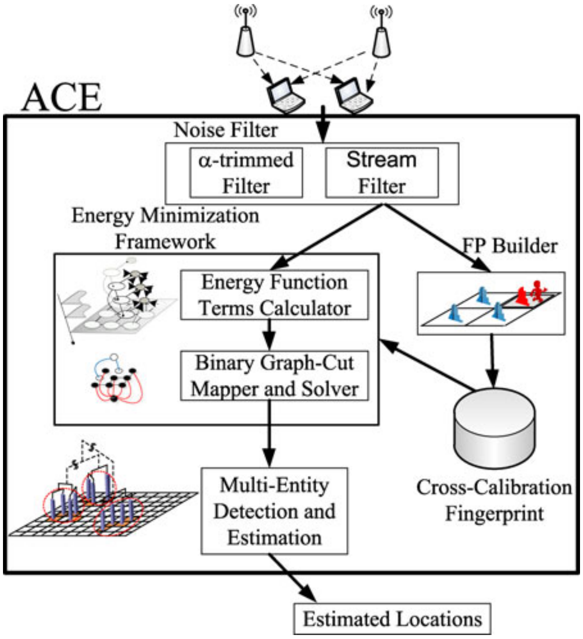

ACE
Noise Filter
α-trimmed Filter
Stream Filter
Energy Minimization Framework
FP Builder
Energy Function Terms Calculator
Binary Graph-Cut Mapper and Solver
Multi-Entity Detection and Estimation
Cross-Calibration Fingerprint
Estimated Locations

Fig. 2. ACE system architecture.

4) We evaluate the system in two typical WiFi testbeds and compare it to the state-of-the-art DF WLAN localization techniques (Section 5).

We also present the related work in Section 6 and discuss the other points related to the system in Section 7. Finally, we conclude the paper and give directions for future work in Section 8.

2 The ACE System

In this section, we give the details of *ACE*. We start by an overview of the system architecture followed by the details of the system modules.

2.1 Overview

Fig. 2 shows the system architecture. The system collects the signal strength readings from the monitoring points for processing. There are two phases of operation:

1) *Offline training phase*: to estimate the system parameters based on the collected signal strength readings and construct the device-free fingerprint. During this phase, a human stands at different locations in the area of interest and the RSS at each MP is recorded. Note that our formulation requires only one human for calibration in the offline phase, regardless of the number of humans during the system operation (Section 3.2). This significantly reduces the calibration overhead as compared to the state-of-the-art DF systems.
2) *Online tracking phase*: to estimate the multi-entities' locations based on the received signal strength from each stream and the fingerprint prepared in the offline phase using the energy-minimization framework.

The *Noise Filtering* module reduces the noise in the RSS readings and filters outlier streams.

The *Fingerprint Builder* module constructs the fingerprint (training data) during the offline phase.

The *Energy-Minimization Framework* calculates the probabilities used in the energy-minimization framework, constructs an equivalent graph, and estimates the most probable active locations (i.e., environment map) based on solving a binary graph-cut problem.

The *Multi-Entity Detection and Estimation* module uses clustering techniques to estimate the number of entities and the location of each entity. A non-zero estimated number of entities is equivalent to a detection event in the area of interest.

2.2 System Model

Without loss of generality, let $\mathbb{X}$ be a two-dimensional physical space. At each location $x \in \mathbb{X}$, we can get the signal strength from k streams. We denote the k-dimensional signal strength space as $\mathbb{S}$. Each element in this space is a k-dimensional vector, $s = (s_1, \dots, s_k)$, whose entries represent the signal strength readings from a different (AP, MP) pair.

Given that m humans are standing in the area of interest, $m \geq 0$, these humans will affect the different streams. Therefore, the problem becomes:

Problem 1. *Given a RSS vector s, we want to both estimate the number of humans $\hat{m}$ and, if $\hat{m} > 0$, the locations of these humans $\{x_i | 0 < i \leq \hat{m}, x_i \in \mathbb{X}\}$, such that the probability $P(x_1, x_2, \dots, x_{\hat{m}} | s)$ is maximized.*

In Section 3, we assume a discrete $\mathbb{X}$ space. We discuss the continuous space case in Section 4.

2.3 Noise Filtering

The aim of this module is to preprocess collected RSS readings during the offline and online phases to reduce the noise effects and detect outliers. We use two techniques: RSS filtering and stream filtering.

2.3.1 RSS Filtering

RSS is a noisy quantity due to the time varying wireless channel [39]. To reduce the noise effect, we apply an *α-trimmed Mean* filter [37] to the measured RSS values. An α-trimmed filter has the advantage of handling both impulse and gaussian noise, as compared to mean and median filters that can handle only one of them. In addition, it is simple to implement: Given a window of q RSS samples, the α-trimmed filter sorts the samples (such that $RSS_1 \leq RSS_2 \leq \cdots \leq RSS_q$) and then discards the α extreme samples and averages the remaining samples. The output of the α-trimmed mean filter is

$$f(q; \alpha) = \frac{1}{q - 2\lceil \alpha q \rceil} \sum_{i=\lceil \alpha q \rceil + 1}^{q - \lceil \alpha q \rceil} \mathrm{RSS}_i, \tag{1}$$

where $0 \leq \alpha < 0.5$. We set α to 0.2 as it is a reasonable value for the window size we use in our system (Section 5).

2.3.2 Streams Filter

Even after smoothing the RSS values, using the alpha-trimmed filter, the readings of a single stream may have significantly changed between the offline and online phases due to changes in the environment. To detect this change and filter outlier streams, we use the Analysis of Variance (ANOVA) to test whether the mean of the RSS of a particular stream have significantly changed between the offline and online phase. If there is a statistically significant difference, the stream is filtered from the current calculations.

3 Energy-Minimization Framework

In this section, we assume a discrete $\mathbb{X}$ space with n locations. Let $\{\alpha_i^t, 0 < i < n\}$ be a set of bernoulli random variables, where α_i^t takes the value of 1 if a human is standing at location $i \in \mathbb{X}$ at time t, and 0 otherwise. Therefore, the problem of estimating the number of entities $\hat{m}$ and their locations, given the received signal strength vector s (Problem 1), is equivalent to finding the assignment of α_i^t's that maximizes

$$P(\mathbb{M}^t \mid s), \tag{2}$$

where $\mathbb{M}^t = (\alpha_1^t, \alpha_2^t, \dots, \alpha_n^t)$. We refer to $\mathbb{M}^t$ as the *environment map* at time t. In this case, $\hat{m} = \sum_{i=1}^n \alpha_i^t$ and the most probable locations of the $\hat{m}$ entities are the locations whose α_i^t's are assigned to one.

Traditional work on probabilistic WLAN localization, *both device-based and device-free*, e.g., [26], [40], use Bayesian inversion to estimate $P(\mathbb{M}^t | s)$. However, these systems typically assume only one entity in the area of interest. Moving to more than one entity makes this Bayesian inversion approach intractable as the complexity of estimating $P(s | \mathbb{M}^t)$ increases exponentially with the number of tracked entities [26] (due to the need to try all combinations of humans' poses in the area of interest).

Alternatively, we use an energy-minimization framework that leverages this joint estimation problem of α_i^t's to enhance the accuracy while, at the same time, leads to an efficient solution. In particular, we represent the *spatial* constraints on the human position by a Conditional Random Field model favoring coherence between adjacent locations. The *temporal* relation between the human locations is captured by a second order Hidden Markov Model (Fig. 3). Estimation is finally performed by mapping the problem to a binary graph-cut problem that can be efficiently solved in $O(n)$ on average and $O(n^3)$ in the worst case.

A CRF is an undirected graphical model that defines a log-linear distribution over label sequences given a particular observation sequence [17]. It was introduced as a framework for labeling and segmenting data that models the conditional probability $P(Y|X)$, where X and Y are the observations and the labels respectively. CRFs have the advantage of relaxing the strong independence assumptions made by Hidden Markov Models (HMMs) [30] for a large number of variables (such as those in the environment map). In addition, CRFs avoid the label bias problem [14], a weakness exhibited by maximum entropy Markov models (MEMMs) [16] and other conditional Markov models based on directed graphical models. Therefore, CRFs outperform both MEMMs and HMMs on a number of real-world sequence labeling tasks [14], [17], [19].

By combining a HMM for temporal relations with a CRF for spatial relations, we gain the benefit of both worlds in

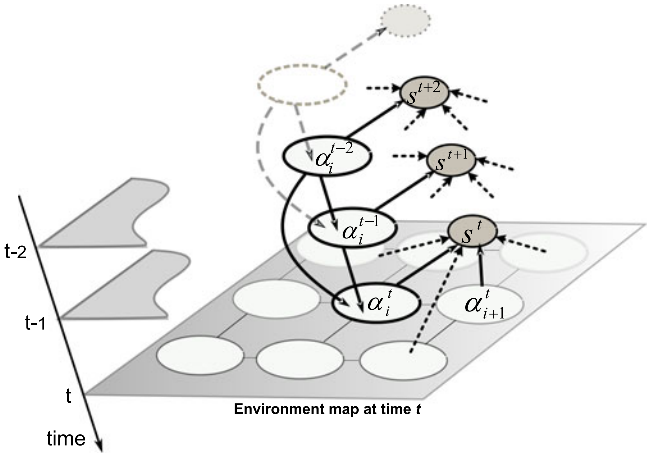


Fig. 3. Combined CRF-HMM model. This graphical model illustrates both the signal strength likelihood together with the spatial and temporal priors. The same temporal chain is repeated at each discrete location. Spatial dependencies are illustrated for a four-neighborhood system. The entire environment map affects the RSS vector s^t.

terms of accuracy and efficiency. In this section, we describe the energy-minimization framework construction and how we efficiently solve it.

3.1 Framework Construction

Our model extends the model in Equation (2) to capture the temporal constraints. In particular, our goal becomes to find the environment map at time t, $\mathbb{M}^t$, that maximizes:

$$P(\mathbb{M}^t \mid s^t, \mathbb{M}^{t-1}, \mathbb{M}^{t-2}) \tag{3}$$

assuming a second order temporal dependence in the Markov model as we discuss in details later.

Based on CRF theory [16], our combined model estimates the probability of Equation (3) as

$$P(\mathbb{M}^t \mid s^t, \mathbb{M}^{t-1}, \mathbb{M}^{t-2}) \propto \exp - \{E^i\} \tag{4}$$

where $E^i = E(s^t, \mathbb{M}^t, \mathbb{M}^{t-1}, \mathbb{M}^{t-2})$ is defined as an energy function that captures the required spatial and temporal constraints on the DF tracking problem at location i . That is, we want to estimate the current environments map given the previous two environment maps and the current signal strength vector measured at the monitoring points. This is obtained by the joint maximization of the posterior in Equation (4), which is equivalent to the minimization of energy:

$$\hat{\mathbb{M}}^t = (\hat{\alpha}_1^t, \hat{\alpha}_2^t, \ldots, \hat{\alpha}_n^t) = \text{argmin } E^i. \tag{5}$$

Energy terms. For our DF tracking problem, each E^i is composed of three components that capture the observation measurement and the spatial and temporal relations between entities' poses:

$$\begin{aligned} E^i &= E(s^t, \mathbb{M}^t, \mathbb{M}^{t-1}, \mathbb{M}^{t-2}) \\ &= V^{\text{Tm}}(\mathbb{M}^t, \mathbb{M}^{t-1}\mathbb{M}^{t-2}) + V^{\text{Sp}}(\mathbb{M}^t, s^t) + U^{\text{SS}}(\mathbb{M}^t, s^t). \end{aligned} \tag{6}$$

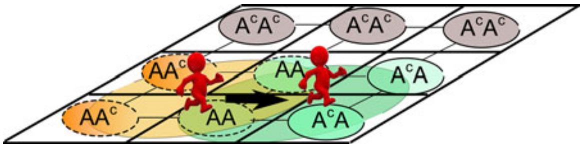


Fig. 4. Temporal transitions at a location (assuming the human affects four locations inside the circle). (a) An entity moves towards the right from time $t-2$ to time $t-1$. (b) Between the two time instances, locations may remain in their own active or inactive states (denoted A and A^c, respectively) or change state; thus defining four different kinds of temporal transitions: $A \rightarrow A$, $A \rightarrow A^c$, $A^c \rightarrow A$, $A^c \rightarrow A^c$. Those transitions influence the label that a location in the environment map is going to assume at time t.

The term $V^{Tm}(\mathbb{M}^t, \mathbb{M}^{t-1}\mathbb{M}^{t-2})$ is a temporal prior term that represents a second-order Markov chain that imposes a tendency to temporal continuity on the environment map.

The term $V^{Sp}(\mathbb{M}^t, s^t)$ presents a spatial prior term which imposes a tendency to spatial continuity of the environment map, favoring coherent assignments between adjacent locations.

Finally, the $U^{SS}(\mathbb{M}^t, s^t)$ term is a likelihood term that evaluates the evidence for location labels based on the RSS distributions in the case of human absence and presence.

This energy model captures both the signal strength likelihood together with the spatial and temporal priors. Fig. 3 shows the graphical representation of the model. Details of these factors are given in the next subsections.

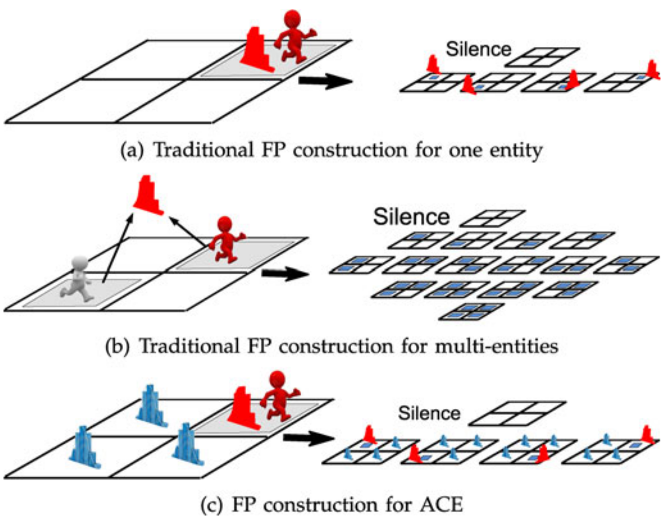
3.1.1 Temporal Prior Term

This term reflects the temporal smoothness captured by the second order HMM. Fig. 4 shows the four different *temporal* transitions a location assignment (label) can undergo in an environment map, based on a two time instances analysis. For instance, an active location may remain active (locations labeled AA in Fig. 4) or move to the inactive state (locations labeled AA^c), etc. It is important to note that a first-order Markov chain is inadequate to convey the nature of temporal coherence in this problem; a higher order Markov chain is required. For example, since a person who started walking will probably continue walking, a location that was inactive at time $t-2$ and is active at time $t-1$ is far more likely to remain active at time t than to go back to the inactive state. A second-order Markov chain is used to balance performance and complexity. We quantify the effect of the order of the chain in Section 5.2.5.

These intuitions are captured probabilistically and incorporated in our energy-minimization framework by means of a second order Markov chain, as shown in the graphical model of Fig. 3. The temporal transition priors ($P(\alpha_i^t \mid \alpha_i^{t-1}, \alpha_i^{t-2})$) are learned during the training phase. This leads to the following joint temporal prior term:

$$V^{\text{Tm}}(\mathbb{M}^t, \mathbb{M}^{t-1}\mathbb{M}^{t-2}) = \beta \sum_{i=1}^{n} -\left[\log P\left(\alpha_i^t \mid \alpha_i^{t-1}, \alpha_i^{t-2}\right)\right], \tag{7}$$

where $\beta < 1$ is a discount factor to allow for multiple counting across non-independent locations. The optimal value of β, as well as the other parameters of the CRF, are obtained discriminatively from the training data using the iterative scaling algorithm [17].



(a) Traditional FP construction for one entity

(b) Traditional FP construction for multi-entities

(c) FP construction for ACE

Fig. 5. Difference between fingerprint (FP) construction for traditional DF systems and ACE. Left figure represents an example while the figure to the right represents all required combinations (FP complexity). (a) FP construction for one entity in a traditional DF system: One histogram, representing the active state RSS, is stored in only one location (where the user is standing). (b) FP construction for two entities in a traditional DF system: Two humans are needed along with trying all their poses combinations in the area of interest $\binom{n}{2}$. A total of 2^n combinations are required to capture the fingerprint of all possible number of humans and their locations. (c) FP construction in ACE: Only one human is needed to construct the FP regardless of the actual number of humans to be tracked (due to the environment map formulation). A human standing at one location (x) captures the RSS active histogram at this location ($P(s^t \,|\, \alpha^t_x = 1)$) and affects the inactive histograms at all other FP locations, ($P(s^t \,|\, \alpha^t_i = 0, \forall i \neq x)$), (cross-calibration). This leads to two histograms at every FP location.

3.1.2 Spatial Prior Term

This term should favor coherent environment maps, i.e., adjacent locations have similar labels. We adopt a variation of the Ising model commonly used for segmentation applications [4] where the spatial energy term can be represented as

$$\begin{aligned} V^{\mathrm{Sp}}(\mathbb{M}^t, s^t) &= \sum_{\{c_i,c_j\}\in\mathbb{N}} V^{\mathrm{Sp}}_{\{c_i,c_j\}}\left(\alpha^t_{c_i}, \alpha^t_{c_j}, s^t\right) \\ &= \gamma \sum_{\{c_i,c_j\}\in\mathbb{N}, \alpha^t_{c_i}\neq\alpha^t_{c_j}} \left(\frac{1+e^{-\|P(s^t|\alpha^t_{c_i})-P(s^t|\alpha^t_{c_j})\|^2}}{2}\right), \end{aligned} \tag{8}$$

where $\mathbb{N}$ is the set of pairs of neighboring locations. The term $P(s^t \mid \alpha^t_{c_i})$ represents the conditional probability of receiving the signal strength vector s^t when the human is present at location c_i ($\alpha^t_{c_i} = 1$) or not present ($\alpha^t_{c_i} = 0$). This can be estimated during the training phase as described in Section 3.2. Intuitively, the closer the two probabilities, i.e., the more similar the two adjacent locations, the higher the value of $e^{(.)}$, and hence the higher the overall term. The constant γ is a strength parameter for the coherence prior that can be estimated based on the training data.

3.1.3 Likelihood for Signal Strength

The term $U^{\mathrm{SS}}(\mathbb{M}^t, s^t)$ is the log-likelihood of the received signal strength. The term is defined as :

$$U^{SS}(\mathbb{M}^t, s^t) = \delta \sum_{i=1}^{n} \left[-\log P\left(s^t|\alpha^t_i\right)\right], \tag{9}$$

where $\delta < 1$ is a discount factor to allow for multiple counting across non-independent locations whose optimal value is obtained discriminatively from the training data.

RSS likelihoods are learned during the offline training phase as described in the next section.

3.2 Fingerprint Construction

During the offline phase, *ACE* needs to estimate both the RSS likelihood, $P(s^t|\alpha^t_i)$, and the temporal transition priors, $P(\alpha^t_i \,|\, \alpha^{t-1}_i, \alpha^{t-2}_i)$. This is the functionality of the *Fingerprint Builder Module*.

3.2.1 RSS Likelihood

Fig. 5 shows the difference between the fingerprint for a traditional DF system and that of *ACE*. In particular, we use a *cross-calibration* technique, where an entity standing at location x contributes to the *active* RSS likelihoods of x ($P(s^t \,|\, \alpha^t_x = 1)$) and the *inactive* RSS likelihoods of the all remaining $n-1$ FP locations ($P(s^t \,|\, \alpha^t_i = 0, \forall i \neq x)$). This has two advantages: (1) It reduces the coverage sparsity problem in the presence of few streams and (2) it converts the intractable exponential number of cases of building the fingerprint for traditional DF systems [26] to a linear complexity problem, as only one human is needed for training, regardless of the number of humans to be tracked. In summary, at each location, we have two histograms for the RSS corresponding to the active and inactive states respectively using the cross-calibration technique. The active histogram is built only using the readings collected when the human is standing at the same location. The inactive one is built by merging all readings collected when the human is standing at the other $n-1$ locations into one set and using it to build the histogram. The fingerprint is the collection of these two histograms over all locations. We smooth the generated histograms by convolution with separable gaussian kernels to avoid the zero-probability problem of missing values during training.

3.2.2 Temporal Transition Prior

Although there are eight possible transitions (Fig. 6), due to probabilistic normalization ($P(\alpha^t_i = 1 \,|\, \alpha^{t-1}_i, \alpha^{t-2}_i) = 1 - P(\alpha^t_i = 0 \,|\, \alpha^{t-1}_i, \alpha^{t-2}_i)$), the temporal priors have only four degrees of freedom. These temporal priors are learned from the training data.

3.3 Most Probable Map Estimation

In this section, we show how to obtain the optimal environment map by solving the energy-minimization problem in Equation (5) efficiently through mapping it to a binary graph-cut problem. We start by a brief background on graph-cuts, followed by how to map the DF energy-minimization problem to a graph problem.

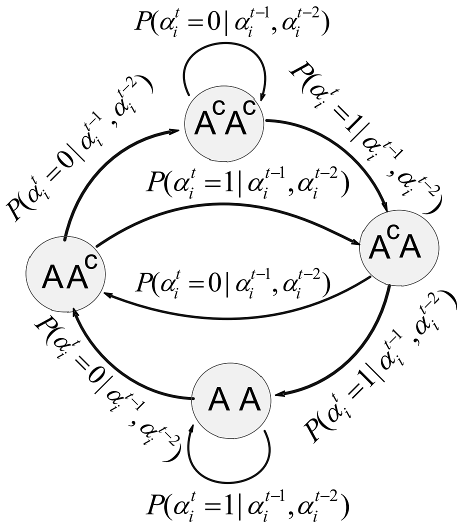

$P(\alpha_i^t=0\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
A^cA^c
$P(\alpha_i^t=0\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
$P(\alpha_i^t=1\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
$P(\alpha_i^t=1\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
AA^c
A^cA
$P(\alpha_i^t=0\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
$P(\alpha_i^t=0\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
$P(\alpha_i^t=1\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$
A A
$P(\alpha_i^t=1\,|\,\alpha_i^{t-1},\alpha_i^{t-2})$

Fig. 6. Finite state diagram for the possible temporal transitions at any location. The sum of arcs originating from any node is one, leading to only four degrees of freedom.

3.3.1 Binary Graph-Cuts

Let $\mathcal{G}=(\mathcal{E},\mathcal{V})$ be a directed graph with nonnegative edge weights that has two special vertices (terminals): the source s and the sink t. An $s-t$-cut (or a binary graph-cut) $C=\{S,T\}$ is a partition of the vertices of $\mathcal{V}$ into two disjoint sets S and T such that $s\in S$ and $t\in T$. The cost of the cut is the sum of costs of all edges that go from S to T:

$$c(S,T)=\sum_{u\in S,v\in T,(u,v)\in\mathcal{E}} c(u,v).$$

The minimum $s-t$-cut problem is to find a cut C with the smallest cost. Ford and Fulkerson [7] proved that this is equivalent to computing the maximum flow from the source to sink. This problem can be solved in a low order polynomial in n [5].[1] This way, a binary graph-cut can be considered as a binary labeling of the graph nodes to be either s or t.

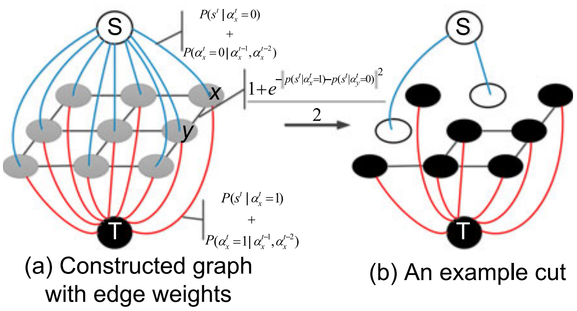
3.3.2 DF Tracking as a Binary Graph-Cut Problem

Not every energy-minimization function can be solved using a graph-cut approach. According to [12], the following theorem gives a necessary and sufficient condition for a function to be solved using the binary min-cut algorithm.

Theorem 1. *Let E be a function of n binary variables in the form of*

$$E(x_1,\ldots,x_n)=\sum_i E^i(x_i)+\sum_{i<j} E^{ij}(x_i,x_j).$$

Then, E is graph-representable if, and only if, each term E^{ij} satisfies the inequality


S
$P(s^t\,|\,\alpha_x^t=0)$
+
$P(\alpha_x^t=0\,|\,\alpha_x^{t-1},\alpha_x^{t-2})$
x
$\frac{1+e^{-\left|p(s^t|\alpha_x^t=1)-p(s^t|\alpha_y^t=0)\right|^2}}{2}$
y
$P(s^t\,|\,\alpha_x^t=1)$
+
$P(\alpha_x^t=1\,|\,\alpha_x^{t-1},\alpha_x^{t-2})$
T
S
T

(a) Constructed graph with edge weights

(b) An example cut

Fig. 7. Mapping the DF energy-minimization problem to a binary graph-cut problem.

$$E^{ij}(0,0)+E^{ij}(1,1)\leq E^{ij}(1,0)+E^{ij}(0,1). \tag{10}$$

Note that the condition only involves the binary terms, i.e.,7 those that involve the relation between two variables. This maps only to the spatial consistency term in our DF energy function (Equation (8)).

Corollary 1. *The DF energy-minimization function is graph-representable.*

Proof. The proof follows directly by mapping the terms of Equations (8) to (10) noting that the LHS of Equation (10) is zero in the DF tracking problem and the two RHS terms are positive. □

The above corollary tells us that we can find a polynomial time efficient solution to the DF energy-minimization problem using the binary graph-cut mapping. Fig. 7 shows how our energy-minimization problem can be mapped to a binary graph-cut problem. We construct a graph that has $n+2$ nodes, where n nodes are the original discrete environment map locations and two additional nodes are added to represent the source and sink nodes. There are two types of edges. Those between the original discrete environment map locations (n-edges) and those between each node and the source and sink terminal nodes (t-edges). The edge weights are assigned in the following way to guarantee that the min-cut solution to this graph is equivalent to minimizing the energy function in Equation (5) [12]:

1) The t-edge between the source and node x is assigned a weight of $P(s^t\,|\,\alpha_x^t=0)+P(\alpha_x^t=0\,|\,\alpha_x^{t-1},\alpha_x^{t-2})$.
2) The t-edge between node x and the sink is assigned a weight of $P(s^t\,|\,\alpha_x^t=1)+P(\alpha_x^t=1\,|\,\alpha_x^{t-1},\alpha_x^{t-2})$.
3) The n-edge (x,y) between node x and node y is assigned a weight of $\frac{1+e^{-\|P(s^t\,|\,\alpha_x^t=1)-P(s^t\,|\,\alpha_y^t=0)\|^2}}{2}$.

Theorem 2. *The binary graph-cut solution on the constructed graph is a solution to the corresponding energy-minimization problem in Equation (5).*

Proof. The proof can be found in the Appendix, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2014.2320265. □

Therefore, any node connected to the source (sink) node after the cut is considered inactive (active).

[1] Note however that generalizations of the minimum s-t-cut problem to involve more than two terminals are NP-hard. We prove in the next subsection that our problem can be mapped to a binary graph-cut problem.

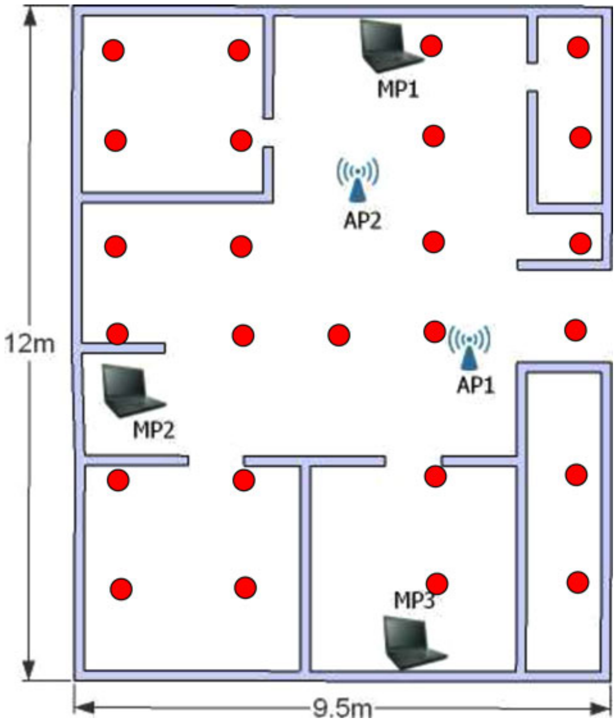


Fig. 8. Testbed 1 layout.

3.4 Computational Complexity

The binary graph-cut algorithm requires $O(n^3)$ operations, where n is the number of fingerprint locations. However, we use the algorithm in [5] as it provides an iterative fast algorithm. Although the algorithm has the same complexity in the worst case, its average complexity is $O(n)$. This has been confirmed in our experiments.

3.5 Discussion

Using the proposed technique, we could reduce the training complexity from $O(2^n)$ to $O(n)$. This is a significant reduction in the calibration overhead which turns the multi-entity tracking problem to a feasible problem.

The proposed framework also treats the detection and tracking problem in a homogenous manner. In particular, detection can be regarded as a special case of the system, where a non-zero estimate of the number of entities is equivalent to a detection event.

4 Multi-Entity Detection and Estimation

The output of the binary graph-cut operation is a set of candidate locations. However, these locations cannot be used directly as a human present at a location typically affects the signal strength at *more than one* neighboring location (Fig. 4) leading to overestimating the actual number of humans and their locations. This effect on neighboring locations decreases as we move away from the actual human location. Therefore, the *Multi-entity Detection and Estimation Module* applies clustering to the output of the binary graph-cut algorithm, such that the number of output clusters determines the number of entities and the center of mass of each cluster gives the coordinates of human corresponding to this cluster. This not only solves the problem of overestimating the number of entities, but also in locating the entities in the continuous space by the weighted averaging of all the samples within a cluster. *This allows ACE to achieve accuracy that is finer than the fingerprint grid spacing*. To further enhance accuracy, we apply clustering to the last w environment maps by merging them into one map.

4.1 Approach

We used a hierarchical clustering technique as it gives us an intuitive means to estimate the number of clusters. In particular, leaf nodes represent individual candidates. Each internal node represents a possible cluster. As we go up in the tree, clusters are combined to form a bigger cluster using euclidean distance between clusters centers as a similarity measure. The root of the tree corresponds to one cluster that contains the entire set of candidate nodes. Starting from the root of the tree, if the degree of inconsistency between two clusters is high, based on a parameter r, we split them as two separate clusters. This process is repeated recursively for each of the split clusters until the degree of inconsistency is below r. The final number of clusters represents the estimated number of humans and the center of mass of each cluster is the estimated human location, where the center of mass is obtained by assigning to each location inside the cluster a probability that is proportional to how many times it appeared in the window w as a candidate location.

4.2 Clustering Complexity

The hierarchical clustering requires $O(c^3)$ operations, where c is the number of candidate locations. Typically, c is $\ll n$. Therefore, clustering has a low overhead. We quantify this effect in Section 5.

5 Performance Evaluation

In this section, we analyze the performance of *ACE* and compare it to the best state-of-the-art DF WLAN localization systems: a probabilistic single-entity system (*Nuzzer*) [26], and two multi-entity tracking systems: (*SPOT*) [25] and (*SCPL*) [34]. We start by describing the experimental setup and data collection. Then, we analyze the effect of different parameters on the system performance.

5.1 Testbeds and Data Collection

We evaluate our system in two different testbeds (Figs. 8 and 9). The first testbed covers a residential apartment with an area of $114\ \mathrm{m}^2$ (about 1,228 sq. ft.) while the second testbed represents an office building with an area of $130\ \mathrm{m}^2$ (about 1,400 sq. ft.). The two testbeds were covered by TP-link TL-WA500G APs and D-Link Airplus G+DWL-650 wireless NICs.

For data collection, we used a sampling rate of one hertz similar to [34]. We had six RSS data streams for both testbeds. A total of 25 fingerprint locations, uniformally distributed over the testbed, are sampled for both testbeds. An *independent test set* at 17 (22) test locations for the first (second) testbed, was collected at different times of day, different locations from the training set, and by different persons. The training set was based on our cross-

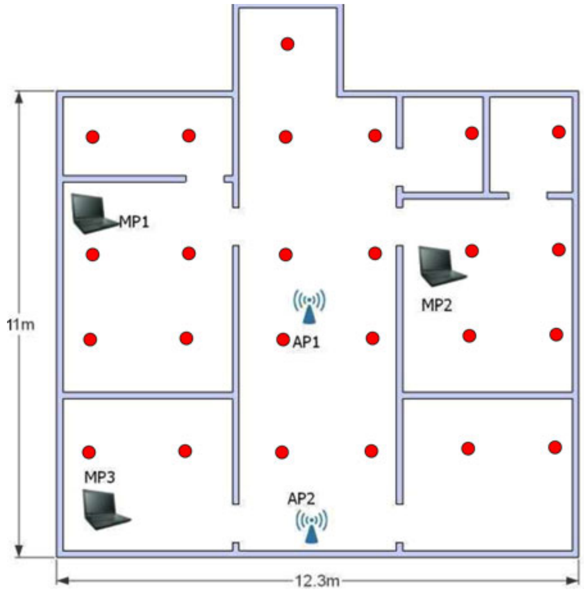


Fig. 9. Testbed 2 layout.

calibration method (Section 3.2), and involved only one man with 1.78 m height and 70 kg weight. While the independent testing set involved other *three different persons* (a 1.64 m height 63 kg weight woman, a 1.55 m height 59 kg weight teenager, and a 1.81 m height 89 kg weight man). In all experiments, persons were moving in their normal real life paths at an average speed of 1.32 meters per second. Other human speeds can be tracked with no problem given the one hertz sampling rate.

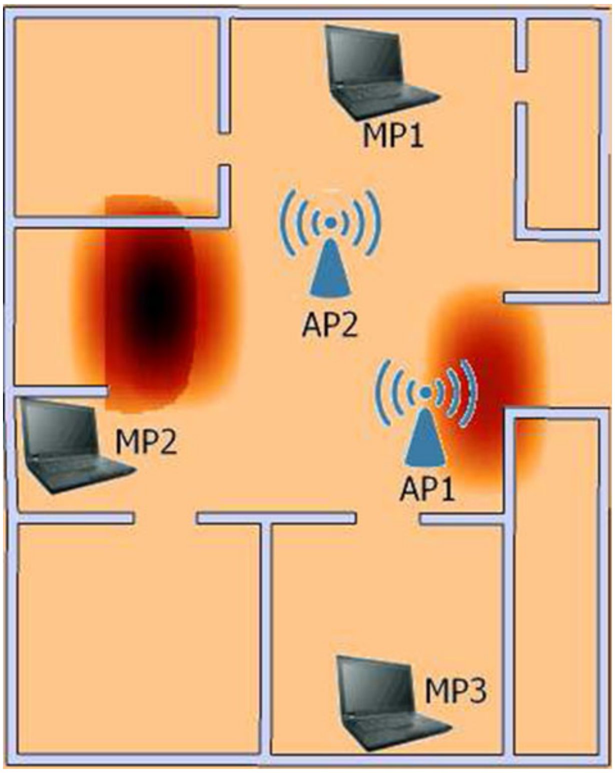


Fig. 10. A heatmap highlighting the system output. Two close entities are present on the left and another entity is present on the right.

TABLE 1
Default Parameters Values

Parameter	Default value	Meaning
k	6	Num. of used streams
n	25	Num. of FP locations
w	13	Clustering window size
o	2	HMM order
r	0.25	Clust. inconsistency thr.

We give the details of the results of the first testbed and summarize the results of the second. Fig. 10 shows an example of the output of the system.

5.2 Parameters Effect

In this section, we study the effect of changing the system parameters on the performance of *ACE*. The average distance error is used as the main metric.

We have adapted a variation of multi-entity tracking metric called OSPA [24] to calculate the distance error, at any time instant t, between the sets of ground truth locations $Q_t = \{q_t^1, q_t^2, \ldots, q_t^g\}$ and the estimated entities' poses $\hat{Q}_t = \{\hat{q}_t^1, \hat{q}_t^2, \ldots, \hat{q}_t^m\}$. The number of ground truth and estimated entities are g and m respectively. For the case of $g \leq m$, the metric is defined as

$$d(Q_t, \hat{Q}_t) = \frac{1}{m}\left(\min_{\pi \in P_m} \sum_{i=1}^{g} d\big(q_t^i, \hat{q}_t^{\pi(i)}\big) + (m-g) * h\right), \quad (11)$$

where P_m includes the set of permutations of length g with elements $\in \{1, 2, \ldots, m\}$, $d(q_t^i, \hat{q}_t^{\pi(i)})$ is the euclidean distance between q_t^i and $\hat{q}_t^{\pi(i)}$. h is a cut-off threshold.[2] In our case, q is the entity's actual location and $\hat{q}$ is the center of her estimated location. For the case $m < g$, the definition in Equation (11) becomes $d(\hat{Q}_t, Q_t)$.

Table 1 shows the default values of the different parameters.

5.2.1 Clustering Window Size (w)

Fig. 11 shows the effect of changing the clustering window size (Section 4) on accuracy. The figure shows that choosing a too short window will degrade the system accuracy. On the other hand, choosing a very long window will increase the latency of the location estimation. However, this will result in better accuracy due to leveraging more information.

An improvement of 33 percent can be achieved between $w = 1$ and $w = 13$, after which the accuracy saturates. Therefore, an application should balance the latency-accuracy tradeoff based on its requirements. Note that the averaging operation performed by the clustering module allows *ACE* to achieve accuracy that is finer than the grid spacing.

5.2.2 Clustering Inconsistency Threshold (r)

Fig. 12 shows that for small values of r, i.e., $r < 0.15$, the system tends to generate one cluster, regardless of the number of entities in the area of interest, underestimating the true number of humans. As r approaches its maximum

[2] We use the length of the diagonal of the area-of-interest as the threshold h.

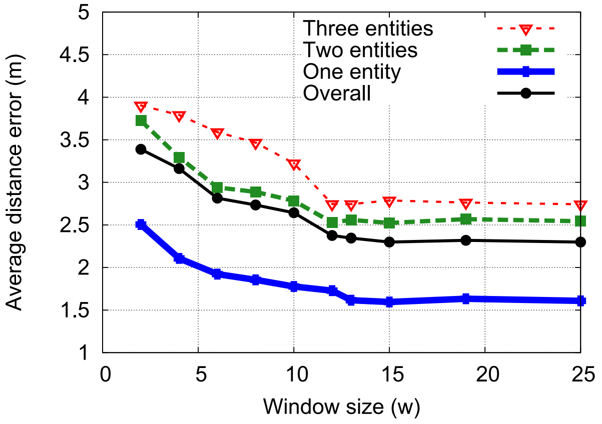


Fig. 11. Effect of changing the clustering window size (w) on accuracy.

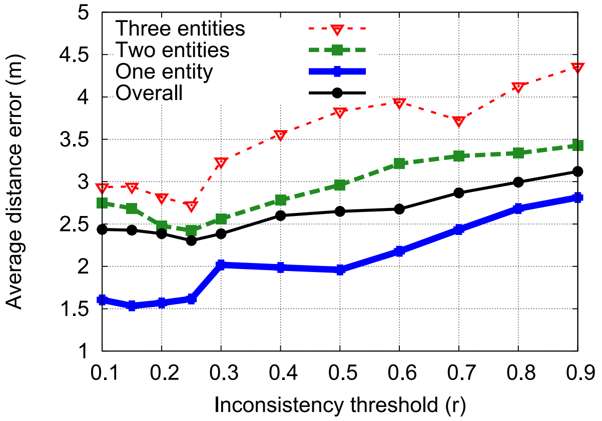


Fig. 12. Effect of changing the clustering inconsistency threshold (r) on accuracy.

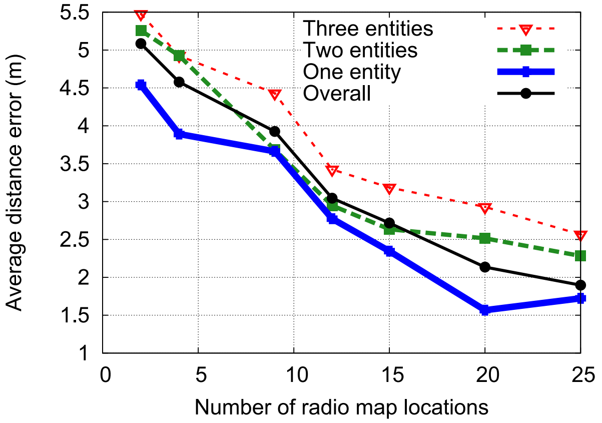


Fig. 13. Effect of changing the fingerprint density (n) on accuracy.

value, i.e., one, the system generates a lot of clusters, overestimating the actual number of humans. This quantifies the advantage of the clustering module. An optimal value for r occurs around 0.25.

5.2.3 *Fingerprint Density* (n)

The denser the fingerprint is, the more accurate results we can achieve (Fig. 13). The curves level out as we approach 25 fingerprint locations, which corresponds to a density of one location every $4.56\ \text{m}^2$.

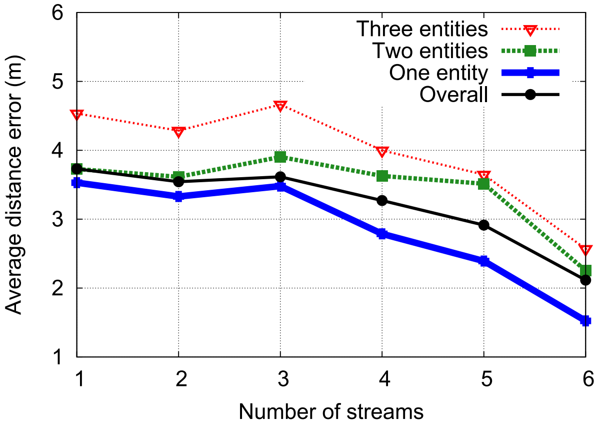


Fig. 14. Effect of changing the number of streams (k) on accuracy.

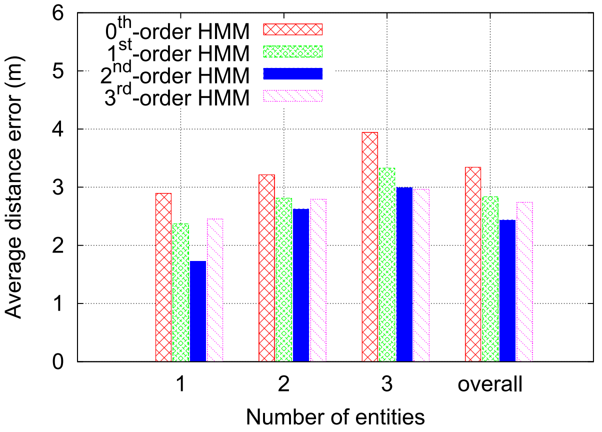


Fig. 15. Effect of changing the HMM order (o) on accuracy.

5.2.4 *Number of Streams* (k)

Fig. 14 shows that, as expected, increasing the number of streams leads to increasing the system accuracy. However, due to the cross-calibration technique employed by *ACE* (Section 3.2), it can tolerate a low number of streams as compared to the other state-of-the-art DF localization systems (as quantified in the next section).

5.2.5 *HMM Order* (o)

Fig. 15 shows that a second order model enhances performance over lower order models by at least 11 percent in overall performance. A third order model does not perform much better than a second order model, with the increase in complexity. Actually, in some cases, e.g., when one entity is present in the area of interest, a third order model performs worse than a second order mode due to over-training. This justifies the use of a second order HMM.

5.3 Comparison with Other DF Systems

5.3.1 *Accuracy*

Fig. 16 shows the CDF of distance error for the single and multiple entities cases of ACE.

Table 2 summarizes the performance of different techniques for the two testbeds. The results show that *ACE* has the best performance under the two testbeds with an enhancement of at least 11.81 percent in median error over

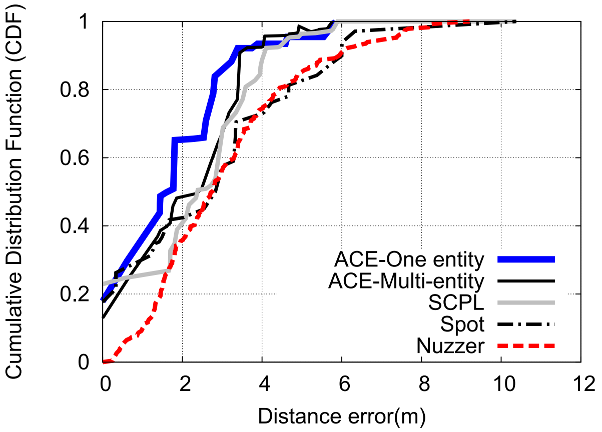


Fig. 16. CDF of distance error for Testbed 1.

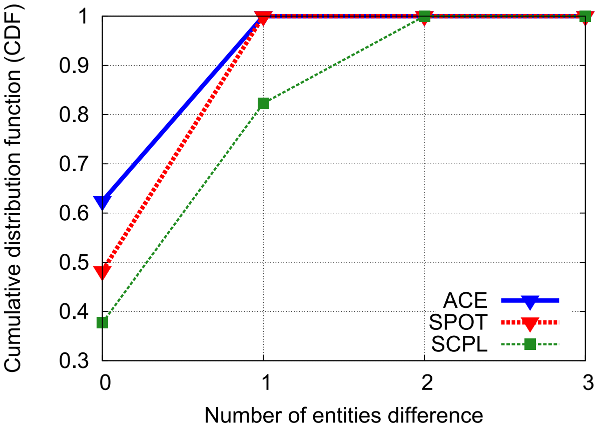


Fig. 17. CDF of num. of entities estimation error for Testbed 1, and a comparison with other systems.

the best state-of-the-art techniques and at least 8.98 percent in average error. In addition, it can perform multi-entity tracking with high accuracy. All techniques perform better in Testbed 2 due to the closer separation of training point in Testbed 2. Note that due to the possibility of estimating the number of entities incorrectly, the performance of the *ACE* single-entity case may not always be better than the *ACE* multi-entity case.

Fig. 17 also shows that *ACE* can estimate the number of entities in the area of interest with at most one difference error.

5.3.2 Running Time

Fig. 18 shows the running time of the different components of *ACE* compared to the other multi-entity techniques: *Spot* [25] and *SCPL* [34]. The results show that the overall *ACE* operations take less 1.9 ms per location estimate for both testbeds. The clustering component consumes the largest time, followed by the min-cut algorithm, and finally calculating the probabilities.

Table 2 summarizes the running time for the different techniques. Although *SCPL* [34] shows a comparable performance to *ACE* in accuracy, it has a higher computational complexity in the number of fingerprint locations. All remaining algorithms (including *ACE*) nearly have the same complexity (as $c \ll n$). However, the running time does differ. This is due to the proportionality constants for the small n and m values in our experiment.

ACE takes higher running time than *SPOT* (less than 31.57 percent on average for both testbeds). However, it significantly outperforms *SCPL* and *Nuzzer*, with at least 49 percent enhancement on average in running time. This highlights that *ACE* significant gain in accuracy and reduction in training overhead comes at a negligible increase in running time.

6 Related Work

Ubiquitous and Context-Aware services are grasping more attention these days. Location is considered a key context information that can be utilized in many ways. A lot of Device-based localization systems have been proposed to provide motion detection and tracking of an entity carrying a device either with the use of special hardware like accelerometers or sensors [3], [23], or by using the existing network infrastructures like wireless networks [11], [15], [31], [40] and GSM [2], [29]. DFP comes with the goal of bypassing the need that the entity being localized to carry a device or even participate actively in the localization process. In DFP, the already installed WiFi settings in the environment is

TABLE 2
Performance Summary for the Different Systems under the Two Testbeds

System	Testbed 1			Testbed 2			Running time Complexity	Multi-entity fingerprint complexity
	Average error	Median error	Running time	Average error	Median error	Running time		
ACE-One ent.	1.72m	1.33m	1.95ms	1.59m	1.43m	1.9ms	$O(n.m + c^3)$	$O(n)$
ACE-Multi-ent.	2.56m (48.83%)	2.11m (58.46%)	2.56ms (31.28%)	2.2m (38.36%)	1.44m (0.69%)	2.4ms (27.3%)		
Spot [25]	2.81m (63.37%)	2.54m (90.97%)	0.435 ms (-77.69%)	2.53m (59.11 %)	1.75m (32%)	0.463ms (-75.63%)	$O(n.m)$	$O(n)$
SCPL [34]	2.79m (62.2%)	2.42m (81.95%)	610ms (311%)	2.42m (52.21%)	1.61m (12.58%)	590ms (309%)	$O(\binom{n}{3})$	$O(n)$
Nuzzer [27]	2.85m (65.71%)	2.77m (108.2%)	3.53ms (81%)	2.48m (55.97%)	1.63m (13.98%)	2.85ms (49.84%)	$O(n.m + n.c)$	$O(2^n)$

The results of ACE-multiple entities are the average of one, two and three entities' results. Number between parenthesis represent percentage of ACE-One entity advantage. c is the number of candidate locations after the graph-cut phase in ACE and first phase of Nuzzer. c is typically $<< n$.

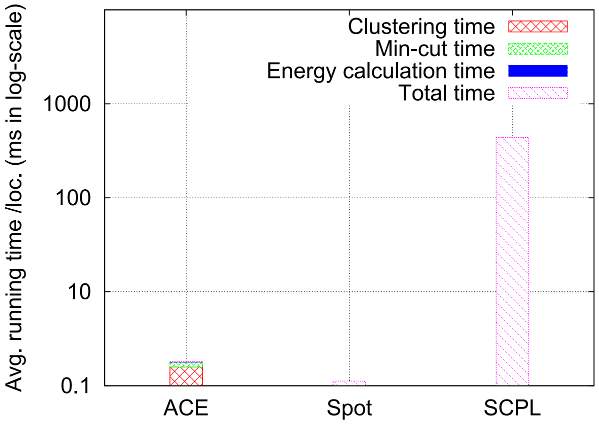


Fig. 18. Running time for the different components of ACE and a comparison with other systems running time.

exploited to estimate the locations of moving objects in the area of interest.

Device-free tracking systems have been introduced over the year including: radar-based [8], [19], [38], camera-based [14], [20], and sensors-based [32].

In the radar-based systems, pulses of radio waves are transmitted into the area of interest and based on measuring the received reflections, objects could be tracked. Several technologies have been presented in this class including ultra-wideband (UWB) systems [38], doppler radar [19], and MIMO radar systems [8].

Camera-based tracking systems are based on analyzing a set of captured images to estimate the current locations of objects of interest [14], [20]. The analysis consists of two main processes: background subtraction and temporal correspondence. However, regular cameras can fail to work in the dark or in the presence of smoke, and they can cause privacy concerns.

Sensor-based systems use especially installed sensor nodes to cover the area of interest. For example, RTI technology [22], [32], [33] applies radio tomographic techniques to the readings of a dense array of sensors to obtain accurate DF tracking. In this method, the relationship between an entity' location and the RSS variation can be mathematically modeled. [32] proposed a linear model to capture the attenuation of RSS values caused by entities when the Line-of-Sight is blocked. In [9], it is shown that the accuracy of RTI can be consistently improved by using channel diversity. Another technique was presented in [10] to provide a real-time RTI-based localization with online re-calibration.

All the technologies above share the requirement of installing special hardware to be able to perform DF tracking, which reduces their scalability in terms of cost and coverage area. In contrast, WLAN DF tracking tries to avoid the previous drawbacks by exploiting the already installed WLANs.

WLAN DF localization was first introduced in [40] along with feasibility experiments in a controlled environment. Several papers followed the initial vision to provide different techniques for detection and tracking [1], [13], [21], [25], [26], [28], [37]. However, all these techniques focus on the problem of a *single entity*.

The closest systems to ours are *Spot* [25] and *SCPL* [34], [35]. *Spot* focuses only on the spatial constraints and does not handle the problem of overestimating the number of entities. In addition, *SCPL* focuses only the temporal transitions in human trajectories and models them as a state transition process. SCPL also uses traditional fingerprinting of a single-entity for counting and localizing multiple entities. *ACE* innovates in the area of multi-entity DF tracking problem by combining temporal an spatial constraints in a unified framework, using a nivel cross-calibration technique, handling RSS and streams outliers, as well as handling the overestimation of the number of entities. Table 3 shows how *ACE* compares to the different systems.

7 Discussion

7.1 Dynamic Changes in the Environment

An important aspect of the practical deployment of DF localization techniques is handling the dynamic changes in the environment. *ACE* currently needs manual re-calibration of the area-of-interest, which can be cumbersome and costly. To reduce this effort, different approaches can be used to capture these dynamic changes including automatic radio map generation using CAD tools, e.g., [6]; leveraging cameras as in [36]; and dynamically updating the stored parameters, e.g., using anomaly detection techniques as in [13]. Other approaches were applied in the context of real-time DF localization to reduce the calibration, e.g., [10].

7.2 Impact of Multiple Entities on RSS Values

The effect of multiple entities on the RSS attenuation is challenging because of the multi-path effect. Briefly, multi-path can cause nonlinear interference when multiple entities coexist in the same radio space. In addition, the RSS changes even when the entity is several meters away from the LoS

TABLE 3
Comparison of Different RF-Based DF Localization Systems

	MIMO Radar-based Systems	**Radio Tomographic Imaging (RTI)**	**Nuzzer System**	**ACE System**
Special hardware required	Yes	Yes	No	No
Number of special nodes	Few	Many	None	None
Number of streams	N/A (echo based)	Large (756)	Small (6)	Small (6)
Covering large areas	Limited by its short range (high frequency)	Limited by the number of deployed nodes (LOS)	Yes	Yes
Computational Complexity	Low	High	Moderate	Low
Accuracy	Very High	Very High	Moderate	High
Affected by muli-path effect	Limited	Yes	Limited (F.print)	Limited (F.print)
Multi-entity tracking	Yes	Yes	No	Yes
Multi-entity overhead	Low	Low	Intractable	Moderate (F.print)

link. Similar challenges have been addressed previously in environments with dense sensors, e.g., [32], [34] and can be used with *ACE*.

7.3 New Crowd-Based Applications

Traditional application scenarios for DF localization systems typically assume that the area of interest is silent (i.e., has no activities and no entities). In our experiments, RSS readings have been acquired when there were people moving in the same floor where *ACE* was deployed, but not in the experiment area.

We believe, however, that when the crowds are close to each other in the area of interest, they could form a big high likelihood area in the reconstructed fingerprint map. When the crowds are scattered, their shadowing effects on streams will be different and they could form lots of high likelihood areas in the reconstructed fingerprint map. The reconstructed fingerprint map could represent the distribution of the crowd. This can enable a new set of device-free applications, including indoor analytics and crowd management.

8 Conclusions

We presented the design, analysis, and implementation of *ACE*: a system for accurate and efficient multi-entity device-free WLAN localization. *ACE* leverages probabilistic techniques to provide a smooth and consistent environment image. It uses a cross-calibration technique and an energy-minimization framework to reduce the calibration overheard to linear in the number of locations, which turns the DF multi-entity tracking to a tractable problem. We showed that the selected energy-minimization terms lead to an efficient solution by mapping the energy function to a binary graph-cut problem. We further showed how to perform clustering on the generated environment map to remove outliers and enhance accuracy.

Implementation on standard WiFi hardware in two different testbeds show that *ACE* can achieve 1.3 m median distance multi-entity tracking error, which is better than the stat-of-art techniques by at least 11.8 percent, and up to 33 percent, in both testbeds. In addition, it can estimate the number of entities correctly to within one entity difference 100 percent of the time. This highlights the promise of *ACE* for a wide range of multi-entity DF tracking applications.

Currently, we are expanding ACE in multiple directions including robustness to environment changes, entity identification, and automating the construction of the fingerprint.

References

[1] H. Abdel-Nasser, R. Samir, I. Sabek, and M. Youssef, "MonoPHY: Mono-stream-based device-free WLAN localization via physical layer information," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2013, pp. 4546–4551.

[2] I. Anderson and H. Muller, "Context awareness via GSM signal strength fluctuation," in *Proc. 4th Int. Conf. Pervasive Comput.*, 2006, pp. 27–31.

[3] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Proc. Int. Conf. Pervasive Comput.*, 2004, vol. 3001, pp. 1–17.

[4] Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images," in *Proc. IEEE 8th Int. Conf. Comput. Vis.*, 2001, pp. 105–112.

[5] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[6] A. Eleryan, M. Elsabagh, and M. Youssef, "Synthetic generation of radio maps for device-free passive localization," in *Proc. Global Telecommun. Conf.*, 2011, pp. 1–5.

[7] L. Ford and D. Fulkerson, *Flows in Networks*. Princeton, NJ, USA: Princeton Univ. Press, 1962.

[8] A. M. Haimovich, R. S. Blum, and L. J. Cimini., "MIMO radar with widely separated antennas," *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 116–129, 2008.

[9] O. Kaltiokallio, M. Bocca, and N. Patwari, "Enhancing the accuracy of radio tomographic imaging using channel diversity," in *Proc. IEEE 9th Int. Conf. Mobile Ad hoc Sensor Syst.*, 2012, pp. 254–262.

[10] O. Kaltiokallio, M. Bocca, and N. Patwari. (2012). Follow@-grandma: Long-term device-free localization for residential monitoring *Proc. IEEE 37th Conf. Local Comput. Netw. Workshops*, pp. 991–998 [Online]. Available: http://dblp.uni-trier.de/db/conf/lcn/lcnw2012.html#KaltiokallioBP12

[11] K. Kleisouris, B. Firner, R. Howard, Y. Zhang, and R. P. Martin, "Detecting intra-room mobility with signal strength descriptors," in *Proc. 11th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2010, pp. 71–80.

[12] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 147–159.

[13] A. E. Kosba, A. M. Saeed, and M. Youssef, "RASID: A robust WLAN device-free passive motion detection system," in *PerCom 2012*, pp. 180–189, 2012.

[14] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-camera multi-person tracking for easyliving," in *Proc. IEEE 3rd Int. Workshop Visual Surveillance*, 2000, pp. 3–10.

[15] J. Krumm and E. Horvitz, "LOCADIO: Inferring motion and location from wi-fi signal strengths," in *Proc. 1st Annu. Int. Conf. Mobile Ubiquitous Syst.: Netw. Serv.*, 2004, pp. 4–13.

[16] S. Kumar and M. Hebert, "Discriminative random fields: A discriminative framework for contextual interaction in classification," in *Proc. IEEE 9th Int. Conf. Comput. Vis.*, 2003, vol. 2, pp. 1150–1157.

[17] J. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn.*, 2001, pp. 282–289.

[18] A. Lin and H. Ling, "Doppler and direction-of-arrival (DDOA) radar for multiple-mover sensing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, pp. 1496–1509, Oct. 2007.

[19] T. B. Moeslund, A. Hilton, and V. Krger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understanding*, vol. 104, nos. 2/3, pp. 90–126, 2006.

[20] M. Moussa and M. Youssef, "Smart devices for smart environments: Device-free passive detection in real environments," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2009, pp. 1–6.

[21] N. Patwari and J. Wilson, "RF sensor networks for device-free localization: measurements, models, and algorithms," *in Proc. IEEE*, vol. 98, no. 11, pp. 1961–1973, Nov. 2010.

[22] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel, "Inferring activities from interactions with objects," *IEEE Pervasive Comput.*, vol. 3, no. 4, pp. 50–57, Oct. 2004.

[23] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3452–3457, Jul. 2011.

[24] I. Sabek and M. Youssef, "Multi-entity device-free WLAN localization," in *Proc. IEEE Global Commun. Conf.*, 2012, pp. 2018–2023.

[25] I. Sabek and M. Youssef. (2012). SPOT demo: Multi-entity device-free WLAN localization *Proc. 7th ACM Int. Workshop Wireless Netw. Testbeds, Exp. Eval. Characterization*, pp. 87–88 [Online]. Available: http://doi.acm.org/10.1145/2348688.2348708

[26] M. Seifeldin, A. Saeed, A. Kosba, A. El-Keyi, and M. Youssef, "Nuzzer: A large-scale device-free passive localization system for wireless environments," *IEEE Trans. Mobile Comput.*, vol. 12, no. 7, pp. 1321–1334, Jul. 2013.

[27] M. Seifeldin and M. Youssef, "A deterministic large-scale device-free passive localization system for wireless environments," in *Proc. 3rd Int. Conf. Pervasive Technol. Related Assistive Environ.*, 2010, pp. 1–8.

[28] T. Sohn, A. Varshavsky, A. Lamarca, M. Y. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. G. Griswold, and E. D. Lara, "Mobility detection using everyday gsm traces," in *Proc. 8th Int. Conf. Ubiquitous Comput.*, 2006, pp. 212–224.

[29] Sutton, Charles, Mccallum and Andrew, *Introduction to Conditional Random Fields for Relational Learn*. Cambridge, MA, USA: MIT Press, 2006.

[30] M. Wallbaum and S. Diepolder, "A motion detection scheme for wireless LAN stations," in *Proc. 3rd Int. Conf. Mobile Comput. Ubiquitous Netw.*, 2006.

[31] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 5, pp. 621–632, May 2010.

[32] J. Wilson and N. Patwari, "See-through walls: Motion tracking using variance-based radio tomography networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 5, pp. 612–621, May 2011.

[33] C. Xu, B. Firner, R. S. Moore, Y. Zhang, W. Trappe, R. Howard, F. Zhang, and N. An. (2003). "SCPL: Indoor device-free multi-subject counting and localization using radio signal strength" in *Proc. 12th Int. Conf. Inf. Process. Sens. Netw.*, pp. 79–90 [Online]. Available: http://doi.acm.org/10.1145/2461381.2461394

[34] C. Xu, B. Firner, Y. Zhang, R. Howard, and J. Li, "Trajectory-based indoor device-free passive tracking," in *Proc. 2nd Int. Workshop Mobile Sens.*, 2012.

[35] C. Xu, M. Gao, B. Firner, Y. Zhang, R. Howard, and J. Li. (2012). "Towards robust device-free passive localization through automatic camera-assisted recalibration," *Proc. 10th ACM Conf. Embedded Netw. Sensr. Syst.*, pp. 339–340 [Online]. Available: http://doi.acm.org/10.1145/2426656.2426697

[36] J. Yang, Y. Ge, H. Xiong, Y. Chen, and H. Liu, "Performing joint learning for passive intrusion detection in pervasive wireless environments," in *Proc. IEEE 29th Conf. Comput. Commun.*, 2010, pp. 1–9.

[37] Y. Yang and A. E. Fathy, "See-through-wall imaging using ultra-wideband short-pulse radar system," in *Proc. IEEE Antennas Propag. Soc. Int. Symp*, 2005, pp. 334–337.

[38] M. Youssef and A. Agrawala, "Small-scale compensation for WLAN location determination systems," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Mar. 2003, pp. 1974–1978.

[39] M. Youssef, M. Mah, and A. Agrawala, "Challenges: device-free passive localization for wireless environments," in *Proc. 13th Annu. ACM Int. Conf. Mobile Comput. Netw.*, 2007, pp. 222–229.

[40] M. A. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. ACM 3rd Int. Conf. Mobile Syst., Appl., Serv.*, 2005, pp. 205–218.

Ibrahim Sabek received the BSc degree in computer systems and engineering from Alexandria University, Egypt, in 2010, and is currently working toward the MSc degree in computer science at the same university. He is a teaching assistant at Alexandria University. He spent 2 years with Microsoft Research Lab in Cairo as a research assistant from August 2010 till June 2012. In November 2011, he joined the Wireless Research Facility as a research assistant at the Egypt-Japan University of Science and Technology (E-JUST), Egypt. He is the winner of Best Undergraduate Research Award 2009 from the Exchange of Students Program between IAESTE Germany and Alexandria University. He is an invited reviewer of the *IEEE Wireless Communications* and Signal Processing Communities. His research interests include location determination systems, machine learning, and computer vision. He is a student member of the IEEE.

Moustafa Youssef received the BSc and MSc degrees in computer science from Alexandria University, Egypt, in 1997 and 1999, respectively, and the PhD degree in computer science from the University of Maryland in 2004. He is an associate professor at Alexandria University and the Egypt-Japan University of Science and Technology (E-JUST), Egypt. His research interests include mobile wireless networks, mobile computing, location determination technologies, pervasive computing, and network security. He has eight issued and pending patents. He is an associate editor for the *ACM Transactions on Spatial Algorithms and Systems*, an area editor of the *ACM SIGMOBILE Mobile Computing and Communications Review*, and served on the organizing and technical committees of numerous conferences. He received the the 2003 University of Maryland Invention of the Year Award, the 2010 TWAS-AAS-Microsoft Award for Young Scientists, the 2012 Egyptian State Award, the 2014 COMESA Innovation Award, among others. He is also an ACM distinguished speaker. He is a senior member of the IEEE.

Athanasios V. Vasilakos is currently a professor at the University of Western Macedonia, Greece. He has authored or coauthored more than 200 technical papers in major international journals and conferences. He is the author/coauthor of five books and 20 book chapters in the areas of communications. He served as the general chair, technical program committee chair, and the TPC member for many international conferences. He was an editor or/and guest editor for many technical journals, such as the *IEEE Transactions on Network and Service Management*, *IEEE Transactions on Systems, Man, and Cybernetics*-Part B: Cybernetics, *IEEE Transactions on Information Technology in Biomedicine*, *IEEE Transactions on Computers*, *ACM Transactions on Autonomous and Adaptive Systems*, and the *IEEE Journal on Selected Areas in Communications* special issues of May 2009, January 2011, and March 2011, the *IEEE Communications Magazine*, ACM/Springer *Wireless Networks*, and the ACM/Springer *Mobile Networks and Applications*. He is the founding editor-in-chief of the *International Journal of Autonomous and Adaptive Communications Systems* and the *International Journal of Arts and Technology*. He is the general chair of the Council of Computing of the European Alliances for Innovation. He is a senior member of the IEEE.