Fusion of Radio and Camera Sensor Data for Accurate Indoor Positioning

Savvas Papaioannou, Hongkai Wen, Andrew Markham and Niki Trigoni

Department of Computer Science, University of Oxford, Oxford, OX1 3QD, UK
{firstname.lastname}@cs.ox.ac.uk

Abstract—Indoor positioning systems have received a lot of attention recently due to their importance for many location-based services, e.g. indoor navigation and smart buildings. Lightweight solutions based on WiFi and inertial sensing have gained popularity, but are not fit for demanding applications, such as expert museum guides and industrial settings, which typically require sub-meter location information. In this paper, we propose a novel positioning system, RAVEL (Radio And Vision Enhanced Localization), which fuses anonymous visual detections captured by widely available camera infrastructure, with radio readings (e.g. WiFi radio data). Although visual trackers can provide excellent positioning accuracy, they are plagued by issues such as occlusions and people entering/exiting the scene, preventing their use as a robust tracking solution. By incorporating radio measurements, visually ambiguous or missing data can be resolved through multi-hypothesis tracking. We evaluate our system in a complex museum environment with dim lighting and multiple people moving around in a space cluttered with exhibit stands. Our experiments show that although the WiFi measurements are not by themselves sufficiently accurate, when they are fused with camera data, they become a catalyst for pulling together ambiguous, fragmented, and anonymous visual tracklets into accurate and continuous paths, yielding typical errors below 1 meter.

I. INTRODUCTION

In the last five years we have witnessed an unprecedented interest in indoor positioning technologies, with a variety of solutions developed in academic and industrial research labs. Although the field has reached a significant level of maturity there is still no dominant solution. As a consequence, positioning services are still lacking in many buildings. For a solution to be widely implemented and adopted it must satisfy two key requirements: low cost and high accuracy. Current low cost systems exploit or reuse existing infrastructure (e.g. WiFi) or can even operate without infrastructure (e.g. inertial tracking). These inexpensive, ubiquitous positioning approaches however cannot provide the sub-metre accuracy that high-end, expensive solutions like UWB or ultrasound can. The dichotomy between cost and accuracy has fragmented the technology landscape, leading to a plethora of competing solutions that cannot satisfy both requirements simultaneously. We present a fresh approach that seeks to unify the two disparate camps, providing high positioning accuracy with very low cost.

Our fundamental observation is that many applications that require high positioning accuracy are in large public or commercial spaces, such as airports, shopping centres, museums and industrial plants. These large public spaces are typically extensively covered with CCTV cameras for reasons of safety and security. In this paper, we propose the use of existing camera infrastructure for the originally unintended task of indoor positioning. However, although camera based tracking can provide excellent position information in ideal conditions, the challenges provided by a real deployment are numerous. In particular, most security cameras are installed to provide a large field of view, typically resulting in a bird's eye view of the scene. This top-down perspective makes it difficult to distinguish facial features and accurate identification is made even more challenging when the room is not well lit (e.g. in museums) or when people wear similar uniforms or helmets (e.g. in industrial plants). Furthermore maintaining tracking as people move behind obstacles, exit the field of view, or cross paths is an exceedingly difficult task.

Instead of trying to overcome the limitations of camera-based tracking through increasingly sophisticated tracking algorithms, we exploit opportunistic and ubiquitous radio signals (e.g. WiFi/Bluetooth Low Energy). Although, radio-based tracking is limited in terms of accuracy, it can be used to add context to trajectories obtained from camera based tracking. At a coarse level, this provides an identifier, which can be used instead of advanced video processing techniques like face recognition. At a finer level, the sequence of radio signal strengths, albeit noisy, can be used to disambiguate between multiple possible trajectories, aiding to merge and split discontinuous traces.

Motivated by the noisy nature of both visual and non-visual sensor modalities, we explore whether we can effectively combine them to overcome each other's weaknesses. We present a generic vision+radio tracking framework, RAVEL (RAdio and Vision Enhanced Localization) that can be used both in receiver-centric (i.e. people carrying smartphones) and transmitter-centric (i.e. low-cost tags in warehouse, industrial or construction sites) applications.

So far, there have been very few attempts at combining infrastructure cameras with radio for positioning. The most relevant work, EVLoc [1], which also fuses radio and camera sensor data, tackles a different problem, that of matching N wireless devices with N visual detections, rather than inferring the location of a device, using only that device's radio data and anonymous camera data from the scene. Unlike our work, they assume a known, calibrated radio model, and have performed tests in rather ideal conditions. A complementary approach is to fuse inertial

measurements with visual trajectories [2], [3], but we argue that radio measurements are the fundamental primitive that is common to many applications ranging from smartphone positioning to tracking low power tags in a warehouse; inertial data is more informative, but less ubiquitous and requires application specific signal processing to yield usable motion traces.

To the best of our knowledge, this is the first paper that proposes a practical solution of radio-aided visual tracking, and tests it in a truly complex and realistic scenario. Specifically, our contributions are:

- We provide a fresh perspective on the problem of low-cost high-accuracy positioning in large open-plan indoor spaces, enabled by the fusion of anonymous visual data and radio data.
- We design a novel multi-hypothesis probabilistic approach (RAVEL) to fuse radio and camera data that is robust to noisy and incomplete measurements.
- We show how the radio propagation model for a particular environment can be learnt online, requiring no site-specific surveying.
- We evaluate the proposed approach in a museum setting, and compare it with multi-hypothesis tracking which only uses visual data.

The remainder of this paper is structured as follows: Sec. II outlines the requirements of a system for fusing visual and radio data to accurately track users in indoor environments. Sec. III presents the overview of our radio-aided visual tracking system, while Sec. IV discusses the details of the proposed algorithms. Sec. V evaluates the proposed system in a real-world museum scenario. Sec. VI overviews related work and Sec. VII concludes the paper and discusses future work.

II. SYSTEM REQUIREMENTS

Here we lay out the key requirements and challenges for a practical visual- and radio-based positioning solution.

Lightweight: The proposed solution should be lightweight enough to run in real-time on resource-constrained devices with minimum training effort. This excludes computationally-expensive approaches, such as part-based object detectors (e.g. [4]), or face recognition techniques.

Accurate: We aim for *sub-meter* tracking accuracy for a particular user, as required by many applications. Existing radio-based positioning approaches typically cannot achieve such accuracy, while visual tracking techniques tend to generate *anonymous* and often segmented trajectories that cannot be assigned to a particular user.

Robust: The proposed solution should be usable in busy and dynamic indoor environments, and robust to noisy visual / radio data. In practice, fixed obstacles such as furniture may easily block line of sight to both the camera and the radio access points, resulting in occlusions and attenuation respectively. Changes in illumination and arbitrary movements of people can also have catastrophic effects on approaches relying on perfect visual detections.
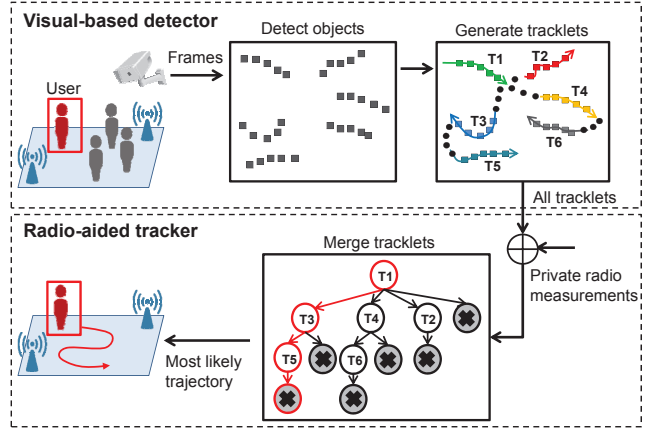

Fig. 1: Overview of the proposed RAVEL system.

III. SYSTEM OVERVIEW

We are now in a position to present the proposed Radio And Vision Enhanced Localization (RAVEL) system, which consists of two components: a *visual based detector* and a *radio aided tracker* shown in Fig. (1). These components can run in the same device (e.g. a server collecting measurements from multiple transmitters) or across different devices (e.g. smart cameras running the visual-based detector and disseminating traces to mobile devices, each running their own radio-aided tracker to preserve user privacy). The rest of the paper considers the latter, more challenging case of distributed tracking, but the algorithms presented are equally applicable to centralized tracking.

A. The Tracking Problem

Let us assume the indoor environment is monitored by a calibrated (known extrinsic and intrinsic parameters) stationary camera (the proposed approach can be easily extended to multi-camera scenarios). For a given time window of size W, $W \in \mathbb{Z}$, the camera captures a series of frames $[f_1, ..., f_W]$ within its field of view (FOV). We assume that each frame f_i contains a number of camera detections of moving objects within the FOV, denoted as $C_i = \{c_i^1, ..., c_i^j, ...\}$, $1 \leq i \leq W$, $1 \leq j \leq |C_i|$. A camera detection c_i^j is represented as a bounding box of the detected object, or simply as the coordinates of the center of the bounding box, i.e. $c_i^j = (x_i^j, y_i^j)$.

We also assume that at each time i, the mobile device carried by a particular user *receives* a set of radio measurements $r_i = \{r_i^1, ..., r_i^m, ...\}$, where r_i^m is the Received Signal Strength (RSS) measurement of the m-th radio basestation at time i. The problem is is to *estimate the trajectory of a user given the sequence of anonymous camera detections* $[c_1, ..., c_W]$ *and personal radio measurements* $[r_1, ..., r_w]$.

B. Visual-based Detector

The visual-based detector processes the captured camera footage as follows. For each frame f_i, a set of anonymous camera detections $C_i = \{c_i^1, ..., c_i^j, ...\}$ of the interesting
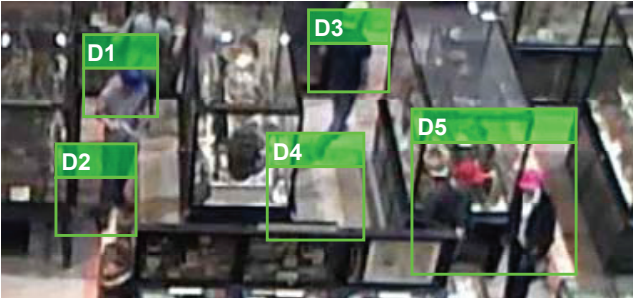
Fig. 2: Three cases of noisy detections generated by our visual-based detector: a) multiple detections are generated for one moving target (D1 and D2), b) a detection contains no moving targets at all (D4), and c) one detection contains multiple moving targets (D5).

objects (i.e. moving people) are firstly extracted. In our implementation, we use a lightweight MoG-based background subtraction approach [5] to detect moving people, which does not require heavy training and with some optimization it can run in real-time in embedded camera networks [6]. However, the computed set of detections C_i can be very *noisy* and *unreliable*. Fig. (2) shows a frame with three typical cases of noisy detections observed in our experiments, namely multiple detections from a single person (splitting); single detections corresponding to multiple people (merging); and null detections (empty bounding box, caused by a visual artifact such as a moving shadow). In this case, linking the detections into coherent trajectories is not a trivial task. Note that the visual-based detector only links detections into short trajectory segments (referred to as *tracklets* hereafter) when they unambiguously belong to the same target, as discussed in Sec. IV-A. The tracklets could be broadcasted to phones in a number of ways, including overloading the WiFi beacon frames (i.e. beacon stuffing) [7], a technique that could also be used to preserve the user's privacy.

C. Radio-aided tracker

The tracklets generated by the visual-based detector tend to be very short, and are still *anonymous*: we have no knowledge of which tracklet should belong to a given user. Therefore, the proposed system broadcasts the generated tracklets (which are just streams of two dimensional coordinates) to the radio-aided tracker, which runs on the mobile devices carried by the users. With the received anonymous tracklets, the radio-aided tracker cross references the user's private radio footprint, i.e. radio measurements from the known basestations, to determine which tracklets should be selected and merged to produce the complete trajectory of this particular user. To achieve this, our tracker uses a multi-hypothesis tracklet merging approach. In a nutshell, it builds *tracklet trees* that encode all hypotheses of user paths. We are now in a position to present the details of the proposed algorithms.

IV. PROPOSED ALGORITHM

In this section, we discuss the core algorithms of the RAVEL tracking system: the *tracklet generation algorithm* (Sec. IV-A) used in the visual-based detector, and the *tracklet merging algorithm* (Sec. IV-B), which is the key competency of our radio-aided tracker.

A. Tracklet Generation Algorithm

Given a fixed window of frames $[f_1, ..., f_W]$, and extracted sets of camera detections $[C_1, ...C_W]$, the task of the tracklet generation algorithm is to link the elements of C_i-s into unambiguous trajectory segments, referred to as tracklets. The proposed algorithm leverages a realistic model of human motion to ascertain that a sequence of detections from consecutive frames belong to the same person with high certainty, and can thus be grouped together into a tracklet τ. It starts by considering detections in the first frame and proceeds in chronological order until all detections are grouped into a set of tracklets $\mathcal{T} = \{\tau_1, ..., \tau_N\}$.

More specifically, at the beginning, or when an existing tracklet cannot be further extended with new camera detections, a new tracklet is initiated with the first available (not-visited) camera detection, say c_i. To extend the tracklet with a second detection, we first check if there is a detection c_{i+1} in the next frame such that $\|c_i - c_{i+1}\| <$ DT where DT is the maximum displacement of a target in the period between two consecutive frames. If there is no such detection, or if there are more than one such detections, c_i becomes a singleton tracklet. Otherwise the two detections must belong to the same person, so they are grouped together in the same tracklet.

Once we have at least two detections in the same tracklet, we can extend it with another detection taking into account the fact that humans normally avoid abrupt changes of direction and speed [8]. For example, consider the last two detections in the tracklet (say, c_i and c_{i+1}), and the potential to extend them with one of the available detections in the next frame, say c_{i+2}. In order to assess its suitability, we measure changes in the direction and in the speed of motion, and combine them to obtain a metric of confidence that the three detections concern the same person:

$$Q(c_{i+2}; c_i, c_{i+1}) = w_d Q_d + w_s Q_s \qquad (1)$$

where Q_d is the cost of direction change while Q_s is the cost of speed change, weighted by w_d and w_s respectively. In our case, Q_d and Q_s are defined as:

$$\begin{aligned} Q_d &= 1 - \frac{(c_{i+1} - c_i) \cdot (c_{i+2} - c_{i+1})}{\|c_{i+1} - c_i\|\|c_{i+2} - c_{i+1}\|} \\ Q_s &= 1 - 2\frac{\sqrt{\|c_{i+1} - c_i\|\|c_{i+2} - c_{i+1}\|}}{\|c_{i+1} - c_i\| + \|c_{i+2} - c_{i+1}\|} \end{aligned} \qquad (2)$$

If at most one detection is found with cost Q less than a predefined threshold then we include that detection in the tracklet and continue to the next frame; otherwise we stop extending this particular tracklet. The above procedure is repeated until all detections are grouped into tracklets.
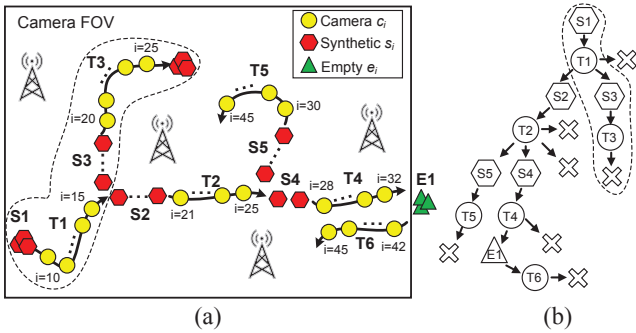
Fig. 3: (a) The tracklets generated by our algorithm, where the gaps are filled by synthetic and empty detections. (b) The built tracklet tree with nodes containing camera (circle nodes), synthetic (hexagon nodes), and empty (triangle nodes) detections. Each camera node is also attached with a EOT node.

B. Tracklet Merging Algorithm

Given the generated tracklets, the tracklet merging algorithm, which is the core of our radio-aided tracker, attempts to decide which tracklets should be merged together to produce the complete trajectory of the particular user. It first builds a set of tracklet trees that encode all possible trajectory hypotheses, and then searches for the trajectory hypothesis that is most consistent with the user's radio data. **Build the tracklet trees:** Let us denote the set of tracklets generated within a window of frames $[f_1, ..., f_W]$ as $\mathcal{T} = \{\tau_1, ..., \tau_N\}$. We first define $\mathcal{T}_{start}$ as the subset of tracklets in $\mathcal{T}$ that start early (before a certain time threshold) and thus could be used to start a user's trajectory. For each of those candidate tracklets, the algorithm initiates a tracklet tree. The algorithm then expands these trees as follows: for a given tree node (parent tracklet), its children become all the tracklets that start soon after it finishes and for which their first detection is at close proximity to the parent tracklet's final detection. This spatio-temporal threshold constrains the size of the tracklet trees and can be adjusted to allow this module to run in real-time on mobile devices.

Now the tracklets in $\mathcal{T}$ are topologically connected into paths, but these paths are still incomplete. For example, consider tracklets T1 and T3 in Fig. (3a), which form a path in the tree shown in Fig. (3b) (ignore the nodes other than T1 and T3 for now). For this path, we can see that: a) it does not start from the beginning of time window (T1 starts at time 10); b) it ends early (T3 ends at time 25 while the window size is 45); and finally c) there is a gap of missing detection in between (the final detection in T1 is at time 15, and the first detection in T3 is at 20).

To address this, we introduce two extra types of detections, *synthetic detections* $s_i \in S$ and *empty detections* $e_i \in E$, in addition to the *camera detections* $c_i \in C$. The synthetic detections are used to address the problem of missing camera detections when the user is actually within the camera's FOV, which are caused by occlusions or

pauses of the user. On the other hand, the empty detections are introduced to account for situations where the user exits and possibly reenters the camera's FOV, and temporarily has no known coordinates.

Concretely, our algorithm completes paths with synthetic and empty detections according to the following rules: *First tracklet rule:* If the first tracklet of a path does not start at time 1 (of the window), we precede it with empty detections if it starts at the boundary of the scene, or with synthetic detections (positioned at the location of its first detection) if it does not start at the boundary. *Gap rule:* If there is a gap between a parent and a child tracklet in the tree, we fill the gap with synthetic detections positioned at interpolated points between the final detection of the parent tracklet and the first detection of the child tracklet. In the unusual case that the parent tracklet ends at the boundary of the FOV and the child tracklet begins at the boundary, we fill the gap with empty (instead of synthetic) detections. *Last tracklet rule:* If the last tracklet of a path does not end at the end of the window, we extend it with empty detections if it ends at the boundary of the scene, or with synthetic detections (positioned at the location of its last detection) if it does not end at the boundary. We also add an end of trajectory (EOT) node to every node containing camera detections, indicating the user may stop there.

After applying these rules, every path of the built tracklet tree represents a possible trajectory that contains W consecutive detections whether camera, synthetic or empty. For example, the highlighted path in Fig. (3b) corresponds to the highlighted trajectory in Fig. (3a), which indicates that the user moves from the bottom left corner of the FOV to the top middle. We refer to such a trajectory as a *hypothesis*, denoted as $H = [h_1, ...h_W]$, where $h_i \in C \cup S \cup E$. **Search the trackless trees:** Once we have identified all trajectory hypotheses, we proceed to identify the most likely one. Specifically, the likelihood score of a hypothesis is defined as the sum of two parts:

$$L(H, R, \lambda) = L^v(H) + L^r(H, R, \lambda) \qquad (3)$$

where $L^v(H)$ is the *visual-based likelihood score* of the hypothesis H, while $L^r(H, R, \lambda)$ is the *radio-based likelihood score* given observed radio measurements R and a radio propagation model λ. Therefore, our algorithm evaluates the likelihood score of a hypothesis in two steps: 1) it firstly estimates the user trajectory $X = [x_1, ..., x_W]$ based on vision data in hypothesis H, and evaluates the likelihood of vision data; 2) it then evaluates the likelihood of the radio data given the estimated user trajectory X. The first step is already applied by existing multiple hypothesis tracking approaches (e.g. [9], [10]), which neglect radio data and select the hypothesis that maximizes $L^v(H)$ only. We could of course derive the overall likelihood score in one step, using a Bayesian filter to jointly fuse vision and radio data. Our design choice to separate these steps stems from our desire to reuse existing implementations of vision-only trackers and extend them flexibly with a new step that additionally exploits radio data. Now we explain how $L^v(H)$ and $L^r(H, R, \lambda)$ are evaluated in detail.

Evaluate $L^v(H)$: For each hypothesis H, our algorithm maintains a filter, to estimate the trajectory X of the user. This can be implemented using a variant of a Bayesian filter, such as a Kalman filter [11]. Let x_i^- be the estimated position (state) of the user given the detections $h_{1:i-1}$, with covariance P_i^-. For a non-empty detection h_i, we define its incremental visual-based score as:

$$\Delta L_i^v = \begin{cases} \log[p_v f(h_i; x_i^-, P_i^-, \theta)] & , h_i \in C \\ \log[(1-p_v)f(h_i; x_i^-, P_i^-, \theta)] & , h_i \in S \end{cases} \quad (4)$$

where p_v is the constant likelihood of having a camera detection when the user is in the FOV. f is a function that evaluates the likelihood of h_i given the estimated x_i^-, P_i^- and the parameters θ of the filter, i.e. it assesses how h_i agrees with the state estimated by the filter. Then $L^v(H)$ is the normalized sum of all ΔL_i^v:

$$L^v = |H|^{-1} \sum_{i=1}^{W} \Delta L_i^v \quad (5)$$

where $|H|$ is the number of non-empty detections in the hypothesis H.

Evaluate $L^r(H, R, \lambda)$: Let us assume that radio model λ in the indoor environment can be described with the log-normal shadowing model with parameters $\{P_0, n, \sigma\}$, and the Received Signal Strength (RSS) $r(a)$ at a point which is a meters to the known basestation is given by:

$$r(a) = P_0 - 10n\log_{10}(a) + \chi_{\sigma^2} \quad (6)$$

where P_0 is the RSS measurement at a reference distance of 1 meter, n is the path loss factor, and $\chi_{\sigma^2} \sim \mathcal{N}(0, \sigma^2)$ is the random shadowing variation. Let x_i^+ be the state estimated with detections $h_{1:i}$ from the previous step, which is the best guess on the location of a user at time i. For a non-empty detection h_i, we define its incremental radio-based likelihood score as:

$$\Delta L_i^r = \sum_{m=1}^{M} \log f_{\mathcal{N}}[r(dist(x_i^+, B_m)); r_i^m, \sigma^2] \quad (7)$$

where $dist(x_i^+, B_m)$ is the distance between x_i^+ and basestation B_m, and $r(dist(x_i^+, B_m))$ is the expected RSS value at x_i^+ (given by Eqn. (6)). $r_i^m \in r_i$ is the received RSS measurement of B_m on the user's mobile device. $f_{\mathcal{N}}$ is Gaussian probability density function. Therefore, the score of h_i is determined by how its corresponding state estimate x_i^+ fits the measured r_i. Then the score for entire hypothesis H and radio model λ is the normalized sum of all ΔL_i^r:

$$L^r = |H|^{-1} \sum_{i=1}^{W} \Delta L_i^r + \log L_\lambda \quad (8)$$

where L_λ is the likelihood of radio model $\lambda = \{P_0, n\}$ given prior distributions on these two parameters. Our algorithm does not require perfect knowledge of λ, but attempts to learn the best model by a search through the parameter space. For simplicity, we assume the parameter σ is known, and only vary P_0 and n. In our experiments,
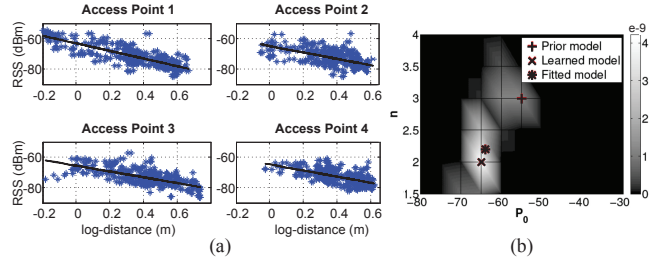

Fig. 4: (a) Fitted radio model from 4 APs. (b) Learning the radio model parameters by searching the parameter space.

we assume P_0 and n are independent, and follow the raised cosine priors governed by known hyper-parameters. Sec. V will show how in practice our algorithm finds the best radio model that is very close to the actual one.

To sum up, for each hypothesis H and possible radio model λ, our algorithm evaluates the likelihood score $L(H, R, \lambda)$ as shown above, and finds the best solution $\{H^*, \lambda^*\}$ which is given by: $\{H^*, \lambda^*\} = \arg\max_{H,\lambda} L(H, R, \lambda)$.

V. SYSTEM EVALUATION

A. Experimental Setup

We have conducted a real world experiment in a three-story Museum building to evaluate the performance of the proposed approach. In our experiment we placed the camera (i.e. off-the-shelf webcam) 10m above the ground in the third floor of the building facing down covering a 11m × 12m area. All people to be tracked were walking on the first floor while looking at the museum exhibits. The duration of our experiment was 40 minutes with the camera taking video at 2fps with a resolution of 960 × 720 px. The total number of people in the scene was varying as the museum visitors were entering and leaving the scene. The minimum number of people in one frame was 8 and the maximum 20 with 4 of them having WiFi enabled smartphones. The objective of the experiment is to determine the accuracy with which people with WiFi data can track themselves in a busy environment with several other people. The WiFi measurements were taken by smartphones receiving beacons from 4 APs at a default rate of 2 samples/sec.

In order to obtain accurate ground truth trajectories from the video footage we supplied all four people carrying WiFi smartphones with hats of different colors. Then we used a mean-shift tracker [12] to track the colored hats and label the ground truth trajectories of all 4 people for our entire 4800 frame dataset. The color features were only used to acquire accurate ground truth; and both our approach and the competing approach use only gray-scale images. We noticed that in the absence of these distinctive hats, appearance features are not informative enough to tell apart one person from another, due to the dim lighting and the fact that with the downward-facing camera we did not get a view of distinctive face/body features.

Algorithms: The competing approach, referred to as *Vision-only tracker*, only uses the smoothness of motion

to connect visual detections into trajectories; it is a multi-hypothesis tracking approach widely used by the vision community [9], [10]. On the other hand, our proposed approach, *RAVEL* (Radio And Vision Enhanced Localization) ravels out ambiguous visual threads by exploiting both the smoothness of motion and radio signal strength data.

Performance metrics: We distinguish between two main operating contexts: 1) offline case, in which the competing/proposed algorithms are given data over a time window of size W, and are tasked to estimate the trajectory of a target during that period; and 2) online case, in which the task is to estimate the current location of the target given a window of historical data of size W.

For the offline case, we use two key metrics to evaluate our approach - offline location error and overlap error. *Offline location error* is an accuracy metric that reflects the distance between the estimated trajectory (using the competing or proposed techniques) and the ground truth trajectory (derived by using distinctive visual markers). In the results below, we measure it as the average Euclidean distance between the ground truth and estimated detections over time. In measuring the distance between trajectories, we only use timestamps (frames) for which both the estimated and ground truth trajectories report the target to be within the field of view.

The second offline metric, referred to as *overlap error*, measures the overlap between the ground truth and estimated trajectories with respect to the camera's field of view. Let FP (false positives) be the ratio of frames in a window of size W for which the estimated trajectory reports a detection in the FOV, whereas the ground truth trajectory suggests that the target is outside the FOV. Let FN (false negatives) be the ratio of frames in which the estimated trajectory reports an empty detection (outside the FOV), whereas the ground truth trajectory includes a detection within the FOV. The *overlap error* between the two trajectories is defined as the sum of false positive (FP) and false negative (FN) ratios, and it reflects the percentage of time in which the estimated algorithm misclassifies the target to be in the FOV when they are not or vice versa.

For the online case, our accuracy metric is referred to as *online location error*, and is the Euclidean distance between the last pair of detections of the estimated and ground truth trajectories (those at the end of the a historical window).

B. Results

No a priori knowledge of the radio model: The first set of experiments is conducted to validate our assumption that we do not require a priori knowledge of the parameters of the radio propagation model. Our proposed algorithm allows us to learn the model from camera and WiFi data, by considering different radio models (from a prior distribution) and choosing the one that maximizes the fitness function described in Eqn. (3). This is in contrast to EV-Loc which assumes a priori knowledge of the radio propagation model and obtains it by collecting training data from the environment (WiFi signal strength data at known distances from the access points). We refer to the radio model learnt

from our proposed approach as *Learned model* and the model derived from training data as *Fitted model*.

Our proposed algorithm, RAVEL, assumes a non-environment specific raised cosine prior distribution on the two parameters (P_0 and n) of the radio propagation model (see Eqn. (6)), with a generous variance for each parameter. The prior distributions for P_0 and n are based on smartphone-based WiFi specs and a number of studies on typical values of n, e.g. [13]. Fig. (4a) shows the received signal strength measurements (600 samples are shown) as we vary the log-distance from four access points and the fitted ground truth radio propagation model. The fitted model ($P_0 = -64$ and $n = 2.2$) is obtained by averaging the P_0 and n values derived from the training data of the four access points.

Fig. (4b) shows that the fitness of the best hypothesis (given by Eqn. (3)) as we vary the parameters of the radio propagation model. Note that the Learned model, derived from the proposed RAVEL algorithm by maximizing the fitness of the best hypothesis, is very close to the *Fitted model* obtained from a wealth of training data (an expensive survey). More importantly this Learned model only assumes a rather broad and uninformative prior, the mode of which (referred to as *Prior model*) is quite far from the Fitted model. Hence, with very little prior knowledge of the radio propagation model, we are able to accurately infer it and avoid the time consuming step of environment-specific training.

Offline location error: The second set of experiments focuses on the offline case, and aims to compare the offline location error of RAVEL and the competing Vision-only tracker. We first use default values for the window size (120 frames taken over 60 secs) and for the WiFi sampling rate (2 WiFi signal strength values per sec), and plot the CDF of the offline location error over 160 windows (40 windows for each of the four people carrying a smartphone with WiFi). Fig. (5a) shows that RAVEL, which leverages WiFi information, achieves a median error of 0.56 m and a 90 percentile error of 1 m, significantly outperforming the competing Vision-only tracker, which has a median error of about 1 m and a 90 percentile error of 4.6 m. Note that to give a fair chance to the Vision-only tracker, we initialize it with the correct visual detection corresponding to the person that we are tracking. However, even with a correct start the Vision-only tracker diverges from the correct path due to crossing, splitting and other ambiguities. These ambiguities are typically resolved by RAVEL with the help of WiFi data; illustrative examples of how this happens are provided later in the evaluation.

We then proceeded to examine the impact of the window size on the offline location error. We kept the WiFi sampling rate to the default value of 2 samples per sec, and varied the window size (10-60 secs corresponding to 20-120 frames). Fig. (5b) shows that the proposed RAVEL approach improves its accuracy as the window size increases. This is reasonable since different people will most likely walk different paths over long periods of time resulting in unique WiFi sequences which act as
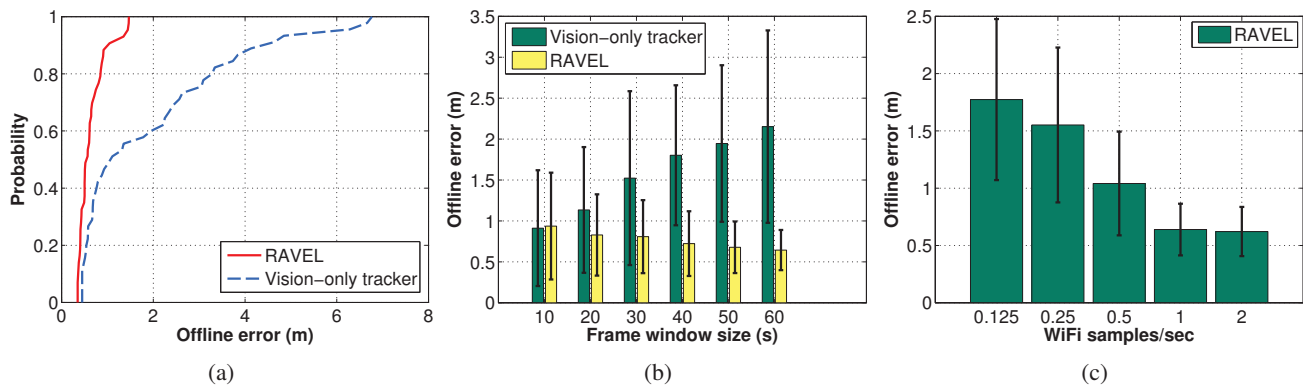
Fig. 5: (a) Cumulative distribution function of the offline location error. (b) Impact of the frame window size on the offline location error. (c) Impact of the WiFi sampling rate on the offline location error.

powerful discriminative signatures for each person. Thus as the frame window size increases the accuracy tends to increase. The converse is true for the Vision-only tracker. For short periods of time we can see that it can be quite accurate and can achieve similar performance with RAVEL. However, for long periods of time a lot of ambiguities are introduced especially in crowded environments which degrades the performance of Vision-only tracking.

The next step was to explore the impact of the WiFi RSS sampling rate on the offline location error of RAVEL with a default window size of 120 frames. As expected, Fig. (5c) shows that the performance of RAVEL improves as we increase the sampling rate. Note that no significant benefits are observed by sampling above 1 WiFi RSS per second, which means that radio sampling does not need to be intensive (i.e. the algorithm could run on battery operated devices) to obtain accurate trajectories. This however may be also due to the fact that people walk relatively slowly in museums; it is possible that different environments may benefit from higher sampling rates.

Offline overlap error: The third set of experiments aims to compare RAVEL with Vision-only tracker in terms of their ability to correctly determine whether a target is inside or outside the camera's FOV. To do so, we use the *overlap error* metric defined above as the percentage of frames in which the estimated trajectory incorrectly places the user inside or outside the FOV when compared to the ground truth. Fig. (6a) shows the CDF of the overlap error of the two algorithms over 160 windows (40 non-overlapping windows times four people). Observe that up to 70% of these windows have no overlap error for RAVEL, as opposed to 50% for the competing approach. The maximum overlap error of RAVEL is 40% as opposed to 100% for Vision-only tracker. This shows that RAVEL's superior performance in resolving ambiguities that result from people entering and leaving the FOV and re-appearing into the FOV from different entry points. Illustrative examples of such cases will be examined below.

Online location error: The last set of experiments focuses on the online case, where we are interested in the most recent location of a target. We still assume that we have
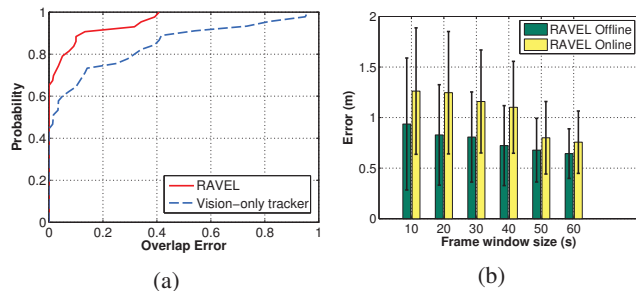

Fig. 6: (a) Cumulative distribution function (CDF) of the overlap error. (b) Online location error.

access to a historical window of visual and WiFi detections, and we investigate the impact of the window size on the online (most recent) location error. As expected, Fig. (6b) shows that the online location error decreases as we increase the window size. This behavior is similar to that of the offline location error (also shown in Fig. (5b)), which measures the average error across all positions of the window. Notice in Fig. (5b) that the online location error is always slightly higher than the offline location error for a given window size, because the last position estimate in a given window has no future detections to benefit from.

Illustrative examples: We close our evaluation by showing in Fig. (7) three illustrative examples showing the performance of the proposed and competing algorithms: In the first example (Fig. (7a)), the person stops at particular museum exhibit, and stays there for a long period of time. Then she starts walking again, and performs a U turn. While the person is looking at the exhibit without moving, no camera detections are generated. Once she starts moving again it becomes difficult to distinguish in which direction she actually moves using camera detections due to ambiguities with other people's trajectories. One hypothesis is that her path merged with a second person's path moving north as illustrated in Fig. (7b); another hypothesis is that she makes a U turn as in Fig. (7c). By taking into account her radio data, RAVEL can identify the correct trajectory. Another interesting case where WiFi measurements are beneficial is illustrated in Fig. (7d). In this case the person leaves the
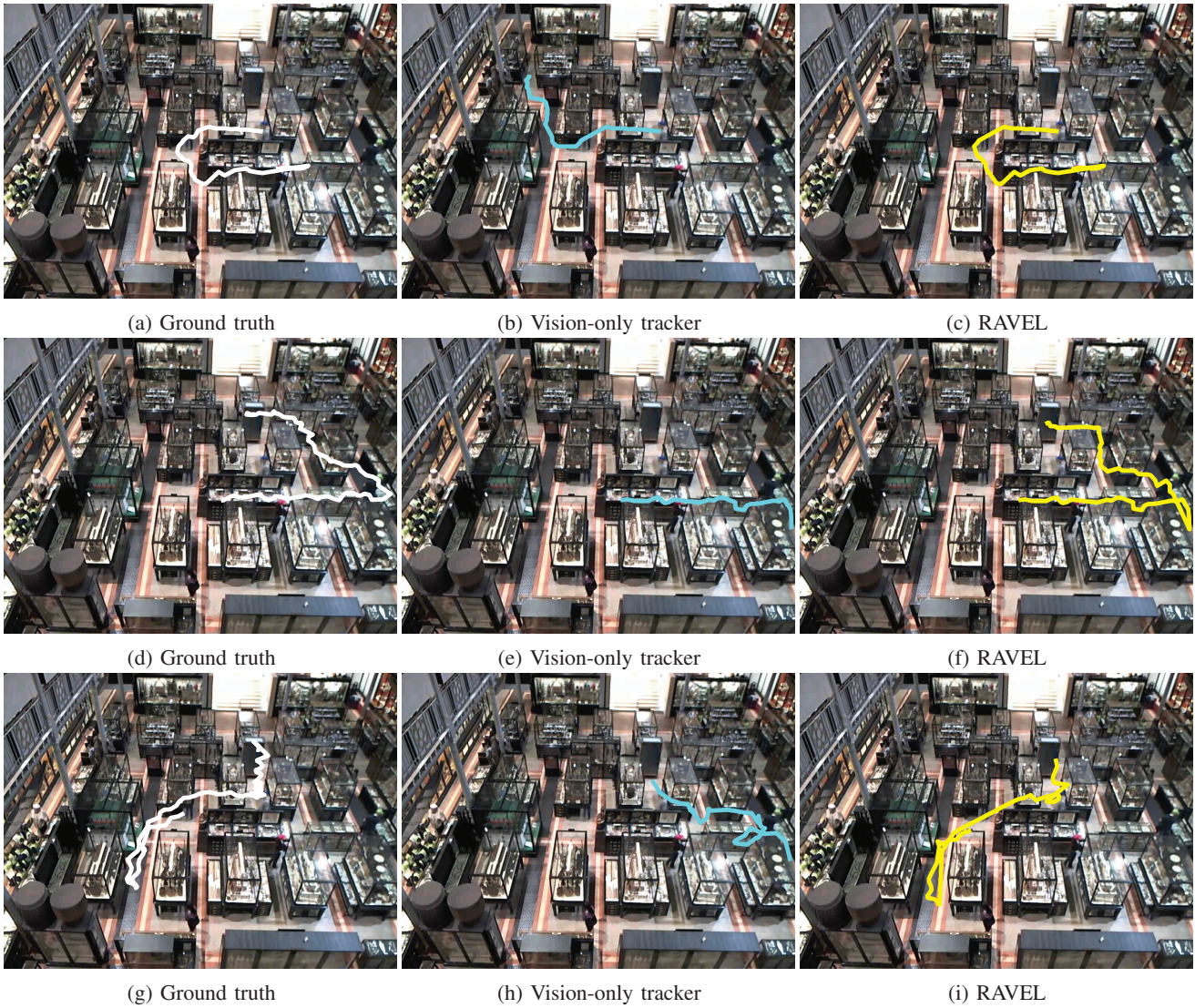
(a) Ground truth	(b) Vision-only tracker	(c) RAVEL
(d) Ground truth	(e) Vision-only tracker	(f) RAVEL
(g) Ground truth	(h) Vision-only tracker	(i) RAVEL

Fig. 7: Illustrative examples showing the performance of the proposed and competing algorithms.

camera's FOV and after some time he re-enters. Fig. (7f) shows that RAVEL can re-establish the identity of the person when he re-enters the FOV and provide the correct trajectory. Finally, Fig. (7g) illustrates the situation of two targets splitting paths which is quite common in crowded environments. More specifically, two people start walking together (i.e. one detection contains two targets) moving south when at some point they split following different paths. In this case the Vision-only tracker has a 50% chance of following the wrong path, whereas RAVEL can use radio data to select the correct path. Although RAVEL allows us to resolve a number of common ambiguities, it is by no means a perfect tracking system; we have observed certain cases in which ambiguities are not correctly resolved either due to a very limited window size, or to a highly crowded and complex scene. In addition, if there are regions where either WiFi or vision are unavailable, in its current implementation no positions are returned. In the future this limitation could be solved by incorporating other modalities

such as inertial sensing. Our qualitative and quantitative results though suggest that RAVEL is significantly more accurate and robust than existing Vision-only tracking systems, requiring only a negligible overhead to extend their implementation.

VI. RELATED WORK

Extensive research has been done on localization techniques, and a recent survey can be found in [14]. Here we provide a brief survey focusing only on state of the art techniques that combine vision with other sensing modalities, such as radio and inertial sensors. Mandeljc et al. [15] recently proposed a fusion algorithm that incorporates radio data into a camera-based probabilistic occupancy map (POM) framework. As in the original POM framework [16], humans are represented as simple rectangles, and detections are generated using a background subtraction technique. The work in [17] generalizes POM to work with arbitrary sensor modalities, and provides an illustrative example with

Ubisense's ultra-wideband radio sensors. However, unlike our work, radio measurements are only applied to the sub-problem of estimating a ground plane occupancy model (that gives the probability of a cell being occupied by any of the humans given sensor data); they are not directly used for identification, i.e. to estimate the probability of a particular person being located in a cell given that the cell is occupied. The latter problem is addressed in [15], [16] using the color distributions of camera detections, without the help of radio. Color distributions and other appearance features are of little use in our context due to the challenging light conditions and the downward facing camera configuration. Hence, these works do not exploit the full power of radio for disambiguating human trajectories. More recent work [17] added a second fusion stage, where anonymous detections are augmented with identity information from radio tags. The cost of mapping an anonymous detection to a radio-based identified detection is evaluated based on the Euclidean distance between the two, and the optimal assignment between two sets of detections in a frame is evaluated using the Hungarian method [18]. Although this approach [17] exploits radio-based detections for identification, it does so on a frame-by-frame basis, and does not jointly address the problem of identification and tracking. More importantly, it assumes knowledge of multiple people's radio data (instead of one person's in our work), and thus addresses a different problem.

Other techniques [2], [3] fuse camera and inertial measurements. More specifically, the authors in [2], [3] use cameras and accelerometers to identify people based on their velocity. The correlation coefficient is being used to calculate the similarity between the velocity of each user estimated from camera and accelerometer measurements. This work is orthogonal to the proposed RAVEL system, and the two papers could be combined in the future to exploit all three sensing modalities (vision, radio and inertial sensors).

Finally the recent EV-Loc [1] system demonstrated how to perform localization by combining electronic and visual signals. Given a number of people equipped with mobile phones, the algorithm estimates the position of each person using both WiFi signal strength and visual tracking. Then an optimization problem must be solved in order to find the best mapping between electronic and visual signals. Finally, the position of each person is calculated as the weighted average of the corresponding electronic and visual estimates. As we already mentioned in the previous sections EV-Loc assumes that the vision system provides clear visual trajectories (i.e. the data association problem is solved) and thus this system concentrates on finding the association between WiFi traces and visual traces given a known radio propagation model. Similar to [17], it assumes knowledge of multiple people's radio data and thus addresses a slightly different problem than the one tackled by RAVEL.

VII. Conclusion

We have introduced RAVEL, a new positioning system that integrates anonymous visual tracking with radio measurements to provide accurate tracking information. Although noisy radio measurements are inadequate for positioning on their own, we demonstrated how they can greatly enhance the performance of visual tracking by handling challenging cases like occlusion, trajectory splitting and entry/exit from the camera field-of-view. The additional advantage of our approach is that it automatically learns the radio propagation model, thus not requiring site-specific calibration. We believe that this work has the potential to provide highly accurate positioning at low cost in challenging indoor environments.

References

[1] B. Zhang, J. Teng, J. Zhu, X. Li, D. Xuan, and Y. F. Zheng, "Ev-loc: integrating electronic and visual signals for accurate localization," in *Proc . MobiHoc*, 2012.

[2] D. Jung, T. Teixeira, and A. Savvides, "Towards cooperative localization of wearable sensors using accelerometers and cameras," in *Proc. INFOCOM*, 2010.

[3] T. Teixeira, D. Jung, G. Dublon, and A. Savvides, "Identifying people in camera networks using wearable accelerometers," in *Proc. PETRA*, 2009.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.

[5] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. CVPR*, 1999.

[6] Y. Shen, W. Hu, J. Liu, M. Yang, B. Wei, and C. T. Chou, "Efficient background subtraction for real-time tracking in embedded camera networks," in *Proc. SenSys*, 2012.

[7] R. Chandra, J. Padhye, L. Ravindranath, and A. Wolman, "Beaconstuffing: Wi-fi without associations," in *Mobile Computing Systems and Applications, 2007. HotMobile 2007. Eighth IEEE Workshop on*, March 2007, pp. 53–57.

[8] I. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *PAMI*, vol. 9, no. 1, pp. 56–73, 1987.

[9] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[10] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, no. 1, pp. 5–18, Jan 2004.

[11] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. D, pp. 35–45, 1960.

[12] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust mean-shift tracking with corrected background-weighted histogram," *Computer Vision, IET*, vol. 6, no. 1, pp. 62–69, January 2012.

[13] S. Seidel and T. Rappaport, "914 mhz path loss prediction models for indoor wireless communications in multifloored buildings," *IEEE Tran. on Antennas and Propagation*, vol. 40, no. 2, pp. 207–217, Feb 1992.

[14] R. Mautz, *Indoor positioning technologies*. Habilitation thesis, ETH Zurich, 2012.

[15] R. Mandeljc, J. Pers, M. Kristan, and S. Kovacic, "Fusion of nonvisual modalities into the probabilistic occupancy map framework for person localization," in *Proc. ICDSC*, 2011.

[16] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multi-camera people tracking with a probabilistic occupancy map," *PAMI*, vol. 30, no. 2, pp. 267–282, 2008.

[17] R. Mandeljc, S. Kovai, M. Kristan, and J. Per, "Tracking by identification using computer vision and radio," *Sensors*, vol. 13, no. 1, pp. 241–273, 2012.

[18] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.