TagAttention: Mobile Object Tracing without Object Appearance Information by Vision-RFID Fusion

Xiaofeng Shi*, Minmei Wang*, Ge Wang†*, Baiwen Huang*, Haofan Cai*, Junjie Xie‡*, Chen Qian*
*University of California, Santa Cruz, CA 95064, USA
†Xi'an Jiaotong University, Xi'an 710049, China
‡Science and Technology on Information Systems Engineering Laboratory
National University of Defense Technology, Changsha Hunan 410073, China

***Abstract*—We propose to study mobile object tracing, which allows a mobile system to report the shape, location, and trajectory of the mobile objects appearing in a video camera and identifies each of them with its cyber-identity (ID), even if the appearances of the objects are not known to the system. Existing tracking methods either cannot match objects with their cyber-IDs or rely on complex vision modules pre-learned from vast and well-annotated datasets including the appearances of the target objects, which may not exist in practice. We design and implement TagAttention, a vision-RFID fusion system that archives mobile object tracing without the knowledge of the target object appearances and hence can be used in many applications that need to track arbitrary un-registered objects. TagAttention adopts the visual attention mechanism, through which RF signals can direct the visual system to detect and track target objects with unknown appearances. Experiments show TagAttention can actively discover, identify, and track the target objects while matching them with their cyber-IDs by using commercial sensing devices, in complex environments with various multipath reflectors. It only requires around one second to detect and localize a new mobile target appearing in the video and keeps tracking it accurately over time.**

I. Introduction

As the key components of the Internet of Things (IoT), many moving objects (the 'Things') carry their cyber-identities (IDs) such as unique sequence numbers or network addresses. We study the *mobile object tracing* problem, which allows a mobile system to report the shape, location, and trajectory of the mobile objects appearing in a video camera and identifies each of them with its cyber-ID, even if the appearances of the objects are not known to the system. Mobile object tracing is one essential problem of mobile computing with emerging applications such as cashier-free stores (identify and track the customers and the merchandise in their shopping carts), autonomous cars (identify other vehicles and traffic signs), electronic article surveillance (EAS), virtual/augmented reality, TV motion sensing games, and lost child/object searching. In most of these applications, the appearances of the objects (customers, merchandise, vehicles, lost objects) may not be known in advance to the system, or the objects are in a huge amount whose appearances are too many to learn.

Mobile object tracing requires the following specific tasks.

978-1-7281-2700-2/19/$31.00 2019 © IEEE

- *Object detection*: detect each mobile object from the video frames and highlight its shape and boundary.
- *Identify matching*: match each mobile object with its cyber-ID.
- *Movement tracking*: obtain the location and moving trajectory of each target object.

These tasks have been individually studied in many areas including computer vision, wireless sensing, and human computer interaction. For example, computer vision may be able to segment a moving object from video frames – most of these methods require the object's appearance is pre-registered. However, computer vision provides no information about the cyber-ID. Wireless sensing methods can tell the cyber-IDs of the objects in an area but their appearances and detailed behaviors are not known. However, combining these two types of methods and achieving fast speed, cost efficiency, and accuracy are still challenging, especially in many applications where the appearances of the moving objects are not known in advance.

Computer vision is a powerful tool for object detection [28], segmentation [22] [15], and tracking [12], [41] from images and videos. Most modern computer vision methods can effectively track objects ***only if*** the object's appearance is pre-registered [1], [12], [41]. In addition, they cannot process any cyber-ID information and fail to identify objects with similar appearances. On the other hand, tracking approaches based on RFID can only estimate the coarse location of objects due to the uncertainty (such as noise and multipath) in signal measurement [6], [8], [13], [14], [31], [36], [43], [46], [47] and fail to highlight the object appearances or localize them precisely in video frames.

An intuitive solution is combining computer vision and RFID technologies to simultaneously obtain the location of the target objects from the visual channel and the identities from the wireless channel [10], [20], [21], [24], [42]. However, existing vision-RFID fusion methods cannot achieve mobile object tracing with ***zero*** human's assistance. They all require to pre-learn the appearances of the objects, either from a vast and well-annotated dataset that describes the target objects or from users' annotation when the targets initially appear in the scene. If the object appearances are unknown, these solutions are *NOT* able to detect and track the objects from the video

frames and match them with their cyber-IDs. In fact, in many applications the system does not know the appearances of the target objects in advance.

In this paper, we argue that the wireless communication between the target and the reader through the RF channel can essentially *assist* the visual channel to actively find the target mobile object *without* knowing the objects' appearance. We consider the raw visual sensing information (such as video frames obtained from cameras) as the bottom-level information and the abstraction of the objects (such as their cyber-IDs and coarse motion trajectories which can be obtained from the RF channel) as the top-level information. We propose the TagAttention, which adopts the "bottom-up" and "top-down" visual attention model to fuse the visual and wireless sensing channels for mobile object tracing. The "bottom-up" visual attention model detects the optical flows (patterns of apparent motion of the objects) from the RGB frames and the "top-down" step detects, segments and tracks the visual regions by matching the motion of targets in the video with the ID and wireless channel information. The intention to use attention model in our framework is that physical layer properties of wireless signals, such as signal phases, can "direct" the vision model to focus its attention only to the moving targets. TagAttention could automatically detect, localize, and identify any tagged object in the video when it appears in the camera and then keep tracking it. It only requires around one second to detect and localize a new mobile target appearing in the video and keeps tracking it accurately over time. To our knowledge, **no prior method can achieve this task**.

The main advantages of TagAttention include: 1) It can actively discover rigid tagged mobile objects and automatically track them without pre-knowledge of the objects' appearance, hence it requires zero human's assistance to label visual data; 2) It is fast and cost-efficient; 3) it does not need manually annotated datasets for training; 4) it uses only commercial devices for sensing; 5) it works well in complex environments with many multipath reflectors.

The balance of this paper is summarized as follows. Section II presents the related work. Section III illustrates the design of TagAttention. In Section IV we present the evaluation results. The limitations of the proposed system are discussed in section V. We conclude the paper in section VI.

II. Related Work

A. Visual Tracking Systems

Object tracking in computer vision research is usually defined as predicting bounding boxes for certain objects in every video frame. One category of the solutions uses correlation filters, such as MOSSE [2] filter. More recently, the target patch searching can be accomplished in an end-to-end manner by deep neural networks [12], [19], [45]. Video segmentation aims at learning fine-grained appearance masks of the objects. Representative state-of-the-art methods use spatio-temporal context [41] or optical flows [5] learned by deep neural networks. However, all the above methods require either a large well-annotated dataset to train their models, or users' initial annotation to tell the model what to track (deep learning based methods, which yield state-of-the-art performance, usually require both). Actively finding and identifying the targets that are not registered or learned by the models remains unsolved.

B. Vision-RFID Fusion

In recent years, attempts have been made to fuse vision and RF signals so that the systems can both track and identify mobile targets by matching the information from both channels [24] [21] [10] [20] [42]. Mandeljc *et al*. [24] propose to detect and track anonymous humans from videos by matching the IDs in RF-channel to the detected human instances based on the location information. ID-Match [21] is a novel vision-RFID fusion system for human identification from a group through an RGB-D camera and an RFID sensor. However, both of the above-mentioned methods rely on the human detection or human pose estimation module accomplished by specifically-trained computer vision models. Therefore they cannot be used to identify objects other than humans.

Beyond tracking and identification of humans, TagVision [10] fuses signals of RFID tags on objects and 2D surveillance video by calculating probabilistic matching scores of the signal phases and object motions. However, it relies on a vision-based blob detection model, which can only track specific objects moving on a static 2D plane by which the camera model is calibrated in advance. Thus, the system may quickly fail when tracing various targets in complex and dynamic 3D scenarios [42]. A recent work proposes IDCam [20], which fuses RFID and 3D camera to trace a tagged item that is held by a user's hand. The system requires a precise detection of the user's hand, which is accomplished by a carefully tuned visual detection and tracking module. In addition, TaggedAR [42] is proposed to detect and identify stationary objects by rotating the sensors and pairing RF-signals with the depth of the target objects.

Existing fusion solutions cannot achieve tracing arbitrary mobile objects in 3D space. They either only trace particular targets (such as a human body) with sophisticated models or trace objects on a calibrated 2D plane. They cannot identify and track objects with unknown arbitrary appearances in complex 3D environments, which is our design objective of this work.

III. Design of TagAttention

A. Overview

In TagAttention, we use a commercial RFID reader carrying one antenna and an RGB-D camera on top of the antenna to capture the sensing data. In addition, each tracing target carries an RFID Tag that can be read by the RFID reader through the antenna. Fig 1 shows an overview of our attention-based fusion system. The inputs of our fusion model are the RGB intensity and distance maps (each pixel of the distance map represents the distance from the 3D voxel to the sensor origin) captured by the RGB-D camera, and the RFID EPCs (denoting the cyber-IDs of the objects) and their corresponding phase signals obtained by the RF reader.

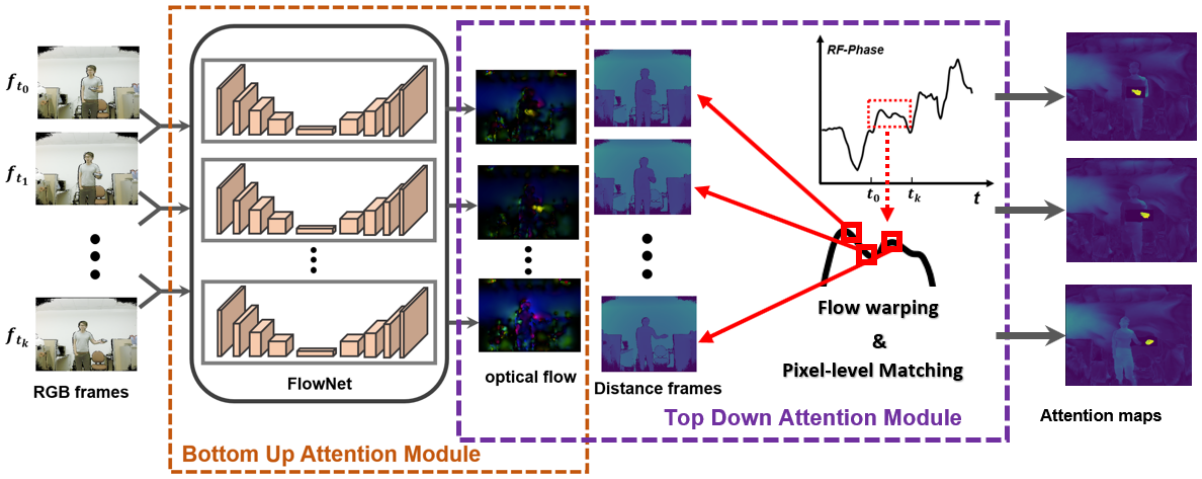


Fig. 1. Overview of TagAttention. The system is mainly comprised of the bottom-up and top-down attention modules.

We consider the raw video inputs as the bottom-level information and the abstraction of the objects (such as their cyber-IDs and motion trajectories) as the top-level information. Given two consecutive RGB frames, the *bottom-up* visual attention mechanism estimates the pixel-level optical flow (the optical flow is the motion velocity of the image pixels along the image's axes in the current video frame) to measure the motions of pixels from the visual frames. Since the produced optical flow can highlight moving pixels from raw video, it works as a bottom-up visual attention mechanism [7], where the system naturally notice the salient visual components of potential importance from visual inputs.

Meanwhile, the *top-down* visual attention module in TagAttention functions as a detector of the targets given the RF signals that match the visual targets. In the top-down attention module, we obtain the consecutive distance by unwrapping the phases of RFID tags, and map it with the per-frame optical flows. By combining the *bottom-up* and *top-down* modules together, we can obtain an attention map for each timestamp, which represents the pixel-level consistency between the motion trajectories in the video and the distance changing of the RFID tag. The attention map is a 2D matrix with the same size as the video frame resolution, in which each element represents the magnitude of attention (measured by the probabilistic matching score of the two sensing channel in our design) on the corresponding image pixel.

Finally, a tracker is designed to actively discover the target objects and output their corresponding shape and location (represented by a pixel-wise mask for the object, we use 'mask' in the following) from the video based on the per-frame attention maps.

Compared to the existing fusion methods, TagAttention can actively highlight ubiquitous target objects in a video *without* any pre-knowledge of the object's appearance. Thus, this tracing model can be applied on a much wider variety of visually-complex scenarios in which target objects are not visually pre-registered.

B. RF Signal Preprocessing

In TagAtthention, the RFID tags are matched to the objects in the video through the correlation of the motion trajectories of the objects. The distance L from the reader antenna to the tag can be calculated as follows:

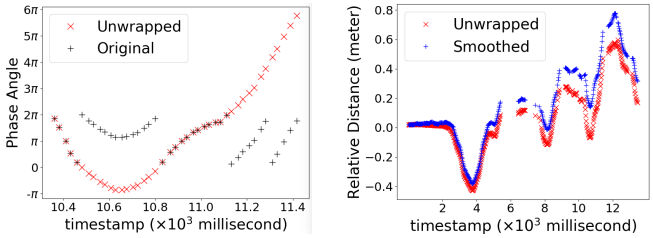
$$L = \frac{\phi_L \cdot c}{4\pi f}, \tag{1}$$

where ϕ_L represents the corresponding phase change over the signal travel distance, c is the speed of light and f is the signal frequency (equals to 920MHz for our reader). Note that with the current COTS devices, we can not calculate the exact distance of the tag. There are two reasons. One is that in addition to the phase ϕ_L over distance, both the reader and tag's circuits will introduce some additional phase rotations to the received phase ϕ, i.e., $\phi = (\phi_L + \phi_R + \phi_T) \mod 2\pi$, where ϕ_R and ϕ_T are the additional phases of the reader and tag respectively [11], [16]. Another reason is that our commercial RFID reader (ImpinJ R420) also introduces π radians of ambiguity. In other words, the reported phase can either be the true phase or the true phase plus π radians [16]. Hence for our reader, $\phi_L = n\pi + \phi - (\phi_R + \phi_T)$, where n is a non-negative integer. Since ϕ_R and ϕ_T are constant over the whole reading period, to estimate the motion of the tag over time, we only consider the relative distance changes of the tag, i.e.,

$$\Delta L = L - L_0 = \frac{(\Delta n\pi + \Delta\phi) \cdot c}{4\pi f}, \tag{2}$$

where L_0 is a reference distance which can be set as the first calculation. And $\Delta n = n - n_0$ and $\Delta\phi = \phi - \phi_0$. After this step, we can obtain a relative moving distance of the tag, ΔL, which only related to the changing positions.

To extract the motion trajectory of the objects, we conduct two signal processing progress, namely phase de-periodicity [4] and motion smoothing. As illustrated as the black plus sign in Fig 2 (A), the received phases are wrapped over cycles and fall into the range of 0 to 2π. This characteristic of the phase values makes the motion estimation discontinuous.



(A) De-periodicity (B) Smoothing

Fig. 2. RF phase signal preprocessing and the relative distance trajectory

Hence we first unwrap the received phase values and retrieve the consecutive motion profile. In our design, we adopt two thresholds, $th_1 = 0.5\pi$ and $th_2 = 1.5\pi$, to detect the π and 2π hops. Specifically, let $\Delta\phi_{t_1,t_2} = |\phi_{t_2} - \phi_{t_1}|$ represent the difference between two adjacent phases ϕ_{t_1} and ϕ_{t_2}. The latter phase value ϕ_{t_2} will be added or subtracted by π if $th_1 < \Delta\phi_{t_1,t_2} \leqslant th_2$, and by 2π if $\Delta\phi_{t_1,t_2} > th_2$. The performance can be found in Fig 2 (A).

We also consider the motion smoothing to get rid of the environment and device noises. Since the received phases can be easily impacted by outside environments and equipments, it is hard to tell whether a hop between adjacent received phases is caused by the π or 2π phase wrapping, or by a sudden movement of the object, or by insufficient reading. Hence, we further smooth the phase based on the estimated acceleration of the moving object. The main idea is based on an observation that the rapid and sudden change of velocity, which requires a huge force acting on the object, is unlikely to happen in most real applications. Thus, we calculate the average velocities and accelerations of the object within the reading time slots after de-periodicity. If the acceleration of the object in a certain time slot is higher than a threshold, i.e. the gravity acceleration $g \approx 9.8m/s^2$, we consider the high acceleration is caused by the inappropriate de-periodicity or other environmental noises. To smooth the motion of the objects in such case, we keep the average velocity $\bar{v}_{t_0,t_1}$ in previous time slot constant for the next time slot and approximate the gain of distance at t_2 by $(t_2 - t_1)\bar{v}_{t_0,t_1}$. A smoothing result is shown in Fig 2 (B).

Channel Synchronization. The fusion of the two channels requires the synchronization of two-channel data samples. However, the reading rate of the RFID reader is unstable due to the slotted ALOHA protocol. Therefore, to synchronize the two channels, we first calibrate the camera's and the reader's reading timestamps according to the system's clock and use the Kinect's timestamps as the standard timestamps. Then the preprocessed RF signals (which have been converted as distance trajectories as shown in Fig 2 (B)) are interpolated so that we can sample the signals at the camera's standard timestamps.

C. Bottom-up Attention Module

In TagAttention, the bottom-up attention module captures the salient visual features through the optical flow, i.e. the motion of pixels in two consecutive video frames at t and $t + \Delta t$. The optical flow is defined as an $H \times W$ matrix, where $H \times W$ is the resolution of the video. Each element at

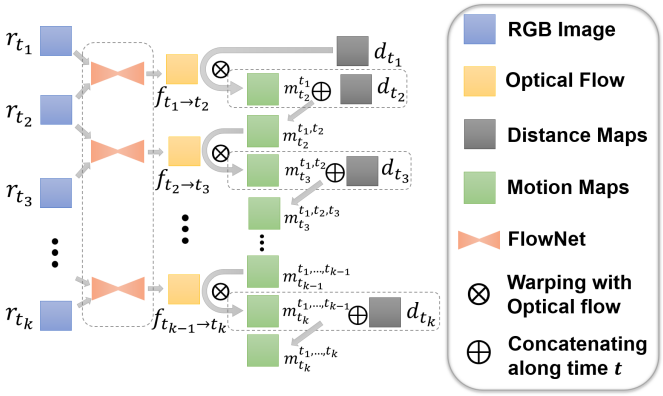


Fig. 3. The distance maps are warped with the optical flows over time to contruct the motion maps.

pixel position (x, y) in the optical flow is a two-dimensional vector $(\Delta x, \Delta y)_{(x,y)}$, which satisfies

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t), \tag{3}$$

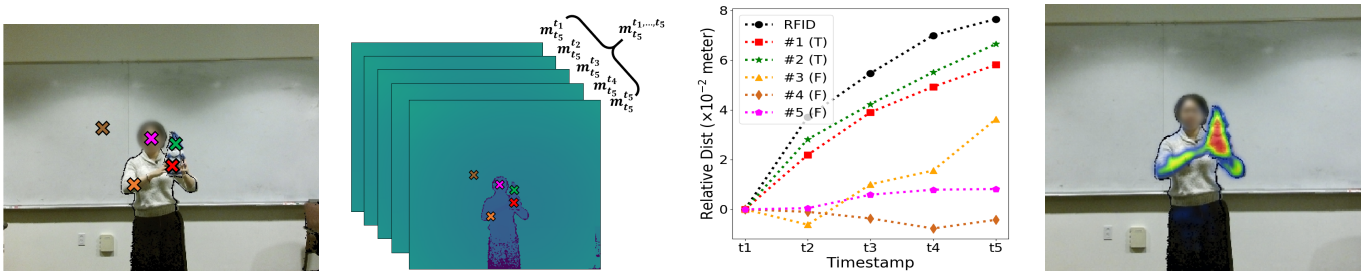
where $I(x, y, t)$ represents the image intensity of the pixel (x, y) at time t.

In our framework, we learn the optical flow through an end-to-end deep neural network, which has been proved to be both more effective and efficient [9] [25] than traditional methods [32]. Specifically, we adopt the FlowNet [9] as the backbone neural network architecture and the training strategy presented by [25] to train the neural network in an unsupervised manner. By feeding the consecutive video frame pairs F_{t_1}, F_{t_2} into the FlowNet, the model predicts the optical flow map $f_{t_1 \to t_2} = \{(\Delta x, \Delta y)\}_{(x,y)}$. The estimated optical flow naturally highlights the pixels on moving objects from the image frames, which works similarly as a visual bottom-up attention mechanism to notice the moving objects since movement is the key clue in TagAttention to fuse the two perceptive channels. The optical flow will be further used to warp the distance maps and propagate the predicted attention maps over frame timestamps. However, as the environment may contain a variety of dynamic factors, for example, the movement of irrelevant objects or changing of light condition, targets can hardly be distinguished from the background directly using optical flow without pre-knowledge of the objects' location and shape. Thus, we will introduce another essential attention mechanism to discover and segment targets in the following sections.

In our system, we can flexibly replace the optical flow module with future advanced optical flow estimators.

D. Top-down Attention Module

1) Motion Estimation: In the top-down attention module, TagAttention finds and highlights the target objects' pixels by matching the motion of each pixel in the visual system with the distance changes measured by the RF phase and calculating their correlation probabilistic scores. To estimate the pixel-level motion (the moving trace of each pixel in Kinect frames), we warp the distance maps D_{map} with the optical flows frame by frame and obtain the motion maps M_{map}. In M_{map}, each



(A) (B) (C) (D)

Fig. 4. (A): Samples of the anchors in an example video frame. (B): The corresponding motion map of the frame in (A) (window size = 5). (C): The motion trajectories of the anchor points: Pixel #1 and #2 are the target anchor pixels, while the rest are random anchor pixels. (D): The generated attention heat map.

pixel denotes the distance trajectory (represented by a vector) of the invariant real-world voxel in 3D space. Specifically, let $d_0, d_1, ..., d_t \in D_{map}$ represent distance maps from the first frame F_0 to the current frame F_t. By feeding the RGB frames $r_0, r_1, ..., r_t \in RGB_{map}$ into the FlowNet, we can estimate the optical flow maps $f_{0\to1}, f_{1\to2}, ..., f_{t-1\to t} \in Flow_{map}$ for each pair of frames. Note that d_t, r_t, $f_{t-1\to t}$ are $H \times W \times 1$, $H \times W \times 3$ and $H \times W \times 2$ matrices respectively, where H and W are the height and width of the video frames, and the third dimension represents the value channels. Then we warp D_{map} with $Flow_{map}$ to estimate the motion maps M_{map} according to Fig 3. In Fig 3, $m_{t_j}^{t_i,...,t_j}$ represents the motion map with size $H \times W \times (t_j - t_i + 1)$, where the third channel is the timestamp channel which records the distance values of each corresponding real-world voxel from t_i to t_j. In $m_{t_j}^{t_i,...,t_j}$, the subscript t_j represents the real-world voxels are projected to the camera frame at time t_j. Hence, $m_{t_j}^{t_i,...,t_j}$ can be calculated as follows:

$$m_{t_i}^{t_i,...,t_j} = ((((d_{t_i} \otimes f_{t_i\to t_i+1}) \oplus d_{t_i+1}) \otimes f_{t_i+1\to t_i+2}) \\ \oplus d_{t_i+2} \cdots \otimes f_{t_j-1\to t_j}) \oplus d_{t_j}, \quad (4)$$

where $\otimes$ represents the warping process with optical flow over all channels of the third dimension of the matrix, and $\oplus$ represents concatenating of two maps along the third channel (i.e. the time channel).

Meanwhile, the RFID reader collects the RF signal for each tag id_k during t_i to t_j, and the signals are converted into relative distance vectors $rd_{id_1}^{t_i,...,t_j}, rd_{id_2}^{t_i,...,t_j}, ..., rd_{id_n}^{t_i,...,t_j} \in RD^{t_i,...,t_j}$. We then match the moving pixels with the RF tag by calculating the correlation probabilistic scores between the motion map $m_{t_j}^{t_i,...,t_j}$ and the RF distance vector $rd_{id}^{t_i,...,t_j}$. Fig 4 presents an example. As shown in Fig 4, (A) shows an RGB frame at time t_5, and (B) represents the motion map $m_{t_5}^{t_1,...,t_5}$ over five timestamps from t_1 to t_5 ($\approx 150ms$) computed by formula 4. In Fig 4 (A), we arbitrarily sample a few pixels as random anchors and illustrate their motion trajectories in (C). As a comparison, we also label two pixels (denoted by red and green) on the target object as target anchors in $m_{t_j}^{t_i,...,t_j}$ and show their estimated relative distance vectors as well over time in (C) [1]. In addition, the motion estimated by RF channel $rd^{t_i,...,t_j}$ is also plotted with the black line in (C). To eliminate the overall bias caused by the π or 2π rotations of RF signal phases, the motion vectors are translated so that the initial relative distance of motion trajectory in the window is 0, namely, for each timestamp t_k within $[t_i, t_j]$, $\hat{m}_{t_j}^{t_k} = m_{t_j}^{t_k} - m_{t_j}^{t_i}$ and $\hat{rd}_{id_n}^{t_k} = rd_{id_n}^{t_k} - rd_{id_n}^{t_i}$. Hence, we obtain the unbiased motion map $\hat{m}_{t_j}^{t_i,...,t_j}$ and RF motion vectors $\hat{RD}^{t_i,...,t_j}$ for comparison and matching (as shown in Fig 4 (C)). From Fig 4 (C), we notice the motions of the two anchor pixels located at the target object in the motion map match well to the motion of the RFID tag estimated by RF signals, while other random anchor pixels fail to match.

Ideally, the motions of the pixels on a rigid target in the unbiased motion map $\hat{m}_{t_j}^{t_i,...,t_j}$ from the visual channel should perfectly match with the unbiased motion vector of the corresponding RFID tag, since they all measure the relative distance from the anchor point of the object to the sensors within timestamp t_i to t_j in the physical 3D space. However, both measurements could be inaccurate, causing the possible misalignment of the two traces. For example, in the visual channel, error exists when warping the distance map as the optical flow may not be perfect; while in the RF channel, the error can be caused by multi-path, random Gaussian noise, low sampling rate and inappropriate De-periodicity. Nevertheless, the tendency of the motions in two channels can match in a long term, as all these noisy factors only cause random and temporary impact on the signals. Hence, we introduce an attention mechanism Att_{RF}, which is robust to the temporary and random noise, to measure the correlation of the motions in different channels.

2) Attention Mechanism: The proposed attention mechanism Att_{RF} is comprised of two attention components: 1) Att_{rbf}, which uses an radial basis function (RBF) kernel to measure the similarity of the motion vectors in Euclidean space; 2) Att_{corr}, which measures the correlation coefficient of the motion vectors. To calculate the attention scores, we first reshape $\hat{m}_{t_j}^{t_i,...,t_j}$ into $\left\{\rho_{(h,w)}^{t_i,...,t_j}\right\}_{H\times W}$, with each element $\rho_{(h,w)}^{t_i,...,t_j}$ representing the motion vector from t_i to t_j of each pixel $p_{(h,w)}$ in the motion map $\hat{m}_{t_j}^{t_i,...,t_j}$. Then the pixel-level

[1]Note that the anchors are artificially selected only for the visualization and illustration purpose.

attention mechanism can be formulated by equation 5 and 6.

$$Att_{rbf} = exp\left(-\frac{\left\|\rho_{(h,w)}^{t_i,\ldots,t_j} - \hat{rd}_{id_k}^{t_i,\ldots,t_j}\right\|^2}{2\alpha}\right), \quad (5)$$

$$Att_{corr} = Relu\left(\frac{cov(\rho_{(h,w)}^{t_i,\ldots,t_j}, \hat{rd}_{id_k}^{t_i,\ldots,t_j})}{\sigma(\rho_{(h,w)}^{t_i,\ldots,t_j})\sigma(\hat{rd}_{id_k}^{t_i,\ldots,t_j})}\right), \quad (6)$$

where we use the rectifier activation function $Relu(x) = max(0, x)$ to suppress negative correlations, α is the RBF kernel parameter, $cov(\cdot)$ represents the covariance of the two vectors and $\sigma(\cdot)$ represents the variance of the vector. To combine the two types of attention mechanism together, we used formula 7, which calculates the weighted sum of the two attention scores.

$$Att_{RF} = \beta Att_{rbf} + (1-\beta)Att_{corr}, \beta \in [0, 1] \quad (7)$$

We empirically set $\alpha = 5 \times 10^{-4}$ and $\beta = 0.8$ in our implementation. According to the formulas, Att_{RF} is in the range of $[0, 1]$. Hence we approximately consider Att_{RF} to describe the probability that the pixel (h, w) at timestamp t_j matches with the target object that is labeled by a certain RFID tag. Thus, for each target object, we construct the attention map matrix a_t, which is of the same size as the input image matrix. Each element in a_t represents the attention probabilistic score Att_{RF} of the corresponding pixel. Fig 4 (D) shows an example of the attention map with a heat map.

E. Attention Propagation

The top-down attention module enables the system to predict an attention probabilistic map for each video frame. However, the prediction can be accurate only when the target objects move during the attention window, since we assume the top-down attention is triggered based on the movement of the targets. When the target object is static, the distance values of the object pixels keep unchanged in the RGB-D camera. However, due to the dynamical factors of the environment (such as the movement of other objects), the phase values of the corresponding tag may still subtly change over time. In such case, the noise of the environment dominates the attention probabilistic scores of the pixels according to formulas 5 and 6. In addition, distance measurement or localization of objects through RF signals within a pixel level error bound (about several millimeters) is rather challenging [3], [6], [23], [37], [38], [46], especially when using commercial RFID readers and a single antenna in our system [27], [29], [30], [34], [36], [44]. Therefore, it is nearly impossible to precisely match every pixel with the corresponding RF signals based on the relative motion at a single frame. Fortunately, the visual channel provides tremendous semantic information of the target objects and the environments, which enables us to track and segment the target objects cross multiple timestamps based on the correlation of objects' appearances.Though there maybe some mismatches at a few frames, the overall trend of motions of the two channels can finally match with each other in a long term.

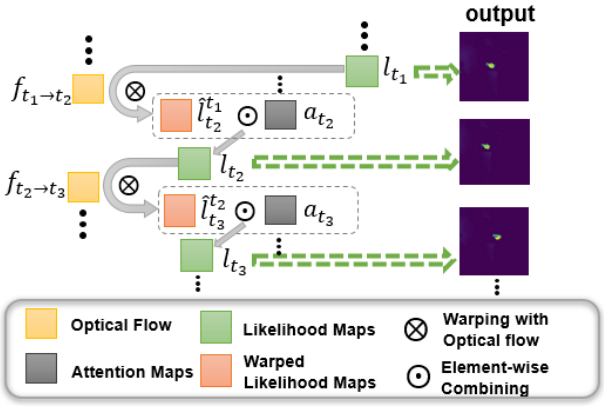


Fig. 5. Mask propagation by warping the probabilistic maps with optical flows over time

Hence, in order to improve the robustness of our tracking system, we propose an attention propagation mechanism as illustrated in Fig 5. Specifically, for each target object instance id_k, we initialize the likelihood map $l_{t_0} = \log a_{t_0}$ ($\log a_{t_0}$ represents the element-wise log operation of the attention map matrix a_{t_0} in our notation) at the first frame F_{t_0}. For each following frame F_{t_i}, we warp the likelihood map l_{t_i} with the optical flow $f_{t_i \to t_{i+1}}$ to reconstruct the warped likelihood map prediction at frame $F_{t_{i+1}}$, which is denoted as $\hat{l}_{t_{i+1}}^{t_i}$. Then the likelihood map $l_{t_{i+1}}$ at frame $F_{t_{i+1}}$ is calculated by Eq. 8,

$$l_{t_{i+1}} = \hat{l}_{t_{i+1}}^{t_i} + \Theta(v_{t_{i+1}} - v_0) \times \log(a_{t_{i+1}}), \quad (8)$$

where $v_{t_{i+1}} = \frac{\left|rd_{id}^{t_{i+1}} - rd_{id}^{t_{i+2-k}}\right|}{t_{i+1} - t_{i+2-k}}$ denotes the absolute velocity of the motion of the target measured by the RF signal within the time window in which $a_{t_{i+1}}$ is computed, k is the window size (count of the timestamps in the window), $\Theta(x) = 1$ if $x > 0$ otherwise $\Theta(x) = 0$, and $v_0 > 0$ represents a velocity threshold. In our implementation, we set $v_0 = 0.1m/s$, meaning the top-down attention is only triggered by the mobile targets that move at a temporary absolute velocity higher than $0.1m/s$ at current timestamp.

F. Tracking by Attention

In the previous attention modules, only the pixels of the target object in video frames would have consistently high attention probabilistic score over different timestamps, thus yielding high likelihood value in the current likelihood map l_{t_i}. Therefore, we can simply use a threshold to cut off the likelihood and segment the target in current frame F_{t_i}. However, according to formula 8, the likelihood value of each pixel keeps decreasing over time as more frames are processed, which makes it infeasible to set a fixed cutting-off threshold. Therefore, we design an automatic thresholding method to segment the target from the video frames based on the likelihood map.

Specifically, we first convert the likelihood map l_{t_i} to the normalized probabilistic map p_{t_i} by calculating $p_{t_i}(h, w) =$

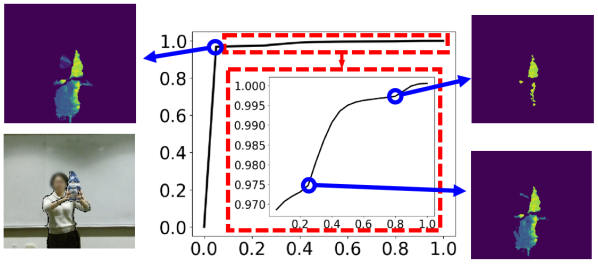


Fig. 6. CDF of the normalized probabilistic values in an example probabilistic map p_{t_i}. Blue circles represent the Conner points in the CDF plot. We can easily segment the image by cutting off the image at the corner points.

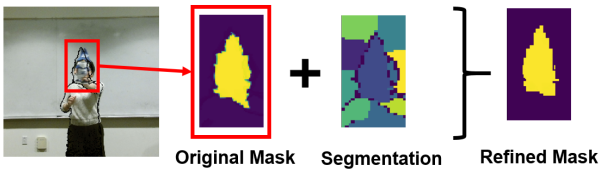


Fig. 7. Mask refinement.

$e^{l_{t_i}(h,w)}$ in element-wise of the 2D matrix l_{t_i}. Then we normalize p_{t_i} crossing all pixels using min-max normalization. By observing the value distribution of the pixels in the probabilistic map p_{t_i}, we can easily find that the probabilistic values are highly hierarchical: the background pixels, which usually comprise the major regions of the frame image, have significantly smaller probabilistic values (close to 0) than the target objects; the"soft" body components that temporarily move in consistency with the target rigid body would have relatively smaller probabilistic values, and the values of these body pixels keep decreasing when the motion consistency no longer holds; while the target object would have consistent highest values. Fig 6 shows an example of the cumulative distribution function (CDF) of the pixel values in p_{t_i}. Based on this observation, we can use multiple ways to segment the frames according to the normalized probabilistic map, such as value clustering or simply cutting off the CDF of the value distribution at the "corners" (showing as a sudden change of the gradient) on the CDF plot (as labeled in Fig 6). In our implementation, we choose the last corner point in the CDF to cut-off the image to extract the target mask.

Another issue of the tracking system is that the errors in the predicted optical flow accumulate over the warping steps, resulting in the possible misdetection of the target after a few iterations of attention propagation. To solve this problem, we refine the shapes of the target masks according to the 3D segmentation of scene based on K-means clustering [17], [33]. Fig 7 illustrates an example of the segmentation and refinement. Then the refined likelihood maps are used in formula 8 for attention propagation.

IV. Evaluation

A. Implementation

In our experiments, we utilize a similar sensor setting as [21] to obtain the visual frames and RFID signals. As shown in Fig 8, a Kinect v2 camera is deployed on the top of an RFID antenna. The antenna is connected to a commercial RFID reader ImpinJ R420. We choose the center of the antenna as the origin O of 3D localization reference system and measure the coordination $(\Delta X, \Delta Y, \Delta Z)$ of the depth sensor on the Kinect. Thus, the XYZ 3D point cloud in Kinect reference system could be translated by $(\Delta X, \Delta Y, \Delta Z)$ to obtain the coordination of pixels in the RF reference system.

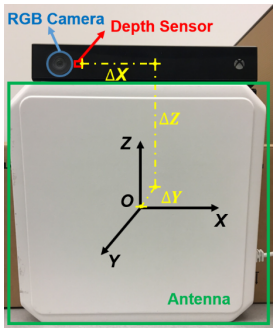


Fig. 8. Deployment of sensors

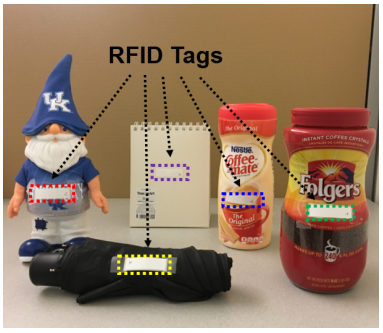


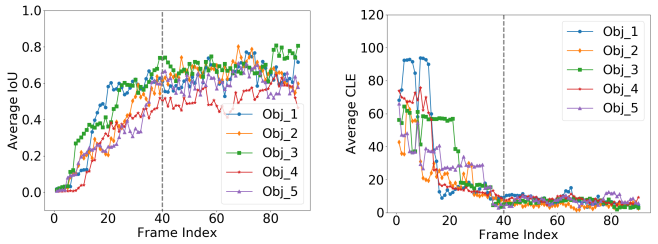
Fig. 9. Examples of target objects.

In our implementation, the FlowNet [9] module for optical flow estimation is implemented with Tensorflow, and we used the loss functions and parameter settings suggested by [25] for training. The neural network is first pre-trained on the synthetic dataset FlyingChairs [9] without using the ground truth data, then fine-tuned on Kinect video frames collected arbitrarily in dynamic environments. The Top-down attention module is also implemented jointly with FlowNet in Tensorflow, but no training is required for this part. The whole system is tested with one Titan X GPU and 8 vCPUs @ 2.6 GHz. Without any decent optimization in the implementation, the average overall processing time for each video frame is around $95ms$, which demonstrates the potential of the proposed method to be applied to online tracking systems.

B. Experiment Setup

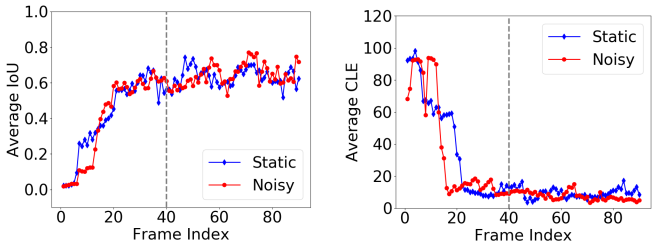
To evaluate the performance of the tracing system, we ask 2 volunteers to move everyday objects continuously with arbitrary traces in front of the sensors. Examples of the objects that we tested are shown in Fig 9. The objects tested are of different shapes, sizes, materials and textures. We stick an RFID tag on each of the objects. When collecting sensing data, the Kinect records the RGB image frames and 3D coordination of the pixels. Meanwhile, the RFID reader records the tag EPCs (considered as the cyber IDs of the targets) and phase information.

Tracing cases: We consider two tracing cases in our evaluation, namely *single moving target tracing* and *multiple moving targets tracing*. In the single moving target tracing case, we conduct the experiments in two totally different environments. One is in a relatively static meeting room with several furniture (e.g., tables and chairs) in it. In this environment, we test tracing of 5 different objects and repeat for 4 times for each object. Besides, to investigate the impact of noise factors such as multipath effects of the RF signals, we also evaluate our system in a noisy and crowded office room, which has narrow



(A) IoU of single target (B) CLE of single target

Fig. 10. The performance of single target tracing. Panel (A) and (B): Average IoU (A) and CLE (B) of each tested target object.



(A) IoU (B) CLE

Fig. 11. Tracing performance of a signal target in noisy environments.

open space, multipath reflectors (tables, chairs, cubicle walls), metal and electronic furniture (cabinet, servers, workstations), various wireless signals (WiFi, LTE), and magnetic fields (whiteboard) in it. We also ask another volunteer to keep walking around to make some dynamic noises. The experiment in such scenario is repeated for 5 times. **Note that both environments are comparable to or more complex than the real world scenarios.**

We also evaluate the system for tracking multiple moving targets and assign the correct ID to each of them in a noisy environment (the office room scenario). Some of the tested targets are of the similar appearance. Thus, a pure vision-based detection system cannot distinguish them.

C. Evaluation Metrics

We use the Intersection over Union (IoU) and Center Location Error (CLE) to evaluate the tracing performance. IoU is calculated as Eq. 9:

$$IoU = \frac{S(B_t \cap B_p)}{S(B_t \cup B_p)}, \tag{9}$$

where $B_t \cap B_p$ and $B_t \cup B_p$ represent the intersection and union of the ground truth bounding box B_t and predicted bounding box B_p of the target object in video frames respectively, and $S(X)$ represents the area of the region X. CLE measures the Euclidean distance (in number of image pixels) between the centers of the ground-truth bounding box and predicted bounding box in pixels, in contrast to the overall input/output frame resolution 512×424. Since TagAttention only outputs the segmentation masks of the targets (i.e., the set of pixels representing the target regions in video frames), we use the smallest unrotated rectangles that cover all the masked pixels in video frames as the predicted bounding boxes B_p. Ideally, we should have used the IoU of object masks as a more precise metric. However, it is rather difficult and time-consuming to manually obtain the ground truth object masks for all video frames. Therefore, we obtain the ground truth bounding box B_t by manually annotating the target objects with an unrotated rectangle.

D. Single Object Tracing

1) Tracing in Static Environment: Fig 10 shows the performance of tracing single target in static scenarios. In Fig 10, plot (A) and (B) show the average IoU and CLE metrics of the five different target objects respectively, where the X axis represents the timestamps of the 90 video frames, and the Y axis represents the average IoU or CLE value.

The evaluation results in Fig 10 illustrate the process in which TagAttention gradually and actively discover the targets and keep tracking them over time. We find TagAttention achieves low IoU scores and high center location errors in the first 20 video frames (at the very beginning frames, the IoUs are always close to 0), showing initially TagAttention cannot track anything as it knows little information about what to trace. This property contrasts to the existing tracking systems, in which they find the targets' location well at the initial stage by human's assistance or an object detection module that is well-trained on large datasets to learn the target. However, we notice the IoU score keeps increasing and the error keeps decreasing until around the $40th$ frame, showing TagAttention can gradually find the location of the targets based on the consistency of the target motion trajectories observed from both sensing channels. Moreover, after around the $40th$ frame, TagAttention becomes confident of the objects' location and mask. Then it keeps tracking the objects for the following frames, yielding high IoUs, low CLEs.

2) Tracing in Dynamic and Narrow Environments: To evaluate the impact of environmental noises, such as multipath effects, to our tracing system, we conduct the tracing experiments in a dynamic and crowded office room. Fig 11 shows the performance in comparison with the tracing results of the same target in the previous static environment. Fig 11 (A) and (B) shows the average IoU and CLE results respectively. From the results, we notice the tracing performances in two different scenarios are equivalent, which shows the system is robust to multipath of the signals. In fact, since TagAttention only estimates the coarse motion of the targets rather than accurate localization using the RF signals, the system does not suffer as much from inaccurate phase measurement. In addition, the smoothing methods introduced in Section 3.2 to preprocess the RF signals and the mask refinement strategies introduced in Section 3.6 also help to minimize the impact of signal noise in real-world scenarios.

However, the system still requires a line-of-sight (LoS) path to guarantee the correctness of formula 1 and assumes the LoS path can dominate the multipath effect. Although extracting the LoS path from the received signal using commercial readers is out of the scope of this paper, we will investigate how existing solutions [39] [35] [34] can improve our system in future work.

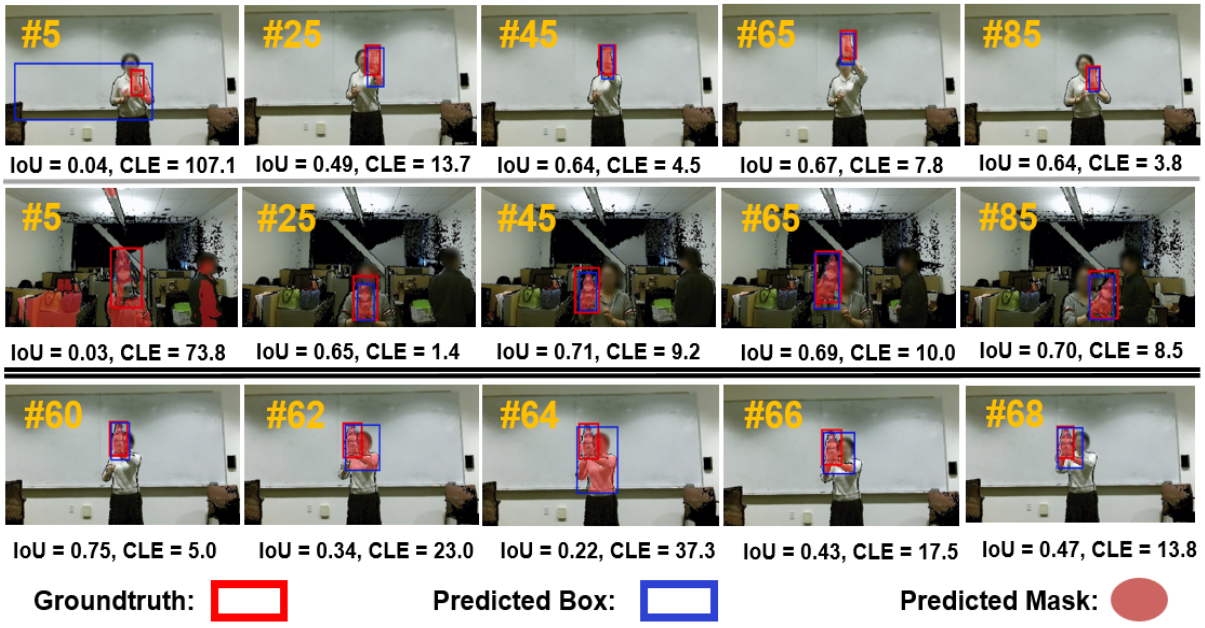
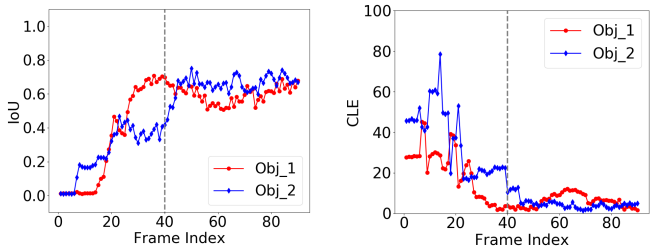


Fig. 12. Examples of single object tracing results.

To better illustrate the actual tracing quality and investigate where the errors come from, we show some selected tracing results of the single object scenarios in Fig 12. Specifically, the first row in Fig 12 shows the meeting room scenario, the second row shows the office scenario, and the third row shows how the system reacts with errors that occur at certain frames[2]. In Fig 12, the number at the left-up corner of each image indicates the frame index in the tracing scenarios. The IoU and CLE of the tracing performance are also presented below each frame image. From Fig 12, we find most tracing errors is caused by the ambiguous boundary between the target and its surroundings. Since TagAttention requires no prior knowledge of the appearance of the target, it cannot distinguish the target and its surrounding body parts (i.e. the hand and wrist of the volunteer) that move consistently with the target. In these cases, the system considers the target as well as part of its surroundings as an entire rigid body. Since the bounding box IoU score is sensitive to the redundant areas, especially for small objects, we observe a low IoU score for these predictions, whereas the tracing performance is still acceptable.

From the third row of Fig 12, we also notice that a sudden decrease of tracing performance occurs at the $62nd$ frame after TagAttention has already found an accurate position of the targets. We find this phenomenon happens occasionally during tracking. It is mainly caused by the flow warping error in the tracking module of TagAttention. Usually, in such cases, the optical flow measured by FlowNet is inaccurate at a certain frame. Consequently, when propagating the attention maps, the target image region "leaks the attention values" to some irrelevant image pixels. Then in the mask refinement module, the tracer mistakenly considers these irrelevant pixels are of the same rigid body as the target object because these pixels are also spatially close the target. Hence, it starts tracking more body parts than the target rigid body (for example, the entire human body in frame # 64 in the last row of Fig 12). However, after a few frames, as the irrelevant body parts move inconsistently with the target, the attention values of corresponding pixels decrease quickly. Then the tracer can recapture the accurate position of the target and track only the target part (for example, the $66th$ and $68th$ frame in the last row of Fig 12).

[2]The black dots and shadows in the video frames are caused by multiple reasons when using Kinect for sensing, including the limitation of Kinect's sensing range, reflection or refraction of infrared light by the materials, and occlusion boundaries among objects. All images are cropped to enlarge the target and fit the template.



(A) Average IoU (B) Average CLE

Fig. 13. Tracing results of the two-object scenarios

E. Multiple Object Tracing

TagAttention can trace multiple mobile targets simultaneously by their cyber IDs without introducing much extra computation. In fact, the most computationally intensive part in TagAttention is the optical flow module, which estimates the optical flow map through a deep neural network. However, the optical flow of the video can be reused by any top-down attention parts to detect and track different targets. Specifically, when the RFID tags of multiple targets are detected, their

Fig. 14. Examples of multi-object tracing

EPCs and the corresponding phase signals are recorded and processed independently. After the optical flow and the pixel-wise motion map of the video frames are calculated, TagAttention can use these phases signals to compute the attention values of the pixels and produce their corresponding likelihood maps in parallel.

We evaluate the performance of TagAttention in multiple target tracking scenarios. Fig 13 shows the average IoU and CLE scores of different targets in the two-object tracing scenarios. From Fig 13, we find the performance of TagAttention for each individual target is similar to the single object tracing cases. Specifically, the tracer takes less than 35 frames to discover the accurate location of each individual targets and keep tracking them for the following frames.

In addition, we show some selected tracing frames of two-object and four-object tracing scenarios in Fig 14. At the $5th$ frame, the tracer cannot recognize and detect any targets. After more motion data is collected, TagAttention produces fine-grained bounding box and segmentation mask for each target, and labels the targets by the corresponding tag IDs. Especially in the four-object scenarios, we find the system can distinguish the two cylindrical bottles (ID_2 and ID_3) by their IDs, even though the two bottles are very similar in appearance.

V. Discussion and Future Work

Tracing of the mobile target without human's supervision is a critical but challenging problem in wireless sensing and robotics. TagAttention solves detecting and tracking mobile targets with RFID tags in an active manner. Meanwhile, we acknowledge the following limitations of the current system and propose possible solutions.

Static target labeling: With current system deployment, detecting static targets seems difficult to TagAttention. By setting the minimal velocity sensitivity threshold v_0, TagAttention ignores the slower motion as we hope the system to be resistant to environmental noises. To enable the system to detect static or slowly-moving targets, we consider using multiple antennas deployed separately to coarsely localized the target and then projecting the location to the camera reference system to obtain the fine-grained location of the target.

Limited sensing range of sensors: Our system is also limited by the sensing range of the sensors. For example, commercial RGB-D sensors like Kinect can only measure the depths of voxels within around 5 meters, while commercial RFID readers read tags at the maximal distance of around 10 meters. However, our proposed attention-based RF-vision fusion model is independent with the sensing technologies. Thus, it is still possible to use noncommercial readers and state-of-the-art sensing technologies [40] to push the sensing range limit of RFID. In addition, the sensing range of RGB-D cameras is limited by the Infrared sensor rather than the RGB camera. Hence, we may use multiple RF antennas and only the RGB camera to overcome the sensing range limit of the overall system.

Object rotation and blockage: Our current system requires the LoS path of the RF signal, which limits its applications in the scenarios where target objects are significantly rotated or blocked by temporary obstacles. The problem can be potentially resolved by deploying multiple antennas and locating the tag with the orientation-aware model [18]. In addition, we can also consider the correlation of visual features of the discovered target over the consecutive frames. For example, since TagAttention can already find the correct mask before the rotation or blockage happens, we may use optical correlation filters [2] [19], which are pretrained on conventional video tracking datasets, to continuously track the targets when they are rotated or partially blocked.

VI. Conclusion

This paper presents TagAttention, a mobile object tracing system by vision-RFID fusion without the knowledge of object appearances. Different from all existing systems, TagAttention can actively "discover" the target objects without any pre-knowledge of the objects' appearance, precisely identify the objects even that they belong to the same generic categories, and track the targets instantly when they appear in the video frames.

VII. Acknowledgments

This work is partially supported by National Science Foundation Grant 1717948 and 1750704. We thank Roberto Manduchi and the anonymous reviewers for their suggestions and comments.

REFERENCES

[1] M. Andriluka, U. Iqbal, A. Milan, E. Insafutdinov, L. Pishchulin, J. Gall, and B. Schiele. Posetrack: A Benchmark for Human Pose Estimation and Tracking. In *Proceedings of IEEE CVPR*, 2018.

[2] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual Object Tracking Using Adaptive Correlation Filters. In *Proceedings of IEEE CVPR*, 2010.

[3] M. Bouet and A. L. Dos Santos. RFID tags: Positioning principles and localization techniques. In *2008 1st IFIP Wireless Days*, pages 1–5. IEEE, 2008.

[4] H. Cai, G. Wang, X. Shi, J. Xie, M. Wang, and C. Qian. When Tags 'Read' Each Other: Enabling Low-cost and Convenient Tag Mutual Identification. In *Proceedings of IEEE ICNP*, 2019.

[5] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang. Segflow: Joint Learning for Video Object Segmentation and Optical Flow. In *Proceedings of IEEE ICCV*, 2017.

[6] L.-X. Chuo, Z. Luo, D. Sylvester, D. Blaauw, and H.-S. Kim. RF-Echo: A Non-Line-of-Sight Indoor Localization System Using a Low-Power Active RF Reflector ASIC Tag. In *Proceedings of ACM MobiCom*, 2017.

[7] C. E. Connor, H. E. Egeth, and S. Yantis. Visual Attention: Bottom-up Versus Top-down. *Current biology*, 14(19):R850–R852, 2004.

[8] H. Ding, J. Han, C. Qian, F. Xiao, G. Wang, N. Yang, W. Xi, and J. Xiao. Trio: Utilizing tag interference for refined localization of passive RFID. In *Proceedings of IEEE INFOCOM*, 2018.

[9] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning Optical Flow with Convolutional Networks. In *Proceedings of IEEE ICCV*, 2015.

[10] C. Duan, X. Rao, L. Yang, and Y. Liu. Fusing RFID and Computer Vision for Fine-grained Object Tracking. In *Proceedings of IEEE INFOCOM*, 2017.

[11] EPCglobal. EPC^{TM} radio-frequency identity protocols class-1 generation-2 UHF RFID protocol for communications at 860 MHz–960 MHz, 2005.

[12] D. Gordon, A. Farhadi, and D. Fox. Re ^{3}: Real-Time Recurrent Regression Networks for Visual Tracking of Generic Objects. *IEEE Robotics and Automation Letters*, 3(2):788–795, 2018.

[13] J. Han, H. Ding, C. Qian, D. Ma, W. Xi, Z. Wang, Z. Jiang, and L. Shangguan. CBID: A Customer Behavior Identification System Using Passive Tags. In *Proceedings of IEEE ICNP*, 2014.

[14] J. Han, C. Qian, X. Wang, D. Ma, J. Zhao, W. Xi, Z. Jiang, and Z. Wang. Twins: Device-free object tracking using passive tags. *IEEE/ACM Transactions on Networking*, 2016.

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of IEEE ICCV*, 2017.

[16] J. Impin. Speedway Revolution Reader Application Note: Low Level User Data Support. *Speedway Revolution Reader Application Note*, 2010.

[17] M. Jaimez, C. Kerl, J. Gonzalez-Jimenez, and D. Cremers. Fast Odometry and Scene Flow from RGB-D Cameras Based on Geometric Clustering. In *Proceedings of IEEE ICRA*, 2017.

[18] C. Jiang, Y. He, X. Zheng, and Y. Liu. Orientation-aware Rfid Tracking with Centimeter-level Accuracy. In *Proceedings of ACM/IEEE IPSN*, 2018.

[19] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. High Performance Visual Tracking With Siamese Region Proposal Network. In *Proceedings of IEEE CVPR*, 2018.

[20] H. Li, E. Whitmire, A. Mariakakis, V. Chan, A. P. Sample, and S. N. Patel. IDCam: Precise Item Identification for AR Enhanced Object Interactions. In *Proceedings of IEEE RFID*, 2019.

[21] H. Li, P. Zhang, S. Al Moubayed, S. N. Patel, and A. P. Sample. Id-match: A Hybrid Computer Vision and Rfid System for Recognizing Individuals in Groups. In *Proceedings of ACM CHI*, 2016.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of IEEE CVPR*, 2015.

[23] Y. Ma, X. Hui, and E. C. Kan. 3D Real-time Indoor Localization via Broadband Nonlinear Backscatter in Passive Devices with Centimeter Precision . In *Proceedings of ACM MobiCom*, 2016.

[24] R. Mandeljc, S. Kovačič, M. Kristan, J. Perš, et al. Tracking by Identification Using Computer vision and Radio. *Sensors*, 13(1):241–273, 2012.

[25] S. Meister, J. Hur, and S. Roth. UnFlow: Unsupervised Learning of Optical Flow with A Bidirectional Census Loss. In *Proceedings of AAAI*, 2018.

[26] R. Miesen, F. Kirsch, and M. Vossiek. Holographic localization of passive UHF RFID transponders. In *Proceedings of IEEE RFID*, 2011.

[27] A. Parr, R. Miesen, and M. Vossiek. Inverse sar approach for localization of moving RFID tags. In *Proceedings of IEEE RFID*, 2013.

[28] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-time Object Detection with Region Rroposal Networks. In *Proceedings of NIPS*, 2015.

[29] L. Shangguan, Z. Li, Z. Yang, M. Li, and Y. Liu. OTrack: Order tracking for luggage in mobile RFID systems. In *Proceedings of IEEE INFOCOM*, 2013.

[30] L. Shangguan, Z. Yang, A. X. Liu, Z. Zhou, and Y. Liu. Relative localization of RFID tags using spatial-temporal phase profiling. In *Proceedings of USENIX NSDI*, 2015.

[31] L. Shangguan, Z. Zhou, X. Zheng, L. Yang, Y. Liu, and J. Han. ShopMiner: Mining Customer Shopping Behavior in Physical Clothing Stores with COTS RFID Devices. In *Proceedings of ACM SenSys*, 2015.

[32] D. Sun, S. Roth, and M. J. Black. Secrets of Optical Flow Estimation and Their Principles. In *Proceedings of IEEE CVPR*, 2010.

[33] D. Sun, E. B. Sudderth, and H. Pfister. Layered RGBD Scene Flow Estimation. In *Proceedings of IEEE CVPR*, 2015.

[34] G. Wang, C. Qian, K. Cui, H. Ding, H. Cai, W. Xi, J. Han, and J. Zhao. A (Near) Zero-cost and Universal Method to Combat Multipaths for RFID Sensing. In *Proceedings of IEEE ICNP*, 2019.

[35] G. Wang, C. Qian, J. Han, W. Xi, H. Ding, Z. Jiang, and J. Zhao. Verifiable Smart Packaging with Passive RFID . In *Proceedings of ACM UBICOMP*, 2016.

[36] G. Wang, C. Qian, L. Shangguan, H. Ding, J. Han, N. Yang, W. Xi, and J. Zhao. HMRL: Relative Localization of RFID Tags with Static Devices. In *Proceedings of IEEE SECON*, 2017.

[37] J. Wang and D. Katabi. Dude, where's my card?: RFID positioning that works with multipath and non-line of sight. In *Proceedings of ACM SIGCOMM*, 2013.

[38] J. Wang, J. Xiong, H. Jiang, X. Chen, and D. Fang. D-Watch: Embracing "bad " Multipaths for Device-Free Localization with COTS RFID Devices. In *Proceedings of ACM CoNEXT*, 2016.

[39] J. Wang, J. Xiong, H. Jiang, X. Chen, and D. Fang. D-watch: Embracing "Bad" Multipaths for Device-free Localization with COTS RFID Devices. *IEEE/ACM Transactions on Networking (TON)*, 25(6):3559–3572, 2017.

[40] J. Wang, J. Zhang, R. Saha, H. Jin, and S. Kumar. Pushing the Range Limits of Commercial Passive RFIDs. In *Proceedings of USENIX NSDI*, 2019.

[41] S. Wug Oh, J.-Y. Lee, K. Sunkavalli, and S. Joo Kim. Fast Video Object Segmentation by Reference-Guided Mask Propagation. In *Proceedings of IEEE CVPR*, 2018.

[42] L. Xie, C. Wang, Y. Bu, J. Sun, Q. Cai, J. Wu, and S. Lu. TaggedAR: An RFID-based Approach for Recognition of Multiple Tagged Objects in Augmented Reality Systems. *IEEE Transactions on Mobile Computing*, 2018.

[43] L. Yang, Y. Chen, X. Li, C. Xiao, M. Li, and Y. Liu. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *Proceedings of ACM MobiCom*, 2014.

[44] L. Yang, Q. Lin, X. Li, T. Liu, and Y. Liu. See through walls with COTS RFID system! In *Proceedings of ACM MobiCom*, 2015.

[45] S. Yoo, K. Yun, J. Y. Choi, K. Yun, and J. Choi. Action-Decision Networks for Visual Tracking with Deep Reinforcement Learning. In *Proceedings of IEEE CVPR*, 2017.

[46] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba. RF-Based 3D Skeletons. In *Proceedings of ACM SIGCOMM*, 2018.

[47] Y. Zhao, Y. Liu, and L. M. Ni. VIRE: Active RFID-based localization using virtual reference elimination. In *Proceedings of IEEE ICPP*, 2007.