RFusion: Robotic Grasping via RF-Visual Sensing and Learning

Tara Boroushaki, Isaac Perper, Mergen Nachin, Alberto Rodriguez, Fadel Adib
{tarab,iperper,mergen,albertor,fadel}@mit.edu
Massachusetts Institute of Technology

ABSTRACT

We present the design, implementation, and evaluation of RFusion, a robotic system that can search for and retrieve RFID-tagged items in line-of-sight, non-line-of-sight, and fully-occluded settings. RFusion consists of a robotic arm that has a camera and antenna strapped around its gripper. Our design introduces two key innovations: the first is a method that geometrically fuses RF and visual information to reduce uncertainty about the target object's location, even when the item is fully occluded. The second is a novel reinforcement-learning network that uses the fused RF-visual information to efficiently localize, maneuver toward, and grasp target items. We built an end-to-end prototype of RFusion and tested it in challenging real-world environments. Our evaluation demonstrates that RFusion localizes target items with centimeter-scale accuracy and achieves 96% success rate in retrieving fully occluded objects, even if they are under a pile. The system paves the way for novel robotic retrieval tasks in complex environments such as warehouses, manufacturing plants, and smart homes.

CCS CONCEPTS

• **Computer systems organization** → **Robotics**; **Sensor networks**; • **Computing methodologies** → **Robotic planning**.

KEYWORDS

Robotic Grasping, Mechanical Search, RF-visual Sensing, Radio Frequency Perception, Reinforcement Learning, RFID Localization

ACM Reference Format:

Tara Boroushaki, Isaac Perper, Mergen Nachin, Alberto Rodriguez, Fadel Adib. 2021. RFusion: Robotic Grasping via RF-Visual Sensing and Learning. In *The 19th ACM Conference on Embedded Networked Sensor Systems (SenSys'21), November 15–17, 2021, Coimbra, Portugal.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3485730.3485944

1 INTRODUCTION

The past few years have witnessed mounting interest in sensing technologies for robotic perception and automation. Remarkable progress in vision systems has enabled robots to perceive, locate, and grasp items in unstructured environments like warehouses, homes, and manufacturing plants [12, 33, 54]. This progress has motivated large companies like Amazon, Uniqlo, and FedEx to adopt piece-picking robots [6, 16, 31], and it has led to the emergence of various piece-picking robotic start-ups with a collective valuation of multi-billion dollars [27]. All of this is driven by the enormous potential of picking robots in applications spanning e-commerce fulfillment, warehouse automation, and agile manufacturing.

A standard robotic task across many of these use cases involves retrieving target items in cluttered environments [4, 5, 37, 51]. E-commerce robots need to retrieve customer orders from under a pile. Similarly, furniture assembly robots need to retrieve certain tools (e.g., wrench), before they can start assembling furniture [22]. The process of searching for and retrieving a target item is efficient when the item is visible. However, if the target item is fully occluded, robotic systems that rely on cameras cannot perceive it, and resort to a time consuming search process.

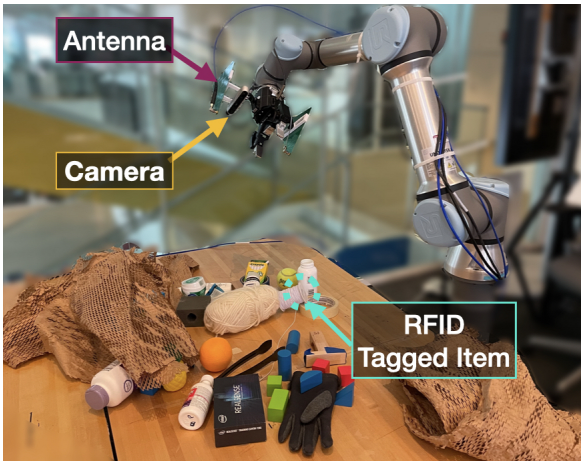


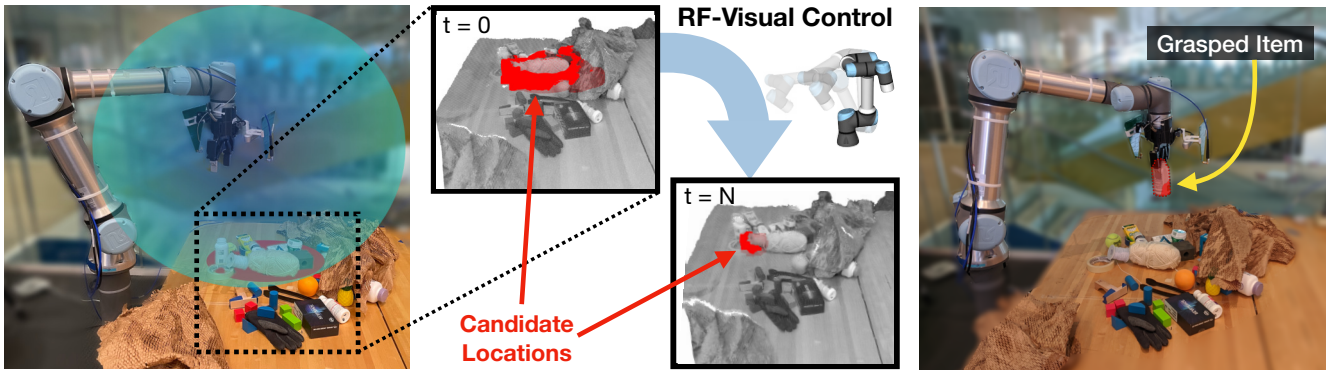
Figure 1: RFusion. The system fuses RF and visual sensor data (from wrist-mounted camera and antenna) to locate, maneuver toward, and grasp items in line-of-sight, non-line of sight objects, and fully-occluded settings.

Motivated by the desire to efficiently search for and retrieve fully occluded objects, researchers have considered the use of radio frequency (RF) localization [2, 7, 10, 45]. Because RF signals can traverse everyday occlusions, these systems can identify and locate items of interest through occlusions and instruct the robot to navigate toward them, making the search process more efficient. However, existing systems that leverage RF localization for grasping make restrictive assumptions (about the objects and the environment) which limit their practicality. For example, state-of-the-art proposals like RFGrasp [2] and RFCompass [45] can only work with simple objects and require prior knowledge of the object's shape and/or orientation. In addition, these systems require a separate, dedicated infrastructure for RF localization and calibration. As a result, they can only work in constrained environments that have already been instrumented with the required infrastructure. Our work is motivated by a similar desire to bring the benefits of RF localization to robotic grasping, but aims to deliver an infrastructure-less, shape-independent, and orientation-independent grasping system.

We present RFusion, a fully-integrated robot that enables practical and efficient grasping in line-of-sight, non-line-of-sight, and fully-occluded settings. Similar to past work that leverages RF localization, RFusion assumes that target objects are tagged with RFIDs and uses the RFID signals to sense the objects of interest through occlusions. Unlike past systems, it is the first to integrate both the RF localization module and the camera into the robotic arm itself (as shown in Fig. 1), eliminating the requirement for instrumenting the environment with a separate infrastructure as well as any associated calibration process. With the camera and RFID localization antenna both integrated onto the robot's end-effector, the system can exploit the arm's mobility to localize and grasp the target item

SenSys'21, November 15–17, 2021, Coimbra, Portugal

ACM ISBN 978-1-4503-9097-2/21/11.
https://doi.org/10.1145/3485730.3485944



a) Geometric Fusion **b)** Narrowing Candidate Locations **c)** Grasping

Figure 2: RF-Visual Sensor Fusion, Control, and Grasping. (a) shows geometric fusion, combining RF measurements with visual point clouds to identify candidate locations for the target RFID-tagged item. (b) demonstrates how RL-based RF-Visual control narrows down the candidate locations (marked in red). (c) illustrates the robot successfully grasping the target item after decluttering its vicinity.

independent of its shape or orientation.

The key challenge in designing RFusion arises from the uncertainty about the target object's location and the environment. In particular, since the target object may be hidden or outside the camera's field of view, the robot can not visually perceive it. Moreover, since the robot has a single wrist-mounted antenna (not a separate RFID localization infrastructure), it cannot directly localize the RFID. In principle, one could instruct the robot to maneuver its arm to scan the 3D environment in order to accurately localize the RFID; however, such an approach would be time-consuming and inefficient [7]. Yet, in the absence of accurate localization, the uncertainty region would be larger and the robot would need to grasp many objects in a pile before it finally picks up the target object [4, 5]. In other words, the uncertainty again results in a time-consuming and inefficient retrieval process.

RFusion overcomes these challenges via two new primitives:

(a) Dense RF-Visual Geometric Fusion (§3): RFusion's first primitive aims to reduce uncertainty about the target item's location by iteratively fusing RF and visual information. In particular, rather than treating RF and vision as independent perception modules, RFusion geometrically fuses them. Fig. 2(a) illustrates this idea. The antenna mounted on the robot's wrist estimates the time-of-flight to the RFID on the target item and maps it to a round-trip distance (by accounting for the speed of propagation of RF signals). Given the round-trip distance from the antenna to the target item, we can localize the RFID to a spherical surface centered around the antenna, depicted as a transparent blue sphere in Fig. 2(a). In parallel, RFusion uses the depth map obtained from its wrist-mounted camera to construct a 3D point cloud, corresponding to the occupied regions in its visual field of view, shown in Fig. 2(b). By geometrically intersecting the RF-computed sphere with its 3D point cloud, RFusion can narrow down the target object's location to only a handful of candidates, shown in red in Fig. 2(a-b). RFusion extends this high-level idea with techniques to account for regions outside its field of view and for the resolution of localization.

(b) RF-Visual Reinforcement Learning (§4): Given the candidate locations from geometric fusion, one might assume that instructing the robot to simply approach the region where the candidate locations are most concentrated may be optimal. In practice, such an approach may lead to poor localization accuracy due to *Dilution of Precision (DoP)*. DoP is a well-known concept in GPS positioning and is the main reason why GPS does not perform well in canyons (or in streets with tall buildings). When a GPS receiver is in a canyon, it only obtains measurements from satellites close to each other in the sky, resulting in much poorer localization precision than when it obtains measurements from satellites that are further apart from each other. A similar problem arises for RFusion. If the robot moves its gripper directly toward the candidate locations, its vantage points (analogous to satellites) would be close to each other, resulting in poor localization accuracy. Yet, if the robot moves its gripper further away from the candidate locations to obtain better localization accuracy, it would increase its overall trajectory for grasping since it needs to make its way back to the object for grasping. This results in poor overall efficiency.

To address this tradeoff, we cast it as a trajectory minimization problem, and solve it using a custom-designed RF-visual reinforcement learning network. Reinforcement learning is suitable for this problem since it enables minimizing a cumulative cost (trajectory) after multiple actions (vantage points). To simplify the learning problem, RFusion decomposes the candidate locations into non-overlapping uncertainty regions that account for the resolution of RF sensing and the camera's depth map. This decomposition enables us to efficiently train the network entirely in simulation on a large number of scenarios. Once trained, the network can be used by a physical robot to identify the optimal next vantage point, instruct its gripper to move to the corresponding location, and iterate through the process, narrowing down the candidates until it has localized the target object. This can be seen in Fig. 2(b) where the candidate regions (in red) shrink over time as the robot collects measurements from new vantage points.

We build on the above two primitives to design an end-to-end system capable of grasping complex objects, even if they are occluded under a pile. RFusion uses the vantage points identified by its learning network to localize the target item's RFID. Then, it feeds the location estimate as an attention mechanism to a vision-based grasping network. If the target item is under a pile, the robot can automatically remove items stacked above it, set them aside, and proceed to retrieving the item of interest. Once RFusion successfully retrieves the target object (as shown in Fig. 2(c)), it senses the proximity of the object's RFID to its wrist-mounted antenna

and declares task completion. We elaborate on these techniques in §4.3-§5 and describe how RFusion can recover from errors arising from RF polarization mismatch and outlier measurements.

We built an end-to-end prototype of RFusion. Our implementation reproduces a state-of-the-art RFID localization system [30] on BladeRF software radios [38]. The system also uses a UR5e robot [44] with an Intel RealSense D415 depth camera and log-periodic antennas mounted on its wrist (as shown in Fig. 1).

We ran over 400 real-world experimental trails to evaluate the system. Our evaluation demonstrates the following:

- RFusion can accurately grasp complex items in line-of-sight and non-line-of-sight settings, under occlusions, and across different orientations. It achieves centimeter-scale median localization accuracy in each of the x/y/z dimensions, and succeeds in 96% of trials across different scenarios.
- We implemented a scanning baseline using RFusion's wrist-mounted antenna (i.e., an antenna-array baseline). The baseline achieves the same localization accuracy as RFusion's full implementation, albeit achieves only 50% of RFusion's efficiency (i.e., it requires 2× more travel distance for localization and grasping).
- We also performed ablation studies to assess the benefits of RF-visual's geometric fusion and reinforcement learning network. Our studies demonstrate that these components contribute equally to RFusion's efficiency gains over the baseline.

Demo Video: rfusion.media.mit.edu

Contributions: RFusion is the first robotic system that grasps occluded objects using a single fully-integrated in-hand RF-visual end-effector. The system introduces two key innovations – dense RF-visual geometric fusion and RF-visual reinforcement learning – to plan efficient trajectories for localization, grasping, and retrieval. The paper also contributes a prototype implementation and evaluation of RFusion in practical real-world environments spanning line-of-sight, non-line-of-sight, and fully-occluded settings.

2 SYSTEM OVERVIEW

RFusion is a system that can efficiently search for and retrieve RFID-tagged items in its workspace. It consists of a robotic arm with a wrist-mounted antenna and a wrist-mounted depth camera (RGB-D). The system works with standard passive UHF RFID tags and common robotic arms with 6 degrees of freedom. It can retrieve RFID-tagged items in line-of-sight as well as those occluded under a pile or in low-lighting conditions.

RFusion can operate with any number of UHF RFID tags in the environment. We focus on scenarios where the robot needs to retrieve a specific item (or an ordered list of items); thus, we assume that the robot knows the identifier of the target item's RFID. In practice, the mapping between an item and its identifier is listed in (online) databases provided by the manufacturer or distributor. We note that RFusion does not need to know exactly where the RFID is attached to the item, nor does it know where the RFID or target item are in its workspace.

The system starts by using its wrist-mounted antenna to selectively query the RFID on the target item. It then uses the RFID's measured response to compute the round-trip distance to the tag by leveraging state-of-the-art RFID positioning techniques. Since a single round-trip distance is not sufficient for localization, the robot fuses RF and visual information in order to efficiently localize, maneuver toward, and grasp the target object. It operates in 3 steps:

- **Dense RF-Visual Geometric Fusion (§3):** Given the round-trip distance to the RFID, the robot maps that distance to a spherical ring centered around the wrist-mounted antenna. Subsequently, it geometrically intersects this spherical ring with the RGB-D data obtained from the wrist-mounted camera, resulting in a list of candidate locations.
- **RF-Visual Reinforcement Learning (§4):** Next, the robot needs to move its gripper to a new location in order to collect new RF and visual measurements. To do this, we trained a reinforcement learning network that uses the history of RF and visual measurements to determine the optimal next vantage point to which the gripper should move. The robot moves to this new location, takes measurements, and repeats this step until it has sufficient confidence in its estimate location of the RFID-tagged item.
- **RF-Visual Grasping (§5):** Once a sufficiently-accurate location has been determined, RFusion uses the location estimate to grasp the object. After grasping, the wrist-mounted antenna can make an additional RFID measurement to verify that the target item has indeed been successfully grasped, and if not, attempt more grasps until the item of interest has been picked up.

3 DENSE RF-VISUAL GEOMETRIC FUSION

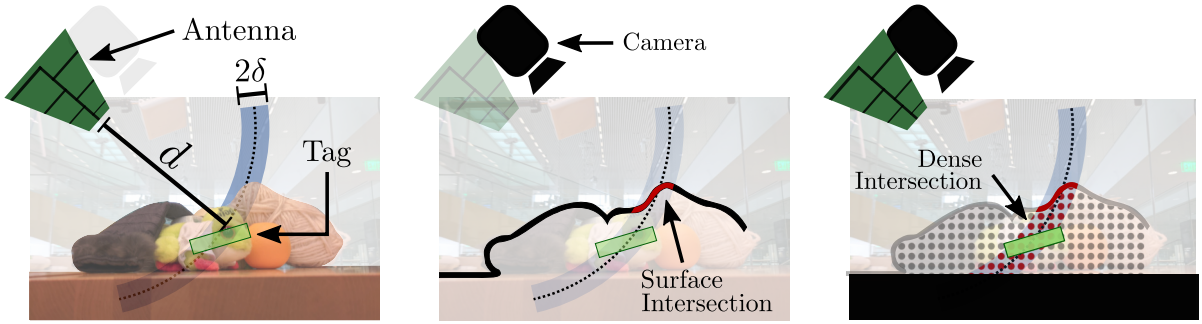
In this section, we describe RFusion's first component. This component fuses RF and visual information to reduce the uncertainty about the target item's location, even when the item is fully occluded. For simplicity of exposition, we describe RF-visual fusion using the example shown in Fig. 3. The figure depicts a scenario with a pile of objects on a table. The RFID-tagged target item (depicted using a green rectangle) is in the middle of the pile, hence occluded from the camera's field of view. The figure also shows the wrist-mounted antenna (pictured using a green trapezoid) and wrist-mounted camera. We describe the fusion process in three steps:

Step 1: RF-based Spherical Ring: First, RFusion uses the wrist-mounted antenna to estimate the time-of-flight to the RFID on the target item (similar to [30]). It then maps the time-of-flight to the round-trip distance d by accounting for the speed of propagation of RF signals. This distance measurement limits the tag's possible location to the surface of a sphere centered on the antenna. Part of this sphere is shown as the dotted arc in Fig. 3a. Since the distance estimate has a certain resolution δ,[1] it constrains the tag's location to the blue 3D region shown in the figure. Geometrically, this means that we can narrow the tag's position P_{Tag} to a spherical ring centered around the in-hand antenna's location P_{Ant}, such that the true antenna-tag round-trip distance must be within δ of the estimated round-trip distance d. Eq. 1 formalizes this idea:

$$\left|2\|P_{Tag} - P_{Ant}\| - d\right| < \delta \quad (1)$$

Step 2: Constructing a Dense 3D Point Cloud: Although RFusion can narrow the possible tag region down to the blue sphere in Fig. 3a, this region can be large. Next, RFusion fuses visual information to further reduce its uncertainty about the target's location. The wrist-mounted camera captures a depth image of the work space and maps this image to a 3D point cloud. This point cloud is depicted with a thick black line in Fig 3b. Note that because

[1] δ represents that uncertainty of a given RF measurement. We choose 3.5cm which corresponds to the 90th percentile of the round-trip accuracy of the system



(a) RF Distance Measurement **(b) Surface Depth Map** **(c) Dense Point Cloud**

Figure 3: Dense RF-Visual Geometric Fusion. (a) The antenna takes a distance measurement d which has a margin of error δ and maps it to a spherical ring. (b) The depth camera is only able to observe the surface points of a pile, so the surface intersection region does not include the tagged item. (c) The known region between the surface and the largest plane (the table) is filled to create a dense point cloud. Now the intersection region includes the location of the tagged item.

the figure only shows a cross-sectional view, the intersection only appears as a line. One challenge is that the camera only obtains what is called a 2.5D image, which measures the distance to the closest surface for each of its pixels. In principle, RFusion can directly intersect this surface point cloud with the blue spherical ring, resulting in the intersection region represented in red. However, such an approach would allow only finding tag candidate locations that are already in the line-of-sight of the camera and would miss all objects that are behind occlusions (such as the green tag here which sits below the surface of the pile).

To address this problem, RFusion extracts the largest plane in its visual field of view, which, in our case, corresponds to the tabletop as shown in the bottom of Fig. 3c in black. This plane represents the furthest possible surface on which the target item can be. Fig. 3c shows how RFusion transforms its original surface point cloud into a dense point cloud by filling in the empty space from the surface point cloud to the largest plane (tabletop). The largest plane surface is also extended to allow intersections outside the field of view; this enables RFusion to localize objects not in in its initial field of view. We now have a dense point cloud $p_{ptcloud}$ that contains all candidate locations based on the visual information.

Step 3: Dense RF-Visual Fusion: Next, RFusion geometrically intersects the spherical ring from Eq. 1 with the dense point cloud to identify all the candidate locations for the RFID-tagged item. The intersection is depicted using the dotted red region in Fig. 3c, demonstrating a significant reduction in the robot's uncertainty about the target object's location. Mathematically, we can narrow down the tag's location to the set I of all points $p_{ptcloud}$ that satisfy:

$$\forall p_{ptcloud} \in I: \ \left|2\|p_{ptcloud} - P_{Ant}\| - d\right| < \delta \tag{2}$$

We make two additional remarks:

- In very simple scenarios, the geometric fusion algorithm alone may be sufficient to localize the target item. For example, if the table has a small number of target items and the spherical ring intersects with only one of them, RFusion can localize the item immediately using geometric fusion. In contrast, neither vision alone nor RF alone is sufficient to localize in one shot.
- RFusion performs two more optimizations to reduce the computational complexity of the subsequent stages. Specifically, it sub-samples the dense point cloud by 60% relative to the highest possible resolution of the camera. Moreover, to eliminate outliers from errors in the vision sensor, it clusters the viable point cloud intersections I using k-mean clustering.[2] Instead of all points in I, the set of cluster centers C represents the possible item locations in the subsequent stages of RFusion's design.

4 RF-VISUAL REINFORCEMENT LEARNING

In the previous section, we discussed how RFusion uses geometric fusion to narrow down the possible object location to a region of candidates. However, to grasp and retrieve an item, a single accurate location is needed. Hence, RFusion needs to collect measurements from different vantage points. One approach is to simply instruct the robot to move toward the center of the region containing the candidate locations, and to collect measurements along the way. However, such an approach would result in bad localization accuracy due to the narrow spacing between antenna measurement positions. Another approach is to maximize the antenna aperture by obtaining measurements from the furthest possible locations in the workspace. While this will improve the localization accuracy, it will also increase the traveled distance to the grasping point, making the system inefficient. On top of this, RFusion can only partially observe the environment through its wrist mounted camera, which yields additional uncertainty about the environment.

To address these problems, RFusion introduces a reinforcement learning (RL) framework that uses visual data and RF measurements to identify an efficient trajectory for collecting vantage points. A general RL framework uses an agent that takes observations of the environment, determines an appropriate action, and applies that action in the environment. In this section, we first explain how RFusion leverages RF-Visual Encoding to enable sample-efficient training of the RL agent. After that, we discuss how this encoding is factored into the RL Network for training. Finally, we will describe the localization process based on visual and RF data.

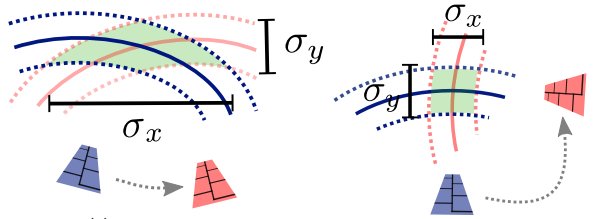
4.1 RF-Visual Encoding

The goal of RFusion's RL agent is to choose optimal vantage points to take measurements from in order to minimize the traveled distance prior to grasping and maximize the localization accuracy. At a high level, this means that the RL agent should find antenna positions that both move the robot arm closer to the target item (whose location isn't fully known) and reduce the localization uncertainty.

[2]Empirically, we found that $k = 4$ performs well. Note that the clusters change across subsequent fusion iterations.

To efficiently train the RL agent on this task, we developed a feature representation of the environment's observation that encodes both visual and RF data. Specifically, we have three feature categories that provide the necessary information for optimizing the trajectory: 1) RF Features 2) Visual Features and 3) Positional Features.

RF Features: To obtain an accurate position estimate, one needs to take many distance measurements over a wide aperture due to dilution-of-precision (DOP) [11, 46]. In Fig. 4, two distance measurements from two vantage points are illustrated in red and blue. The solid lines represent the distance measurements (d_1, d_2), and the dotted lines represents the error margin of δ. The green area in the middle represents the uncertainty about the tag location that is caused by the resolution of distance estimation. The uncertainty can be approximated by (σ_x, σ_y) along x and y axes. In Fig. 4(a), the distance measurements are obtained from nearby vantage points and the uncertainty area is large. In contrast, in Fig. 4(b), the distance measurements are obtained from further vantage points, so the uncertainty area is smaller. Here, we illustrated the DoP concept in 2D for simplicity, but it easily extends to 3D.



(a) Poor DOP (b) Good DOP

Figure 4: RF-DOP. Dilution-of-precision measures the uncertainty of positioning due to the antenna vantage points. Solid lines represent the distance measurements (d_1, d_2,) and dotted lines represent the error margin δ. The green areas represent the uncertainty about the tag location that is caused by the possible errors in measurements. In (a), the antenna vantage points are close, leading to a large uncertainty σ_x. (b) shows how well located vantage points lead to a good DOP and lower uncertainty.

Our above formulation described the DOP assuming we have full knowledge of the tagged item's ground-truth location. However, when we are optimizing the robot's trajectory for localization, the tagged item's location is unknown. To deal with this uncertainty, we approximate the DOP using the geometric center of the candidates identified using RF-visual fusion in §3.[3] The DOP is composed of three values, $(\sigma_x, \sigma_y, \sigma_z)$, which represent the RF uncertainty along the corresponding axes. Intuitively, a large σ along a given axis indicates a high uncertainty in that direction, and thus, a need to take additional measurements in new positions along that axis. We formally specify the DOP calculation in Eqs. 3-5 where (x_i, y_i, z_i) correspond to the location of the antenna at vantage point i; (x_u, y_u, z_u) correspond to the coordinates of the geometric center of the candidates; N is the number of vantage points that RFusion has taken measurements from; and R_i is half of the round trip distance estimate of i^{th} measurement.

$$A = \begin{bmatrix} \frac{x_1-x_u}{R_1} & \frac{y_1-y_u}{R_1} & \frac{z_1-z_u}{R_1} \\ \frac{x_2-x_u}{R_2} & \frac{y_2-y_u}{R_2} & \frac{z_2-z_u}{R_2} \\ \vdots & \vdots & \vdots \\ \frac{x_N-x_u}{R_N} & \frac{y_N-y_u}{R_N} & \frac{z_N-z_u}{R_N} \end{bmatrix} \tag{3}$$

$$Q = \left(A^T A\right)^{-1} \tag{4}$$

[3]This generalizes to scenarios where there are one or more disjoint candidate regions.

$$DOP_{RF} = \sqrt{tr(Q)} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{5}$$

We summarize the RF Features as RF-DOP using $(\sigma_x, \sigma_y, \sigma_z)$.[4] Notice that if RFusion relies entirely on RF features, it would require at least 3 vantage points before it starts optimizing its trajectory.

Visual Features: By leveraging vision, RFusion can bootstrap its RL network using a single vantage point. We formulate the uncertainty of the geometrically fused region (from §3) using an intuition similar to that of RF-DOP. Fig. 5 shows an example based on the scenario described earlier. The red dots indicate the region of point cloud candidates based on the results from Dense Geometric Fusion. The green box is a bounding box of these candidates, with the box axes aligned with the X,Y,Z coordinate axes. Similar to RF-DOP, a large candidate region along a certain axis suggests a greater uncertainty along that axis. Thus, the length of the bounding box represents our visual uncertainty encoding: $(\Delta X, \Delta Y, \Delta Z)$.

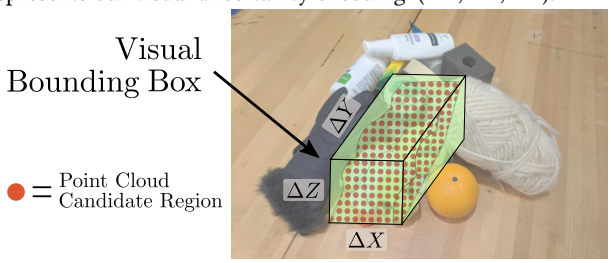


Figure 5: Visual Features. The visual features capture the uncertainty of the point cloud candidate regions using a bounding box. A longer edge represents more uncertainty in the corresponding dimension.

Positional Features: The position of the antenna relative to the target location impacts the optimal trajectory. Intuitively, if the antenna is near the item of interest, a small movement significantly changes the angle of the measurement and sharply decreases DOP. In contrast, if the antenna is far from the item, the same amount of movement results in a smaller change in the angle of the measurement, and the DOP will not improve as much. Thus, to reduce the Visual and RF uncertainty while also encouraging movement toward the final grasping location, we encode the candidate locations with respect to the robot's gripper. Specifically, we calculate a displacement vector as follows:

$$\text{Displacement} = \frac{1}{|C|} \sum_{p_i \in C} p_{\text{gripper}} - p_i \tag{6}$$

where p_{gripper} is the robot's gripper location, and p_i is the center of cluster i in the set of cluster centers C.

4.2 RFusion Network

With a feature representation that encodes position uncertainty from the observations, we now show how RFusion can leverage reinforcement learning to optimize for efficient item retrieval.

In reinforcement learning, an agent is rewarded based on its performance on a task. During training, the agent learns how to maximize its reward, resulting in solving the task or optimization problem. In the case of RFusion, we reward the agent based on the traveled distance to the grasping point and the number of vantage points until it successfully localizes the target item. We can formulate the reward as:

$$R = \zeta - \beta d_{\text{total distance}} - \eta N_{actions}$$

[4]Note that $Q = \left[\sigma_{ij}^2\right]_{ij}$ where i and j correspond to the x, y, and z dimensions.

where ζ, β, and η are scalar weights.[5] $N_{actions}$ is the number of vantage points and $d_{\text{total distance}}$ is the overall traveled distance from the initial pose until the grasping point.

Next, we formalize the above RL network. At any given time t, the RL agent is in state s_t and uses a policy π to decide on a robot action a_t based on the input observations. In RFusion, the state s_t consists of the robot joint values ($\mathbf{x}_t^R \in \mathbb{R}^6$), the round-trip distance from the robot wrist to the RFID location (denoted as d), and the structure of the pile(s) in the workspace. The action space consists of movements on a straight line from the current robot wrist location to a new point on a 3D grid of $40cm \times 60cm \times 5cm$ centered at the current robot wrist location.[6] For training, we simulated scene objects, depth camera measurements, RF measurements, and robot movements, such that the simulated environment's state transitions are based on the agent's action.[7] After each transition, RFusion's observation of the environment is updated, and the new state information s_{t+1} is passed to the RL agent via the RF-Visual feature encoding. Once the item is successfully localized, the agent is rewarded with r_t,[8] and the simulation trial ends. This RL problem can be solved using a standard Q-Network, where $Q(s_t, a_t)$ estimates the expected reward at the state s_t when taking the action a_t according to the policy $\pi(s_t)$. Q-learning is the process used to update the Q values, and it can be defined as:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

where r_t is the reward the agent receives after a_t, α is learning rate, γ is discount factor. The agent is trained across episodes until the target item is localized. Thus, the RL network enables RFusion to learn a policy that optimizes the robot trajectory for successful grasping.

4.3 Robust Localization

Next, we describe how RFusion localizes the target.

Robust Trilateration: Recall that each distance measurement from a vantage point constrains the tag's location to a sphere. With only three distance measurements, we can localize a tagged item by intersecting the three spheres (but the localization accuracy may be poor). Once additional measurements are made, the trilateration problem becomes over-constrained, and can be solved using robust least-squares with outlier rejection [26].

Handling Tag Orientation: Up until now, we have ignored the impact of the tag's orientation on RFusion's ability to sense and localize it. However, RFusion uses compact linearly polarized antennas mounted on its wrist. These antennas cannot get distance measurements when they are perpendicular to the tag direction due to polarization mismatch. To overcome this, RFusion detects and compensates for polarization mismatch.[9] Specifically, it measures the signal-to-noise ratio (SNR) of the tag's received signal. If the SNR is below a threshold (3dB), RFusion rotates the gripper until the SNR improves. This approach ensures that RFusion can accurately measure the tag's location regardless of its initial orientation.

[5] In our implementation, we set the hyper-parameters $\zeta = 5$, $\beta = 1$, and $\eta = 0.1$.
[6] The range of the grid is limited by the expected mechanical reach of the robot, and is evenly divided in each of the x, y, and z directions; hence, the 3D grid steps are 10, 15, 2.5 cm in x, y, z directions.
[7] The full simulation environment used for training is detailed in §6.2.
[8] $r_t = R$ when the object is grasped, and zero otherwise.
[9] We also experimented with circularly polarized antennas. However, these were bulkier and more difficult to mount. Also, their phase estimates were impacted by orientation, making it difficult to use them for orientation-robust localization.

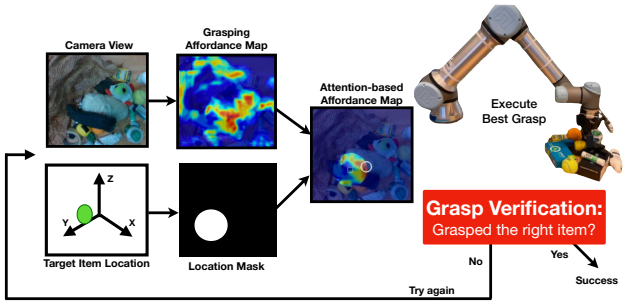


Figure 6: RF-Visual Grasping. RFusion uses the GG-CNN grasping network to calculate a grasping affordance map of the scene based on the RGB-D information. A binary mask is used to focus the attention of the system on objects near the target item. RFusion performs the best grasp from this attention-based affordance map, and declarers task completion or re-attempts grasping based on the ID of the grasped item.

5 RF-VISUAL GRASPING

Once RFusion has localized the target RFID, it proceeds to the grasping phase. Here, it faces two challenges: first, even if the target item is in LOS, the RFID's location alone is not enough for grasping. This is because grasping complex objects (e.g., a screwdriver) doesn't just require any location on the object, but also a grasping strategy. Second, the target item may be occluded under a pile, necessitating a decluttering process prior to retrieval.

To deal with these challenges, RFusion feeds the RFID's estimated location as an attention mechanism to a state-of-the-art vision-based grasping network. This approach allows RFusion to inherit the benefits of that network while extending it to retrieve occluded items. Fig. 6 illustrates this idea through an example scenario. After RFusion localizes the RFID, the robot moves its gripper above the object in the x-y plane. The wrist-mounted camera captures an RGB-D image and feeds it to a deep convolutional neural network called GG-CNN [32]. The network outputs predicted grasping qualities, grasping angle, and gripper width for each pixel in the image. Since RFusion knows the target's location, it knows that it should either directly grasp the target item or remove what is on top of it. To do so, we apply a circular binary mask centered at the item's location to the quality map, and select the highest quality grasp within this mask.[10]

Grasping Verification: RFusion can verify whether it has picked up the target item. To do so, after grasping, the robot can measure the target item's RFID response. Since a grasped item is expected to be in the gripper (and close to the antenna), the measurement can be used to determine if it was successful. If RFusion determines that it has grasped a non-target item (e.g., clutter), the robot can discard the item. The robot can repeat this process until it succeeds.

Recovering from Errors: An unsuccessful grasping attempt (e.g., due to grasping a wrong item or clutter) might inadvertently move the target item. To detect this, RFusion can obtain a new distance measurement before re-attempting to grasp. If the measurement indicates that the target item has moved from its previous location (but still not in the robot's gripper), RFusion can perform another localization round to discover the item's new location.

6 IMPLEMENTATION

6.1 Real-World Setup

Physical Setup: We implemented RFusion on a Universal Robots UR5e robot [44] with a Robotiq 2F-85 gripper [41], an Intel Realsense D415 depth camera [18], and Nuand BladeRF 2.0 Micro

[10] RFusion can also adapt the radius of the mask to the target object's dimensions.

(A) LOS, Scattered (B) LOS, 1 Pile (C) NLOS, 1 Pile (D) NLOS, 2 Piles (E) NLOS, Big + Small Pile

Figure 7: Evaluation Scenarios. The target item is unoccluded in (A) and B, and under a pile in (C)-(E).

software radios [38]. We mounted the camera and two WA5VJB Log Periodic PCB antennas (850-6500 MHz) [20] around the robot's wrist, using a 3D printed fixture as shown in Fig. 1. The system is tested with standard off-the-shelf UHF RFID tags (the Smartrac DogBone RFID tags with Monza R6-P Chips [17]) attached to standard everyday objects, office supplies, and kitchen items including gloves, plastic bottles, toys, tape rolls, stress balls, chalk boxes, and thread skeins. Each RFID tag costs around 3-5 cents.

RFID Localization: To obtain 1D distance estimates from the wrist-mounted antennas to the RFIDs, we reproduced a state-of-the-art RFID localization system on BladeRF software radios [30]. In contrast to the prior system which requires installing multiple antennas to obtain a tag's 3D location, our system is limited to a single transmit-receive antenna pair mounted on the wrist. Hence, we can only estimate the round-trip distance to an RFID (i.e., 1D localization).[11] The 1D localization implementation requires two BladeRF software radios: the first implements the standard EPC Gen2 protocol and transmits a signal within the ISM band (at 910 MHz) to power up the RFIDs; the second is clock-synchronized to the first and transmits a low-power signal, hopping its frequency from 950MHz-1200MHz in 27MHz steps to produce a wide-band channel estimate of the tag's channel. Each hopping sequence requires 150 ms and is used to estimate the round-trip distance to the tag. A Mini-Circuits ZAPD-21-S+ splitter (0.5-2.0 GHz) splits the received signal between the two BladeRFs. Our implementation performs two additional optimizations to improve localization accuracy over previous work. First, we discard channel estimates whose SNR is below 3 dB so that the bad measurements do not negatively impact localization. Second, if more than 10 frequencies are below the SNR threshold, we repeat the wideband estimation up to 4 times before moving the gripper to another vantage point.

Control Software: The system was developed and tested on Ubuntu 16.04 and ROS Kinetic. We used a ROS-native inverse kinematic solver, MoveIt [13], to control the UR Robot Driver package [43]. The solver is invoked whenever the robot needs to move to a new pose, as provided by the RL-network §4 or the grasping network §5. We used ROS for all inter process communications between the controller, camera, and BladeRFs.

6.2 Simulation

Recall from §4 that RFusion can be trained entirely in simulation. We built a simulation environment to train the RL-network to optimize the robot trajectory for localization and grasping. We used CoppeliaSim [3] for the simulation environment and Bullet Physics 2.83 for the physics engine. We simulated the tool of the UR5 robot, attached to a simulated camera and antenna.

We created 200 different scenes with 0-3 piles of 35 objects in the simulation. For example, in the case of 3 piles, we randomly specify 3 pile locations in the workspace. Each object is randomly allocated to a pile and dropped from a random pose centered around the pile center. For each training episode, we randomly choose one of the 200 scenes and then a random object in the scene to be the item of interest, which gives us 7000 possible different training episodes (200 scenes $\times$ 35 objects). In each iteration of an episode, a round trip distance is calculated based on the simulated antenna and tag's positions. Gaussian noise is added to this distance to simulate real RF measurement errors. The simulated camera also captures a depth image from the scene. This depth image is used to create a point cloud for RF-Visual fusion. The observation of the scene is fed to the RFusion Network, and the simulated robot's tool moves based on the result of the Q-Network. When the location estimates converge or too many measurements are made, the episode ends, the Q-Network is rewarded, and a new episode starts.

[11] Since Tx and RX are separate, the sphere described in §3 is a spheroid with 2 foci.

6.3 Learning Networks

RF-Visual RL Network: We implemented the Q-Network with PyTorch [40]. Adam optimizer was used for training. The reward discount factor was set to 0.9. The buffer size and minimum buffer size for training were set to 1500. The batch size of 128 was used for training. The network was trained for 11000 iterations. The training was performed on a machine running Ubuntu 16.04 with graphics card RTX 2080Ti, Intel Core i9, and 64 GB DDR4 RAM.

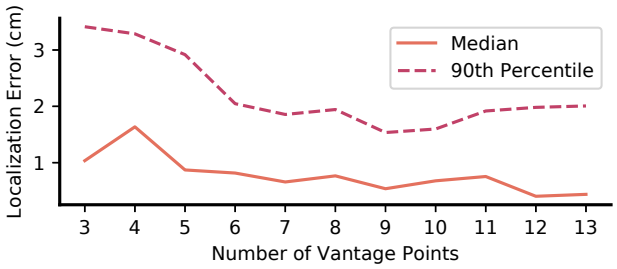
RF-Visual Grasping Network: For grasping, we implemented GG-CNN network, whose code and pre-trained weights were adopted from [35]. The RF-based attention described in §5 is implemented as a 5 cm circular binary mask centered at the RFID estimated location.

7 EVALUATION

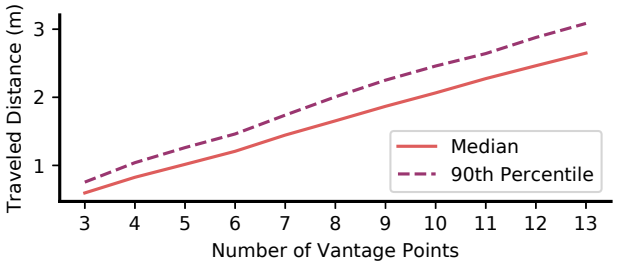
We evaluated RFusion in a multi-path-rich indoor environment, placing our target items on top of a wooden table under piles of objects. The environment is a standard office building, fully furnished with tables, chairs, and computers. Similar to prior localization systems for piece-picking robots [2, 4, 5], the robot's workspace measures roughly 0.7m $\times$ 1m $\times$ 0.3m atop a wooden table.

Evaluation Scenarios: We evaluated RFusion in five different categories of scenarios with varying levels of complexity as shown in Fig. 7. Each scenario had one or more target items (each tagged with an RFID) in addition to 5-15 non-target items to distract from or hide the targets. The scenarios included both line-of-sight (LOS) and non-line-of-sight (NLOS) settings where there were 0-2 distinct piles of objects on the table. In NLOS settings, the item of interest was fully occluded by at least another larger item in the pile.

Ground truth: We used the OptiTrack [39] system to obtain ground truth tag location. In NLOS scenarios, we recorded ground truth location before covering the item of interest with other objects. Because the tag is rectangular, we calculated localization accuracy by measuring the distance to the closest point on the tag.



(a) Localization Error vs Number of Vantage Points



(b) Traveled Distance vs Number of Vantage Points

Figure 8: Impact of the Number of Vantage Points. The figure plots (a) the localization error and (b) the traveled distance vs the number of vantage points used.

8 MICRO-BENCHMARKS

We performed micro-benchmark experiments to understand the impact of different parameters on RFusion's overall performance.

8.1 Number of Vantage Points

Recall from §4.3 that RFusion needs at least three vantage points to compute an RFID's 3D location. In our first micro-benchmark experiment, we varied the number of vantage points in order to understand the impact of increased vantage points on localization accuracy and traveled distance.

We performed 16 experimental trials in 10 different scenarios, spanning all five categories described in §7. In each of these trials, RFusion was allowed to obtain RFID channel measurements from up to 13 vantage points. For a given number of vantage points, we used all the available measurements to estimate the RFID's location as discussed in §4.3. We computed the localization error as the Euclidean distance between the RFusion-estimated location and the ground-truth RFID location. We also computed the traveled distance as the total distance traversed by the gripper from its initial vantage point all the way to the grasping point.

Fig. 8 plots the localization error and traveled distance as a function of number of vantage points. The solid red line shows the median and the purple dotted line shows the 90^{th} percentile. We make the following remarks:

- Fig. 8(a) shows that as the number of vantage points increases, the median and 90^{th} percentile error both decrease until a certain level, then plateau. In particular, the 90^{th} percentile drops from 3.5 cm to around 2 cm, and the median drops from 1 cm to 0.43 cm with 13 vantage points. This is expected since more measurements enable higher localization accuracy but with diminishing returns. More importantly, the plot shows that the RFusion's median localization accuracy is around 1 cm even with only three vantage points, yet the 90^{th} error requires 6 measurements before it reaches 2 cm and plateaus.
- Fig. 8(b) shows that as the number of vantage points increases, the median and 90^{th} percentile traveled distance consistently increase. This is expected because collecting vantage points requires moving the gripper to additional locations, thus increasing the overall traveled distance.

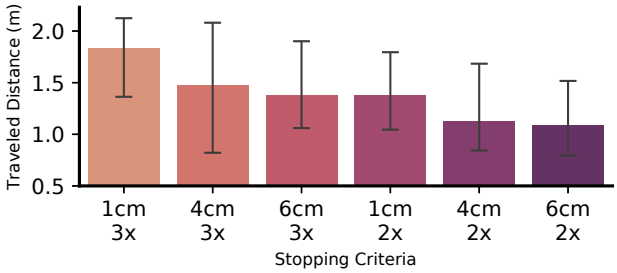


Figure 9: Stopping Criteria. The figure plots the median traveled distance for different stopping criteria. Error bars denote the 10^{th} and 90^{th} percentile.

The above findings show that one could always collect measurements from 6-7 vantage points to get higher localization accuracy. However, doing so would incur additional overhead on the traveled distance, reducing the overall efficiency of RFusion's grasping and retrieval process. This motivates the need for a mechanism that enables RFusion to determine when the target item has been sufficiently localized for grasping while avoiding unnecessary overhead.

8.2 Selecting a Stopping Criterion

The goal of our second micro-benchmark experiment was to identify the stopping criterion for RFusion's RL network. The stopping criterion defines when the system accepts its RFID-estimated location as correct, prompting it to stop collecting measurements from additional vantage points and proceed to the grasping phase. One option is to always collect measurements from a fixed number of vantage points, e.g., six vantage points to meet the 90^{th} percentile error of 2-cm as per the above micro-benchmark. However, fixing the number of vantage points to six would not be ideal since it incurs unnecessary overhead (traveled distance). As shown in §8.1, more than half the trials require no more than 3 vantage points to achieve 1 cm localization accuracy. Ideally, we would like to identify a stopping criterion that achieves accurate localization (1-2 cm) using the minimum number of vantage points.

We investigated six potential stopping criteria using the same 16 trials from the above experiment. We defined each stopping criterion by the amount of change in the tag's estimate location across 2-3 consecutive vantage points. For example, a stopping criterion of (1 cm, 3×) corresponds to a change in the estimate location of less than 1 cm across the past three consecutive vantage points. We tested the following stopping criteria: (1 cm, 3×), (4 cm, 3×), (6 cm, 3×), (1 cm, 2×), (4 cm, 2×), (6 cm, 2×). We computed both the traveled distance and the localization error across the trials whenever each stopping criterion is met.

Fig. 9 shows the median traveled distance of the system for the different stopping criteria. The error bars indicate the 10^{th} and 90^{th} percentile. We make the following remarks:

- Stopping criteria that require three consecutive vantage points, rather than two, incurs an additional distance overhead of 20-75%. This is expected since more vantage points requires the gripper to move to a new pose, increasing the overall traveled distance.
- When the number of consecutive vantage points is fixed, stopping criteria that require a smaller change in location estimate also

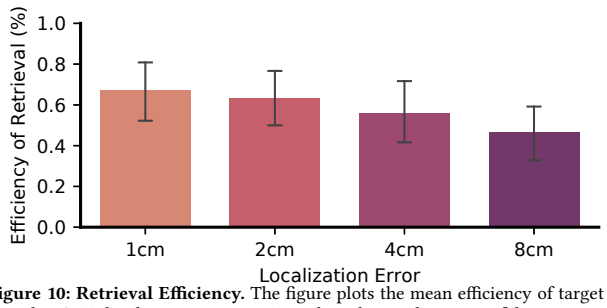


Figure 10: Retrieval Efficiency. The figure plots the mean efficiency of target retrieval vs RFID localization accuracy. Error bars denote the 95% confidence interval.

require a longer traveled distance. For example, a 1 cm threshold requires 20% more traveled distance over a 4 cm threshold. This is expected since a smaller change is a more stringent requirement.

- Interestingly, we noticed that across all six stopping criteria, RFusion achieved a localization accuracy of around 1 cm, making them all equally desirable from a localization perspective.

Since the median localization accuracy across the different stopping criteria was similar, and the traveled distance was lower (i.e., more efficient) for more relaxed criteria, we selected (4 cm, 2×) as our stopping criterion for RFusion's final implementation[12] and used it for all the subsequent performance evaluation in §9.

8.3 Retrieval Efficiency vs Localization Error

The goal of our final micro-benchmark experiment was to assess the impact of localization error on the efficiency of item retrieval. Higher retrieval efficiency indicates that RFusion succeeds in grasping the target item using a smaller number of grasping attempts. Mathematically, we define retrieval efficiency as $\frac{1}{\#\text{ of grasping attempts}}$. Note that the denominator only includes attempts where the robot successfully picks up *any* item in the workspace (i.e., we discard attempts where it fails to grasp any item). This allows us to isolate the performance of our contribution from that of the pre-trained grasping network [32].[13]

We ran 120 trials in total. Across all trials, the environment was cluttered with many distractor objects, but the target item was unoccluded from the camera (i.e., in its line-of-sight). This allowed us to assess the impact of localization error rather than pile complexity on the retrieval efficiency. Moreover, to avoid randomness in localization arising from RL-network and to focus on retrieval efficiency, we did not use the network to perform RFID localization in this micro-benchmark experiment. Instead, we obtained the tag's ground truth location from the OptiTrack system, then artificially added different amounts of errors (1, 2, 4, or 8 cm) to the RFID's location before feeding the erroneous location into RFusion's grasping network. This enabled us to control the errors and understand their impact on retrieval efficiency. In each trial, RFusion's grasping network performed grasping given the erroneous location, and measured the retrieval efficiency as described above.

Figure 10 plots the mean retrieval efficiency with different levels of localization error, as well as the 95% confidence intervals. As expected, a smaller localization error results in the highest efficiency of retrieval, and the grasping efficiency drops off as the accuracy of the estimated location decreases. The 1 cm error has the highest

[12] We chose (4 cm,2×) over (6 cm,2×) because its 90th percentile accuracy is better.
[13] An efficiency metric that accounts for all grasping attempts would be the product of the reported retrieval efficiency and the grasping efficiency of the grasping network.

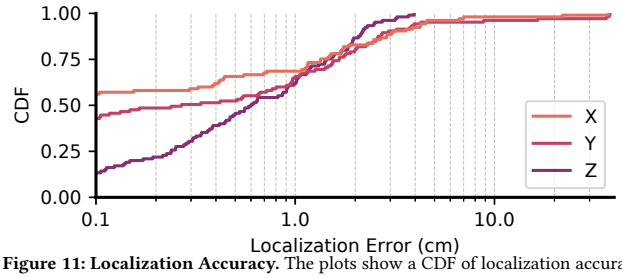


Figure 11: Localization Accuracy. The plots show a CDF of localization accuracy along the X, Y, and Z dimensions.

mean efficiency at 67%. The mean efficiencies for 2, 4, and 8 cm are 63%, 56%, and 46% respectively. Interestingly, this result shows that even with 8 cm localization error, RFusion would only need two attempts (on average) before it successfully picks up the target item. At the same time, a higher localization accuracy of 1-2 cm would further boost its efficiency.

9 PERFORMANCE RESULTS

To evaluate RFusion's overall performance, we ran experiments across all five scenarios described in §7.

9.1 Localization Accuracy

We first evaluated RFusion's accuracy in localizing RFID-tagged items. We conducted 105 trials. In each trial, the RFID-tagged target item was placed in a randomly chosen position in the workspace. The robot started from a randomly chosen initial pose, and performed geometric fusion, RF-visual control, and grasping as described in §3- §5. We computed the localization error as the euclidean distance between the RFusion-estimated location and the ground-truth location obtained from the OptiTrack system.

Figure 11 shows the CDF of the localization error along the x, y, and z dimensions across all the scenarios and trials. The figure shows that RFusion's median errors along x, y, and z dimensions are 0.1 cm, 0.26 cm, and 0.6 cm respectively. Note that the error along z axis is slightly larger because the robot wrist could only move and take measurements within a 20cm range in z direction, while it was able to take measurements within a 75cm and 1m range in x and y direction. The figure also shows that RFusion achieves 90th percentile errors of 3.07 cm, 2.85 cm, and 2.16 cm along the x, y, and z dimensions respectively. These results demonstrate that RFusion can achieve centimeter level positioning accuracy, matching or exceeding that of state-of-the-art systems [29, 30]. However, unlike these prior systems that require an infrastructure to localize, RFusion can perform accurate localization in an infrastructure-less manner, by relying entirely on the wrist-mounted antenna.

9.2 Traveled Distance

Recall from §4.2 that RFusion's controller optimizes the robot's traveled distance for localization and grasping. Our second experiment evaluated the distance traveled by RFusion's gripper and compared it to a baseline and to partial implementations of RFusion.

We conducted 140 trials. In each trial, we placed an RFID-tagged target item in a randomly chosen location within the robot's workspace. We computed the distance traveled by the gripper from its initial position until the first grasping point.

(a) Baseline Comparison: Since antenna arrays are standard in RFID localization, we implemented a baseline that uses RFusion's

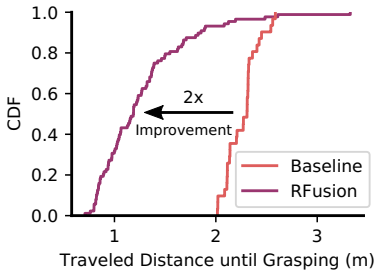
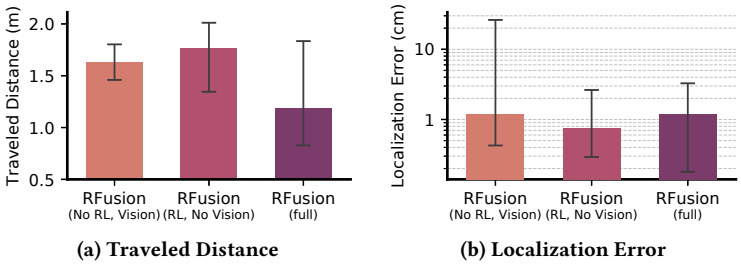


Figure 12: Traveled Distance vs Baseline. The figure plots the CDF of traveled distance for RFusion and the baseline.



(a) Traveled Distance **(b) Localization Error**

Figure 13: Decomposing RFusion's Gains. The figure plots the median (a) distance and (b) localization error for RFusion and its partial implementations. Error bars denote the 10^{th} and 90^{th} percentile.

wrist-mounted antenna to obtain measurements from a 2D array of vantage points. The baseline moves the gripper to 9 predefined vantage points spanning the maximum aperture reachable given the robot's control constraints (the aperture is 40 cm in the y dimension and 55 cm in the x-dimension).[14] Once the system has obtained measurements from all 9 vantage points, it uses them to localize the RFID using the same method described in §4.3, and then moves its gripper to the estimated grasping point. We evaluated the baseline in the same scenarios as RFusion, with each experimental trial mirroring the same initial conditions as an RFusion trial. We computed the localization error and the traveled distance for both RFusion and the baseline as described earlier.

Fig. 12 shows the CDF of the traveled distance of RFusion (purple) and the baseline (in orange). We make the following remarks:

- The median traveled distance for RFusion and the baseline are 1.18 m and 2.31 m respectively. This shows that RFusion improves the traveled distance by a median of 2× over the baseline.
- The 90^{th} percentile traveled distance of RFusion and the baseline is 1.8 m and 2.45 m respectively. This shows that RFusion not only improves the median but also the 90^{th} percentile by 36%.
- We also computed the localization error for both RFusion and the baseline. Interestingly, the baseline achieves slightly better localization error (1.02 cm) than RFusion (1.18 cm), likely due to the baseline's larger aperture. This is expected according to the micro-benchmark experiment in §8.1, which demonstrated that more vantage points slightly improve localization performance.

These results show that RFusion can optimize its trajectory to localize and grasp the item of interest and that this improvement in efficiency does not cost a noticeable loss in localization accuracy.

(b) Decomposing RFusion's Gains: Next, we wanted to quantify the gains from each of RFusion's key primitives (geometric fusion and RL-based control). To do this, we built and evaluated two partial implementations of RFusion:

- *RFusion (RL, No Vision):* In this partial implementation, the robot does not use any visual information for localization. Since RF-DOP (from §4.1) requires at least three measurements, this implementation randomly chooses three initial vantage points for each experimental trial. After the third vantage point, it uses the same RL-based controller as RFusion to determine its next vantage points (but relying only on RF features). The stopping criteria in this implementation is the same as RFusion.
- *RFusion (No RL, Vision):* In this partial implementation, the robot starts from a random initial pose and performs geometric fusion similar to RFusion's full implementation. Then, instead of performing RL to identify its vantage points, it intelligently selects four vantage points based on the candidate regions identified in geometric fusion. Specifically, it computes the mean of the candidate regions and identifies four vantage points at the corners of a 0.6 m square centered on the mean. This implementation stops after the fifth measurement (to allow for outlier rejection described in §4.3). In this implementation, localization is constrained using RF-Visual Geometric Fusion similar to RFusion.

Figure 13a plots the median traveled distances for RFusion and the partial implementations. Error bars indicate 10^{th} and 90^{th} percentiles. We make the following remarks:

- The partial implementations RFusion (No RL, Vision) and RFusion (RL, No Vision) have a median travel distance of 1.63 m and 1.76 m respectively. Interestingly, both partial implementations outperform the baseline of Fig. 12, which has a median of 2.31 m. This demonstrates the value of each sub-component individually.
- RFusion travels a median distance of 1.18 m from initial position to the grasping point, outperforming both the partial implementations and the baseline. This demonstrates the value of combining both sub-components in RFusion's overall design.
- RFusion and both its partial implementations achieve a 90^{th} percentile travel distance of 1.8-2.01 m, again outperforming the baseline's (2.45 m).

Fig. 13 plots the median localization error for RFusion and the partial implementations. Error bars indicate 10^{th} and 90^{th} percentiles. We make the following remarks:

- RFusion and the partial implementations all achieve similar median localization accuracies of 0.75-1.1 cm.
- RFusion and its partial implementation that uses RL both achieve a 90^{th} percentile localization accuracy of 2-3 cm. In contrast, RFusion (No RL, Vision) has a much worse 90^{th} percentile localization error of 26.22 cm. This is because this partial implementation has a pre-defined stopping criterion and a fixed number of vantage points, which prevent it from adapting to scenarios where the system is not confident in its RFID location estimate.
- Interestingly, the localization error is slightly lower in RFusion's partial implementation with no vision (0.75 cm) than its full implementation (1.1 cm). This is because without a camera, the robot needs to travel a longer distance to be confident in its estimate of the object location,[15] and the longer distance results in higher localization accuracy (as per Fig. 8). A natural question

[14] To avoid errors caused by polarization, we fixed the tag's orientation to align it with the wrist-mounted antenna. We only did that for the baseline and not for RFusion, since our system has a mechanism to recover from orientation errors.

[15] This result shows that RFusion's RL network is robust to potential RF errors and uncertainties as it achieves high accuracy even with RF alone.

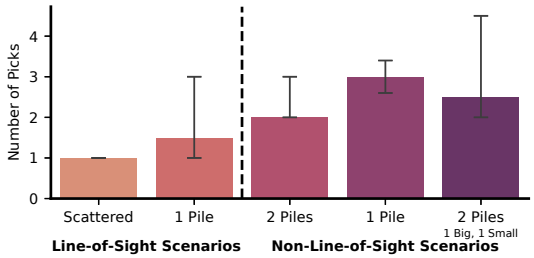


Figure 14: Retrieval Efficiency. The figure plots the median number of picks to grasp the target item in 5 different scenarios. Error bars denote 10^{th} and 90^{th} percentile.

is whether we should incentivize the full implementation to take more measurements to achieve higher accuracy. Based on §8, there is negligible benefit from marginally higher accuracy, but significant efficiency benefits in shortening the traveled distance.

Overall, these results demonstrate that RFusion benefits from both of its sub-components and can avoid large localization errors by travelling longer distances if needed. Moreover, the median traveled distance of RFusion is significantly less than the partial implementations, demonstrating the benefits of combining RL-based control and RF-Visual geometric fusion in one system. These results were demonstrated across a wide range of locations in the work space, scenario types, and tag orientations, which also demonstrates the performance in practical, unstructured environments.

9.3 Retrieval Efficiency

Next, we evaluated RFusion's efficiency in retrieving a target item from its work space. Recall from §8.3 that the grasping network uses the target item's location as an attention mechanism. Moreover, because the antenna can estimate its distance to the RFID, it can determine when it has picked up the target item. Hence, RFusion makes successive grasping attempts until it determines that the target object has been picked up. In scenarios where it picks up an item other than its target, the item is moved to the side; this allows RFusion to deal with scenarios when the object is under a pile.

To evaluate retrieval efficiency, we used the same experimental trials described in §9.1 and measured the number of picks required until RFusion grasps the object of interest. Unlike RF localization, which is minimally impacted by the complexity of the scenario, the number of picking attempts is dependent on complexity. For example, if the target object is under a pile, RFusion needs to first grasp and set aside the occluding item(s) on top of it before it can grasp the target. Hence, here we report the number of picks separately for each of the evaluated scenarios.

Fig. 14 plots the median number of picks across the five different scenarios (and the error bars represent the 10^{th} and 90^{th} percentile number of picks). We sort the scenarios in increasing order of complexity on the x-axis, and the dashed vertical line separates the LOS and NLOS scenarios.[16] We make the following remarks:

- In the simplest scenario, where the items are scattered and in line-of-sight, RFusion almost always requires only one pick. This shows that combining RFusion's high localization accuracy with attention-based grasping results in high retrieval efficiency.
- In the more cluttered LOS scenario (with 1 pile), RFusion requires a median of 1.5 picks to extract the target item. This is because even though the item is in line-of-sight, the pile is dense. Hence, in some scenarios, the robot may need to pick up a nearby item to declutter the target's vicinity prior to picking it up. Alternatively, even a small localization error might result in focusing the robot's attention to an adjacent item. Nonetheless, the grasping efficiency remains very high with a median overhead of 0.5 picks.
- Across all NLOS scenarios, the median is always at or higher than 2 picks. This is expected because here the target item is occluded by at least one other item. Thus, in order to retrieve the target item, the robot first needs to grasp and remove the occluding item. Impressively, in the scenario where there are two piles, RFusion successfully achieves the optimal picking strategy of only 2 picks.
- Finally, the scenario with one large NLOS pile has the highest median picks for retrieval (3 picks). Indeed, the median extraction process for this scenario is more difficult than that of two piles. This is because all scenarios have similar number of items (10-15), but when there is a single large pile, all items are stacked on top of each other. Thus, RFusion may need to grasp and remove multiple occluding items before it can retrieve its target item.

These results demonstrate that RFusion's efficiency of item retrieval is primarily dependent on the number of items covering the target item rather than on the complexity and number of piles in the workspace. This is because RF-Visual Geometric Fusion and RFID localization quickly eliminate distractor piles, thus allowing the system to efficiently retrieve the target item.

Here, it is also worth comparing RFusion to prior systems that rely only on vision for retrieving occluded objects. When dealing with piles of the same size and complexity as those tested here, state-of-the-art vision systems require a median of 6.5 picks [5], i.e., twice that of RFusion. This demonstrates that RFusion's ability to leverage RFID localization doubles the retrieval efficiency.

9.4 Robustness to Orientation

Recall that RFusion relies on a linearly-polarized antenna mounted on the robot's wrist and that it rotates and adjusts the gripper and antenna's orientation to achieve good SNR for estimating the RFID's channel. In this section, we report RFusion's robustness to variations in the orientation of RFID-tagged objects.

To evaluate the robustness to orientation, we ran 48 experimental trials, placing the RFID-tagged item in a different location and orientation in the workspace. We varied the tag's orientation from 0-180° at 30° intervals. (Since the tag is linearly polarized, 0° and 180° are the same; similarly 30° and 210°.) We also implemented a variant of RFusion that does not adjust its gripper orientation (i.e., the variant does not have the orientation module described in §4.3). We evaluated this variant in the same scenarios as RFusion's full implementation (i.e., with the same initial conditions for the robot pose, tag position, and tag orientation). We computed the localization accuracy of both systems across orientations.

Table 1 shows the median localization error across tag orientations for both RFusion and its variant implementation. It shows that RFusion achieves centimeter-level accuracy regardless of the tag's orientation. In contrast, the implementation without antenna rotation fails to localize any tag at 60°, 90°, or 120°. This is expected since when the tag and antenna polarization are close to orthogonal, the SNR of the channel estimate is low, resulting in poor localization accuracy. The results shows the value of RFusion's ability to

[16] Similar to §8.3, we only counted the grasps where the robot successfully picks up *any* item to factor out inefficiencies from the pre-trained grasping network [32].

detect and recover from orientation mismatches and its ability to perform robust, orientation-agnostic RFID localization.

	Tag Orientation					
	0°	30°	60°	90°	120°	150°
RFusion	1.09	1.35	1.62	1.41	2.72	2.6
w/o Rotation	1.81	0.41	**Failed**	**Failed**	**Failed**	1.1

Table 1: Localization Error (in cm) vs Tag Orientation.

9.5 End-to-End Success Rate

Finally, we evaluated RFusion's end-to-end success rate. A trial is considered successful if RFusion can successfully retrieve the target item in any given experimental trial. To penalize highly inefficient scenarios, we abort the retrieval task and declare the task unsuccessful if RFusion fails to grasp any item within three consecutive grasping attempts. We ran 50 trials across the five scenarios explained in §7. Table 2 shows RFusion's success rate across these scenarios, demonstrating that RFusion aborts grasping in only 2 out of 50 trials, thus achieving an overall success rate of 96%.

LOS:	Scattered	1 pile	**NLOS:**	1 pile	2 piles	Big+small pile
	$10/10$	$10/10$		$9/10$	$9/10$	$10/10$

Table 2: End-to-end Success Rate.

10 RELATED WORK

RFusion builds on past work in robotic grasping and RFID localization, and is the first to enable grasping occluded objects using a fully-integrated RF-visual sensing robot.

(a) Vision-based Robotic Grasping. The past decade has witnessed remarkable progress in deep-learning vision models that enable robots to grasp complex objects [21, 34, 52, 53]. The majority of grasping systems aim at improving the efficiency and speed of grasping *all* objects, and typically focus on maximizing the number of picks per hour. Recently, researchers have started tackling the problem of *mechanical search*, where the goal is to retrieve a specific target object from a pile (e.g., to retrieve a tool or prioritize fulfillment) [4, 5, 37, 51]. However, these systems often suffer from low efficiency when the object of interest is under a pile; in such scenarios, the target object is invisible to the robot's camera, necessitating a time-consuming search and decluttering process until the item becomes visible for recognition and grasping. RFusion directly builds on this line of work and extends it to operate efficiently even when the target object is fully occluded.

(b) RFIDs for Robotic Grasping. RFusion builds on prior work that explored the potential of using RFIDs to aid in robotic grasping. Pioneering research in this space focused on using the received signal strength (RSS) to locate the object of interest [8–10]. These systems performed a raster scan where the antenna is moved near the object of interest as it measures the RSS of the RFID on the object; subsequently, the object is localized by mapping it to the location where the RSS is highest. However, because RSS is significantly impacted by the orientation of the tag and the presence of multipath, these systems required the object of interest to be in a known orientation in line-of-sight and could not demonstrate grasping in dense clutter or when the object was under a pile. As techniques for RFID positioning improved in subsequent years, researchers explored the potential of using antenna arrays and time-of-flight in order to locate and grasp RFID-tagged objects, even in multipath-rich and non-line-sight environments [2, 45]. However, these prior systems could only operate in environments instrumented with an RFID localization and/or calibration infrastructure. For example, RF-Compass requires the robot and RFID localization system to be calibrated with respect to each other using a dedicated Vicon motion capture system [45]. Moreover, these systems either couldn't deal with orientation or required careful placement of multiple RFIDs on the target object (which does not represent practical RFID deployments). Similar to these prior systems, RFusion builds on state-of-the-art RFID localization systems that measure time-of-flight in multipath rich and non-line-of-sight environments [30]. In contrast to these past systems, the antenna is integrated directly onto the robot's end-effector, eliminating the need for a complex setup with a separate motion capture system, and overcoming orientation issues with a single RFID tag by using antenna motion.

(c) RFID Localization using Mobile Platforms. Prior work has investigated the problem of RFID localization by placing a reader on mobile platforms [14, 36, 42, 45, 46]. In these systems, the reader is typically mounted on a Roomba robot and moved over few meters to collect measurements (similar to a synthetic aperture radar) and perform antenna-array based localization [46]. As we showed in §9.2, such an approach would be much less efficient than RFusion.

(d) Sensor Fusion for RFID Localization. RFusion is related to prior work that performs sensor fusion to localize RFIDs [9, 15, 19, 47]. In these systems, the RFID is typically already visible to a camera and the goal is to improve its localization accuracy by matching visual information and the measured RFID channel (generally for a moving tag). As a result, these methods cannot be applied for the RFusion task when the RFID of interest is static and/or fully-occluded. Researchers have also considered other forms of sensor fusion (inertial+RF) for RFID localization [50], but these systems require specialized tags in motion and, unlike RFusion cannot work with off-the-shelf UHF RFID tags. As a result, they are not suitable for the problem of grasping (static) objects tagged with off-the-shelf RFIDs. Finally, RFusion builds on prior work on RF-optical fusion in the sensors community [1, 24, 25, 28, 48, 49] and RL in the robotics community [23, 52], and develops the first RL framework capable of fusing RF and vision for robotic grasping and localization.

11 CONCLUSION

This paper presents RFusion, the first robot that fully integrates RF and vision to grasp and retrieve items in line-of-sight, non-line-of-sight, and fully-occluded settings. RFusion's design introduces novel primitives for RF-visual sensing and learning. By applying these primitives to the robotic grasping problem, our prototype demonstrates remarkable efficiency in performing highly complex robotic tasks in scenarios where neither RF nor vision alone would be efficient. As the research evolves, it would be interesting to combine the system with other actions (e.g., allowing the robot to push occluding objects) and explore how alternative reinforcement learning networks (e.g., continuous actor-critic instead of discrete) may further boost its efficiency. More fundamentally, RFusion's primitives are general and highlight new directions for sensor fusion and multi-sensory learning, with exciting applications spanning robot manipulation, scene understanding, and autonomous navigation.

Acknowledgement - We thank the anonymous reviewers and the Signal Kinetics group for their feedback. This research is sponsored by an NSF CAREER Award (CNS-1844280), the Sloan Research Fellowship, NTT DATA, Toppan, Toppan Forms, the MIT Media Lab, and the Abdul Latif Jameel Water and Food Systems Lab (J-WAFS) at MIT.

REFERENCES

[1] Ashwin Ashok, Chenren Xu, Tam Vu, Marco Gruteser, Richard Howard, Yanyong Zhang, Narayan Mandayam, Wenjia Yuan, and Kristin Dana. 2013. Bifocus: Using radio-optical beacons for an augmented reality search application. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. 507–508.

[2] Tara Boroushaki, Junshan Leng, Ian Clester, Alberto Rodriguez, and Fadel Adib. 2021. Robotic Grasping of Fully-Occluded Objects using RF Perception. In *2021 International Conference on Robotics and Automation (ICRA)*. IEEE.

[3] Coppelia Sim. 2021. https://www.coppeliarobotics.com/coppeliaSim.

[4] Michael Danielczuk, Anelia Angelova, Vincent Vanhoucke, and Ken Goldberg. 2020. X-ray: Mechanical search for an occluded object by minimizing support of learned occupancy distributions. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 9577–9584.

[5] Michael Danielczuk, Andrey Kurenkov, Ashwin Balakrishna, Matthew Matl, David Wang, Roberto Martín-Martín, Animesh Garg, Silvio Savarese, and Ken Goldberg. 2019. Mechanical search: Multi-step retrieval of a target object occluded by clutter. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 1614–1621.

[6] Jeffrey Dastin. 2019. Amazon rolls out machines that pack orders and replace jobs. Reuters. https://www.reuters.com/article/us-amazon-com-automation-exclusive/exclusive-amazon-rolls-out-machines-that-pack-orders-and-replace-jobs-idUSKCN1SJ0X1.

[7] Travis Deyle, Cressel Anderson, Charles C Kemp, and Matthew S Reynolds. 2008. A foveated passive UHF RFID system for mobile manipulation. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3711–3716.

[8] T. Deyle, C. Anderson, C. C. Kemp, and M. S. Reynolds. 2008. A foveated passive UHF RFID system for mobile manipulation. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3711–3716. https://doi.org/10.1109/IROS.2008.4651047

[9] T. Deyle, H. Nguyen, M. Reynolds, and C. C. Kemp. 2009. RF vision: RFID receive signal strength indicator (RSSI) images for sensor fusion and mobile manipulation. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5553–5560. https://doi.org/10.1109/IROS.2009.5354047

[10] T. Deyle, C. J. Tralie, M. S. Reynolds, and C. C. Kemp. 2013. In-hand radio frequency identification (RFID) for robotic manipulation. In *2013 IEEE International Conference on Robotics and Automation*. 1234–1241. https://doi.org/10.1109/ICRA.2013.6630729

[11] Shing H. Doong. 2009. A closed-form formula for GPS GDOP computation. *GPS Solutions* 13, 3 (July 2009), 183–190. https://doi.org/10.1007/s10291-008-0111-2

[12] Guoguang Du, Kai Wang, Shiguo Lian, and Kaiyong Zhao. 2020. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review* 54, 3 (Aug 2020), 1677–1734. https://doi.org/10.1007/s10462-020-09888-5

[13] Ettus Research, CDA-2990. [n.d.]. https://moveit.ros.org/.

[14] M. Gareis, C. Carlowitz, and M. Vossiek. 2020. A MIMO UHF-RFID SAR 3D Locating System for Autonomous Inventory Robots. In *2020 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*. 1–4. https://doi.org/10.1109/ICMIM48759.2020.9298989

[15] Michael Goller, Christoph Feichtenhofer, and Axel Pinz. 2014. Fusing RFID and computer vision for probabilistic tag localization. In *2014 IEEE International Conference on RFID (IEEE RFID)*. IEEE, Orlando, FL, USA, 89–96. https://doi.org/10.1109/RFID.2014.6810717

[16] Kana Inagaki. 2019. Packing T-shirts? There's a Uniqlo robot for that. Financial Times. https://www.ft.com/content/79434838-2142-11ea-b8a1-584213ee7b2b.

[17] Smartrac Shortdipole Inlay. 2021. www.smartrac-group.com.

[18] Intel RealSense. 2019. https://www.intelrealsense.com.

[19] P. Kamol, S. Nikolaidis, R. Ueda, and T. Arai. 2007. RFID Based Object Localization System Using Ceiling Cameras with Particle Filter. In *Future Generation Communication and Networking (FGCN 2007)*, Vol. 2. 37–42. https://doi.org/10.1109/FGCN.2007.194

[20] Kent Electronics. 2021. http://www.wa5vjb.com.

[21] Sung-Kyun Kim and Maxim Likhachev. 2016. Planning for grasp selection of partially occluded objects. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 3971–3978.

[22] Ross A Knepper, Todd Layton, John Romanishin, and Daniela Rus. 2013. Ikeabot: An autonomous multi-robot coordinated furniture assembly system. In *2013 IEEE International conference on robotics and automation (ICRA)*. IEEE, 855–862.

[23] Jens Kober, J Andrew Bagnell, and Jan Peters. 2013. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research* 32, 11 (2013), 1238–1274.

[24] Hanchuan Li, Peijin Zhang, Samer Al Moubayed, Shwetak N Patel, and Alanson P Sample. 2016. Id-match: A hybrid computer vision and rfid system for recognizing individuals in groups. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 4933–4944.

[25] Tianhong Li, Lijie Fan, Mingmin Zhao, Yingcheng Liu, and Dina Katabi. 2019. Making the invisible visible: Action recognition through walls and occlusions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 872–881.

[26] Jing Liu, Pamela C Cosman, and Bhaskar D Rao. 2017. Robust Linear Regression via ℓ_0 Regularization. *IEEE Transactions on Signal Processing* 66, 3 (2017), 698–713.

[27] Market Study Report LLC. 2020. Global Material Handling Robots Market Size to surpass USD 13.8 Billion in value by 2026. Global News Wire. https://www.globenewswire.com/news-release/2020/03/30/2008157/0/en/Global-Material-Handling-Robots-Market-Size-to-surpass-USD-13-8-Billion-in-value-by-2026.html.

[28] Chris Xiaoxuan Lu, Muhamad Risqi U Saputra, Peijun Zhao, Yasin Almalioglu, Pedro PB de Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. 2020. milliEgo: single-chip mmWave radar aided egomotion estimation via deep sensor fusion. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (Sensys)*. 109–122.

[29] Zhihong Luo, Qiping Zhang, Yunfei Ma, Manish Singh, and Fadel Adib. 2019. 3D backscatter localization for fine-grained robotics. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. 765–782.

[30] Yunfei Ma, Nicholas Selby, and Fadel Adib. 2017. Minding the billions: Ultra-wideband localization for deployed rfid tags. In *Proceedings of the 23rd annual international conference on mobile computing and networking (MobiCom)*. 248–260.

[31] Christopher Mims. 2020. As E-Commerce Booms, Robots Pick Up Human Slack. Wall Street Journal. https://www.wsj.com/articles/as-e-commerce-booms-robots-pick-up-human-slack-11596859205.

[32] Douglas Morrison, Peter Corke, and Jurgen Leitner. 2018. Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. In *Robotics: Science and Systems XIV (RSS)*.

[33] Douglas Morrison, Peter Corke, and Jürgen Leitner. 2019. Multi-View Picking: Next-best-view Reaching for Improved Grasping in Clutter. In *2019 IEEE International Conference on Robotics and Automation (ICRA)*.

[34] Douglas Morrison, Peter Corke, and Jürgen Leitner. 2019. Multi-view picking: Next-best-view reaching for improved grasping in clutter. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 8762–8768.

[35] Morrison, Douglas and Corke, Peter and Leitner, Jürgen. 2018. GG-CNN Code. https://github.com/dougsm/ggcnn.

[36] A. Motroni, P. Nepa, P. Tripicchio, and M. Unetti. 2018. A Multi-Antenna SAR-based method for UHF RFID Tag Localization via UGV. In *2018 IEEE International Conference on RFID Technology & Application (RFID-TA)*. IEEE, Macau, 1–6. https://doi.org/10.1109/RFID-TA.2018.8552780

[37] Tonci Novkovic, Remi Pautrat, Fadri Furrer, Michel Breyer, Roland Siegwart, and Juan Nieto. 2020. Object finding in cluttered scenes using interactive perception. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 8338–8344.

[38] Nuand, BladeRF 2.0 Micro. 2021. https://www.nuand.com/bladerf-2-0-micro/.

[39] Optitrack. 2017. http://www.optitrack.com.

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019), 8026–8037.

[41] Robotiq. 2019. https://robotiq.com/products/2f85-140-adaptive-robot-gripper.

[42] Longfei Shangguan and Kyle Jamieson. 2016. The design and implementation of a mobile RFID tag sorting robot. In *Proceedings of the 14th annual international conference on mobile systems, applications, and services (MobiSys)*. 31–42.

[43] Universal Robots ROS Driver. 2020. https://github.com/UniversalRobots/Universal_Robots_ROS_Driver.

[44] Universal Robots, UR5e. 2021. https://www.universal-robots.com/products/ur5-robot/.

[45] Jue Wang, Fadel Adib, Ross Knepper, Dina Katabi, and Daniela Rus. 2013. RF-compass: Robot object manipulation using RFIDs. In *Proceedings of the 19th annual international conference on Mobile computing & networking (MobiCom)*. 3–14.

[46] Jue Wang and Dina Katabi. 2013. Dude, where's my card? RFID positioning that works with multipath and non-line of sight. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 51–62.

[47] Zhongqin Wang, Min Xu, Ning Ye, Ruchuan Wang, and Haiping Huang. 2019. RF-Focus: Computer vision-assisted region-of-interest RFID tag recognition and localization in multipath-prevalent environments. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 3. ACM New York, NY, USA, 1–30.

[48] Lei Xie, Jianqiang Sun, Qingliang Cai, Chuyu Wang, Jie Wu, and Sanglu Lu. 2016. Tell me what I see: Recognize RFID tagged objects in augmented reality systems. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 916–927.

[49] Jingao Xu, Hengjie Chen, Kun Qian, Erqun Dong, Min Sun, Chenshu Wu, Li Zhang, and Zheng Yang. 2019. ivr: Integrated vision and radio localization with zero human effort. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 3. ACM New York, NY, USA, 1–22.

[50] Qian Yang, David G. Taylor, Muhammad B. Akbar, and Gregory D. Durgin. 2019.

Analysis of Kalman Filter-Based Localization for HIMR RFID Systems. *IEEE Journal of Radio Frequency Identification* 3, 3 (Sept. 2019), 164–172. https://doi.org/10.1109/JRFID.2019.2921001

[51] Yang Yang, Hengyue Liang, and Changhyun Choi. 2020. A Deep Learning Approach to Grasping the Invisible. *IEEE Robotics and Automation Letters* 5, 2 (2020), 2232–2239.

[52] Andy Zeng, Shuran Song, Stefan Welker, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. 2018. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4238–4245.

[53] Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, Francois R Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, et al. 2018. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 1–8.

[54] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker Jr. au2, Alberto Rodriguez, and Jianxiong Xiao. 2017. Multi-view Self-supervised Deep Learning for 6D Pose Estimation in the Amazon Picking Challenge. arXiv:1609.09475 [cs.CV]