

AIDE: Augmented Onboarding of IoT Devices at Ease

Huanle Zhang[‡], Mostafa Uddin[§], Fang Hao[§], Sarit Mukherjee[§], Prasant Mohapatra[‡]

[‡]University of California, Davis [§]Nokia Bell Labs

{dctzhang, pmohapatra}@ucdavis.edu {mostafa.uddin, fang.hao, sarit.mukherjee}@nokia-bell-labs.com

ABSTRACT

In order to use and manage IoT devices, a prerequisite is to onboard them so that they can be initialized and connected to the infrastructure. This requires mapping each physical device with its digital identity. Doing so manually is tedious, error-prone and not scalable. In this paper, we propose AIDE, a mechanism that provides Augmented onboarding of IoT Device at Ease. AIDE offers a streamlined on-boarding process by automatically associating devices at different locations with their corresponding Received Signal Strength (RSS) profiles, which can be applied to a wide range of wireless technologies such as WiFi, BLE and Zigbee. AIDE does not require additional infrastructure or hardware support, and can work by simply using a COTS smartphone as receiver. The mechanism employs a carefully designed measurement approach and a post-processing algorithm to mitigate multi-path effect and improve measurement accuracy. Preliminary experiments in different indoor environments show that AIDE achieves about 90% on-boarding accuracy when devices are 6 feet away from the measurement point, and 100% accuracy when devices are directly approachable.

ACM Reference Format:

Huanle Zhang, Mostafa Uddin, Fang Hao, Sarit Mukherjee, Prasant Mohapatra. 2019. AIDE: Augmented Onboarding of IoT Devices at Ease. In *The 20th International Workshop on Mobile Computing Systems and Applications (HotMobile '19), February 27–28, 2019, Santa Cruz, CA, USA*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3301293.3302354>

1 INTRODUCTION

The Internet of Things (IoT) continue to expand its reach into homes, industry, hospitals, and other environments, as more and more devices are connected with the purpose of gathering and sharing data. Apart from the convenience aspect, there are several potential benefits of IoT that can lead to increased energy efficiency, improved safety and security, and higher product quality. However, to achieve the benefits of IoT devices, it is critical to have an efficient *on-boarding* process that can initialize and provision the devices for accessing the network infrastructure. Unfortunately, often the process to on-board IoT devices is time consuming and labor intensive, which becomes the barrier to streamlined IoT adoption and deployment [1]. Furthermore, the complexity of deploying large

number of devices may also increase the vulnerability and security risk of the infrastructure.

To better understand the limitation of the current manual on-boarding process, consider a scenario where an enterprise has acquired many smart light bulbs and installed them on ceiling, wall or floor. These devices can be controlled through wireless communication such as BLE, WiFi and Zigbee. But before the system administrator or operator can operate these light bulbs, s(he) needs to know the device ID (MAC address or physical address) of each light bulb. Note that although the human-readable manufacturer names may be contained in the beacon packet, these names can only help to separate different types of devices (e.g. light bulbs vs. thermostats), or devices from different manufacturers. It is difficult to know (physically) which light bulb has which device ID just based on beacon packets in the case where all light bulbs are from the same manufacturer. To on-board these light bulbs manually, the operator may either try to find the MAC address on the original package of each device and enter them into the system one by one, or s(he) can try to onboard each light bulb one at a time, and turn it on/off and try to verify which device is under control. We can see that such manual on-boarding process is very tedious and error-prone, and can be very inefficient when the number of devices is large. In addition, for devices that do not give visual feedback about its operational status, e.g., sensors that do not show on/off status, it can be difficult to verify their device IDs without testing each of them in isolation.

In order to on-board IoT devices at large numbers, we need a streamlined mechanism to register each device to the infrastructure based on its unique digital identity (i.e., MAC or physical address). In addition to seamless registration, it is also essential to know, which digital identity corresponds to which physical device. Knowing this information enhances usability [2] and safety [1] in interacting with the surrounding IoT devices. In this paper, we refer to such methodology as *augmented on-boarding*.

Our basic idea is to differentiate the seemingly identical devices based on their Received Signal Strength (RSS) values. In a deployed environment, devices are typically separated from each other by a certain distance. For example, light bulbs may be installed on ceiling with several feet in between. Similarly, hand held devices can be separated from each other by moving them apart. Hence when we measure the RSS values of different devices, generally we should be able to find some difference in their signal strength due to their location differences. Note that RSS is available in almost all COTS receivers regardless of what wireless communication technology is used, e.g., WiFi, BLE and Zigbee, which makes RSS-based solution IoT-protocol independent.

One naive solution to identify a target device is to measure the RSS values by holding a smart phone closest to this device and then

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

HotMobile '19, February 27–28, 2019, Santa Cruz, CA, USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6273-3/19/02...\$15.00

<https://doi.org/10.1145/3301293.3302354>

identify this device as the one with the highest RSS value. However, there are a number of challenges that make such naive solution not working well. First, RSS value drops exponentially with the increase in distance, which makes it difficult to reliably compare two signals beyond a certain distance range. Therefore, in order to create reliable RSS contrast, we need to conduct measurement at the close proximity of the target device. However, in many cases, due to physical constraint (e.g., devices on ceiling) or obstruction (e.g., furniture on the way), target devices may not be approachable. Furthermore, RSS measurements are affected significantly by the multi-path effect. A slight change in location or direction may cause significant changes in measurement results. To further complicate the matter, RSS values vary significantly across devices. Even for the same type of devices, their RSS values vary due to other factors such as battery levels or age.

Due to above signal and physical constraints, the naive approach of selecting maximum RSS measurement to identify devices shows only $\sim 65\%$ accuracy in our experiments. In this paper, we propose AIDE, a more carefully designed measurement approach that systematically samples across multiple locations, and then use a voting-based algorithm to process the RSS measurement results for different devices at different locations to infer the device identities. Through preliminary experiments in several different indoor environments, we find that our solution can significantly improve the measurement accuracy over the naive approach. In the case that the target device is directly reachable, we can achieve 100% accuracy. In the case, the target devices are installed on ceiling and not directly reachable, we can achieve about 90% accuracy. However, in order to make augmented on-boarding applicable in practical settings we need near perfect accuracy. As our first steps towards that goal, AIDE shows promising results in our evaluation.

2 USE-CASE SCENARIOS

Large-Scale Device Onboarding for Industry: IoT devices have been increasingly adopted by industries for many different applications. In the introduction we have shown one such example, where an enterprise that deploys smart light bulbs can use our solution to streamline on-boarding process. In addition, consider a retail store that uses IoT to improve the shopping experience, e.g., sending beacon alerts to customers or using smart shelves to show product information [4, 5]. This requires a large number of IoT devices deployed at various locations of the store. When such devices are initially deployed, they need to be registered in the system, so that correct device ID to location mapping can be established. In order to do so, the current de-facto process is to either enter each device ID into the system manually, if this can be found from device's original package; or through a trial-and-error process where the operator can try to connect to each device one-by-one, change its status (e.g. turn them off or change light color), and then observe which device is changed and hence make the association. However, such manual process may be error-prone and inefficient. Instead, if we use the proposed AIDE mechanism, the store operator can simply use a phone to do device's beacon measurement close to the shelf where each device is installed. Then after all the measurement is done, the AIDE app that runs on the smart phone will automatically associate all device IDs with their corresponding shelf locations.

Inventory Management in Hospitals: Our on-boarding solution can be used in managing day-to-day inventory in the medical sector. Consider a scenario, where a patient is admitted to an emergency care. In this environment, for efficient utilization of space and easy movement of the physicians, often patients are assigned to hospital beds that are close to each other, separated only by curtains (i.e., vertical treatment room [3]). Once a patient is admitted, (s)he wears a wrist band with bar-code that represents the identity of that patient. This identity is used by the hospital to maintain the record about the patient. Now assume that the hospital has an inventory of BLE heart-rate monitoring devices. Since these devices are typically acquired in batches, many of them are from the same manufacturing companies and have the same model numbers. One of the heart-rate monitors will be attached to the patient after (s)he is admitted. The de facto process requires to first register all devices in the inventory manually by entering their MAC addresses and serial numbers etc. into the database, and also attach a printed label with its unique ID to this device. Then when the device is assigned to the patient, again manually associate the device label with the patient's record. In this way, the hospital can monitor the patient status and at the same time keep record of their inventory. However, such manual process may be error-prone and inefficient. Instead, if we use the proposed AIDE mechanism, the physician or nurse can simply hold a smart phone close to the heart-rate monitor, and the AIDE app that runs on the smart phone will automatically identify the device based on its beacon signal, despite having other beacon signals from similar heart-rate monitoring devices of nearby patients. Later this device identity can be associated with the patient's record. Although the hospital environment requires stringent 100% accuracy, we have seen promising results from our experiments that this may be achievable when the devices are directly approachable.

Interactive Indoor Map: In an enterprise environment such as an office building, we may be surrounded by a large number of smart devices and appliances. As the usage of these devices grows, it becomes important for the employees to be able to interact with these devices seamlessly. One way to enable such interaction is to use a smart phone app with an interactive indoor map of the building [2], where the IoT devices are marked on the map. Users can then click on the devices on the map to control them. In this scenario, it is important to have a streamlined process to on-board all such devices whenever they are installed and replaced. If this is done manually, one would have to try to connect and control each device one by one and try to assign device IDs on the map. With AIDE, one can instead use a smart phone to collect measurement data at the proximity of each device for a few seconds, and then the algorithm will automatically assign all device IDs on the map in one shot. In this usage scenario, AIDE can help to associate the physical device to its beacon and MAC address.

3 CHALLENGES

In the proposed on-boarding solution, we passively measure RSS from the wireless communication of surrounding devices. Unlike many other metrics such as CSI and AoA, RSS is considered as the most generic and easily accessible measurement metric. In that regard, any COTS mobile device that is compatible with IoT wireless

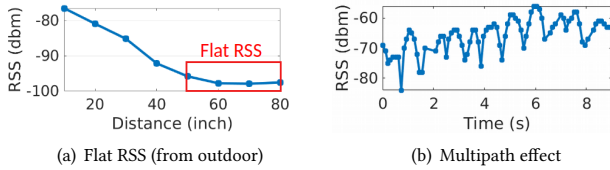


Figure 1: Signal constraint. (a) Flat RSS beyond some distance; (b) Noisy RSS due to multipath effect

communication protocol can be used as a receiver for our measurement. Thus, without any modification (software and hardware) in the already deployed IoT devices, and without any infrastructure support (e.g., access points), we can use any COTS smartphone for our on-boarding solution. Despite the practicality of RSS measurement, there are a number of challenges due to the characteristics of signals, and the physical settings at indoor environment.

RSS measurement can vary due to a number of reasons that include transmission power, distance, multi-path effect, etc. In the following list, we describe different challenges that we face for varying nature of RSS measurement and the complex layouts of indoor structure.

- (1) Different devices have different transmission powers. Assume, we have two devices of same type (device 'A' and 'B') side-by-side, and their transmission powers differ because one has (device 'A') higher battery capacity than the other (device 'B'). Note that, increase of transmission power increases the RSS value of the received signal. Given the close proximity of device 'A' and 'B', even if we measure RSS of both devices at the position of device 'B', we may see higher absolute RSS value of device 'A' compared to device 'B'. Thus we cannot rely on absolute RSS value to infer the proximity of devices.
- (2) Beyond certain distance, change in RSS is indistinguishable. Figure 1(a) shows a trace (collected outdoor at open-space on top of Crawford Hill, NJ) in which the RSS does not decrease much beyond ~ 50 inches. Therefore, measuring RSS in close proximity helps in distinguishing target devices. However, it may not always be possible to get close to the target devices or devices may not be approachable. For example, if devices are deployed on ceiling, we cannot get very close to the target devices. In these circumstances, it is challenging to use RSS to distinguish target devices from distance, especially when the target devices are close to each other. In other words, it is more difficult to create sufficient contrast in RSS values of target devices to distinguish them when the measurements are conducted farther away from the devices.
- (3) RSS data at indoor environment is noisy because of multipath effect. Figure 1(b) shows a trace of RSS when we walk with a receiver directly toward a transmitter located at 80 inches away. Although the RSS increase is the general trend, the data fluctuates significantly. Due to the multi-path effect, measuring at larger distance may show higher RSS value compared to a shorter distance from the target device. Therefore, without proper techniques to combat multipath effect, the accuracy of onboarding based on RSS may degrade.

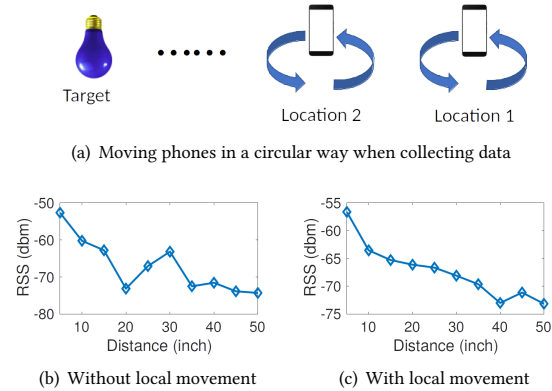


Figure 2: To mitigate multipath effect, we move our phone in a circular way as (a) shows. (b) and (c) plot one trace with and without local movement respectively

4 PROPOSED SOLUTION

Before describing the proposed solution, we first present the RSS measurement technique in mitigating multi-path effect. Second, we describe the RSS measuring procedure, and finally we describe the algorithm to identify devices. By putting them together, we propose an augmented on-boarding solution, AIDE.

4.1 Mitigating Multipath Effect

In Figure 1(b), we see how multi-path can have both constructive (multi-path components are in phase) and destructive (multi-path components are out of phase) interference effect on RSS measurement. In such phenomenon, for constructive case we see relatively higher RSS value, and relatively lower RSS value for destructive case. Therefore, instead of fixing the phone, we move our phone in a circular way (i.e., local movement) when we collect RSS data as Figure 2(a) shows. By doing this, we average RSS (spatially) within a small region, and thus we mitigate the multi-path effect in our measurement. Note that the radius of the circular movement has to be at least 2.5 inches, which is half of the wavelength (i.e., $\lambda = c/f = 3 \times 10^8 / 2.4 \times 10^9$ meters ≈ 2.5 inch). Thus, we can have measurement across full wavelength. To show the effectiveness of our local movement method, we measure RSS at different distances from a transmitter, and average RSS data at each location. Figure 2(b) and Figure 2(c) plot the results with and without local movement, respectively. They clearly show that our local movement method results in a smoother and more consistent RSS curve over distance.

4.2 Measuring Procedure

Figure 3 depicts our measuring procedure. In this example, we want to on-board device IDs of three light bulbs on the ceiling. To on-board these devices, we collect RSS from all three light bulbs at fixed-locations, called *measurement locations*. There are three constraints in selecting a *measurement location*: First, each measurement location corresponds to a target device, whose device ID we are interested in finding. Therefore, in Figure 3, we have three measurement locations for three target devices. Second, a

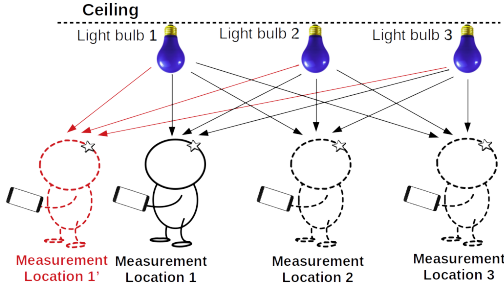


Figure 3: Measuring procedure in AIDE. We measure RSS at fixed positions closest to each target device. At each measurement location, we move our phone in a circular way when collecting RSS

measurement location of a target device is the position that is closest to that device compared to the other target devices. Third, a measurement location should be as close as possible to the target device. For example, in Figure 3, the measurement location 1 is the closest one (right below) to light bulb 1 compared to the left-most measurement location 1'. Hence location 1 should be used even though both locations satisfy the second constraint. This third constraint allows us to avoid the flat-like RSS region from Figure 1(a), and to have enough RSS contrast among multiple target devices.

For approachable case, a measurement location can be at the position of the target device, where as, for unapproachable case, a measurement location can be as close as possible to the target device. For example, as shown in Figure 3, the measurement location for light bulb 1 on ceiling (target device), which is unapproachable, is right below at *measurment location 1*. At each measurement location, we collect RSS of surrounding devices, both target and non-target, for a few seconds. Here non-target devices are the set of devices that the user is not interested in on-boarding or devices that may not be visually present (e.g., devices deployed in other rooms). Note that here we only focus on devices that are seemingly identical (e.g., same type and from same manufacturer). Devices of different types can be differentiated based on their device ID structure (i.e., MAC address) and device's name extracted from the beacon message. Furthermore, we also filtered out the already on-boarded devices from our measurement using their device IDs. After collecting the data, we derive statistical metric (i.e., mean, median, 95 percentile (close to maximum) and 5 percentile (close to minimum)) for each device or device ID to build RSS profile. Once we build the profiles for all device IDs, we apply our device identification algorithm to map the device ID to each measurement location, which physically represents a target device.

4.3 Identification Algorithms

Problem Formulation: For better understanding, let's first formulate the problem before describing the algorithms. Assume, we have N measurement locations for N target devices. For each measurement location, we have RSS profile for M number of device IDs that include both the target and the non-target devices ($M \geq N$). Correspondingly, we have an M -by- N matrix D , in which d_{ij} represents the RSS profile of i^{th} ($i = 1, 2, \dots, M$) device ID at j^{th} ($j = 1, 2, \dots, N$) measurement location.

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1N} \\ d_{21} & d_{22} & \dots & d_{2N} \\ \dots & \dots & \dots & \dots \\ d_{M1} & d_{M2} & \dots & d_{MN} \end{bmatrix} \quad (1)$$

Given the RSS profile matrix D , our objective is to associate the right device ID i for the measurement location j . Before describing the proposed algorithm, we describe two intuitive algorithms. Later, in evaluation, we compare these two algorithms with our propose algorithm.

Naive Algorithm. For each measurement location, this algorithm selects the device ID that has the strongest RSS. The outcome of this algorithm may vary due to the different transmission powers of different devices.

Greedy Algorithm. This algorithm improves on Naive Algorithm. It first finds the largest RSS in D , say RSS d_{ij} . Then it assigns measurement location j with device ID i . Afterwards, the row i and column j in D is set to $-\infty$. The procedure repeats N times until N devices at N measurement locations are identified. Compared to Naive Algorithm that considers a measurement location to be independent of other measurement locations, this algorithm starts with the largest RSS (normally higher confidence) and also avoids assigning same Device ID to multiple measurement locations.

Voting-based Algorithm: We propose a voting-based algorithm to consider the *likelihood* of each device ID at each measurement location. Each device i receives a vote for location j , reflecting its likelihood of being at location j . The vote is calculated as $\sum_{k=1}^N (d_{ij} - d_{ik})$. This is derived by comparing device i 's RSS at location j with other locations. A higher vote for device i at location j means that device i has greater signal strength at location j compared to that at other locations. Since each device only compares its signal strength at different locations, the vote is not affected by the difference of transmission powers between devices. Also note that the vote is jointly determined by measurement result from all locations, which makes the result more robust than the result of the greedy algorithm where a single RSS value is used.

$$V = \begin{bmatrix} \sum_{j=1}^N (d_{11} - d_{1j}) & \dots & \sum_{j=1}^N (d_{1N} - d_{1j}) \\ \sum_{j=1}^N (d_{21} - d_{2j}) & \dots & \sum_{j=1}^N (d_{2N} - d_{2j}) \\ \dots & \dots & \dots \\ \sum_{j=1}^N (d_{M1} - d_{Mj}) & \dots & \sum_{j=1}^N (d_{MN} - d_{Mj}) \end{bmatrix} \quad (2)$$

Based on the vote matrix V , we search for the largest vote summation of N elements in V . These N elements are from unique devices (i.e., different rows) and unique measurement locations (i.e., different columns). Currently, we use a brute-force method, in which we traverse every combination of N devices out of M devices, and for those N devices we traverse every combination of N measurement locations. The result is given by the combination (device-wise and location-wise) that has the largest summation. The complexity of the brute-force algorithm is exponential. We plan to explore heuristic algorithms that have polynomial complexity.

4.4 Putting All Together: AIDE

Figure 4 shows a visual prototype of using AIDE in smartphone system that associates the visual objects (images or icons) with

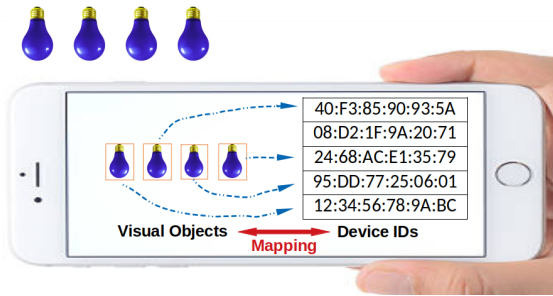


Figure 4: AIDE associates visual objects with device IDs using a smartphone

received device IDs. During the data collection phase, a user clicks a device object on screen and collects RSS at a position close to that device. The user repeats this procedure for all devices to on-board. Afterwards, AIDE automatically binds each visual object with its corresponding device ID. Then the user can control these devices, e.g., setting the brightness level of a light bulb. Note that in general, the system needs to (1) associate the physical device to its device ID, and (2) associate the physical device to its visual representation (e.g. image or icon) in the app. The mechanism we presented so far focuses on Step (1). In this simple prototype, Step (2) is done by requiring the user to click on the device image while measuring this device. This can also be done automatically by relying on the phone’s camera to recognize and track the devices using machine learning [6], which we plan to investigate as part of our future work.

5 EVALUATION

For evaluation, we conduct preliminary experiments using BLE devices. However, AIDE supports other wireless communications such as WiFi and Zigbee because it only requires RSS information.

5.1 Experimental Setup

We deploy BLE devices at three sites: a small meeting room, a medium conference room and an office corridor. Devices are placed at various locations, some not directly approachable, e.g., on ceiling, some approachable, e.g., on table or floor. We create different topologies on the ceiling including line, grid and random, and also consider the scenario with mixed target and non-target devices. We implement the measurement app using a Google Pixel 2 smartphone. At each measurement location, we collect RSS data for 30 seconds, and use the mean, median, 95 percentile, or 5 percentile value. For better usability of AIDE, it is important to reduce the time of data collection. However, reducing the duration of collection time may affect the accuracy of measurement, especially when the distance between measurement locations and devices are large. As part of the future work, we are exploring this tradeoff.

5.2 Accuracy Versus Device Distance

In this evaluation, we investigate the impact of distance between devices on measurement accuracy when the devices are not approachable (i.e. on the ceiling). Here we use a pair of devices, with distance of either 2 feet or 4 feet in between. The phone is placed 6 feet below the measured device. Thus, the maximum difference of

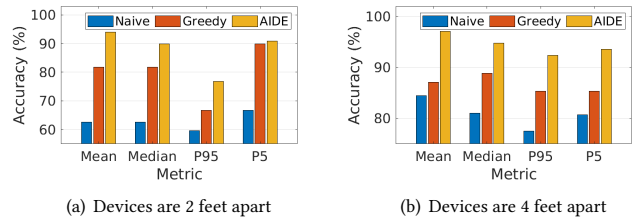


Figure 5: Accuracy of onboarding two devices on ceiling. AIDE achieves 93.4% and 97.1% in 2 feet case and 4 feet case respectively

distances between the pair of devices and the phone is only 0.3 feet (for 2 feet case) and 1.2 feet (for 4 feet case), respectively.

Figure 5 shows the overall accuracy comparison between naive, greedy and our voting-based algorithms. It clearly shows that voting-based algorithm outperforms the greedy algorithm which in turn outperforms the naive algorithm. The voting-based algorithm consistently achieves high accuracy using the Mean metric, with 93.4% and 97.1% accuracy in 2 feet and 4 feet device distance respectively. Given the fixed distance between the measurement location and the target device, we see the accuracy increases with the increase of distance between neighboring target devices. In the rest of evaluation, we use Mean in our algorithm, and compare to the other algorithms with whichever metric (e.g., 5 percentile) gives its highest accuracy.

5.3 Devices in Different Topology

We deploy multiple devices on ceiling to form two different topologies to study the accuracy. i) **Line Topology:** Where we deploy 4 devices in a line at different indoor environments. The distance between neighboring devices is 2 feet and thus the maximum difference of distance between neighboring devices to the phone is 0.3 feet. ii) **Grid Topology:** Where we deploy 6 devices into a 2-by-3 grid on ceiling in the medium conference room. The distance between neighboring devices is 4 feet and thus the maximum difference of distance between neighboring devices to the phone is 1.2 feet. Again the phone is placed 6 feet below the ceiling.

Table 1 tabulates the accuracy for both topologies. The accuracy of the voting-based algorithm is greater than the greedy algorithm, which in turn is better than the naive algorithm. More specifically, the voting-based algorithm achieves an average 86.2% accuracy for these two topology, with an average improvement of 15.7% and 28.2% compared to the greedy algorithm and the naive algorithm respectively. The accuracy in Table 1 differs from Figure 5 because here we need to separate out multiple devices instead of two. The accuracy of the grid topology is lower than the line topology because we need to onboard 6 devices in the grid topology whereas only 4 in the line topology.

5.4 Other Testing Scenarios

Target and Non-target Devices. We randomly deploy 5 target devices on ceiling at the conference room and 2 non-target devices on ceiling at corridor outside the room, imitating the case where some devices are not visually present. In this setting, the voting-based algorithm achieves 92.0% accuracy of identifying device IDs

Algorithm	Topology: Line 2 feet apart on ceiling	Topology: Grid 4 feet apart on ceiling
Naive	53.8% (median)	62.2% (mean)
Greedy	76.5% (mean)	64.4% (median)
AIDE	87.9%	84.4%

Table 1: Accuracy of onboarding multiple devices that are shaped into a line and a grid

for the target devices. We also calculate the percentage that target and non-target devices are falsely categorized as non-target (i.e., false negative) and target devices (i.e., false positive), which is 6.0% and 15.0% respectively. In general, the false ratio does not depend on the number of target and non-target devices, but rather depends on the relative signal strength of these devices. False ratio is lower when target devices have larger difference of signal strength at the measurement locations compared to non-target devices.

Approachable Scenario. In approachable setting, where we deploy 5 devices at various positions such as on table, floor and TV top in the medium conference room. In this case the voting-based algorithm achieves 100% accuracy. Here the result is as expected, as we have measured RSS very close to the target device, and thus the likelihood of that device at its measurement location is significantly higher than at other measurement locations.

6 RELATED WORK

Previously researchers have addressed the challenges of associating the physical device and the device identity under different circumstances. For instance, recently [7], researchers have used on-board inertial-sensors to correlate between motion information sensed by the sensors and the physical object detected by the camera [8]. This solution assumes target devices to be in motion, and to have on-board motion sensors. In other circumstances [9, 10], RF-aided localization techniques have been used, which require additional infrastructure support (i.e., anchor points, directional antennas [10], etc.). Furthermore, using only RSS for localization has an average estimation error of 2 meters for BLE [11], which makes it challenging to distinguish devices that are less than 1 meter apart. Unlike previous works, our proposed solution does not require infrastructure support or special hardware requirements for IoT devices.

7 DISCUSSION AND FUTURE WORK

In this paper, we have proposed AIDE that targets at an emerging necessity to on-board IoT devices in more intuitive and easy way. At the center of this solution is the voting-based algorithm that process RSS measurement to associate device identification at different physical locations. Through evaluation, we have shown that the proposed algorithm can achieve over 90% accuracy in different physical settings.

Although RSS profiles are subject to environmental changes, our data measurement procedure mitigates the effect because we collect data with the phone making circular movement for a period of time. During our data collection in the building, people occasionally walked nearby, but our system still shows promising performance. In fact, as long as there is a direct line-of-sight path between the target device and its corresponding measurement location, any blockage between this measurement location and other devices

actually improves the accuracy because the signal strength of other devices at this measurement location is reduced, which makes vote value higher for the target device for this location compared to other locations. As a result, voting based algorithm is more likely to produce the correct device mapping. Currently, we allocate 30 seconds at each measurement location. We plan to reduce the measurement time length by designing an indicator that automatically prompts to the user when to stop measuring at each location, and thus mitigate user’s burden of data collecting.

In general, our algorithm is not affected by the transmission power because it does not directly use absolute RSS values. Instead it is based on the difference of RSS at different measurement locations. Presence of WiFi devices may potentially affect the measurement for BLE devices due to channel overlap. However, during our experiment the inference of WiFi signal does not seem to have much impact. Nevertheless, we plan to explore the environment with mixture of WiFi and BLE more carefully in our future work.

Considering that BLE signals transmit at different channels (i.e., hopping) and each channel has its own characteristics, we want to explore techniques such as channel separation [12] and leverage different channels separately to improve the accuracy. In addition, we want to explore whether machine learning can result in a better RSS profile representation than the metric mean which is used in our current implementation. We also plan to study the system performance with other wireless standards such as Wi-Fi, Zigbee, etc. Finally, as part of our future work, we plan to implement and integrate the visual part of AIDE to build a user-friendly augmented on-boarding solution.

REFERENCES

- [1] Jennifer Gilburg. Zero Touch Device Onboarding for IoT Control Platforms. RSA Conference, 2017.
- [2] CUBICASA. Integrate Floor Plans into Your Smart Home and Home Automation Applications. <https://www.cubi.casa/floor-plans-smart-homes-iot/>. Accessed on Jan. 9, 2019.
- [3] Hamedani AG, Liu SW, Brown DF, Asplin BR, and Camargo Jr. Vertical Patient Flow, an Innovative Approach to Crowding In US Academic Emergency Departments. *Annals of Emergency Medicine*, 2010.
- [4] Cameron Klotz. IoT Applied in Retail 5 Real Examples. <https://hardwaremassive.com/resources/guest-articles/iot-applied-in-retail-5-real-examples>. Accessed on Jan. 9, 2019.
- [5] Alexander Coolidge. Kroger Tests "Smart Shelf" Technology. <https://www.cincinnati.com/story/money/2015/10/02/next-shelves-giving-cues-kroger/73218252/>. Accessed on Jan. 9, 2019.
- [6] Jongwon Choi, Hyung Jin Chang, Tobias Fischer, Sangdoon Yun, Kyuewang Lee, Jiyeoup Jeong, Yiannis Demiris, and Jin Young Choi. Context-aware Deep Feature Compression for High-speed Visual Tracking. In *CVPR*, 2018.
- [7] Carlos Ruiz, Shijia Pan, Adeola Bannis, Xinlei Chen, Carlee Joe-Wong, Hae Young Noh, and Pei Zhang. IDrone: Robust Identification through Motion Actuation Feedback. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2018.
- [8] Shijia Pan, Carlos Ruiz, Jun Han, Adeola Bannis, Patrick Tague, Hae Young Noh, and Pei Zhang. UniverSense: IoT Device Pairing through Heterogeneous Sensing Signals. In *ACM HotMobile*, 2018.
- [9] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. Accurate Indoor Localization with Zero Start-up Cost. In *ACM MobiCom*, 2014.
- [10] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. Avoiding Multipath to Revive Inbuilding WiFi Localization. In *ACM MobiSys*, 2013.
- [11] Dongyao Chen, Kang G. Shin, Yurong Jiang, and Kyu-Han Kim. Locating and Tracking BLE Beacons with Smartphones. In *ACM CoNEXT*, 2017.
- [12] Rahul Majethia and Krishnan Rajkumar. Mining Channel State Information from Bluetooth Low Energy RSSI for Robust Object-to-Object Ranging. In *ACM IOT*, 2018.