2022 2nd Workshop on Data-Driven and Intelligent Cyber-Physical Systems for Smart Cities Workshop (DI-CPS) | 978-1-6654-7042-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/DI-CPS56137.2022.00008

Establishing Trust in Vehicle-to-Vehicle Coordination: A Sensor Fusion Approach

Jakob Veselsky, Jack West, Isaac Ahlgren, George K. Thiruvathukal, Neil Klingensmith
Loyola University Chicago
Abhinav Goel, Wenxin Jiang, and James C. Davis
Purdue University
Kyuin Lee and Younghyun Kim
University of Wisconsin - Madison

***Abstract*—Autonomous vehicles (AVs) use diverse sensors to understand their surroundings as they continually make safety-critical decisions. However, establishing trust with other AVs is a key prerequisite because safety-critical decisions cannot be made based on data shared from untrusted sources. Existing protocols require an infrastructure network connection and a third-party root of trust to establish a secure channel, which are not always available.**

In this paper, we propose a sensor-fusion approach for mobile trust establishment, which combines GPS and visual data. The combined data forms evidence that one vehicle is nearby another, which is a strong indication that it is not a remote adversary hence trustworthy. Our preliminary experiments show that our sensor-fusion approach achieves above 80% successful pairing of two legitimate vehicles observing the same object with 5 meters of error. Based on these preliminary results, we anticipate that a refined approach can support fuzzy trust establishment, enabling better collaboration between nearby AVs.

I. Introduction

As we add more autonomous and semi-autonomous vehicles (AVs) to our roads, their effects on passenger and pedestrian safety are becoming more important. Despite extensive testing before deployment, AV systems are not perfect at identifying hazards in the roadway [1], [2]. Although a particular AV's sensors and software may not be 100% accurate at identifying hazards, there is untapped pool of information held by *other* AVs in the vicinity that could be used to more quickly and accurately identify roadway hazards before they present a safety threat.

Enabling *coordination* between untrusting AVs is a significant challenge [3], [4], [5]. Because AVs are safety-critical systems, they cannot make decisions based on data from untrusted external sources. Existing vehicle-to-vehicle (V2V) standards lack a workable trust scheme for vehicles and the data they share. Consider the scenario depicted in Figure 1:

A vehicle (③) driving down a narrow side street in a densely-populated city comes to a stop at an intersection with a busy road. The vehicle needs to turn into the busy road, but parked cars and trees prevent the driver from seeing cross traffic. Other vehicles (① and ②) on the busy road have a clear view of traffic conditions, and they could alert vehicle ③) when it is safe to turn.

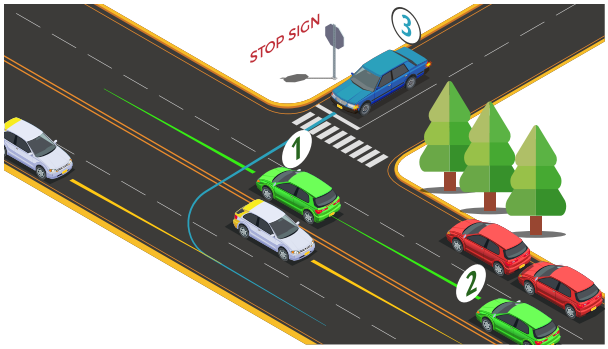

Fig. 1: Vehicles in the roadway gathering visual data from objects. The keys they generate will be based on overlapping observations of moving objects.

If the occlusions in the roadway are too large, vehicle ② may not be visible to vehicle ③, and it could be dangerous to begin the turn immediately after vehicle ① passes. In our experience driving in dense urban environments, situations like these arise frequently, and they could be easily prevented with coordination. But even simple AV coordination—in the scenario, vehicle ② broadcasting its position and velocity to vehicle ③—requires some reliable method to establish trust.

As a first step toward enabling AV coordination, we need a mechanism by which AVs can establish mutual trust. Existing V2V coordination standards like DSRC [6], C-V2X [7], and WAVE [8] rely on a public key infrastructure to authenticate the source of transmissions. But mobile AVs may not always be able to rely on a centralized trust broker, *e.g.,* in areas with spotty cellular coverage. Since we know that the most prevalent class of attack on cyber-physical systems involves a physically remote attacker [9], [10], our goal is to establish mutual trust among vehicles by taking advantage of dynamic observations of the physical environment.

In this work, we lay out the considerations behind building a workable trustworthy V2V coordination system. We then put forward a preliminary architecture for a secure V2V trust system, including a description and evaluation of an early-stage prototype.

The motivation behind this work is that evidence of proximity is a proxy for trust, even – under the right circumstances –

978-1-6654-7042-1/22/$31.00 ©2022 IEEE
DOI 10.1109/DI-CPS56137.2022.00008

in a public space. Here we build on our previous contributions to zero-involvement pairing and authentication (ZIPA), which is a suite of techniques for producing trust and encryption tokens from correlated environmental noise. With little coordination overhead, two nearby vehicles that observe the same scene can prove to one another that they are nearby. The techniques we present are also applicable in other scenarios like CCTV networks in which nearby devices wish to establish trust using visual observations in order to coordinate.

Our proposal addresses the problem of low-latency trust for use in vehicle-to-vehicle coordinations. We begin by discussing the desired properties of a trustworthy V2V coordination system along with some high-level ideas about what the shape of a solution should be. Finally, we present a prototype implementation that achieves the goals that we set out in our design objectives and suggests some future directions.

II. Design Objectives

Here we discuss the challenges, threat model, and the properties of our solution to the trust establishment problem.

A. Challenges of V2V Trust Establishment

Recent attempts to build V2V trust establishment from shared observations of the roadway have met only moderate success [11]. The main difficulty is finding a source of environmental entropy that is highly correlated for legitimate observers in the roadway but unobservable outside of the roadway. The state of the art in V2V trust establishment is Convoy [11], which uses low-cost sensors to generate correlated random numbers from environmental noise. Convoy produces keys at a rate on the order of one bit per second, which may be insufficient for real-time trust establishment. As such, pairing with low-cost sensors is limited to specialized applications where conditions do not evolve quickly and slow key generation is acceptable.

AVs have richer sensors on board like LIDAR and cameras. A challenge in using the sensors on an AV is that two vehicles observing the same scene from different angles will generate different raw image data. To use these observations as a trust token, we need a function that maps two or more "similar-enough" observations of the same environment to the same binary sequence. For example, one camera may capture a complete view of only the North side of a roadway while another may capture a complete view of only the South side of the same roadway as in Figure 1. The whole process is complicated by the fact that observations of the physical environment contain error. Getting accurate observations of the physical environment is central to proving that two vehicles that wish to establish trust are in the same location. Although there may be some overlap in the images they capture, extracting and identifying that overlap without divulging too much information to an eavesdropper is difficult.

B. Threat Model

Recent attacks on critical infrastructure [9], [10] have a shared threat model: the attack was launched over the network; the attacker was not physically nearby. We base our threat model on this vector. Our adversary, Eve, does not have physical access to the communicating vehicles' immediate environment, although she can access archival information such as aerial photographs, satellite images, or publicly available photographs (*e.g.,* Google Street View). Beyond this, Eve can observe broadcast communications between legitimate users Alice and Bob and can broadcast her own messages. However, she cannot gather detailed information about the location and motion of objects in real time.

This threat model excludes physically onboard attackers such as those supposed in the work on Jeep hacking [12]. Onboard attackers are in a position to do far more than spoof trust. We treat physical proximity as a source of trust. The threat model is practical in the case where an adversary is unaware of the exact location of their target. In this threat model, it is unbeknownst to the attacker where to strike.

C. Properties of Our Proposed Solution

A V2V trust scheme needs three properties: it is *peer-to-peer* for practicality, it uses *environmental information* for trust, and is *low latency* for safety.

Peer-to-peer: Vehicles in a peer-to-peer networks transmit information to others that are within broadcast range without incurring long latency to a remote datacenter. Distant vehicles outside the single-hop wireless range are also outside of the local traffic pattern and do not need the information being shared about the immediate surroundings. And by eliminating reliance on datacenters for facilitating information sharing, we can reduce the attack surface of AV networks. Without a centralized hub of coordination, it is more difficult for an adversary to compromise traffic safety on a large scale. However, peer-to-peer coordination makes trust establishment more difficult because each vehicle on the road must be able to establish trust with any other vehicle on the road.

a) Proximity is Trust: Based on our threat model, physically distant participants are untrustworthy. Pragmatically, they also may not possess relevant information, since environmental conditions change. Conventional approaches to trust establishment such as distributing individual secret keys to each car on the roadway are not scalable in peer-to-peer networks because of the coordination overhead they incur. Thus, only physically nearby AVs are candidates for trust.

This design goal is shared by work on context-based zero involvement pairing and trust (ZIPA), a method for transparently validating another device's legitimacy to join a network based on location [13], [14]. Observations are authenticated by real-time environmental observations, allowing participants to generate a shared key by harvesting environmental noise. We use a philosophy motivated by ZIPA to prevent remote attackers from authenticating themselves to legitimate vehicles. It requires all coordinations to be authenticated with observations of shared environmental context.

b) Latency affects Safety: Vehicle-to-vehicle (V2V) networks allow vehicles to share safety-sensitive information. For example, one vehicle can inform another about environmental dangers such as an object on a roadway. At high speeds, *e.g.,*

automobiles or drones, a lengthy trust protocol can endanger the participants. Fast trust matters.

III. Related Work

In other projects, authors have used point clouds and other visual data to approximate indoor localization and zero involvement trust.

a) Indoor Localization: Our work considers the outdoor localization problem. Other researchers have considered the *indoor* problem. Inaccuracies of a GPS, when indoors, pose a problem for devices that rely on GPS data for navigation. One solution lies in cost effective depth sensors (a Kinect for example)[15], [16], [17] which effectively allow for devices to navigate indoor environments. Our work relates to indoor localization by both areas of research requiring accurate location algorithms. However, an indoor localization project solely focuses indoor navigation which currently addresses singular device navigation rather than a system of devices.

b) Zero-Involvement Pairing and Authentication (ZIPA) : ZIPA refers to the idea of encrypting data based on a device's current position [14], [13]. Our work relates to these works by sharing a mutual goal of encrypting data using localization. However, these works are different because they use stationary GPS measurements. Our work aims to deal with coordinates that can change in each frame.

IV. Our Approach to Trustworthy V2V Coordination

Our approach uses the sensors present on an AV to find the location of objects in the roadway. The results are used as the basis to generate a key. Since the estimate of an object's location may be different depending on the vantage point of the observer, it cannot be directly used to authenticate vehicles. To deal with errors in object location estimates, we use a fuzzy extractor to convert an object's approximate position to a bit sequence on which all observers of the scene can agree.

Because a principal objective of this work is to build a key that can only be computed by observers that are physically located in the roadway, we use only moving objects as the basis for the keys. Stationary objects like signposts and buildings could be observable using other means (such as from satellite imagery or historical observations of the roadway) and should not contribute to the key.

A. Algorithm Overview

Our algorithm works in three steps:

1) We segment a point cloud from the Kinect into clusters. Each cluster represents an object in the scene. We then calculate the centroid of each cluster.
2) We estimate the absolute GPS coordinates of the centroid of each object from step 1. This is accomplished by adding the distance to the centroid to the GPS coordinates of the Kinect using the Halversine Equation.
3) We then draw a circle of fixed radius around each centroid from step 2 and select points within that circle to be used as private information.

Algorithm 1: Algorithm to generate fuzzy extractor entities from observations of the roadway.

Input: A GPS reading G
Input: A point cloud $P = \{p_1, p_2, \ldots, p_N\}$
Input: A leniency parameter q
Input: Randomly chosen GPS coordinate points $E = \{e_1, e_2, \ldots, e_L\}$
Output: A subset f of E

```
/* Segment point clouds                                   */
objects[] ← segmentPointCloud(P)
/* Find GPS location of centroid of each object.          */
for k ← 1 to length(objects) do
    /* cent[k] is relative location of centroid (x,y)
       */
    cent[k].x ← mean(objects[k].points.x[])
    cent[k].y ← mean(objects[k].points.y[])
    /* cent[k].gps is estimated GPS coords of centroid
       */
    cent[k].gps ← halversine(G, cent[k])
    /* Fuzzy point selection                              */
    for k ← 1 to length(E) do
        if ||E[k] − cent[k].gps||_2 < q then
            append(f, E[k])
return f
```

Algorithm 2: Encoding algorithm to ensure no private information leakage.

Input: f a list of points gathered from Algorithm 1
Input: EC an Elliptic Curve known to all AVs
Input: A field $\mathbb{F}_p$
Output: K a list of encoded points

```
/* p is the prime number of the field F_p              */
for k ← 1 to length(f) do
    /* f[k].x/y are the GPS coords generated in Algo 1
       */
    /* round refers to a function that rounds to a
       decimal place.                                  */
    i = |round(f[k].x * f[k].y, 5) * p| mod p
    /* EC.G is the generator point associated with
       the given elliptic curve.                       */
    K.append(i * EC.G)
return K
```

B. Segmenting Point Clouds into Objects

The Kinect generates point clouds of approximately two million points at a frequency of 1-2 Hertz. These point clouds must be segmented to identify which points belong to individual objects in the surrounding environment. We segment these objects using the difference of normals algorithm [18].

We remove background and stationary objects such as the ground, buildings, posts, etc. at this stage because their locations are remotely observable from satellite imagery. The remaining data represents dynamic objects in the environment that are changing location between frames.

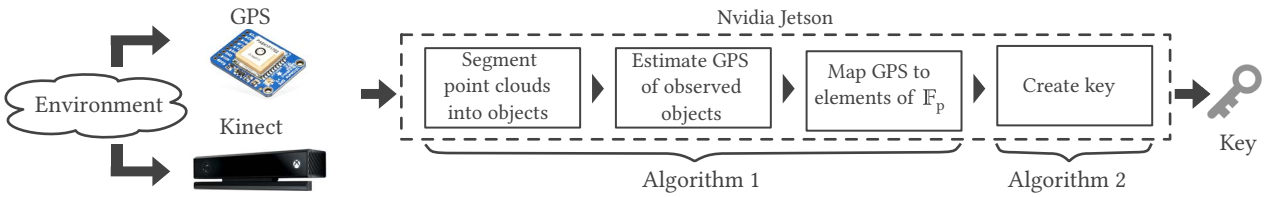

Fig. 2: Sensor Fusion-Based key generation pipeline.

Fig. 3: Our prototype mounted to a vehicle.

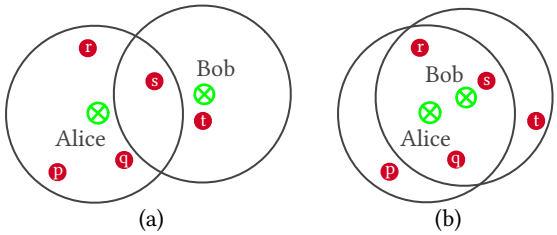

Fig. 4: Alice and Bob use a constellation of points to prove that they are located near one another.

C. GPS Location Estimation

Next, we calculate the centroid of each object in the segmented point clouds generated in §IV-B. The centroids are the private data shared only among legitimate AVs. The coordinates of the centroids are calculated from point clouds generated by the Kinect, which are reported relative to the point of observation (i.e. the location of the Kinect). We calculate the absolute location of each object's centroid by adding the coordinates of the observation point (measured by GPS) to the relative location reported by the Kinect. Observations of the same object from two different vantage points may have error of up to 10 meters, caused by uncertainty in location reported by the GPS.

D. Fuzzy Key Generation

Finally, from estimates of the object's centroid we generate a key that can be publicly shared without leaking information to an eavesdropper Eve, based on our threat model. Figure 4 show two scenarios where Alice and Bob are trying to establish trust. The circles for Alice and Bob enclose a point cloud with a particular centroid (marked with $\bigotimes$). In Figure 4(a), Alice and Bob's estimates of the object's centroid are different enough to represent a case where the overlap of the circles are too small to authenticate. Whereas, in Figure 4(b), Alice and Bob's estimates of the object's centroid are similar enough

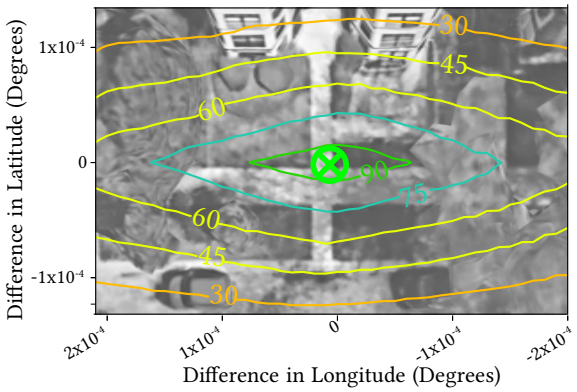

Fig. 5: Contour map of the percentage of matching points based on distance.

to properly authenticate even though their centroids differ. In Figure 4, p, q, r, s and t are elements in the set E elements from Algorithm 1 which Alice and Bob want to exchange. However, Alice and Bob still have one more problem. If either one of them sends their list over a public channel, an adversary can estimate what Alice's private centroid could be. Therefore, Alice and Bob encode their points. Our prototype implementation uses a simple elliptic curve encoder which is described in Algorithm 2. Elliptic curves are fast, secure, and every component of an elliptic curve can be public[19].

V. Preliminary Evaluation

We evaluated our prototype with data gathered on roads near our office building. This evaluation focused on point cloud estimation, fuzzy key generation and fuzzy key quality. The keys produced by our prototype are 256 bits in length.

a) GPS Location Estimation: The estimated GPS coordinates of the centroid computed in Algorithm 1 have some expected error. We measured that error to be in the range of $5 - 10m$. Figure 6 shows the match rate of fuzzy extractor entities f as a function of error in the centroid location estimate. Figure 5 shows a contour map of match rate as a function of position estimate near our department's office building. The green X at the center of Figure 5 represents the true location of an object's centroid, and the contour lines represent the match rate of fuzzy extractor entities at nearby locations.

b) Fuzzy Key Generation: For our real-world data, we evaluated perceived centroids from our prototype and simulated point set agreement over varying fake centroids. Figure 5 shows how estimated centroids, located in the center of the figure, vary over latitude and longitude. Other points,

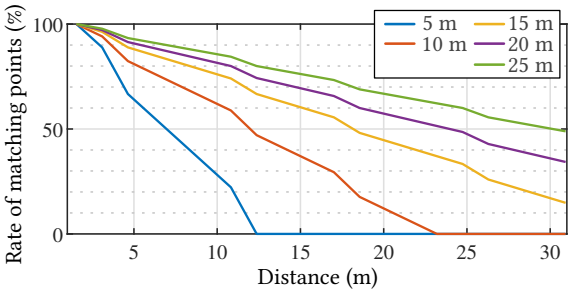

Fig. 6: Matching rate as a function of distance away from the centroid.

Test	Result	H2H[20]
Frequency	✓	✓
Block Frequency	✓	✓
Cumulative Sums	✓	✓
Runs	✓	×
Longest Run	✓	×
Rank FFT	✓	✓
Non-Overlapping	✓	✓
Overlapping	✓	×
Universal	×	×
Approximate Entropy	✓	✓
Serial	✓	×
Linear Complexity	✓	✓

TABLE I: NIST test results with data from our fuzzy extractor (✓indicates pass and × indicates failure).

besides the reference point, were simulated. The contours are overlayed over a to-scale satellite image of where the test subject was standing; a error radius of ten meters was used as a constant for the creation of the figure. The figure's contours depict distance limits where the percentages every point within the contour have a greater percentage than the contour outlining the area. The figure is able to characterize the fuzzy limits of a ten meter error limit over latitude and longitude differences.

c) Randomness Of Generated Bit Streams: We gathered 14 million bits using simulated real-number centroids to properly evaluate randomness of our keys. To evaluate key randomness of our fuzzy key generation algorithm, we used the NIST randomness test suite[21]. As shown in our other work[22], ZIPA schemes have trouble passing the NIST test evaluation. However, Table I demonstrates that our fuzzy key generation algorithm has no trouble passing almost all of the tests. For comparison, we also show NIST test results from Heart 2 Heart [20], which has about average performance relative to other ZIPA schemes. The results of the NIST test imply our algorithm's bit sequences are strong enough to be sent over a public channel.

VI. Conclusions

Our sensor fusion approach is a promising starting point on which to build trust based on applied computer vision in AVs. It supports peer-to-peer coordination using proof of proximity. A full system implementation will require further investigation into implementation details on an real AV. For our techniques to be useful in real AVs, estimates of centroid location must be fast to compute and accurate. This can be accomplished by using more powerful computers and more responsive sensors such as LIDAR or stereo cameras—topics that we are currently investigating.

We also did not discuss algorithms for selecting objects in the roadway that should be used for trust establishment. In our preliminary evaluation, we used a rudimentary algorithm along with a dataset that focused on one or two subjects in the frame of view. But current AVs have more sophisticated techniques for identifying moving objects and forecasting their trajectories, which could be used as inputs to our trust establishment mechanism.

A major advantage of our techniques is that the penalty of trust failure is low: in a crowded roadway, many vehicles are observing the same scene from different vantage points, and a hazard will likely be observable by more than one vehicle. If one legitimate AV with useful information about the hazard fails to establish trust with a second vehicle, that useful information can still be shared by other vehicles with the same information. And AVs can always fall back on their own local sensors to corroborate information they receive from others. In this work, we don't discuss corroboration techniques, but they are an important piece of an AV trust establishment system and a topic of future work.

Our algorithm has several limitations which are open directions for further research. *Low-entropy environments*, such as an open field, will likely lack objects in proximity for our trust establishment algorithm. Therefore, a research direction is to choose among objects for use in key generation, *e.g.,* building on object detection methods to distinguish "suspicious" objects or to focus on moving objects.

References

[1] B. Vlasic and N. E. Boudette, "Self-driving tesla was involved in fatal crash, u.s. says," *The New York Times*, Jun 2016. [Online]. Available: https://www.nytimes.com/2016/07/01/business/self-driving-tesla-fatal-crash-investigation.html

[2] D. Wakabayashi, "Self-driving uber car kills pedestrian in arizona, where robots roam," *The New York Times*, Mar 2018. [Online]. Available: https://www.nytimes.com/2018/03/19/technology/uber-driverless-fatality.html

[3] W. Chen, R. K. Guha, J. Lee, R. Onishi, and R. Vuyyuru, "Methods for context driven disruption tolerant vehicular networking in dynamic roadway environments," Sep. 22 2011, uS Patent App. 12/724,623.

[4] M. B. Takla, "Systems and methods for controlling vehicle-to-everything personal safety message transmission," May 11 2021, uS Patent 11,006,264.

[5] S. P. McFarland *et al.*, "Transport dangerous driving reporting," Aug. 31 2021, uS Patent 11,107,355.

[6] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the united states," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.

[7] ETSI, "Service requirements for V2X services," Mar 2017.

[8] Y. J. Li, "An overview of the dsrc/wave technology," in *Quality, Reliability, Security and Robustness in Heterogeneous Networks*, X. Zhang and D. Qiao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 544–558.

[9] M. D. Shear, N. Perlroth, and C. Krauss, "Colonial pipeline paid roughly $5 million in ransom to hackers," *The New York Times*, May 2021. [Online]. Available: https://www.nytimes.com/2021/05/13/us/politics/biden-colonial-pipeline-ransomware.html

[10] N. Perlroth, N. Scheiber, and J. Creswell, "Russian cybercriminal group was behind meat plant attack, f.b.i. says," *The New York Times*, Jun 2021. [Online]. Available: https://www.nytimes.com/2021/06/02/business/jbs-beef-cyberattack.html

[11] J. Han, M. Harishankar, X. Wang, A. J. Chung, and P. Tague, "Convoy: Physical context verification for vehicle platoon admission," in *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, ser. HotMobile '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 73–78. [Online]. Available: https://doi.org/10.1145/3032970.3032987

[12] A. Greenberg, "The jeep hackers are back to prove car hacking can get much worse," *Wired*, Aug 2016. [Online]. Available: https://www.wired.com/2016/08/jeep-hackers-return-high-speed-steering-acceleration-hacks/

[13] M. McLoone and M. J. Robshaw, "Public key cryptography and rfid tags," in *Cryptographers' Track at the RSA Conference*. Springer, 2007, pp. 372–384.

[14] K. Ghorbani, N. Orouji, and M. R. Mosavi, "Navigation message authentication based on one-way hash chain to mitigate spoofing attacks for gps l1," *Wireless Personal Communications*, vol. 113, no. 4, pp. 1743–1754, 2020.

[15] E. I. Al Khatib, M. A. K. Jaradat, and M. F. Abdel-Hafez, "Low-cost reduced navigation system for mobile robot in indoor/outdoor environments," *IEEE Access*, vol. 8, pp. 25 014–25 026, 2020.

[16] G. Jóźków, C. Toth, Z. Koppanyi, and D. Grejner-Brzezinska, "Combined matching of 2d and 3d kinect™ data to support indoor mapping and navigation," in *ASPRS 2014 Annual Conference, Louisville, Kentucky*, 2014.

[17] H. I. M. A. Omara and K. S. M. Sahari, "Indoor mapping using kinect and ros," in *2015 International Symposium on Agents, Multi-Agent Systems and Robotics (ISAMSR)*. IEEE, 2015, pp. 110–116.

[18] Y. Ioannou, B. Taati, R. Harrap, and M. Greenspan, "Difference of normals as a multi-scale operator in unorganized point clouds," 10 2012.

[19] R. R. Ahirwal and M. Ahke, "Elliptic curve diffie-hellman key exchange algorithm for securing hypertext information on wide area network," *International Journal of Computer Science and Information Technologies*, vol. 4, no. 2, pp. 363–368, 2013.

[20] M. Rostami, A. Juels, and F. Koushanfar, "Heart-to-heart (h2h): authentication for implanted medical devices," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, ser. CCS '13. New York, NY, USA: ACM, 2013, pp. 1099–1112. [Online]. Available: http://doi.acm.org/10.1145/2508859.2516658

[21] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications."

[22] J. West, K. Lee, S. Banerjee, Y. Kim, G. K. Thiruvathukal, and N. Klingensmith, "Moonshine: An online randomness distiller for zero-involvement authentication," in *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (Co-Located with CPS-IoT Week 2021)*, ser. IPSN '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 93–105. [Online]. Available: https://doi.org/10.1145/3412382.3458899