AirID: Injecting a Custom RF Fingerprint for Enhanced UAV Identification using Deep Learning

Subhramoy Mohanti*, Nasim Soltani*, Kunal Sankhe*, Dheryta Jaisinghani*,
Marco Di Felice† and Kaushik Chowdhury*
*Institute for the Wireless Internet of Things, Northeastern University, Boston, USA
†Department of Computer Science, University of Bologna, Bologna, Italy
Email: smohanti@coe.neu.edu, {soltani.n, sankhe.ku}@husky.neu.edu, dheryta@ieee.org,
marco.difelice3@unibo.it, krc@ece.neu.edu

***Abstract*—We propose a framework called AirID that identifies friendly/authorized UAVs using RF signals emitted by radios mounted on them through a technique called as RF fingerprinting. Our main contribution is a method of intentionally inserting 'signatures' in the transmitted I/Q samples from each UAV, which are detected through a deep convolutional neural network (CNN) at the physical layer, without affecting the ongoing UAV data communication process. Specifically, AirID addresses the challenge of how to overcome the channel-induced perturbations in the transmitted signal that lowers identification accuracy. AirID is implemented using Ettus B200mini Software Defined Radios (SDRs) that serve as both static ground UAV identifiers, as well as mounted on DJI Matrice M100 UAVs to perform the identification collaboratively as an aerial swarm. AirID tackles the well-known problem of low RF fingerprinting accuracy in 'train on one day test on another day' conditions as the aerial environment is constantly changing. Results reveal 98% identification accuracy for authorized UAVs, while maintaining a stable communication BER of 10^{-4} for the evaluated cases.**

I. Introduction

The US Federal Aviation Administration predicts the number of UAVs to reach 823,000 by the year 2023 [1] given the many civilian, military and public safety applications [2, 3]. However, intentional attacks or accidents caused by flying UAVs in proximity of airports [4] are some examples of adverse impact. Furthermore, as the UAV market matures, there is a risk of the same models being used by both malicious actors and legitimate users. In such cases, accurate identification of UAVs is of paramount importance.

There exist several sophisticated techniques for UAV detection such as, the use of IR sensors for thermal imaging [5], radar assisted detection [6], and image processing [7] and for device localization [8]. All these approaches pose additional challenges for identifying the type of UAV beyond the simpler binary problem of detecting if a UAV exists. Approaches that involve transmitting a recognized device ID or a predetermined preamble can be foiled by an adversary through software modifications [9].

•**UAV Identification with AirID:** Considering all of the above, we propose to identify a specific, known UAV within a large pool of UAVs of the same model, by using physical (PHY) layer information. This does not require any changes to the upper layers of the protocol stack. We use a technique known as – *RF fingerprinting* [10, 11], which passively identifies a given transmitter using received I (In-phase) and Q (Quadrature) samples. Most prior works have pointed out that RF fingerprinting does not work well when the channel conditions vary, i.e., a machine learning classifier trained in one channel environment fails to perform with acceptable accuracy when the wireless channel changes. To address this problem in context of hovering UAVs, we intentionally *introduce* a processing block at the transmitter side that modifies the I/Q samples before over-the-air transmission (see Fig. 2(a)). The resulting controlled modifications of the I/Q samples add to the channel impairments, which are both compensated at the receiver. Thus, the challenge lies in designing the type and amplitude of injected impairments to keep the Bit Error Rate (BER) of ongoing communications within a threshold.

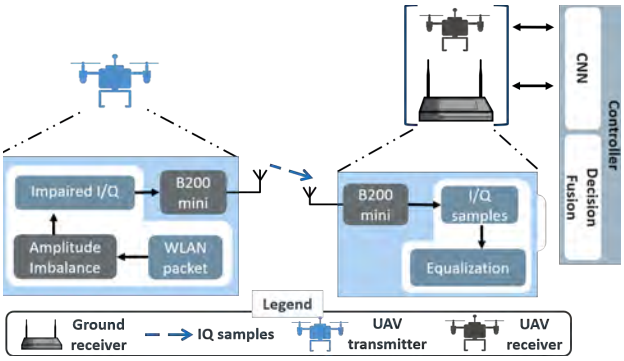
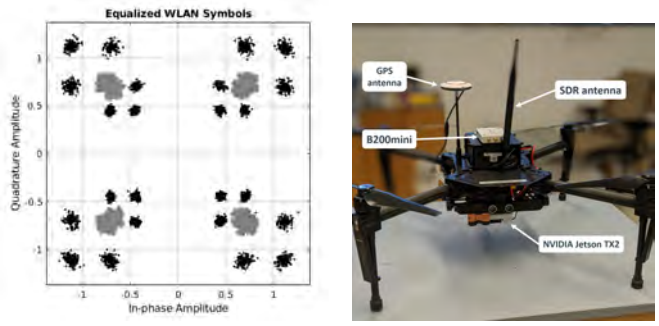


Fig. 1: System architecture for AirID: UAVs transmitting data with impaired I/Q signatures facilitate identification by ground and UAV units, assisted by a neural network enabled centralized controller

• **AirID framework description:** The schematic overview of AirID is shown in Fig. 1. Its building blocks are – (i) Ground and UAV-swarm identifiers, which we refer to as ground receivers and UAV receivers, respectively, for simplicity. These receivers record transmitted I/Q samples from signals emitted by all the UAVs that need to be identified and post-process the received data. This data is then delivered through a wireless backhaul to a central controller. (ii) Multiple distributed UAVs transmit signals, which are preloaded with a I/Q symbol impairment block to introduce a distinct PHY layer signature, for identification. (iii) A central



(a) (b)

Fig. 2: (a) Default QPSK constellation with no added impairment (grey). The impaired QPSK constellation (black) after injecting a unique 4dB amplitude impairment to the original I/Q symbols. (b) DJI Matrice M100 drone, with B200mini SDR and NVIDIA Jetson TX2 module installed.

controller receives the processed data from each receiver, and executes the pre-trained CNN model on all ground and receiver UAV inputs to detect unique RF signatures and performs decision fusion to finalize the identification outcome. AirID solves a number of systems challenges in the course of the implementation, which we identify as follows – (i) selecting lightweight but, capable SDRs and embedded computing hosts with high performance GPUs, which can be easily mounted on UAVs without exceeding the maximum take-off weight and, (ii) making the machine learning model robust to unpredictable variations in the wireless channel, given the highly dynamic channel conditions that UAVs fly in.

The main contributions of AirID are as follows–

- We study the effect of the wireless channel on using RF fingerprinting for UAV transmissions by collecting signals emitted by SDRs mounted on UAVs. We then propose a CNN architecture and an impairment selection algorithm that permits offline training of the CNN on simulated data, follwed by injection of the fingerprint into the UAVs post training. [Section II]
- We demonstrate the feasibility of using UAVs as receivers with real-world flying experiments, even when the CNN model is trained and tested in different environments [Section III]. This experimental dataset is available at [12].
- We implement AirID on actual UAVs that carry SDRs and conduct numerous tests to measure the impact of distance, displacement, and interference within an open space of 100 sq. ft. We demonstrate that with multiple UAV receivers, AirID efficiently fuses decisions to achieve up to 98% identification accuracy, which is ≈2x higher than alternate approaches. [Section IV]

II. Learning Fingerprints for UAV Identification

We begin by explaining the method of data transmission and reception in UAVs and how we design the CNN to identify the received RF fingerprints from UAV transmitters.

A. Mounting SDRs on UAV for communication

As shown in Fig. 2(b), we use the DJI Matrice M100 UAV as it allows customization of the airframe. We choose the Ettus B200mini SDR as the reconfigurable transceiver and an NVIDIA Jetson TX2 as the host computing module that controls the SDRs. We load the Jetson TX2 with complex-valued I/Q samples generated from WiFi 802.11a compliant packets created through MATLAB WLAN toolbox. We transmit these data packets through the B200mini SDR mounted on the UAV. At the receiver side, the same SDRs (both on ground and on UAVs), capture the UAV transmission in the form of I/Q samples that now have the added channel-induced impairment. We next describe experimental results on using RF fingerprinting for the SDRs mounted on the UAVs and transmitting in the 900MHz ISM band.

B. Deep Convolutional Neural Network design

Referring to [13], we partition each long sequence of I/Q samples collected from UAVs, into 60%, 20% and 20% portioons, to form our training, validation and test sets, respectively. For each set, we use a sliding window of size 1024 (as in [14]) with a stride of one, to form "slices". We further separate the I and Q symbols to form a tensor with size (batch_size,slice_size,2) = $(128, 1024, 2)$, where the third dimension (2) is represented by I and Q symbols. We feed these tensors to the CNN as input in the training and testing phase. The output of the neural network is a probability vector of size N, indicating the probability of predicting each device, where N corresponds to the number of UAVs that we aim to identify. We adopt the CNN presented in [14], previously shown to be effective for modulation classification. This architecture, shown in Fig. 3, is a 1-dimensional version of the well known VGG architecture [15] and has 159,173 parameters, 7 convolution layers (each with 64 spatial filters), interleaved with MaxPooling layers. The Conv/MaxPooling stack is followed by 3 fully connected (dense) layers at the end of the sequential architecture. In all convolution layers, filters learn a 3-sample variation over their input features, and generate 64 output features. The output features are down-sampled using MaxPooling layers. The Conv/MaxPooling stack, is followed by 2 fully connected layers with output size of 128 and 128, and a Softmax layer with size N. The network is trained using Adam optimizer with a learning rate of 0.0001. At the end of each epoch, we test the trained network on the validation set. For each validation slice in the input, the class with the highest probability in the output is the predicted class. We measure validation accuracy by dividing the number of correctly predicted slices by the total number of validated slices. We stop training when validation accuracy does not improve for 3 consecutive epochs. After this, we enter the testing phase where we use our 'never-seen-before' test set to demonstrate well the trained CNN can generalize to unseen data. Next, we demonstrate how the CNN trained through unaltered data collected under one channel condition does not perform well when deployed in the field under totally different conditions.

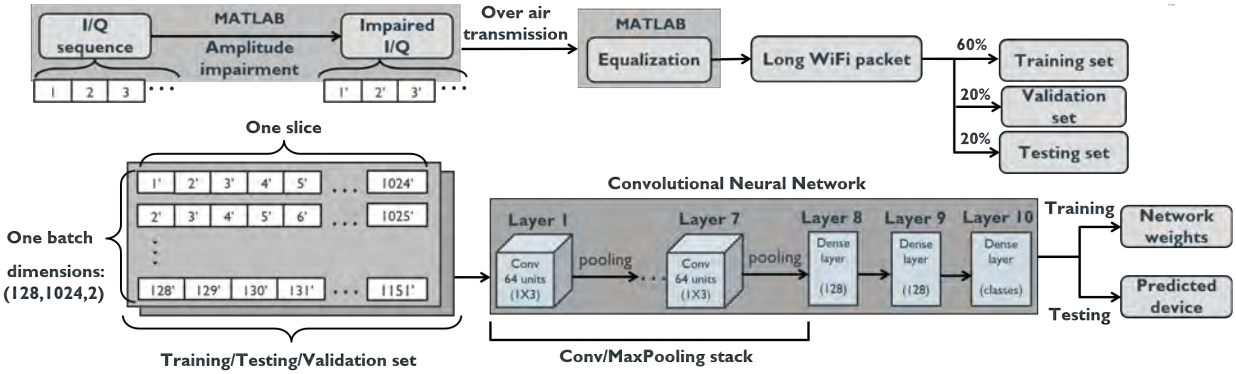


Fig. 3: Tensor forming, training and testing processes in the CNN.

C. Creating Unique RF Fingerprints

In simulation, we induce artificial impairment into I/Q symbols of 802.11a baseband waveform to modify their amplitude and phase intentionally. We choose I/Q imbalance as our selected impairment, since (i) it is independent of the environment, and (ii) do not apply only in context of a specific transmitter receiver pair (as opposed to, say, relative phase offset). I/Q imbalance is often represented as a combination of amplitude and phase impairments. Each such amplitude/phase impairment is unique to a given SDR and imparts it a new identity. This allows the CNN to easily distinguish between modified radios. In Fig. 2(a), the constellation points in black, indicate intentionally designed shifts from the ideal position in grey, due an amplitude scaling of 4 dB that is added to the original I/Q samples. In this work, we focus on only adding amplitude changes in the real/imaginary parts, which results in only I/Q imbalance. Note that an increased levels of impairments negatively affects the BER; thus, neither an infinite number of impairments nor arbitrary levels of impairments are possible.

1) Effect of impairment addition on BER: High levels of amplitude imbalance translates to higher mismatch of gain in the parallel sections of RF chain processing I and Q signal paths. Thus, this lowers the quality of the outcome of the quadrature mixer in the SDR transmitter. This mismatch alters the transmitted I/Q samples, and if not controlled, it can lead to high BER at the receiver. Our goal is to not exceed a BER of 10^{-4} after adding the artificial signatures into the signal. Through MATLAB simulations, we test the BER resulting from 10 different amplitude impairments, from 1 to 10dB, in steps of 1dB, in SNR conditions ranging from low (0dB) to high (24dB). As shown in Fig. 4, high BER causes packets to be discarded for any value of amplitude impairment exceeding 5dB. Thus, we inject an amplitude impairment in the SDR mounted on the UAV within the range 1 to 5dB.

2) Effect of channel in identification accuracy: We transmit the impaired I/Q samples through different simulated channel instances, and process the received data in the neural network, which is trained on only one type of channel. As shown in Fig. 5(a), the CNN, when trained on the received samples for a specific channel, gives 100% accuracy during testing on data passed through the same channel. This proves that impairment

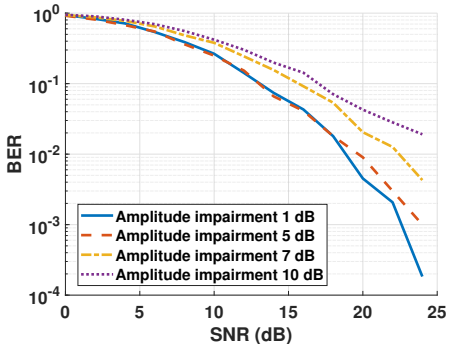
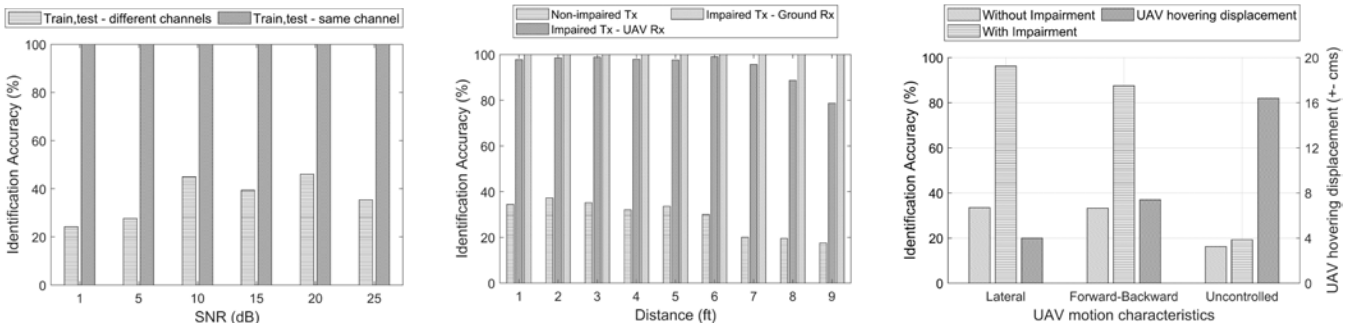


Fig. 4: Effect of different levels of amplitude impairment on BER, at different SNR values.

classes, hence individual SDRs, are detectable in static channel conditions, without affecting the data communication process. However, when this trained CNN is tested with data that has passed through a different channel than what it was trained on, it fails to identify the unique impairments. Thus, we infer that the action of the channel still dominates the artificial modifications to the I/Q samples. To mitigate further the effect of the channel, we perform one additional step of equalization of the received signal through a pre-processing step. On the receiver side, the received waveform is processed in MATLAB using the default WLAN receiver functions performing packet detection, synchronization and carrier frequency offset correction. The data field of the packet is then detected to decode the equalized symbols of the received WLAN data. This equalized data removes the channel effect but preserves the amplitude impairments added by us in the transmitter side, which is then fed to the CNN for training, validating and testing. Next, we perform real world UAV experiments to establish viability of leveraging impairments for identification purposes.

III. Aerial Receivers for UAV Identification

While ground receivers have the advantage of regular connection to a power source, receivers mounted on friendly UAVs allow more flexibility in deployment. We explore this further through a set of experiments, where we study the feasibility of using impairments along with an aerial UAV receiver for this identification purpose. Based on these experimental findings, we propose using multiple UAV receivers for accurate identification results, and show performance evaluation of this



(a) (b) (c)

Fig. 5: (a) Identification accuracy at different SNR levels with CNN trained and tested in same and different channels. The accuracy drops by upto 20% with different channels, (b) A minimum 80% accuracy is achieved by impaired transmissions for ground and UAV receivers, and (c) Identification accuracy with different UAV motion types and UAV hovering displacement.

approach in the next section.

We conduct aerial experiments within a 100 sq. ft. outdoor netted area, under different UAV hovering and motion scenarios, by displacing the UAV laterally, in forward-backward actions within a 10 cm radius, and random motion within a sphere of radius 15 ft. In all of these cases, the SDR in the UAV transmitter continues to transmit impaired I/Q samples. The transmitter UAV is made to hover at a constant height of 10 ft. from the ground, while the receiver UAV is continuously obtaining I/Q samples, processing the data and then feeding I/Q samples to the CNN. The receiver UAV hovers at the same height as that of the transmitter, but at different distances, ranging from 1-9 ft. A second, static ground receiver also collects signals with increasing distances from 1-9 ft from the transmitter.

A. UAV receiver performance in identification

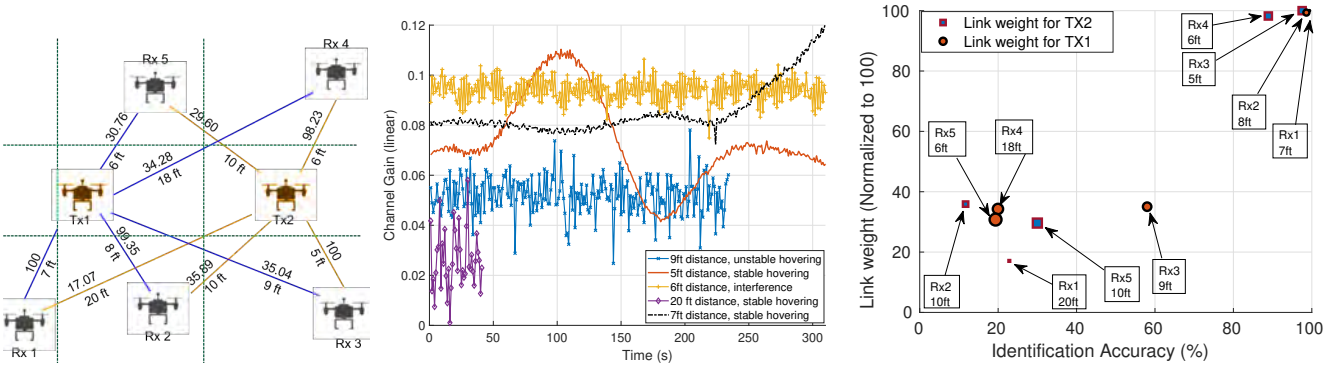
Need for artificial signatures and equalization: Fig. 5(b) shows that with the UAV receiver, the identification accuracy remains ≈100% till a distance of 6ft from the transmitter, and then drops to ≈80% at a distance of 9ft. This decrease in accuracy is due to SNR variation induced by the UAV hovering motion, which adds noise to the transmitted I/Q samples. However, this accuracy is still good enough for correct identification of a UAV. In the case of ground receiver, due to less dynamic channel condition, the identification accuracy remains an impressive 100% till 9ft from the transmitter. In case of UAV transmission without any amplitude impairment, the CNN fails to identify the known uAV. The BER measured in both receivers, despite the introduced I/Q impairment, remains in the range of 10^{-4}, although it worsens to 10^{-3} at longer distances owing to lower SNR. This shows the key benefit of using AirID in UAV identification: the CNN is trained offline on simulated datasets. The impairments are then assigned to different UAVs at test time, and yet these can accurately be identified by the CNN without any further training or transfer learning, which reduces the time for deployment.

Identification accuracy with UAV motion: When a UAV hovers at a particular location, it maintains its position through short correctional movements in the x, y and z plane by moving: (i) sideways along it's forward looking axis (lateral motion), (ii) forward and backward, and (iii) up and down to achieve a stable hovering motion. From Fig. 5(c), it can be observed that for a UAV with a hovering displacement $\Delta(x, y, z)$, beyond the limit of $\pm 10cms$ in either one of the x, y or z coordinates from its intended hovering (x, y, z) position, the channel conditions between the corresponding UAV transmitter and receiver start to degrade, which directly impacts the identification accuracy. We observe that the accuracy is 96% when the UAV is moving laterally, and it goes down to 88% with forward-backward motion. The BER remains below 10^{-4} in all cases. The identification accuracy goes down to 19%, as the UAV hovering displacement goes beyond $\pm 10cms$, which happens when the UAV looses GPS reception and/or in cases of high GPS interference and high wind conditions. Here, the the BER becomes 60%, due to increased variations in channel state information (CSI), which in turn, increases frequency and phase offsets in the received waveform.

Identification accuracy with interference: In a given operating area of receiver UAVs, there are possibilities of unintended interference from nearby ongoing transmissions or intended interference due to jamming from hostile entities. To prove this hypothesis, we make two UAV receivers, Rx5 and Rx3, hover under stable conditions, at distances of 6ft and 5ft from the unknown transmitter UAV. Rx5 suffers from directed jamming of -10 dB at the receiver from a nearby directional antenna on the ground, which drops identification accuracy to 19.34%, as shown in Table I. In comparison, Rx3, which doesn't suffer from any interference or unstable hovering, achieves an identification accuracy of 97.53%

B. UAV identification using Rx groups

We now explore how to utilize multiple UAV receivers and intelligently fuse their individual decisions at the controller, to improve the accuracy. Since CSI accurately quantifies the channel variations experienced by different UAV receivers, we leverage CSI to devise link weight allocation for each transmitter-receiver (abbreviated as 'Tx-Rx') UAV pair. Each receiver records the transmitted signal, equalizes it and forwards it to the controller along with the estimated CSI. The controller relies on CNN to compute a separate identification accuracy from the input provided by each receiver. These are later combined systematically using our designed link weight



(a) (b) (c)

Fig. 6: (a) Bird's eye view of receiver UAV (Rx) swarm and transmitter UAV (Tx) locations, with link weight and distance metrics. (b) Channel gain (linear) values for different UAV Rx in this experiment. The UAV Rx with unstable hovering or interference or large distance from the Tx have high fluctuating channel gains over time, and (c) Relation between link weight and identification accuracy between UAV Tx1, Tx2, and UAV receivers.

based decision fusion algorithm (described next) to decide the optimal identification.

Link weight calculation from CSI: Consider U is a total number of unknown transmitter UAVs, whereas R is a total number of candidate receiver UAVs that are yet to be chosen to identify a particular transmitter. We denote estimated CSI as $CSI_{(r,u)}$ for a link between each UAV receiver $[1 \leq r \leq R]$ and transmitter $[1 \leq u \leq U]$. We evaluate the channel gain of CSI as $G_{(r,u)} = ||CSI_{(r,u)}||_{52}$, where 52 is the total number of OFDM subcarriers used to represent CSI in WiFi 802.11a standard. We keep track of variations in channel gain $G_{(r,u)}(t_i)$ at a discrete time instance t_i over an interval of $[t_1 \leq t_i \leq t_N]$, where N is the total number of channel gain estimates considered for a UAV pair (r, u). Further, we also measure fluctuations in the channel gain over time, given by $\Delta_{(r,u)}(t_i) = G_{(r,u)}(t_{i+1}) - G_{(r,u)}(t_i)$. To quantify these fluctuation statistically, we compute the standard deviation of $\Delta_{(r,u)}$ over an entire time interval, given by $\sigma_{(r,u)} = \sqrt{\frac{\sum_{i=1}^{N}(\Delta_{(r,u)}(t_i)-\mu)^2}{N}}$, where μ is the mean value of channel gain fluctiations. Once we calculate $\sigma_{(r,u)}$ for every UAV pair (r, u), we evaluate their mean to derive the threshold $\Theta_{th} = \frac{\sum_{r=1}^{R}\sum_{u=1}^{U}\sigma_{(r,u)}}{M}$, M is the total number of all possible distinct pairs. For each UAV pair (r, u), we determine a set of discrete time instances $C = \{t_i \mid 1 \leq i \leq N \text{ and } \Delta_{(r,u)}(t_i) > \Theta_{th}\}$. We use the carnality of set C, i.e., $|C|$ to calculate the normalized value of the link weight LW for that (r, u) pair as,

$$LW_{(r,u)} = 100 - \Big[\frac{|C|}{\sum_{i=1}^{N} t_i} * 100\Big] \quad (1)$$

where, $[t_1 \leq t_i \leq t_N]$.

Decision fusion: For a given UAV transmitter, as each corresponding receiver input to the CNN returns a specific identification accuracy, we need to compute a single final outcome. Our decision fusion algorithm employs weighted majority voting with the link weights as input and decides which receiver UAV's prediction carries more weight, which then becomes the finalized result. The decision fusion algorithm in the controller takes in the identification results from each receiver along with the corresponding link weights for that transmitter-receiver pair. Let the number of receiver UAVs be U_{rx}, each giving an identification result, and the number of UAV labels which the CNN has been trained on be U_l. Next, we perform the following steps to run the decision fusion algorithm:

- We create a UAV label matrix L of dimensions $[U_l \times 1]$, an identification result matrix I of dimensions $[1 \times U_{rx}]$ and a link weight matrix LW of dimensions $[1 \times U_{rx}]$
- We formulate a matrix C of the possible options of unknown UAVs by the decisions taken by each allocated receiver, with the dimensions $[U_l \times U_{rx}]$
- Then we compare the identification results from each UAV receiver against a uniform surface of each label, i.e., comparing I with C and generating a comparison matrix Cmp of dimensions $[U_l \times U_{rx}]$
- Next, we perform row wise comparison of Cmp with the link weight matrix LW to find the weighted sum for each UAV label.
- Finally, we take the UAV label with the highest weighted sum, as it is the most accurate identification result among all the matched receiver to a particular unknown transmitter UAV

This identification result with the highest weighted sum is the output of the decision fusion algorithm as the final identification of the UAV transmitter.

IV. Performance Evaluation

A. Experimental Setup

We use 2 UAV transmitters and 5 UAV receivers in an outdoor netted area. Each receiver UAV is placed randomly in one of the virtual grid squares around the transmitter UAVs, as shown in Fig. 6(a). UAV Tx1 and Tx2 are selected to transmit data with predefined amplitude impairments of 4 dB and 2 dB, respectively, thus emulating two unique injected fingerprints for IDs. The receiver UAVs, Rx 1, 2 and 5 are close to Tx1 (distance of 7ft, 8ft and 6ft respectively), Rx 3 and 4 are located close to Tx2 (distance of 5ft and 6ft) and Rx5 is subjected to external interference of -10dB from

TABLE I: Identification accuracy of UAV receivers with BER

Link Pairs		$Distance$	$Accuracy(\%)$	BER
$Tx1$	Rx1	7 ft	99.39	0.00014
	Rx2	8 ft	98.61	0.00043
	Rx5	6 ft	19.34	0.8538

a directional ground transmission. Rx3 is forced to perform unstable hovering by switching off the GPS functionality when gathering data from Tx1. The GPS is turned back on when Rx3 is receiving data from Tx2, ensuring stable hovering.

B. Performance Analysis

We next analyze the channel gain values for each UAV Tx-Rx pair that influence their link weights. A few of those UAV pairs' channel gain values are given in Fig. 6(b). We observe that UAV receivers that are quite close to their paired transmitters with stable hovering and no interference exhibit relatively stable channel gains over time. The situation is opposite for UAVs farther away in distance, prone to interference, or experiencing unstable hovering. These channel gain values influence the link weights for the UAV Tx-Rx pairs, which we see in Fig. 6(c), which showcases the calculated link weights for each Tx-Rx pair and the corresponding identification accuracy per receiver. The controller assigns higher weights to the links between Tx1 and Rx1 and Rx2, and the links between Tx2 and Rx3 and Rx4, because of better CSI estimates due to the proximity of these receivers to the respective transmitters. Note Tx1-Rx5 link gets lower weight as Rx5 suffers from interference. Table I compares the relation between channel conditions and the identification accuracy for UAVs. Since Rx5 is in an interference region, given by the high BER value, its identification accuracy suffers, even though it is situated near the Tx1. Meanwhile, Rx1 and Rx2 return better identification accuracy, since their respective channel conditions are much better, due to stable UAV hovering and absence of interference. From Table II, we see that the decision fusion algorithm chooses the result from Rx1 to correctly identify Tx1.

In case of Tx2, the identification accuracy from Rx5 is very low, since it is affected by external interference and long distance (10 ft). In this scenario, the controller employs weighted majority decision fusion, to filter out the unreliable receiver Rx5, as shown in Table II. Overall, the controller accurately identifies the known UAV transmitters with accuracy of ≈98%, which is at par with the accuracy of ground receivers. Comparing the CSI-based UAV allocation to closest distance based UAV allocation, we see that Rx5 is selected to identify Tx1 (smallest distance of 6ft). Tx2 is paired with Rx3 (smallest distance of 5ft). However, since Rx5 suffers from interference, this leads to very low identification accuracy of 19.34% for Tx1, hence erroneous. Since Rx3 has a stable hovering and no interference, it's accuracy is 97.53%. Here the UAV hovering related channel variations and interference are not taken into account with the distance metric, thus resulting in low accuracy.

V. Conclusion

We devised an over-the-air UAV RF fingerprinting approach using a deep CNN fed by I/Q samples at the PHY layer, for both ground and aerial receivers. We demonstrated the feasibility of a practical UAV identification system through extensive experiments with COTS UAVs. We evaluated the effect of the addition of artificial impairments on the transmitted signal, channel and UAV motion parameters, and designed a decision fusion rule that combines individual identification results from multiple receivers. We show up to 98% accuracy for known UAV identification, while maintaining a stable BER of 10^{-4} on the regular communication link. As part of the future extension of this research, we propose to integrate the mechanism of outlier detection of unauthorized UAVs along with identification of authorized UAVs. This will make AirID more robust to RF impairment spoofing by rogue UAVs.

TABLE II: Decision fusion of identification accuracy results

Link Pairs		$Weight$	Identification Accuracy (%) $AirID$	$ClosestDistance$
$Tx1$	Rx1	100.00	99.39	✗
	Rx2	99.36	98.61	✗
	Rx5	30.77	19.34	19.34
$Tx2$	Rx3	100.00	97.53	97.53
	Rx4	98.23	88.91	✗

VI. Acknowledgment

This work is supported by the US National Science Foundation Award CNS-1923789.

References

[1] FAA, "FAA Aerospace Forecast, Fiscal years 2019-2039," [accessed 13-May-2020]. [Online]. Available: https://www.faa.gov/data_research/aviation/aerospace_forecasts/media/FY2019-39_FAA_Aerospace_Forecast.pdf

[2] S. Puliti, L. T. Ene, T. Gobakken, and E. Næsset, "Use of partial-coverage uav data in sampling for large scale forest inventories," *Remote Sensing of Environment*, vol. 194, pp. 115–126, 2017.

[3] N. H. Motlagh, M. Bagaa, and T. Taleb, "Uav-based iot platform: A crowd surveillance use case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.

[4] CBC, "Drone collides with commercial aeroplane in Canada," [accessed 2-Oct-2019]. [Online]. Available: https://www.bbc.com/news/technology-41635518

[5] HGH Infrared, "Drone/UAV Detection and Tracking," [accessed 21-Sep-2019]. [Online]. Available: https://www.hgh-infrared.com/Applications/Security/Drone-UAV-Detection-and-Tracking

[6] J. Zhao, X. Fu, Z. Yang, and F. Xu, "Radar-Assisted UAV Detection and Identification Based on 5G in the Internet of Things," *Wireless Communications and Mobile Computing*, 2019.

[7] C. Ruiz, X. Chen, and P. Zhang, "Poster abstract: Hybrid and adaptive drone identification through motion actuation and vision feature matching," in *IPSN*, 2017.

[8] S. Kianoush, A. Vizziello, and P. Gamba, "Energy-efficient and mobile-aided cooperative localization in cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 3450–3461, 2015.

[9] B. Mokhtar and M. Azab, "Survey on security issues in vehicular ad hoc networks," *Alexandria Engineering Journal*, 2015.

[10] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "ORACLE: Optimized Radio clAssification through Convolutional neuraL nEtworks," in *IEEE INFOCOM*, 2019.

[11] S. U. Rehman, K. W. Sowerby, S. Alam, and I. Ardekani, "Portability of an RF fingerprint of a wireless transmitter," in *IEEE CNS*, 2014.

[12] Genesys, "Genesys Lab ML datasets," [accessed 01-Sep-2020]. [Online]. Available: http://genesys-lab.org/mldatasets

[13] N. Soltani, K. Sankhe, S. Ioannidis, D. Jaisinghani, and K. Chowdhury, "Spectrum awareness at the edge: Modulation classification using smartphones," in *2019 IEEE DySPAN*, 2019, pp. 1–10.

[14] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE JSTSP*, 2018.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.