Machine Learning-Based Delay-Aware UAV Detection over Encrypted Wi-Fi Traffic

Amir Alipour-Fanid*, Monireh Dabaghchian*, Ning Wang*, Pu Wang*, Liang Zhao†, Kai Zeng*
*Electrical and Computer Engineering Department,
† Information Science and Technology Department
George Mason University, Fairfax, VA 22030
Email: {aalipour, mdabaghc, nwang5, pwang20, lzhao9, kzeng2}@gmu.edu

***Abstract*—In this paper, we propose a machine learning-based framework for fast UAV (unmanned aerial vehicle) detection and identification over encrypted Wi-Fi traffic. It is motivated by the observation that many consumer UAVs use Wi-Fi links for control and video streaming. The proposed framework extracts statistical features derived only from packet size and inter-arrival time of encrypted Wi-Fi traffic, and can quickly identify UAV types. In order to reduce the online identification time, our framework adopts a re-weighted ℓ_1-norm regularization, which considers the number of samples and computation cost of different features. This framework jointly optimizes feature selection and prediction performance in a unified objective function. To tackle the packet inter-arrival time uncertainty when optimizing the trade-off between the identification accuracy and delay, we utilize maximum likelihood estimation (MLE) method to estimate the packet inter-arrival time. We collect a large number of real-world Wi-Fi data traffic of four types of consumer UAVs and conduct extensive evaluation on the performance of our proposed method. Evaluation results show that our proposed method can identify the tested UAVs within 0.15-0.35s with high accuracy of 85.7-95.2%.**

I. Introduction

In the past few years, we have seen a significant growth of the consumer **u**nmanned **a**erial **v**ehicle (UAV) market for personal recreation. Despite its huge potential in spurring economic growth, the significant increase of consumer UAVs raises lots of issues regarding public security, and personal privacy [1]. It was reported that an Army chopper was struck by an illegally flying drone over a residential neighborhood in September 2017 [2]. In April 2016, a UAV was peeping outside a teenager's bedroom window in Massachusetts [3]. In January 2015, a small UAV crashed on the White House lawn bringing the worry about security measures [4]. Hence, there is an urgent need to quickly and efficiently identify an intruder UAV in a restricted area, or assist the forensics investigation to identify its appearance and type.

Although many traditional physical detection mechanisms, such as radar [5], [6], acoustic [7], [8], and vision [9], [10] have been proposed for UAV detection, they cannot always work in practical scenarios, especially in a crowded urban environment. The identification accuracy of radar could be influenced by other small flying objects, such as birds. The radar signals can also be easily blocked by walls, buildings, and other obstacles, which are very common in a civilian environment. The vision detection technique cannot detect the UAV in non-line-of-sight scenarios, which are very common in urban areas. The acoustic detection can be interfered by environment noises.

Nowadays, we observe that many existing consumer UAVs are equipped with Wi-Fi interfaces and communicate with a user handheld device (e.g., smartphone) for command control and video streaming. Based on this observation and motivated by network traffic identification technologies [11], [12], [13], we propose a machine learning-based framework for fast UAV identification over encrypted Wi-Fi traffic.

Identifying UAVs through wireless traffic can well complement the traditional physical detection mechanisms. *First*, Wi-Fi signal sensing and packet capturing are not affected by obstacles, other flying objects, acoustic noise, or light conditions that could affect traditional physical detection mechanisms. *Second*, information about UAV's type can be inferred from Wi-Fi data traffic, which can be very useful for forensics investigation.

Challenges: At the same time, UAV identification through Wi-Fi traffic introduces unique challenges that separate it from traditional network traffic identification and sensing tasks as follows: **1) UAV traffic can be encrypted. Therefore, existing network monitoring and intrusion detection mechanisms that are based on packet header examination or port filtering are not applicable to encrypted UAV traffic.** For example, Wi-Fi controlled UAVs (such as DJI and Bebop drones) use WPA2 to secure the wireless communication. Although SSID in the MAC frame may reveal information about the type or vendor of the drone, it can be easily changed through drone control Apps. **2) Existing machine learning methods cannot be directly used to identify UAV traffic in a timely manner.** For realtime applications, we need to identify the UAV as soon as it is appearing in or approaching to a restricted area. From learning and classification perspective, traditional machine learning methods [11], [12] that only aim at minimizing detection error cannot be directly applied. Identification delay introduced by the online computations on feature generation and future packet arrival time should also be considered. **3) Traditional time series early detection strategies [14] cannot be applied to UAV traffic.** The inter-packet arrival time of UAV traffic is random, so the traditional time series early detection method which is based on fixed time intervals cannot be directly applied.

Approaches: To address the above challenges, we propose

a delay-aware machine learning-based UAV identification framework to strike a tunable balance between UAV identification accuracy and delay. Our classification framework treats the encrypted data flow as a time series and extracts statistical features only based on the packet size and inter-arrival time. By considering the computation time among different features, our framework adopts a re-weighted ℓ_1-norm regularization and integrates feature selection and performance optimization in one objective function. To tackle the packet inter-arrival time uncertainty when estimating the delay cost function, we use maximum likelihood estimation (MLE) method to estimate the packet inter-arrival time. Finally, expected total cost function integrates misclassification and delay cost which are updated online when a new packet arrives and an optimal detection decision is made to minimize the expected total cost function.

Our main contributions are summarized as follows:

- We propose a machine learning-based framework to achieve delay-aware UAV detection and identification over encrypted Wi-Fi traffic. This framework extracts features derived only from information of packet size and inter-arrival time. The proposed method can well complement existing physical UAV detection methods (such as radar, vision, and sound) in practical scenarios.
- In order to reduce the model prediction time for fast UAV identification, our framework adopts ℓ_1-norm regularization and integrates feature selection and accuracy optimization in one objective function, which considers the feature importance and difference of computation time among different features.
- We collect a large amount of real-world encrypted Wi-Fi data traffic of non-UAV and four types of consumer UAVs, and conduct extensive evaluations on the performance of the proposed methods.

II. Related Work

A. UAV Detection and Identification

Existing UAV detection mechanisms mainly focus on physical sensing through various means, including radar, vision, and acoustic. In order to adapt the detection of small size UAVs, X-band radar systems were proposed [15], [16]. However, in metropolitan areas (e.g., a city) with lots of buildings and obstacles, radar based detection may become ineffective due to its line-of-sight requirement. The vision-based UAV detection based on video cameras [9], [17], [18], [10] has the same weakness as radar based techniques, as it also requires line-of-sight connection between the camera and UAV. The acoustic signal-based UAV detection is a method that can solve the non-line-of-sight problem [7], [8].

A recent work [19] has been proposed to detect the approaching of a UAV within a short distance through the observation of received signal strength (RSS) changes of Wi-Fi signals. However, an intruder UAV may just launch inside a restricted area and hover, or standby on a neighboring roof to spy on someone. In these scenarios, this method will not work. Moreover, a changing RSS is not necessarily introduced by UAVs, but could be other moving objects with Wi-Fi interfaces, such as mobile users carrying smartphones or a driving car with Wi-Fi connections.

In another work [20], the authors propose a new RF-based drone detection method based on the physical characteristics of the drone, such as body vibration and body shifting, which impact the wireless signal transmitted by drones during the communication. This method is not useful when the UAV is in the standby mode. Both [19] and [20] require line-of-sight connection between the RF signal monitor and UAV. Our proposed method based on Wi-Fi traffic identification relaxes this strong assumption.

B. Data Traffic Classification/Identification

Classical approaches such as *port-based*, *payload-based* and *deep packet inspection* can be used to identify the type of the non-encrypted network data traffic. There are several works for identifying the encrypted data flow based on protocol data fingerprinting in wired and wireless networks, where commonly a combination of statistical and machine leaning approaches have been used [11], [21], [12], [13].

In [11], a new SVM based method is proposed to identify three types of traffic, HTTP, FTP, and Email. One of the pioneering works in this area applies classification techniques to classify traffic in a wired network into classes of bulk transfer, small transactions, and multiple transactions [12]. Bernaille et al. [13] show that it is possible to distinguish the behavior of an application from the observation of the size and direction of the first few packets of the TCP connection. However, this method requires 1) packet header traces analysis and 2) initial TCP connection packets.

Our work is different from the existing traffic identification works in the following aspects: 1) Our model provides packet-by-packet analysis, hence the decision is made in a timely manner as packets enter the detection system. 2) Our model adaptively finds the optimal number of the packets that are needed for optimal identification with high accuracy, while considering time cost (or delay). 3) When training the model, feature generation time is also considered and critical features are selected. Therefore, in the prediction, useless features are not generated, and thus the prediction runtime is reduced.

III. Problem Formulation

A. System Setup

There is a Wi-Fi signal sensing and packet capturing system that can collect all the Wi-Fi traffic within a physical sensing range in realtime. There can be multiple Wi-Fi users in the sensing range and multiple UAV or non-UAV devices. The sensing system may capture non-encrypted packets, for which we assume the system can tell the application types of the corresponding flow by examining the packet headers or contents. Since these non-encrypted packets can be easily identified by existing methods, we will only focus on encrypted Wi-Fi traffic in this paper. For an encrypted Wi-Fi frame, the information we can obtain about the frame is its source and destination MAC addresses, transmitter and receiver MAC addresses, packet size, and packet arrival time together with other MAC header information, such as frame

type (control, management, or data), sequence number, and duration/connection ID.

For the UAV identification, we only use data frames. We divide the encrypted Wi-Fi traffic into individual flows according to the pair of source and destination MAC addresses. A unique flow includes the packets between a pair of nodes. The traffic in a flow can be bi-directional or unidirectional. In a realtime scenario, these flows usually interleave with each other in time. The goal of this paper is to identify the UAV data flows when frames are captured and decide the UAV type in a quick manner with high accuracy.

B. Delay-aware UAV Identification Problem Formulation

Let's assume that we can obtain a large training dataset with m flow traces with each trace having n consecutive packets. The traces contain UAV and non-UAV flows which are labeled with their corresponding flow types $y_i \in \mathcal{Y}$, where $\mathcal{Y} = \{\text{UAV}_1, \text{UAV}_2, ..., \text{UAV}_{\upsilon-1}, \text{non-UAV}\}$ and $\upsilon = |\mathcal{Y}|$ denotes the number of class types in set $\mathcal{Y}$. UAV_j for $j \in \{1, ..., \upsilon - 1\}$ denotes the UAV type j.

Packet size and *packet inter-arrival time* are two key attributes we extract from these traces for UAV detection. The sequences of packet size and packet inter-arrival time for the ith trace are denoted by $\mathbf{x}_i$ and $\boldsymbol{\tau}_i$, respectively. Let $\mathbf{x}_i = (x_{i,1}, \cdots, x_{i,n})$ where $x_{i,j}$ for $i = 1, ..., m$ and $j = 1, ..., n$ indicates the size of the jth packet in the ith trace. Similarly, let $\boldsymbol{\tau}_i = (\tau_{i,1}, \cdots, \tau_{i,n})$ where $\tau_{i,j}$ denotes the inter-arrival time between the jth and $(j+1)$th packets ($j < n$) in the ith trace. Define a finite set $S = \{((\mathbf{x}_i, \boldsymbol{\tau}_i), y_i)\}_{i \in \{1,...,m\}}$ where the pair $(\mathbf{x}_i, \boldsymbol{\tau}_i)$ represents packets' sizes and their inter-arrival times of the ith trace in set S, respectively.

Let $\tilde{\mathbf{x}}(t_k)$ denotes the received incoming traffic up to its kth packet arrived at time t_k. Assume a set of multi-class classifiers $\mathcal{H} = \left\{h_\gamma^j\right\}_{j \in \{1,...,n\}}$ are trained to classify an incoming traffic flow $\tilde{\mathbf{x}}(t_k)$ where kth packet arrives at time t_k and $\gamma \in \mathcal{Y}$. Test misclassification cost function is defined as $C_1(\hat{y}, \tilde{y}) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, where $(\hat{y}, \tilde{y}) \in \mathcal{Y}$, and $\hat{y} = h_\gamma^j(\tilde{\mathbf{x}}(t_j))$ is the predicted class label, while the true class label of the incoming flow is $\tilde{y}$. Let $C_2(t_p) \in \mathbb{R}$, $p > k$, be the time cost function which indicates the time cost value if UAV detection is postponed up to time instant t_p. Thus, the estimated total cost function is given by

$$J(\tilde{\mathbf{x}}(t_k)) = C_1(h_\gamma^k(\tilde{\mathbf{x}}(t_k), \tilde{y})) + C_2(t_k). \tag{1}$$

Hence, we formulate the delay-aware UAV identification optimization problem as follows:

$$p^* = \underset{p \in \{k,...,n\}}{\arg\min} \quad J(\tilde{\mathbf{x}}(t_p)), \tag{2}$$

where p^* indicates the optimal number of the packets that need to be received from the incoming flow before performing the UAV detection decision, and t_{p^*} denotes the p^*th packet arrival time. When $p^* = k$ is the solution of the optimization problem in Eq. (2), there is no need to collect more packets because $J(\tilde{\mathbf{x}}(t_k)) < J(\tilde{\mathbf{x}}(t_q))$ for $q = k+1, ..., n$. Therefore, the detection is performed instantly at time $t_{p^*} = t_k$. However, when $p^* \neq k$ is the solution of Eq. (2), it means that the total cost is minimized for $k+1 \leq p^* \leq n$, thus the detection process is deferred to collect more packets from incoming flow. For the notation purposes, let the indicator function $\mathbb{I}(p^*) = 1$ when $p^* = k$; and otherwise $\mathbb{I}(p^*) = 0$, where $p^* = k$ corresponds to the UAV detection at time t_{p^*}.

Algorithm 1 Training phase framework for UAV identification

Input: Wi-Fi traffic trace dataset $\{(\mathbf{x}_i(t_n), y_i)\}$, $(y_i, \gamma) \in \mathcal{Y}$, $\upsilon = |\mathcal{Y}|$, $i \in \{1, ..., m\}$, $j \in \{1, ..., n\}$;
Output: $E^j(X_i^j)$, W^j, set of classifiers $\mathcal{H} = \left\{h_\gamma^j\right\}$;

Step 1: Extract packet size and inter-arrival time of encrypted Wi-Fi traffic traces and create dataset $S = \{((\mathbf{x}_i, \boldsymbol{\tau}_i), y_i))\}$;

Step 2: Define subsets $S^j = \left\{((\mathbf{x}_i^j, \boldsymbol{\tau}_i^j), y_i))\right\}$, where $\mathbf{x}_i^j = (x_{i,1}, x_{i,2}..., x_{i,j})$ and $\boldsymbol{\tau}_i^j = (\tau_{i,1}, \tau_{i,2}, ..., \tau_{i,j})$;

Step 3: Determine design matrices $\mathbf{X}^j = [X_1^j, X_2^j, ..., X_m^j]^T$, where $X_i^j = \left[V_1(\mathbf{x}_i^j), ..., V_l(\mathbf{x}_i^j), V_1(\boldsymbol{\tau}_i^j), ..., V_l(\boldsymbol{\tau}_i^j)\right] \in \mathbb{R}^{2l}$;

Step 4: Train a set of classifiers $\mathcal{H} = \left\{h_\gamma^j\right\}$ by solving Eq. (3): $\min_{W^j} \ \mathcal{L}(y, h_\gamma^j(\mathbf{X}^j, W^j) + \sum_{i=1}^{2l} \lambda_i^j |W_i^j|$;

Step 5: Compute expected training misclassification function:
for $i \in \{1 : m\}$
  for $j \in \{1 : n\}$
    for $\gamma \in \{\text{UAV}_1, \text{UAV}_2, ..., \text{UAV}_{\upsilon-1}, \text{non-UAV}\}$
      Compute $P_j(\hat{y} = \gamma | X_i^j; W^j) = h_\gamma^j(X_i^j, W^j)$
      Compute $E^j(X_i^j)$ using Eq. (4).

IV. Delay-aware UAV Identification Framework

A. Learning-Based Model Design

Based on the definition of traffic flow dataset S in Section III-B, we further define the data of *"incomplete traffic flow"*, where only the first $j < n$ packets are available in the dataset, denoted as $S^j = \{((\mathbf{x}_i^j, \boldsymbol{\tau}_i^j), y_i))\}_{i \in \{1,...,m\}, j \in \{1,...,n\}}$ where $\mathbf{x}_i^j = (x_{i,1}, x_{i,2}..., x_{i,j})$ and $\boldsymbol{\tau}_i^j = (\tau_{i,1}, \tau_{i,2}, ..., \tau_{i,j})$ indicate the sequence of packet size and inter-arrival time of the ith trace in the jth subset, respectively. Next, for each dataset S^j, we create a design matrix $\mathbf{X}^j = [X_1^j, X_2^j, ..., X_m^j]^T \in \mathbb{R}^{m \times 2l}$, where $X_i^j \in \mathbb{R}^{2l}$ is a row vector as

$$X_i^j = \left[V_1(\mathbf{x}_i^j), ..., V_l(\mathbf{x}_i^j), V_1(\boldsymbol{\tau}_i^j), ..., V_l(\boldsymbol{\tau}_i^j)\right]$$

and $V_1(\cdot), ..., V_l(\cdot) \in \mathbb{R}$ are functions which compute the statistical features of the input samples (i.e., $\mathbf{x}_i^j$, $\boldsymbol{\tau}_i^j$). l denotes the number of features. A list of statistical feature functions with their corresponding computation formula is shown in Appendix IX-A.

Feature selection based on re-weighted ℓ_1-norm by considering both feature discriminative power and computation time cost: Different features have different significance; we use $W^j = \{W_1^j, W_2^j, \cdots, W_{2l}^j\} \in \mathbb{R}^{2l}$ to denote the weight vector of jth subset for all the $2l$ features. Therefore, $W_i^j = 0$ means the ith feature of jth subset is not useful and can be discarded. Then, given the jth design matrix $\mathbf{X}^j$ and $y \in \mathcal{Y}$, our problem is to learn a predictive mapping $h_\gamma^j(X_i^j, W^j) \to y_i$ for $i = 1, 2, \cdots, m$ such that: 1) the disagreement between $h_\gamma^j(X_i^j, W^j)$ and y_i is minimized; and 2) the number of non-zeros in W^j is minimized.

Other than the fact that different features have different discriminative power for identification, we also observe that

Table I: Empirical and exponential CDF Goodness-of-fit statistics

	UAV type			
	Bebop 1	Bebop 2	Spark	UDI
Kolmogorov-Smirnov (KS)	0.0612	0.0508	0.0773	0.0811
Cramer-von Mis (CvM)	0.0720	0.0633	0.0691	0.0794

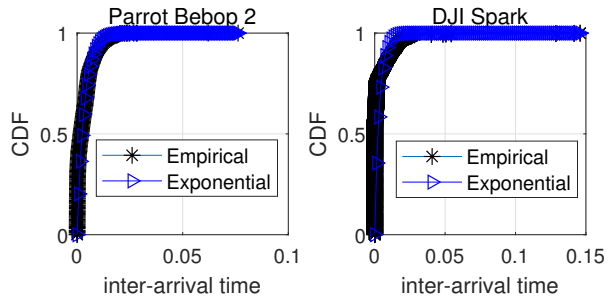


Figure 1: Cumulative distribution function (CDF) plot to compare the fit of exponential distribution to the empirical CDF of packet inter-arrival time

.

different features consume different amounts of computation time (refer to Table III in Appendix IX-A). Therefore, in the jth subset among the features that bring the same discriminative power in UAV identification, we tend to remove the one(s) that consume more time. This requires us to give personalized penalty on each different feature. The more time one consumes, the more penalty it is given. Thus, instead of using conventional ℓ_1-norm regularization [22] that penalizes all the features evenly, we propose the following new objective loss function with the re-weighted ℓ_1-norm:

$$\min_{W^j} \mathcal{L}(y, h_\gamma^j(\mathbf{X}^j, W^j) + \sum_{i=1}^{2l} \lambda_i^j |W_i^j|, \tag{3}$$

where the strength of penalty λ_i^j for ith feature in jth subset is proportional to the computational time for this feature. Therefore, the objective function in Eq. (3) will minimize misclassification error and enforce some W_i^j's to be zeros, especially those that consume more computation time. Since for computing of expected misclassification cost function, we shall need probabilistic output of the classifier, then we choose $h_\gamma^j(\cdot)$ to be one-vs-all logistic regression function [23]. One-versus-all logistic regression is a generalized version of the logistic regression into multi-class classification.

Next, we compute the training expected misclassification cost function on each trace i for every subset j as follows:

$$E^j(X_i^j) = \sum_{y_i \in \mathcal{Y}} P(y_i|X_i^j) \sum_{\hat{y} \in \mathcal{Y}} P_j(\hat{y}|X_i^j; W^j) C^j(\hat{y}|y_i), \tag{4}$$

where $P(y_i|X_i^j) = 1$ if $\hat{y} = y_i$ and 0 otherwise. A set of one-vs-all logistic regression classifiers $\mathcal{H} = \{h_\gamma^j\}$ for $j = 1, ..., n$ and $\gamma \in \{\text{UAV}_1, \text{UAV}_2, ..., \text{UAV}_{1-\upsilon}, \text{non-UAV}\}$ is trained in the training phase. Based on the probabilistic output of one-vs-all logistic function, we can compute $P_j(\hat{y} = \gamma|X_i^j; W^j) = h_\gamma^j(X_i^j, W^j)$. $C^j(\hat{y}|y_i)$ denotes the misclassification cost function of training dataset. $C^j(\hat{y}|y_i)$ = 1 if $\hat{y} = y_i$ and 0 otherwise, and $\hat{y} = \max_\gamma h_\gamma^j(X_i^j, W^j)$. We

Algorithm 2 Delay-aware UAV identification

Input: Incoming traffic flow $\tilde{\mathbf{x}}(t_k)$, $E^j(X_i^j)$, W^j, $\mathcal{H} = \{h_\gamma^j\}$, $i \in \{1, ..., m\}$, $j \in \{1, ..., n\}$, $\gamma \in \mathcal{Y}$, $(\mu_1, ..., \mu_\upsilon)$;

Output: t_{p*}, p^*, $\hat{y} = \max_\gamma h_\gamma^k(\tilde{X}_i^k, W^k)$;

Step 1: Extract packet sizes $\tilde{\mathbf{x}}^k$ and inter-arrival times $\tilde{\boldsymbol{\tau}}^k$ of $\tilde{\mathbf{x}}(t_k)$, then compute statistical feature values $\tilde{X}^k$;

Step 2: Compute Δ_i^k and weight function $f_{w_i}^k$;

Step 3: Compute C_1 using Eq. (5);

Step 4: Identify the trace label which has minimum distance with $\tilde{X}^k$. Pick the corresponding class inter-arrival time from $(\hat{\mu}_1, ..., \hat{\mu}_\upsilon)$, then compute C_2 using Eq. (6);

Step 5: Calculate total cost function J using Eq. (1);

Step 6: Compute Eq. (2), if $p^* = k$ then, $\mathbb{I}(p^*) \leftarrow 1$, perform UAV detection and break; otherwise $k \longleftarrow k + 1$ and go to **Step 1**;

(a) Parrot Bebop 1 (b) Parrot Bebop 2 (c) DJI Spark (d) DBPower UDI

Figure 2: UAV types used in the experiment

summarize the model training phase for UAV identification in **Algorithm 1**.

B. Delay-aware Predictive Model

1) ***Expected missclassification cost function*** C_1*:* In the prediction phase of the incoming flow $\tilde{\mathbf{x}}(t_k)$, in order to compute the expected misclassification cost function C_1, $E^j(X_i^j)$ is weighted based on the incoming traffic's Euclidean distance from every trace in the training dataset. Consider $\tilde{\mathbf{x}}(t_k)$ be the incoming flow and $\tilde{X}^k \in \mathbb{R}^{2l}$ its corresponding feature values. The weight function is defined as a normalized sigmoid function by $f_{w_i}^k = s_i^k / \sum_i^m s_i^k$ where $s_i^k = 1/1 + exp^{-\eta \Delta_i^k}$, and η is some positive constant, and $\Delta_i^k = \bar{D}_i - d_i^k / \bar{D}_i$ is the normalized average distances between $\tilde{X}^k$ and all the traces in the training dataset [14]. $d_i^k = ||\tilde{X}^k - X_i^k||_2$ indicates the Euclidean distance of the incoming flow from ith trace in the dataset. In fact, the weight function $f_{w_i}^k$ plays the role of a *similarity function* which measures how close the incoming traffic flow is to each of the traces in the training dataset. Hence, the expected misclassification cost function for $\tilde{\mathbf{x}}(t_k)$ is defined as follows:

$$C_1(h_\gamma^k(\tilde{\mathbf{x}}(t_k), \tilde{y})) = \sum_{i=1}^{m} f_{w_i}^k E^k(X_i^k). \tag{5}$$

2) ***Estimated time cost function*** C_2*:* For the incoming flow $\tilde{\mathbf{x}}(t_k)$, future packet arrival times are unknown and random. This uncertainty in packet arrival times introduces difficulties in constructing a delay-aware UAV identification algorithm. In order to tackle to this challenge, we propose to estimate the incoming flow's future packet inter-arrival time according to the exponential distribution with parameter $\hat{\mu}_i$ for $i = 1, ..., \upsilon$ achieved by MLE [24] method (refer to Appendix IX-B). In Table I we present the goodness-of-fit statistics to show that exponential distribution provides a good approximation for packet inter-arrival time estimation. We also graphically illustrate the goodness-of-fit for Bebop

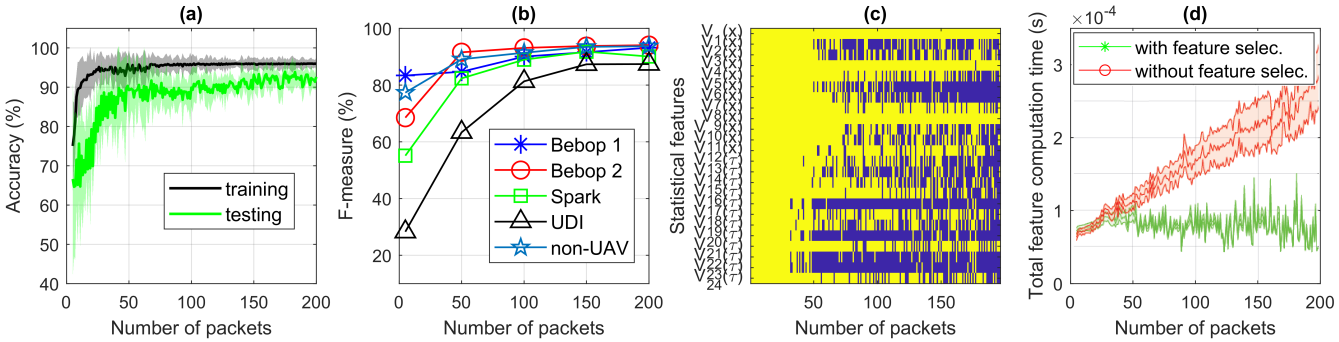


Figure 3: (a) training and testing accuracy on each subset j, (b) F-measure metric, (c) Selected (yellow) and discarded (blue) features where the total number of statistical features is $2l = 24$. (d) Total feature computation time.

2 and DJI Spark in Fig. 1. Due to the space limitation the other UAVs' figures are omitted.

Next, we estimate the packet inter-arrival time of the incoming traffic flow $\tilde{\mathbf{x}}(t_k)$, through the following steps: *First*, the Euclidean distance between $\tilde{\mathbf{x}}(t_k)$ and each trace in the training set is computed. *Second*, the class label of the trace which has a minimum distance from the incoming flow is identified. *Third*, the average inter-arrival time of the identified class is selected from $\hat{\mu}_i$ to estimate the packet inter-arrival time of $\tilde{\mathbf{x}}(t_k)$ using exponential distribution.

Now, let $\tilde{\tau}_{i+1} = t_{i+1} - t_i$ for $i = k, ..., n$ be the packet inter-arrival time of the $\tilde{\mathbf{x}}(t_k)$ estimated by the above steps. Then, the estimated time cost function is obtained as

$$C_2(t_p) = \sum_{i=k}^{p} \tilde{\tau}_{i+1} \quad \text{for} \quad p = k, ..., n \tag{6}$$

where C_2 is a strictly increasing function.

*3) **Estimated expected total cost function** J:* According to Eq. (1), the total cost function J is defined based on C_1 and C_2 which can be computed using Eqs. (5) and (6), respectively. **Algorithm 2** summarizes the total cost function estimation and incoming traffic flow's identification phase.

V. UAV and non-UAV Dataset

We collect real-world traffic flows of four types of consumer UAVs shown in Fig. 2: Parrot Bebop 1 Quadcopter Drone, Parrot Bebop 2 Quadcopter Drone, DJI Spark, DBPower UDI U842 Predator FPV. We use a DELL Latitude laptop embedded with a wireless network interface card (NIC), Intel Corporation Wireless 8260, operating in promiscuous mode to monitor and collect the Wi-Fi network traffic. For each UAV type, we collect the data traffic while the UAV is flying and streaming video to the controller.

In an effort to make a diverse non-UAV dataset, we create a dataset which consists of two main sub-dataset: *First*, we use the Wi-Fi data traffic available online from CRAWDAD database [25]. We choose this dataset because this dataset consists of live and non-live video streaming traffic captured from commonly seen popular applications such as Google Hangouts, ooVoo, Skype, TED and Youtube. *Second*, in order to simulate a realworld UAV identification scenario, we have also captured encrypted Wi-Fi traffic on a university campus Wi-Fi network where a mixed multiple traffic types such as video streaming, social network apps, VoIP, email, web browsing applications are usually running. If the UAV detection system is setup on the campus, our method should be able to differentiate UAV traffic from the non-UAV traffic. The non-UAV and each UAV type dataset contains 3,000 traffic traces with each trace having $n = 200$ consecutive packets.

VI. Performance Evaluation

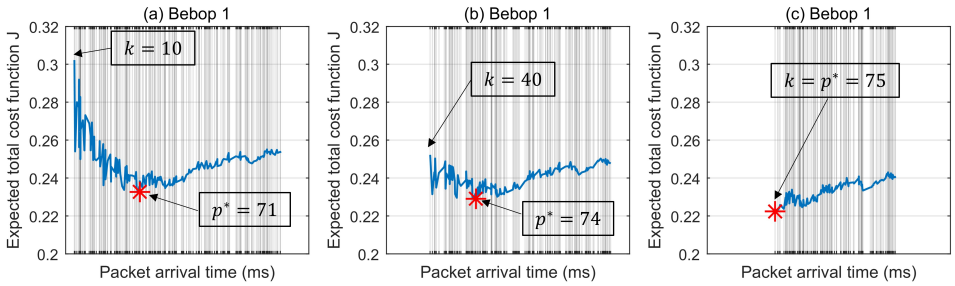
A. Learning-based Model Performance Evaluation

By randomly sampling the dataset, we split the whole dataset into training and testing datasets with the ratio of 70% and 30%, respectively. We create subset S^j for $j = 5, ..., 200$ by adding packet-by-packet information to each subset j according to the step 2 in **Algorithm 1**. Then, we form the design matrix $\mathbf{X}^j$ for $j = 5, ..., 200$ by extracting $2l = 24$ statistical feature values listed in Table III. One-vs-all logistic regression multi-class classification algorithm with re-weighted ℓ_1-norm technique proposed in Eq. (3) is run over each design matrix $\mathbf{X}^j$.

Figure 3(a) illustrates the accuracy of the classification algorithm in training and testing on each design matrix $\mathbf{X}^j$. The shaded areas denote the regions surrounded by one standard deviation above and below the mean accuracies. The results show that a mean testing accuracy of higher than 90% is achieved when the information of fifty or more packets ($j > 50$) are available in the subset. F-measure (weighted harmonic mean of precision and recall) for different UAV types and non-UAV is shown in Fig. 3(b). Average F-measure of higher than 88% is achieved on the test data which indicates an acceptable discriminative power of the trained classifiers.

B. Feature Selection and Computation Time

Our objective function in Eq. (3) jointly minimizes the missclassification error and runtime by discarding useless features. Figure 3(c) shows the set of selected (yellow) and discarded (blue) features for each model trained on jth subset for $j = 5, ..., 200$. As it is shown when the sample size is small (e.g., $n < 30$), all of the features are selected by the model. This is because, on the one hand, small sample size does not provide enough information for the classifier to distinguish different classes with high accuracy, and on



Bebop 1	(k, t_k)	(p^*, t_{p^*})	Pr
(a)	(10,20.56)	(71,139.47)	0.4254
(b)	(40,84.75)	(74,142.94)	0.6891
(c)	(75,146.42)	(75,146.42)	0.9015

Figure 4: Delay-aware UAV identification using Algorithm 2

the other hand, it consumes less amount of time to compute the feature values. However, as the sample size increases, misclassification error is reduced and the feature computation time increases which results in the smaller set of selected features by the algorithm.

In order to evaluate the impact of feature selection method on the prediction time, we select 1,000 traces uniformly at random from the UAV dataset and consider them as incoming flows (i.e., $\tilde{\mathbf{x}}(t_k)$). Then, we compute the feature generation time of the flows for $k = 5, ..., 200$. Figure 3(d) illustrates the mean total feature generation time of the flows in the trace with and without feature selection methods. The results show that as the number of the packets increases, with feature selection, the mean total feature generation time oscillates in a non-increasing trend depending on the number of selected features. However, without feature selection, the total computation time increases when the number of packets increases. Therefore, the proposed feature selection method reduces the prediction runtime.

C. Delay-aware UAV Identification

We consider four types of incoming UAV traffic flows each belonging to a specific UAV type (i.e., Bebop 1, Bebop 2, Spark, UDI), and run the delay-aware UAV detection algorithm on them. Due to the space limitation, we only show the test results for Bebop 1 (in three steps a, b, c) and summarize the outcome in the far right table in Fig. 4. We show the total cost function optimization demo for the four tested UAVs' traffic flows in a demo that can be accessed from [26]. In Fig. 4(a), $k = 10$th packet arrives at time $t_k = 20.56ms$ and based on the received traffic flow till then, total cost function J is estimated. In this case, it is estimated that the minimum total cost function will occur when $p^* = 71$th packet arrives at $t_{p^*} = 139.41ms$. Therefore, the decision for the flow detection is deferred. The far right column of the table indicates that if the UAV identification is performed in $k = 10$, then the identification accuracy will be 42.54% ($Pr = 0.4254$). In Fig. 4(b), $k = 40$th packet arrives at time $t_k = 84.75ms$. In this case, the algorithm estimates that the minimum expected total cost function will occur when $p^* = 74$th packet arrives at $t_{p^*} = 142.94ms$. If the identification is performed in $k = 40$, then with the accuracy of 68.91% ($Pr = 0.6891$) the flow will be detected as a Bebop 1 traffic flow. This process is continued until the arrival of the kth packet for which $k = p^*$. According to Fig. 4(c), this condition is satisfied when $k = 75$th packet arrives at $t_k = 146.42ms$ for which $k = p^* = 75$. In this case, UAV detection is performed and the detection accuracy is 90.15% ($Pr = 0.9015$).

Table II: Tested UAVs identification results

Traffic	$E[p^*]$	$E[t_{p^*}](ms)$	$E[P_r]$ (%)
Bebop 1	87 ($\pm$8)	160.43 ($\pm$10.01)	87.84 ($\pm$1.20)
Bebop 2	95 ($\pm$13)	151.91 ($\pm$18.82)	90.75 ($\pm$1.74)
Spark	93 ($\pm$11)	142.80 ($\pm$15.57)	95.23 ($\pm$0.69)
UDI	141 ($\pm$21)	350.79 ($\pm$23.41)	85.76 ($\pm$2.38)

Next, we select 1,000 traffic traces uniformly at random from the test dataset and test the flows based on the proposed delay-aware UAV detection algorithm. Table II shows the test results where $E[p^*] = \frac{1}{N_\gamma}\sum_{i=1}^{N_\gamma} p_i^*$ denotes the average optimal number of packets, $E[t_{p^*}] = \frac{1}{N_\gamma}\sum_{i=1}^{N_\gamma} t_{p_i^*}$ indicates the average arrival time of p^*th packet, $E[P_r] = \frac{1}{N_\gamma}\sum_{i=1}^{N_\gamma} P_{r_i}$ denotes the average detection accuracy, and N_γ is the number of selected traces for class γ where $\gamma \in \{\text{Bebop 1}, \text{Bebop 2}, \text{Spark}, \text{UDI}\}$. The results show that for the four tested UAV types, our proposed method can detect and identify the UAVs in average within $0.15 - 0.35s$ with high accuracy of $85.7 - 95.2\%$.

VII. Conclusions

To complement existing physical detection mechanisms, In this paper, we proposed a delay-aware machine learning-based UAV identification framework over encrypted Wi-Fi UAV traffic. This framework extracts features from packet size and inter-arrival time and in the model training phase adopts re-weighted ℓ_1-norm regularization with consideration of computation time among various features. To deal with packet inter-arrival time uncertainty when estimating the delay cost function, we utilized model-based MLE method to estimate the packet inter-arrival times of the incoming flow. We collected a large amount of encrypted Wi-Fi traffic of four types of consumer UAVs and conducted extensive evaluation on the performance of our proposed methods. Experimental results show that the proposed method can identify the tested UAVs within $0.15-0.35s$ with the accuracy of $85.7-95.2\%$.

VIII. Acknowledgment

This work is partially supported by National Security Agency under grants No. H98230-16-1-0356 and No. H98230-18-1-0343.

IX. Appendix

A. Statistical Features

Table III: Statistical features (sample size $N = 100$).

Function: Feature Name	Description	Comput. time
$V_1(x)$: mean	$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x(i)$	0.672 μs
$V_2(x)$: median	The higher half value of a data sample.	4.365 μs
$V_3(x)$: MAD[1]	$MAD = median(\|x(i) - median(x)\|)$	8.346 μs
$V_4(x)$: STD[2]	$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x(i) - mean(x))^2}$	1.608 μs
$V_5(x)$: Skewness	$\gamma = \frac{1}{N}\sum_{i=1}^{N}(x(i) - mean(x))/\sigma)^3$	14.917 μs
$V_6(x)$: Kurtosis	$\beta = \frac{1}{N}\sum_{i=1}^{N}(x(i) - mean(x)/\sigma)^4$	14.095 μs
$V_7(x)$: MAX	$H = (Max(x(i))\|_{i=1\ldots N})$	0.464 μs
$V_8(x)$: MIN	$L = (Min(x(i))\|_{i=1\ldots N})$	0.652 μs
$V_9(x)$: Mean Square	$MS = \frac{1}{N}\sum_{i=1}^{N}(x(i))^2$	1.147 μs
$V_{10}(x)$: RMS	$RMS = \sqrt{ms(x)}$	1.273 μs
$V_{11}(x)$: PS[3]	$3(mean(x) - median(x))/\sigma$	8.011 μs
$V_{12}(x)$: MAD[4]	$MAD = \frac{1}{N}\sum_{i=1}^{N}\|(x(i) - mean(x))\|$	2.531 μs

[1] MAD: median absolute deviation
[2] STD: standard deviation
[3] PS: Pearson Skewness
[4] MAD: mean absolute deviation

B. Exponential Distribution Parameter Estimation Using MLE

Let $\{\mathcal{T}_n^i\}$ be a sequence of n independent and identically distributed (IID) exponential random variables. Thus, $\mathcal{T}_j^i \sim \text{Exp}(\mu_i)$ has a probability density function (pdf) of $f_{\mathcal{T}^i}(\tau_j^i) = \mu_i \text{exp}(-\mu_i \tau_j^i)$ for $\tau_j^i \geq 0$ with parameter μ_i, where $j = 1, ..., n$, $i = 1, ..., \upsilon$ and $\upsilon = |\mathcal{Y}|$. Given the data sequence $\{\mathcal{T}_n^i\}$, our goal is to estimate the average packet inter-arrival time (i.e., μ_i). Since $\mathcal{T}_j^i$ for $i = 1, ..., \upsilon$ and $j = 1, ..., n$ are assumed to be i.i.d., then the likelihood function is given by

$$\mathcal{L}(\mu_i; \tau_1^i, ..., \tau_n^i) = \prod_{j=1}^{n} f_{\mathcal{T}^i}(\tau_j^i; \mu_i) = \mu_i^n \text{exp}\left(-\mu_i \sum_{j=1}^{n} \tau_j^i\right) \tag{7}$$

By taking log of both sides in Eq. (7), we obtain the log-likelihood function as

$$l(\mu_i; \tau_1^i, \tau_2^i, ..., \tau_n^i) = n\ln(\mu_i) - \mu_i \sum_{j=1}^{n} \tau_j^i. \tag{8}$$

Then, maximum log-likelihood estimation of μ_i is achieved by solving the first order maximization problem of

$$\hat{\mu}_i = \underset{\mu_i}{\text{argmax}}\ l(\mu_i; \tau_1^i, \tau_2^i, ..., \tau_n^i) \tag{9}$$

as $\frac{d}{d\mu_i} l(\mu_i; \tau_1^i, \tau_2^i, ..., \tau_n^i) = 0$, which results in

$$\hat{\mu}_i = \frac{n}{\sum_{j=1}^{n} \tau_j^i} \quad \text{for} \quad i = 1, ..., \upsilon. \tag{10}$$

References

[1] R. Altawy and A. M. Youssef, "Security, privacy, and safety aspects of civilian drones: A survey," *ACM Trans. CPS.*, vol. 1, no. 2, 2016.

[2] D. Furfaro, L. Celona, and N. Musumeci, "Civilian drone crashes into army helicopter," RT, 2017. [Online]. Available: http://nypost.com/2017/09/22/army-helicopter-hit-by-drone/

[3] RT, "Peeping drone: UAV hovers outside of massachusetts teen's bedroom window," RT, Apr. 2016. [Online]. Available: https://www.rt.com/usa/341404-drone-privacy-teenager-window/

[4] M. D. Shear and M. S. Schmidt, "White House drone crash described as a US worker's drunken lark," New York Times, Jan. 2015.

[5] D. H. Shin, D. H. Jung, D. C. Kim, J. W. Ham, and S. O. Park, "A distributed fmcw radar system based on fiber-optic links for small drone detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 2, pp. 340–347, Feb 2017.

[6] A. Moses, M. J. Rutherford, and K. P. Valavanis, "Radar-based detection and identification for miniature air vehicles," in *2011 IEEE International Conference on Control Applications (CCA)*. IEEE, 2011, pp. 933–940.

[7] A. Sutin, H. Salloum, A. Sedunov, and N. Sedunov, "Acoustic detection, tracking and classification of low flying aircraft," in *Technologies for Homeland Security (HST), IEEE International Conference*, 2013, pp. 141–146.

[8] P. Marmaroli, X. Falourd, and H. Lissek, "A UAV motor denoising technique to improve localization of surrounding noisy aircrafts: proof of concept for anti-collision systems," in *Acoustics*, 2012.

[9] P. A. Prates, R. Mendonça, A. Lourenço, F. Marques, J. P. Matos-Carvalho, and J. Barata, "Vision-based uav detection and tracking using motion signatures," in *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, May 2018, pp. 482–487.

[10] A. Rozantsev, V. Lepetit, and P. Fua, "Detecting flying objects using a single moving camera," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 5, pp. 879–892, 2017.

[11] N. Jing, M. Yang, S. Cheng, Q. Dong, and H. Xiong, "An efficient svm-based method for multi-class network traffic classification," in *30th IEEE International Performance Computing and Communications Conference*, Nov 2011, pp. 1–8.

[12] A. McGregor, M. Hall, P. Lorier, and J. Brunskill, "Flow clustering using machine learning techniques," in *Passive and Active Network Measurement*. Springer Berlin Heidelberg, 2004, pp. 205–214.

[13] R. Bar Yanai, M. Langberg, D. Peleg, and L. Roditty, "Realtime classification for encrypted traffic," in *Experimental Algorithms*. Springer, 2010, pp. 373–385.

[14] A. Dachraoui, A. Bondu, and A. Cornuéjols, "Early classification of time series as a non myopic sequential decision making problem." Springer International Publishing, 2015, pp. 433–447.

[15] A. Moses, M. J. Rutherford, and K. P. Valavanis, "Radar-based detection and identification for miniature air vehicles," in *2011 IEEE International Conference on Control Applications (CCA)*. IEEE, 2011, pp. 933–940.

[16] A. Moses, M. J. Rutherford, M. Kontitsis, and K. P. Valavanis, "UAV-borne X-band radar for collision avoidance," *Robotica*, vol. 32, no. 1, p. 97114, 2014.

[17] A. Rozantsev, V. Lepetit, and P. Fua, "Flying objects detection from a single moving camera," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 4128–4136.

[18] F. Göke, G. Üoluk, E. Sahin, and S. Kalkan, "Vision-based detection and distance estimation of micro unmanned aerial vehicles," in *Sensors*, 2015.

[19] S. Birnbach, R. Baker, and I. Martinovic, "Wi-Fly?: Detecting privacy invasion attacks by consumer drones," in *24th Annual Network and Distributed System Security Symposium, NDSS*, 2017.

[20] P. Nguyen, H. Truong, M. Ravindranathan, A. Nguyen, R. Han, and T. Vu, "Matthan: Drone presence detection by identifying physical signatures in the drone's rf communication," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys. New York, NY, USA: ACM, 2017, pp. 211–224. [Online]. Available: http://doi.acm.org/10.1145/3081333.3081354

[21] G. Xie, M. Iliofotou, R. Keralapura, M. Faloutsos, and A. Nucci, "Subflow: Towards practical flow-level traffic classification," in *IEEE INFOCOM*, March 2012.

[22] F. Bach, R. Jenatton, J. Mairal, G. Obozinski *et al.*, "Optimization with sparsity-inducing penalties," *Foundations and Trends® in Machine Learning*, vol. 4, no. 1, pp. 1–106, 2012.

[23] K. P. Murphy, *Machine learning : a probabilistic perspective*. MIT Press, 2013.

[24] A. Komaee, "Maximum likelihood and minimum mean squared error estimations for measurement of light intensity," in *44th Annual Conference on Information Sciences and Systems (CISS)*, March, pp. 1–6.

[25] S. Sengupta, H. Gupta, N. Ganguly, B. Mitra, P. De, and S. Chakraborty, "CRAWDAD dataset iitkgp/apptraffic (v. 2015-11-26)," Downloaded from https://crawdad.org/iitkgp/apptraffic/20151126, Nov. 2015.

[26] Total cost optimization demo, https://youtu.be/eR9Keh2bO4I.