University of Padova

Department of Mathematics "Tullio Levi-Civita"

Master Thesis in Cybersecurity

Aerial Threat Detection around Critical Infrastructures: Drone Trajectory Analysis and Behavior Profiling

Supervisor
Prof. Alessandro Brighente
University of Padova

Co-supervisor
Prof. Mauro Conti
University of Padua

Master Candidate
Luca Cosuti

Student ID
2057061

Academic Year
2022-2023

> “What is the cost of lies”
>
> — Jared Harris

Abstract

In recent times, the utilization of drones has seen a significant increase across both civilian and military domains. Within the military sphere, these UAVs (Unmanned Aerial Vehicles) exhibit the capacity to engage targets at extended distances, disrupt critical infrastructure, and conduct comprehensive facility assessments via ISR (Inspection, Surveillance, and Reconnaissance) operations. To prevent such events from happening, it is required a calibrated system to detect drones and conduct an analysis of their intentions based on their movement and behavior, while also considering the economic implications of deploying costly missiles for every drone interception. This is the principal focus of this thesis: after detailed research, some credible trajectories that could be followed by attacking drones were simulated and fed to a classifying system with the scope to recognize which patterns were attacks and which were harmless. The classifier is also supported by a trajectory-forecasting system, that can help direct the decision. The entire system has then been tested on real-world data to ensure its feasibility in a realistic scenario

Contents

Listing of figures

Listing of tables

Listing of acronyms

ISR	Inspection, Surveillance and Reconnaissance
UAV	Unmanned Aerial Vehicle
CI	Critical Infrastructure
GCS	Ground Control Station
LSTM	Long Short Term Memory Network
MAE	Mean Absolute Error
SVM	Support Vector Machine
MLP	MultiLayer Perceptron
AIS	Automatic Identification System
IMU	Inertial Measurement Unit

1 Introduction

Unmanned Aerial Vehicles, or UAVs, are remotely-controlled aircraft that do not require an onboard pilot to be managed. They can be found in different shapes and sizes, with different names according to the number of rotors. In recent years there has been a huge increase in usage of such devices in both civilian and military environments due to their reliability, high quality of the onboard camera, and intuitive controls. Figures 1.1 and 1.2 show a traditional, consumer-grade drone and a military drone respectively.

Figure 1.1: DJI Inspire 2

Figure 1.2: General Atomics MQ-1 Predator

Traditionally, a drone ecosystem is composed of multiple elements, both portable and fixed:

1. Drone Hardware: every onboard electronic component used to control the drone: CPU, a miscellaneous of sensors such as gyroscopes and accelerators, firmware.

2. Drone Chassis: every non-electronic component of the drone: it is typically made in plastic or carbon fiber to provide robustness and prevent additional weight.

3. Ground Control Station: the device used to control the UAV remotely, it could either be mobile or stationary.

4. Pilot: whoever is controlling the UAV via the GCS.

5. Communication Channel: the transmission medium used by the pilot to communicate with the drone.

The rise of UAVs owes much to their improved battery life, lightweight design, and increased operational altitude. These technological strides have made drones invaluable in fields like express delivery [1], search and rescue [2], geographic mapping of hard-to-reach places [3], and border surveillance [4]. Industries like agriculture, engineering, and defense have all seen significant benefits from the versatility of drones.

Yet, there's a flip side to this coin. While these advancements open doors for positive applications, they also make UAVs perfectly suitable for misuse. In recent times, we've witnessed an uptick in the not-so-friendly uses of drones, ranging from spying [5] to acts of terrorism [6]. The very features that make UAVs appealing for good purposes, such as their improved battery life and portability, also make them attractive for less savory activities. Drones are now being used to get hacking gear closer to isolated computer systems and even for smuggling illegal goods across borders.

So, while we revel in the positive potential of UAVs, it's crucial to acknowledge the emerging threats they bring. The dual nature of these technological marvels highlights the need for robust cybersecurity measures to keep pace with their expanding use. In response to the escalating concerns surrounding drone-related incidents, the Federal Aviation Administration (FAA) has implemented **RemoteID** [7]. This system enables a flying drone to transmit essential information to any accessible ground station. As per this regulatory framework, a drone in flight must continuously relay its pilot ID, current flight location, and altitude to any monitoring party at intervals of one second or less. This protocol, akin to ADS-B technology employed to track the location and communications of airplanes, is tailor-made for precise drone tracking.

However, while RemoteID serves the purpose of identifying an approaching drone, it falls short in providing insights into the vehicle's intentions. Even with an infrastructure in place and a UAV properly disclosing its approach, a crucial gap exists in understanding the purpose behind the vehicle's proximity. Consequently, beyond merely identifying a drone, it becomes imperative to establish a method for classifying the behavior of these devices as either *innocuous* or *dangerous*. This thesis focuses primarily on addressing this critical aspect.

The underlying objective of this research is to address the pressing need for promptly and accurately ascertaining the intentions of drones, thereby mitigating the risk of potential attacks. It is imperative to acknowledge that not all approaching drones bear malicious intent. Given the complexity and cost associated with neutralizing a flying device, especially in scenarios such as a military facility equipped with anti-aircraft systems requiring expensive ammunition, the importance of distinguishing between threatening and non-threatening drones becomes mandatory. This differentiation allows for a more targeted and cost-effective response, focusing countermeasures only on those posing a genuine threat.

While considerable attention has been devoted to the domains of drone detection and tracking, as evidenced by works such as [8] and [9], and the challenge of discriminating between drones and birds has engaged various research groups in the last few years ([10, 11]), the classification of drones based solely on their behavior remains an under-explored area. This thesis represents one of the pioneering attempts to bridge this gap by introducing a novel approach to drone identification centered on behavioral patterns. The study endeavors to demonstrate the feasibility of deducing UAV intentions through the analysis of flight trajectory data, with a specific emphasis on distinguishing between Intelligence, Surveillance, and Reconnaissance (ISR) activities and innocuous behaviors. In the subsequent chapters, the thesis unfolds its exploration. Chapter 2 reports the current State of the Art methods used for trajectory classification and forecasting, and behavior identification. A few additional analyses complementary to this one are being discussed as well. Chapter 3 reports the specific system that has been defined for this analysis together with the threat model it can present; Chapter 4 defines a Deep Learning model, outlining the subsequent training and testing phases while Chapter 5 reports a comprehensive description of the simulated dataset, by thoroughly explaining the structure of each trajectory. The narrative then progresses to Chapter 6, where the selected method undergoes validation through testing on real-world data. This validation process unfolds in two phases: initially, trajectories are classified as innocuous or offensive, followed by real-time-like forecasting of such data. This part is then followed by Chapter 7 where the conclusions are stated, and Chapter 8 approaches the next steps to further tackle down this problem.

2
Related Works

This study represents a pioneering effort in simultaneously addressing trajectory classification and forecasting tasks within the context of Unmanned Aerial Vehicle movement tracking for behavioral classification purposes. The initial section of the Related Works segment will cover the contemporary state-of-the-art in trajectory classification and forecasting pertaining to objects or humans. Subsequently, the second segment will concentrate on supplementary literature addressing the subject of drone detection and identification. Such analyses are considered complementary to this research, providing a comprehensive overview of the relevant scholarly landscape.

Trajectory classification has yielded noteworthy outcomes through computer vision methodologies, exemplified by Kontopoulos et al.'s work [12], employing clustering-based approaches as illustrated by Lee et al.'s contribution [13], and adopting various machine learning techniques such as Support Vector Machines (SVM) and Multilayer Perceptrons (MLPs), as surveyed by da Silva et al. [14]. Additionally, the proliferation of available data has facilitated the utilization of deep learning models, as evidenced by Kontopoulos et al.'s exploration [15].

Similarly, trajectory forecasting has been subject to rigorous analyses. Salzmann et al. [16] introduced a framework guided by a modular, graph-structured recurrent model addressing human-robot interactions in autonomous systems. Giuliari et al. [17] proposed an innovative approach to enhance trajectory forecasting, substituting Long Short-Term Memory (LSTM) networks with Transformers [18], motivated by the success of Transformers in Natural Language Processing (NLP). Kothari et al. [19] conducted a comprehensive evaluation of deep learning-based methods for modeling social interactions, introducing a novel benchmark with evaluation metrics to quantify the physical feasibility of a forecasting model. Even the topic of human motion trajectory forecasting has been covered, as shown by Rudenko et. al. in their 2020 work [20]. While certain trajectory categories have been predominantly considered in forecasting studies, the field of cyber-physical systems is gradually being incorporated into this domain. Zhang et al. [21] addressed the prediction of maritime vessel movements, a theme also explored by Sorensen et al. [22]. Additionally, Xiao et al. [23] focused on traffic forecasting, contributing insights that can be integrated into the broader trajectory forecasting discourse. A similar approach has been followed by Altché and de La Fortelle [24], a study where they defined an LSTM model for highway trajectory prediction.

Simultaneously, the realm of behavior classification has undergone comprehensive scrutiny. Gutierrez et al. [25] devised an embedded neural network approach for real-time classification of animal behaviors, facilitating the mapping of their movements and variations in behaviors. Aoude et al. [26] introduced a driver behavior classification method based on Hidden Markov Models and Support Vector Machines, while Riley and Veloso [27] proposed an adaptive approach to modulate the behavior of a robot in an adversarial environment. Notably, Zhou et al.'s work in 2019 [28] on ships' classification based on their behavior, derived from Automatic Identification System (AIS) data, aligns closely with the thematic focus of this research. The key distinction between Zhou's work and this very research lies in the incorporation of a real-time forecasting module within this present work, coupled with a fully modular and customizable bespoke model that allows it to scale to any Critical Infrastructure.

The task of drone identification is as intricate as classifying its behavior. Broadly, drone detection primarily employs either acoustic or radar-based methods, both of which will be briefly expounded upon. Acoustic methods involve mapping the sound captured by a sensor to a repertoire of well-known drone-produced noises. However, this approach encounters significant limitations in environments with heightened ambient noise levels, such as urban areas or airports, where distinguishing characteristic drone patterns becomes challenging. Additionally, the effective range of acoustic methods is constrained to approximately 300 meters, proving restrictive when drones are operating at high speeds toward a target, as noted in the works of [29] and [30].

Another drawback of acoustic methods is their reliance on a database of known drone sounds, implementing a signature-based detection approach. Consequently, unknown sounds emitted by new or modified drone models may go undetected, as discussed in [31]..

The second method of drone detection involves the use of radar. High-frequency radar exhibits high accuracy but is effective only within a limited range, typically up to several tens of meters. To address this limitation, a system employing a network of smaller drones communicating with a control station has been proposed. This network can collectively perform detection, tracking, and localization by approaching and monitoring an unidentified drone [32] [33].

On the other hand, low-frequency radar offers a considerably larger detection range but faces challenges in detecting smaller objects such as drones [34] [35]. The drawbacks of radar-based detection include elevated costs, susceptibility to interference from obstacles, diminished detection rates in congested air-spaces, and the complexity of classification or identification [31].

An additional avenue for future work involves an exploration of the field of driver classification, which has seen increased research efforts amid the growing interest and investments in autonomous vehicles. For instance, Bouhoute et al. [36] employ graphical models to represent driver behavior, employing probabilistic hybrid automata and labeled directed graphs. This methodology is notable for utilizing transitions of states to define behavior, providing a structured approach to behavior representation. In the context of drones, these states could serve as a simplistic means of depicting potential threats. The continual tracking of behavior through different states post-detection enables the identification of significant behavioral changes, facilitating the classification of drones as potentially malicious.

In conclusion, drone detection constitutes a dynamic field of study, marked by continuous evolution and improvement. Various methods with distinct challenges contribute to this area. Based on considerations of the adversary's nature and the operational environment, radar-based detection has been selected as the preferred approach.

3
System and Threat Model

3.1 Drone Detection to Deactivation Process

In addressing security threats posed by drones, Solomitckii et al. [37] outline a comprehensive five-stage process for preventing drone attacks: detection, localization, tracking, classification and identification, and deactivation.

- Detection: The initial step involves becoming aware of objects entering a protected area.

- Localization: Following detection, the precise location of the entering object is determined.

- Tracking: Once the location is identified, the system tracks the detected object over time.

- Classification: This step involves determining the type of object being tracked, and distinguishing between relevant and irrelevant types.

- Deactivation: The final stage revolves around deactivating the detected drone, and mitigating potential threats.

3.2 Considered Scenario

This research considers a Critical Infrastructure in need of protection against potential drone attacks. The CI could encompass various facilities, such as military bases, industrial plants, or any establishment involved in safety-related operations or safeguarding sensitive content from external threats. The assumption is that the CI acknowledges the drone-related threats and opts to implement the five-stage Solomitckii approach to counter such threats [37]. Specifically:

- The CI deploys a radar-based system for information gathering, a technology renowned for its accuracy over long distances and under adverse weather conditions.
- The radar system, positioned near the control room, can distinguish drones from other objects and derive the coordinates of specific drones to create individual location tracks.

While the detection and tracking aspects fall beyond the scope of this work, the focus remains on drone identification. The research introduces an approach to building this identification module, addressing a fundamental requirement for Critical Infrastructures. CIs may need to deactivate drones, a crucial step that should only occur upon the detection of a credible threat to avoid causing harm to non-threatening entities. Figure 3.1 provides a simplified overview of the basic system model, emphasizing the importance of drone identification in securing Critical Infrastructures.

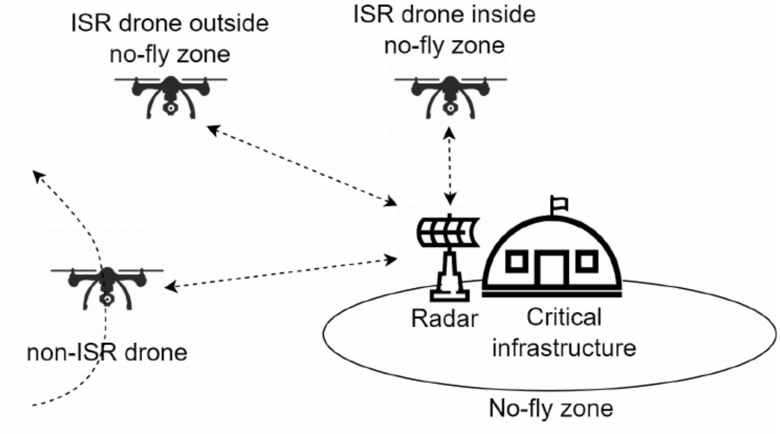


Figure 3.1: System Model with a CI equipped with a radar system to protect against ISR drones.

3.3 Threat Model

When delineating potential drone-induced threats, six primary threat scenarios align with this profile: Intelligence, Surveillance, and Reconnaissance operations (ISR), Air Disruption operations [38], Harassment operations [39], Transportation operations [40], Man-in-the-Middle (MitM) operations [41], and Attack operations [42]. This research specifically focuses on the initial set of threats. It is essential to clarify that this investigation is restricted to civilian drones, exemplified by Unmanned Aerial Vehicles (UAVs) accessible for purchase and general usage. This stipulation is underpinned by two principal considerations: firstly, civilians may engage in ISR operations, as evidenced by instances such as the Frankfurt airport attack that took place earlier this year [43]; secondly, the technological sophistication of military drones surpasses the resources available to private citizens.

In 2013, the Defense Advanced Research Projects Agency (DARPA) commissioned the development of the Autonomous Real-Time Ground Ubiquitous Surveillance Imaging System (ARGUS-IS) project [44]. This project encompassed a wide-area persistent surveillance system, featuring a camera system with a remarkable 1.8 Gigapixels of resolution and two subprocessing systems. Designed to be mounted on military-grade UAVs, this camera system enables detailed video capture, allowing users to collect data and track individual subjects within the field of regard. Notably, the system possesses the capability to detect human movements, such as hand gestures, from an altitude of 5km. Consequently, a drone equipped with this technology could operate above critical infrastructure at considerable heights, potentially being misconstrued as an airplane and not classified as an attack. To maintain a focus on realistic threats, this research exclusively addresses commercially available drones.

The premise assumes that an attacker may customize a drone with a camera and microphone to capture sensitive information regarding Critical Infrastructures. The attacker gathers intelligence by capturing images and videos or via a live video feed. Assumptions regarding whether the drone internally stores this information or transmits it to the attacker via wireless communications are withheld, since it does not realistically modify the approach to solving this task. The collected data can then be utilized to plan subsequent actions or traded to media outlets for publicity purposes. The collected information encompasses details such as the location of objects within the CI area, the presence of individuals, the existence of dedicated anti-drone technologies, and the location of a specific building of interest, among other sensitive data.

For effective ISR operations, the drone is required to maintain proximity to the CI for an extended duration, accumulating comprehensive data before departing the premises. To facilitate a broad application of this model, no assumptions are made regarding either the Critical Infrastructure or the drone itself. It has been established that to avoid swift detection, the drone does not significantly alter its velocity while approaching the infrastructure, nor does it change its altitude. This approach ensures applicability across various drone models, accommodating those capable of flying at considerable altitudes and those constrained to lower heights, while emphasizing the overarching concept of directionality. It is indeed taken into consideration the distance between the drone and the buildings since the constant reduction of such metric would indicate the UAV's approach.

It is necessary to remember that not every drone approaching a CI necessarily signifies an impending attack. For instance, a photographer capturing images and videos of an area adjacent to the infrastructure for wildlife monitoring or search and rescue purposes does not pose a disruptive impact on the CI. Acknowledging that the response required to neutralize a flying device can be costly and operationally challenging (e.g., employing anti-aircraft artillery or missiles) is essential. Thus, distinguishing between indications of an attack and benign activities is essential.

4
Classification and Forecasting Framework

The task of behavior identification consists of classifying a trajectory in one of two categories: attack or innocuous. This means that every traditional approach to trajectory evaluation can be used, so either the methods employed for trajectory classification (as thoroughly described by da Silva et.al. [14] and Bian et.al. [45]) and the techniques for trajectory forecasting, as studied by Shi and Dit-Yan [46] can be applied. On the topic of trajectory forecasting, since the data this research is being based on is quite straightforward it is possible to obtain good results while performing such an analysis: given the dangerous orbits, adding a forecasting component would allow for quicker identification of a potential attack if the unidentified drone were to move accordingly to the predicted orbits. For this very reason, a specific LSTM model has been developed and fine-tuned: the next section will proceed with the description of such architecture. Also, due to the modular nature of the work presented, the whole framework can be fragmented in modules executable one after the other: this pipeline can be better seen in Figure 4.1.

4.1 LSTM

Long Short-Term Memory networks, or LSTMs [47], represent a category of recurrent neural networks designed for handling sequential data, particularly excelling in scenarios where understanding and capturing long-term dependencies are essential (e.g., text comprehension and generation, speech recognition, etc...) [48].

In the context of trajectory forecasting, LSTMs prove to be highly effective as seen in the works of [49, 24]. Their design enables them to model data sequences, making them well-suited for tasks like time series prediction and trajectory forecasting. What sets LSTMs apart is their unique ability to retain information over extended sequences: this capacity to capture and remember temporal relationships in data proves invaluable when predicting future positions based on previously provided trajectory information.

As an additional point, LSTMs perform well when the trajectories vary in length, as they can quickly handle sequences of different sizes without sacrificing information; this property comes in hand when training the model on simulated data characterized by constant length, and testing it on real data which can be less consistent in terms of extent. In the particular case of this research, it follows the structure of the LSTM shown below.

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
    self.model = Sequential()
    self.model.add(LSTM(32, input_shape=(self.num_prev_positions, 2),
        return_sequences=True))
    self.model.add(LSTM(16))
    self.model.add(Dense(8, activation='relu'))
    self.model.add(Dense(2, activation='linear'))
    self.model.compile(loss="mean_squared_error", optimizer='adam')
```

Listing 4.1: LSTM Model

In the case of this research, this model has been trained and tested only on *simulated attack* trajectories. At the same time, the forecasting and classification analyses have been conducted on real-world data containing both attack and innocuous trajectories.

4.2 Trajectory Forecasting

In the trajectory forecasting task, a critical component is defining future positions as labels, essential for the effective training of the model. This process involves presenting the model with known x, y coordinates and their corresponding timestamps as input data. Simultaneously, the labels are determined by the x, y coordinates at timestamps situated further ”in the future.” This unique approach empowers the LSTM model to establish meaningful associations between each current position and its anticipated future position. The richness of training data, encompassing diverse trajectories, significantly contributes to the model's versatility. Exposure to varied x, y coordinates across different trajectories and time instances enables the LSTM to grasp the essence of trajectory patterns rather than fixating on specific instances. This capability enhances the model's ability to generalize and make accurate predictions across a range of scenarios. After rigorous testing, it was determined that assigning each position in the trajectory its corresponding label as the position **three** timestamps in the future strikes a balance between achieving a non-immediate forecasting horizon and maintaining the granularity necessary for precise classification. This strategic choice optimally captures the trajectory's evolution over time, ensuring the model's adaptability to diverse patterns while minimizing localized errors. The decision to use the position three timestamps in the future was found to provide the best results after all, avoiding both over-dependence on trajectory-specific instances and excessive generalization.

4.3 Trajectory Classification

The capabilities of an LSTM model do not limit its application to the field of forecasting, as the model can also be used to classify time series. In the case of this research, the approach used consisted in defining an error metric, computing for each forecasted trajectory the error accordingly with the chosen method, and comparing such errors with one another to figure out eventual dependencies. This approach turns out to be very efficient in distinguishing between attack trajectories and innocuous trajectories, since once provided a model that was trained on attack trajectories only it can be safely assumed that the error in forecasting such trajectory will be lower than the one computed for the innocuous trajectories, so when established the line of distinction between the two categories, the error computed during the trajectory forecasting task allows for a trajectory classification too.

The error metric selected is the Mean Absolute Error (or MAE), defined as follows.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \left|y_i - \hat{y}_i\right|$$

where the absolute value of the difference represents the distance between the predicted value for the label (e.g., what the model supposes will be the position 3 timestamps from the current one) and the real value of such label (the actual position). This metric has been selected as it describes the error between paired observations expressing the same phenomenon, hence perfectly fitting the trajectory forecasting task. Specifically, the Mean Absolute Error (MAE)

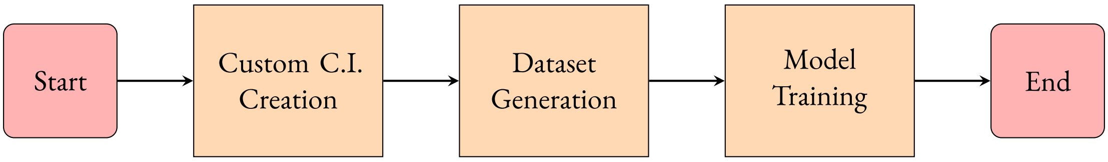


Figure 4.1: Representation of the Framework's pipeline

metric has been exclusively employed for the classification of real-world trajectories. Following the training and testing phases on synthetic data, the model's MAE for each authentic orbit was computed. Orbits with an error exceeding a threshold of **8** were identified as innocent trajectories, while those below the specified threshold were categorized as potential attacks.

This approach facilitated a linear classification of both trajectory types, achieving a perfect accuracy of 100% in classification on each occasion.

4.4 Pipeline Visualization

The selection of a Deep Learning approach for this research was motivated not only by the abundance of simulated data, enabling effective model performance, but also by the modularity inherent in such architectures. The primary objective of this work was to establish a scalable framework that could readily adapt to diverse real-world scenarios. Given the inherent variability of critical infrastructure layouts and the need for tailored calibrations, a one-size-fits-all model would encounter challenges in accommodating every possible condition. To address this, the research adopts the pipeline illustrated in Figure 4.1.

The initial segment of the pipeline involves code utilization to customize the structure of Critical Infrastructure, a topic elaborated upon in Chapter 5. Subsequently, the generation of attack trajectories takes place, adjusting according to the specific structure defined in the prior step of the pipeline. The concluding phase of the framework involves training the model on these dynamically generated trajectories.

This approach facilitates versatility in addressing the problem, as the model can be precisely trained on trajectories crafted to align with the specific premises of each unique case.

5 Dataset

Due to the particular nature of this research, it is quite difficult to obtain logs of drones inspecting an infrastructure for processing; for this reason, the majority of the tests have been conducted on simulated data generated using custom scripts in Python. A hypothetical infrastructure is indicated on a 2D plane as a circle for simplicity of representation, and developed as described below:

- Building center: it is indicated as a red dot in the graphs, it can be considered as the center of the infrastructure. A 360° radar could be placed here to maximize the coverage, or it can also simply be taken as a reference point.

- No Fly Zone: the infrastructure per se. No unauthorized devices have permission to fly on its premises, so whatever enters this area will immediately get shot down. It is indicated as a circular zone for simplicity purposes, but it can be customized for every specific infrastructure. In a real-world case, it could be surrounded by walls, cameras, and other defensive mechanisms.

- Notification Radius: the amount of area surrounding the infrastructure that can be actively covered by a defensive mechanism, e.g., a radar. It is not the entire area the radar can cover, but it is the portion that will trigger the classification model even though the drone is still far from performing an attack. In this specific instance, the radius is 1.5 times the size of the no-fly zone. Since this pipeline is fully modular it can be easily customized, and every trajectory will be generated again with the updated values.

The assumption is that every object detected inside the *no fly zone* must be taken down as they could get physically too close to sensitive buildings, but a drone could be performing a reconnaissance mission of the premises of the CI without drawing near too close: for this reason, a series of trajectories that follow a suspicious behavior limited to the notification radius have been defined and simulated, and they are described below. The trajectories are represented (and plotted) in a similar way to what a radar could show in terms of positioning.

5.1 Trajectories

To provide a diversity of movement, four trajectories have been designed, each representing a different approach to performing an attack. It is believed that the majority of ISR operations can indeed be traced within these 4 orbits. The model has been trained on 400 trajectories properly divided into the aforementioned typologies. Each trajectory is composed of 100 points (each representing one timestamp): this granularity can be changed also according to the size of the infrastructure to defend, the amount of area that the radar can cover, the amount of error we can accept, and other characteristics specific to each case.

Each trajectory is characterized by the following features:

- timestamp: a temporal indication used to show how often the position varies. In the case of these particular trajectories, the resolution is 1 second.
- x_coordinate: the x component of the drone's position on the plane during each timestamp.
- y_coordinate: the y component of the drone's position on the plane during each timestamp.
- distance_to_infrastructure: the minimum distance between the drone and the CI during each timestamp.

To following features are provided as labels to the LSTM:

- Future_X_3: the X component of the position three timestamps from the current one.
- Future_Y_3: the Y component of the position three timestamps from the current one.

It will now follow a thorough description of each trajectory, how they are represented, and what the model can learn from them.

5.1.1 Straight Trajectory

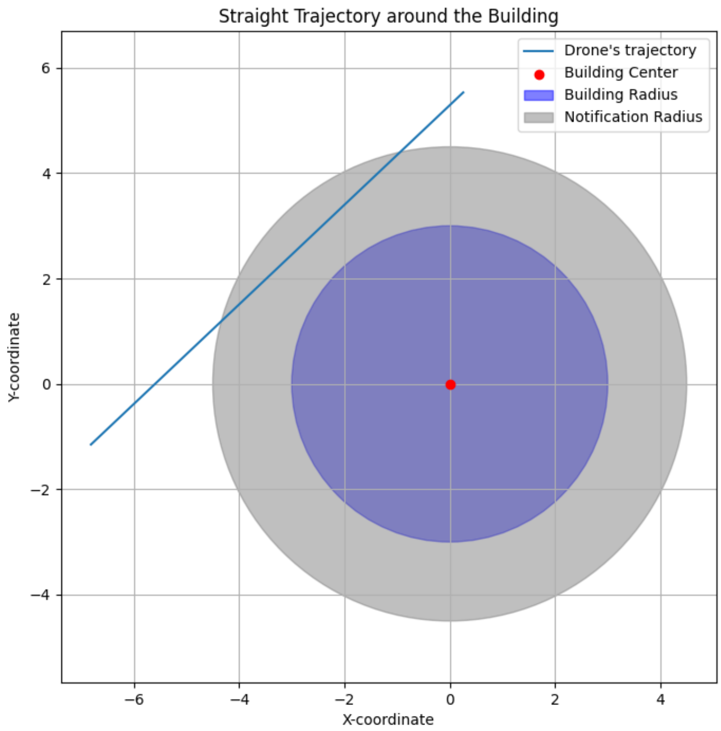


Figure 5.1: Straight Trajectory

The most simple trajectory possible, this orbit indicates an object moving toward the Critical Infrastructure following a straight line without any major divergences. The initial point and final point are outside both the *no fly zone* and the *notification radius*, therefore emulating a vehicle approaching the infrastructure from the outside, getting closer, and then leaving the area. It is possible that in some cases the drone flew over the *no fly zone* in the simulated orbit: those cases are considered as attacks since in a realistic scenario that drone would immediately get shot down. This trajectory also teaches the model how to interpret the consecutive points that make up a straight line.

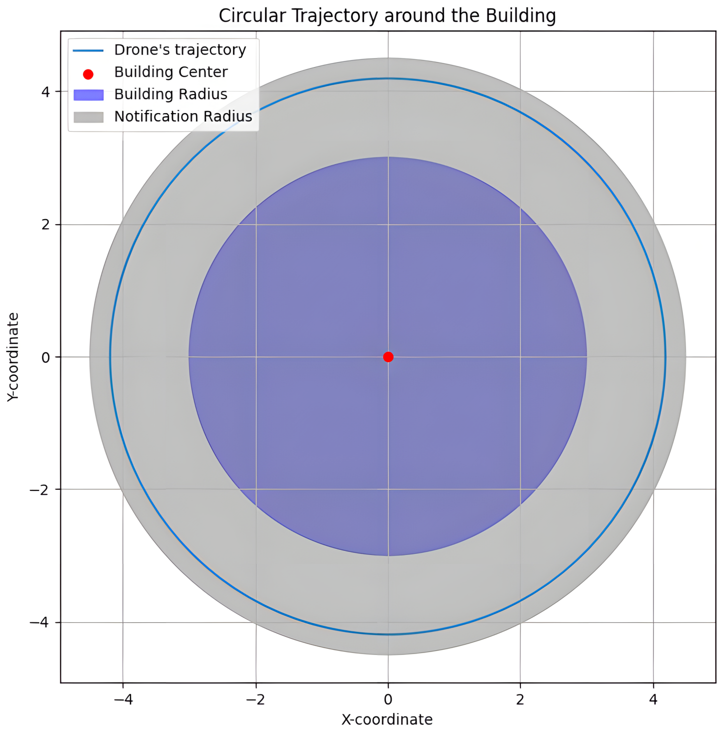


Figure 5.2: Circular Trajectory

5.1.2 Circular Trajectory

Another simple and intuitive trajectory, this circular motion indicates a drone keeping the same distance to the infrastructure while surveilling it. This scenario represents a drone that was able to elude the defensive systems from targeting it as it entered the notification radius, and that got detected while already inside the area covered by the drone. In a realistic case it is unlikely that the UAV would be able to perform an entire 360° analysis of the premises before being recognized and tear down, but the full circular motion can help the model understand the features indicating this consequential sequence of positions.

5.1.3 Spiral Trajectory

An improvement to the simple circular trajectory, this orbit represents a drone being detected as beginning its movement from outside the notification radius of the infrastructure going

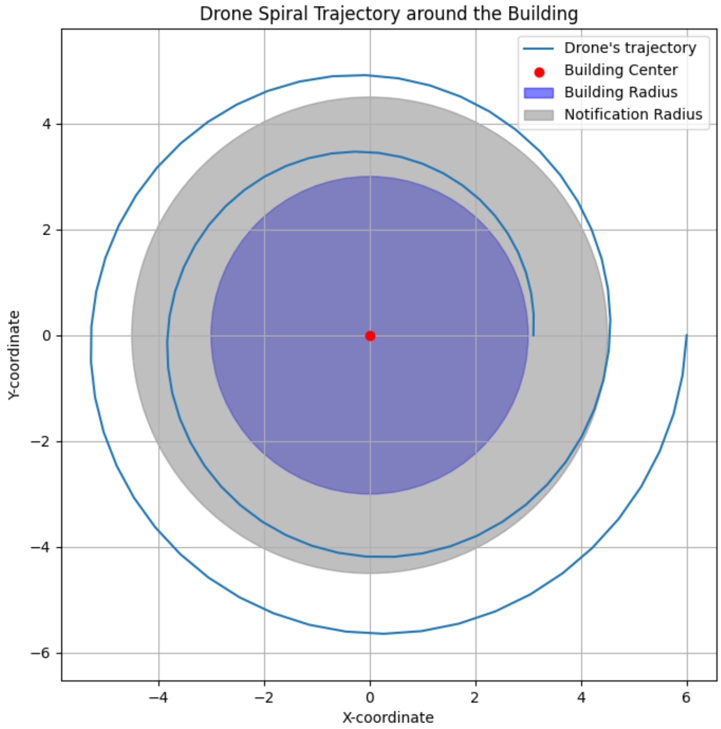


Figure 5.3: Spiral Trajectory

towards the *no fly zone* while exploring the entire infrastructure and its neighboring facilities, reaching a 360° coverage. The main difference with the Circular Trajectory is the noticing of the UAV that takes place while the drone is still not in the notification radius; this allows to better define and comprehend the entire movement of the device, and consequently, it can improve the attack detection time and precision. The number of circles of the spiral is randomly picked between 2 and 4 for each orbit to further diversify the trajectories.

5.1.4 Mixed Trajectory

This is the trajectory that can better indicate a surveilling behavior between the ones listed so far, as it is a combination of them. It shows a potential attacker reaching the CI in the quickest way possible (via a straight line) that then changes the behavior of the drone to slowly discover the infrastructure by covering a circular-like trajectory for some time, and in the end it leaves the premises again rapidly (via a straight line). The circular portion is an half-circle, but if can

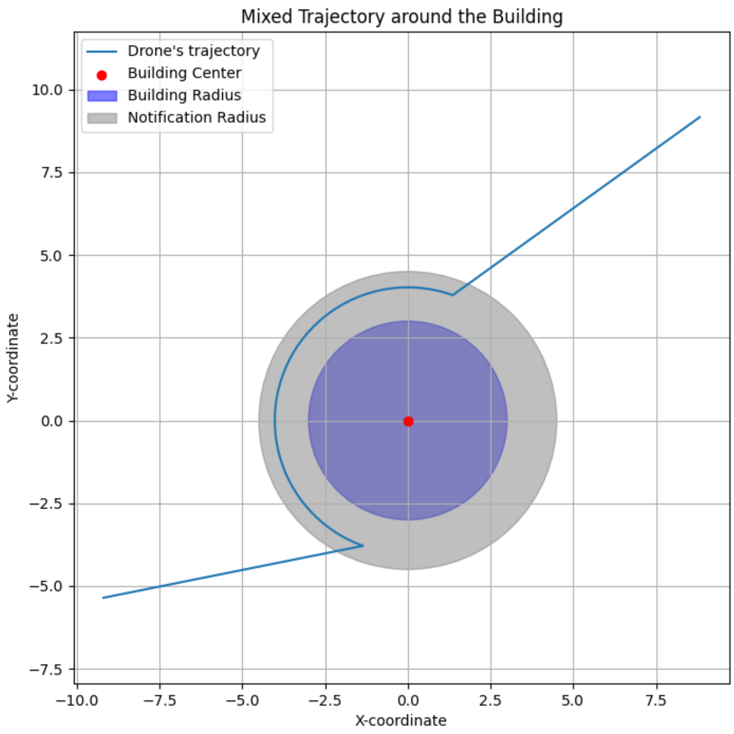


Figure 5.4: Mixed Trajectory

be extended or reduced in dimensions accordingly with the requirements of each specific case. Once again, this is yet another aspect of the developed pipeline that can be customized.

5.2 Real-Data Trajectories

In addition to simulated trajectories, the models underwent testing with real-world data. These trajectories are not publicly available since they were captured by members of the Dutch Army for internal research purposes. The drone used in this specific recording is a DJI Inspire 2 (shown in Figure 1.1, it is a case of military environments using civilian available technology for military purposes). The soldiers employed a truck as a Critical Infrastructure target to base their recordings on a physical facility for further reference. The data collection and processing were conducted using the Signal Quiet Universal Intruder Recognition Equipment (SQUIRE) radar, developed by Thales [50], and a parser has been used to transform the raw format in which the radar provides the data to a format compatible with the main programming languages. For compatibility purposes, the only features that are being extracted from the logs are the x and y coordinates together with the timestamps; in this case, the sampling resolution is not precisely set to 1 second every time due to the conversion error from the parser, but it still is comprised between 1 and 5 seconds hence acceptable. An additional feature is the *distance_to_infrastructure*, computed to ensure continuity with the format of data the model was trained on. The setting on which such trajectories have been can be seen in Figure 5.5: the truck is mimicking the Critical Infrastructure around which the drone is being flown. Figure 5.6 on the other hand shows the position of the radar device with respect to the "infrastructure".

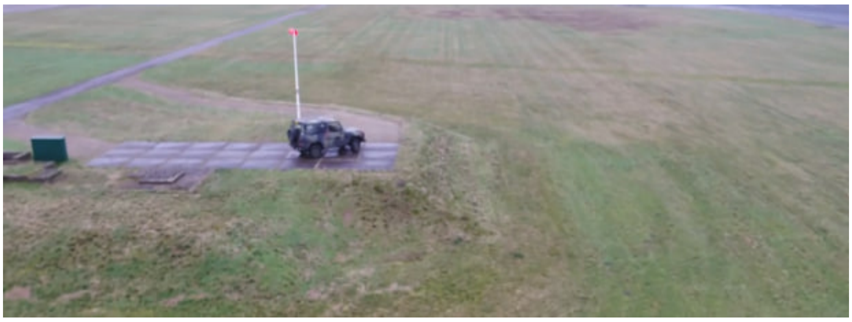


Figure 5.5: Aerial picture of the CI while being scouted by the ISR drone

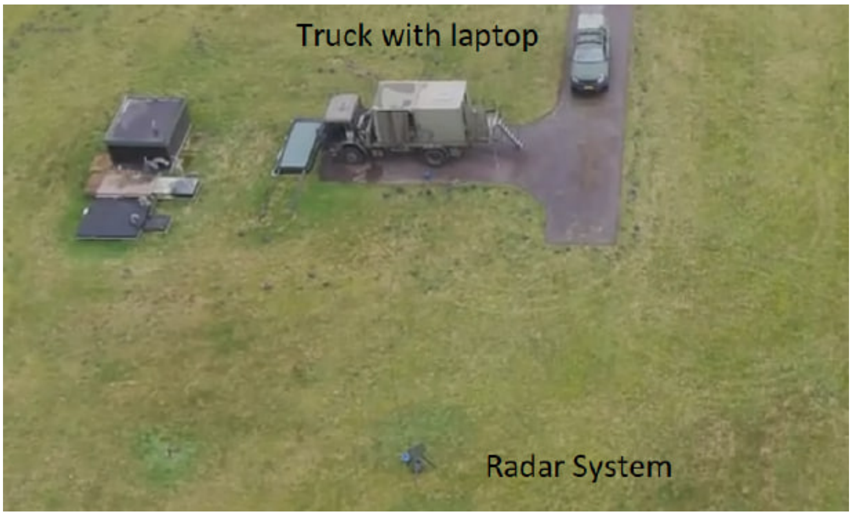


Figure 5.6: Aerial picture of the radar setup

5.2.1 Attack Pattern

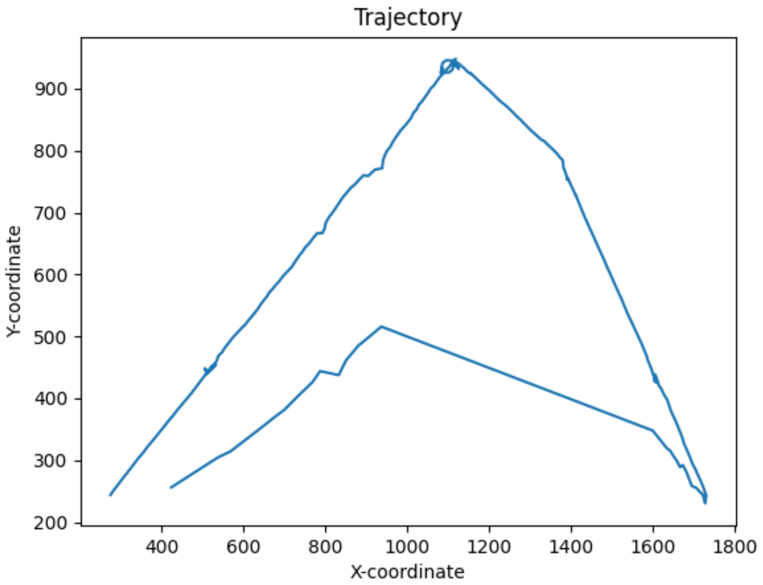


Figure 5.7: Attack Trajectory

The attack pattern shown in Figure 5.7 represents a drone executing a simile-rounded motion around a specific target situated approximately at the coordinates (1100, 650). The UAV follows the orbit in a clockwise direction, initially approaching the infrastructure along a straight line. Subsequently, it pauses in motion for a few timestamps before adopting a different direction toward the critical infrastructure. Following this segment, the drone gradually closes in on the building while retracing its previous path back to the starting point, completing the triangular motion in the process.

5.2.2 Innocuous Behavior

The innocuous orbit illustrated in Figure 5.8 adopts a notably linear approach, missing the segmented and concentrated characteristics observed in the attack trajectory. This linear pattern, lacking the distinct presence of revolutions around a building, leads to the assumption of safety associated with the innocuous trajectory.

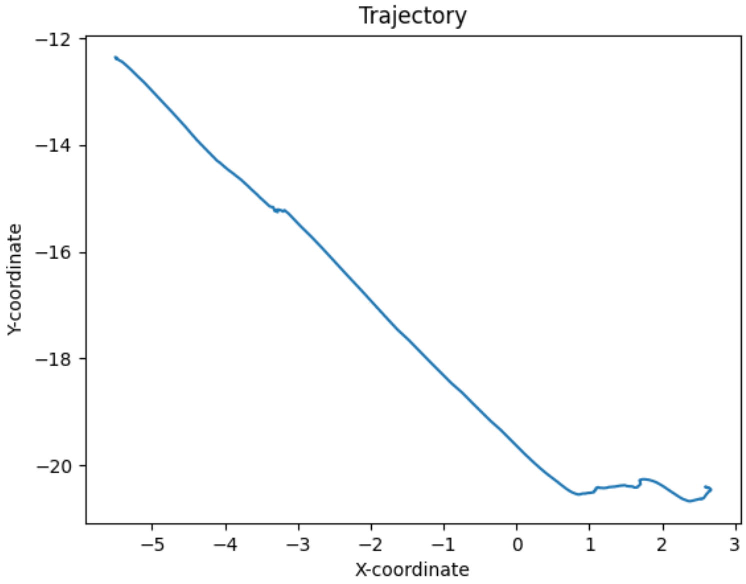


Figure 5.8: Innocuous Trajectory

6
Evaluation

This chapter delineates the processing phase of the research and unveils the core of the pipeline. Initially, the training and testing procedures of the LSTM model on synthetic data will be expounded upon. Subsequently, the model's performance will be assessed on real-world trajectories, encompassing both classification and forecasting tasks. The interdependence between these two analyses will be elucidated. Accompanying the textual descriptions, numerous graphs will be presented to visually underscore the efficacy of these tests. This approach aims to provide a comprehensive understanding of the model's capabilities and insights into its real-world applicability.

6.1 Training

The model underwent training on approximately 400 trajectories, evenly distributed with around 100 instances for each predefined attack category. The dataset was subjected to reshaping to align with the model's structure, and a scaling operation was applied to mitigate dependency on specific feature values, such as the distance from the infrastructure. The dataset has been partitioned into training and test sets, following an 80/20 distribution ratio; the model has been fitted on the data by computing 30 epochs with a batch_size of 16, a validation_split of 0.1 (to further prevent overfitting) and patience = 5 to avoid performing useless calculations. After computing the loss, which in this case is the *mean_squared_error*, the weights of the model are saved and then reloaded to ensure portability, and then it follows the sole testing with the real-world trajectories.

6.2 Testing

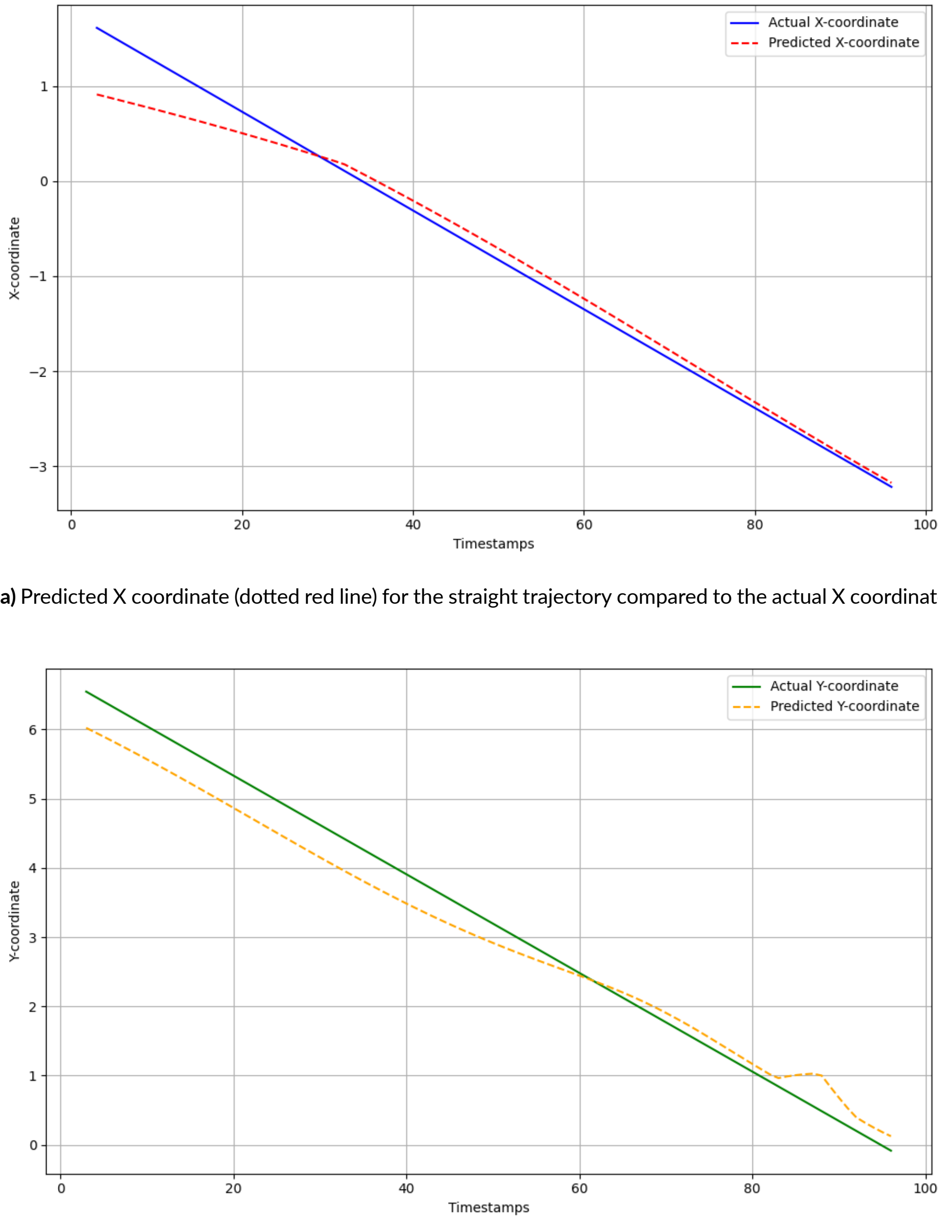
The primary goal of the training of the model was to instruct the network in interpreting variations in feature values without fixating on the specific numerical values. The results are encouraging: the test loss on the data is 0.011, indicating that the LSTM can both learn the data on which it's being trained and it is also not overfitting. Some visual results of the forecasting can be seen in Figures 6.1 and 6.2.

6.3 Real data testing

Following the loading of weights computed during training on synthetic data, the model is presently undergoing testing with real-world trajectories. The variable length of each orbit, representing data received and preprocessed by an actual radar, poses no challenges related to the length of each time series due to the adaptable nature of the LSTM, as previously detailed in Chapter 4. During the analysis phase, the model was tested on a total of 10 trajectories simulating attacks and 5 innocuous orbits. For simplicity, this report focuses on a subset of 8 attack trajectories and 1 innocuous orbit, as all safe trajectories yielded the same result. The trajectories exhibit diverse lengths, ranging from a minimum of 111 timestamps (equivalent to slightly less than 2 minutes of flight data) to a maximum of 300 timestamps (approximately 5 minutes of flight data). On average, these trajectories comprise 200 timestamps, equivalent to around 3 minutes of flight data. Figure 6.3 depicts the forecasting results on the attack trajectory, while Figure 6.4 indicates the performance of the LSTM when it comes to innocuous orbits.

The model demonstrates a quick adaptation to changes in the trajectory pattern, adjusting its direction accordingly. Notably interesting is the fact that the LSTM, drawing knowledge solely from simulated trajectories, showcases remarkable adaptability to changes in the real trajectory patterns. This suggests that the simulated attack strategies were chosen wisely, capturing characteristics that align with the potential actions of a genuine attacker.

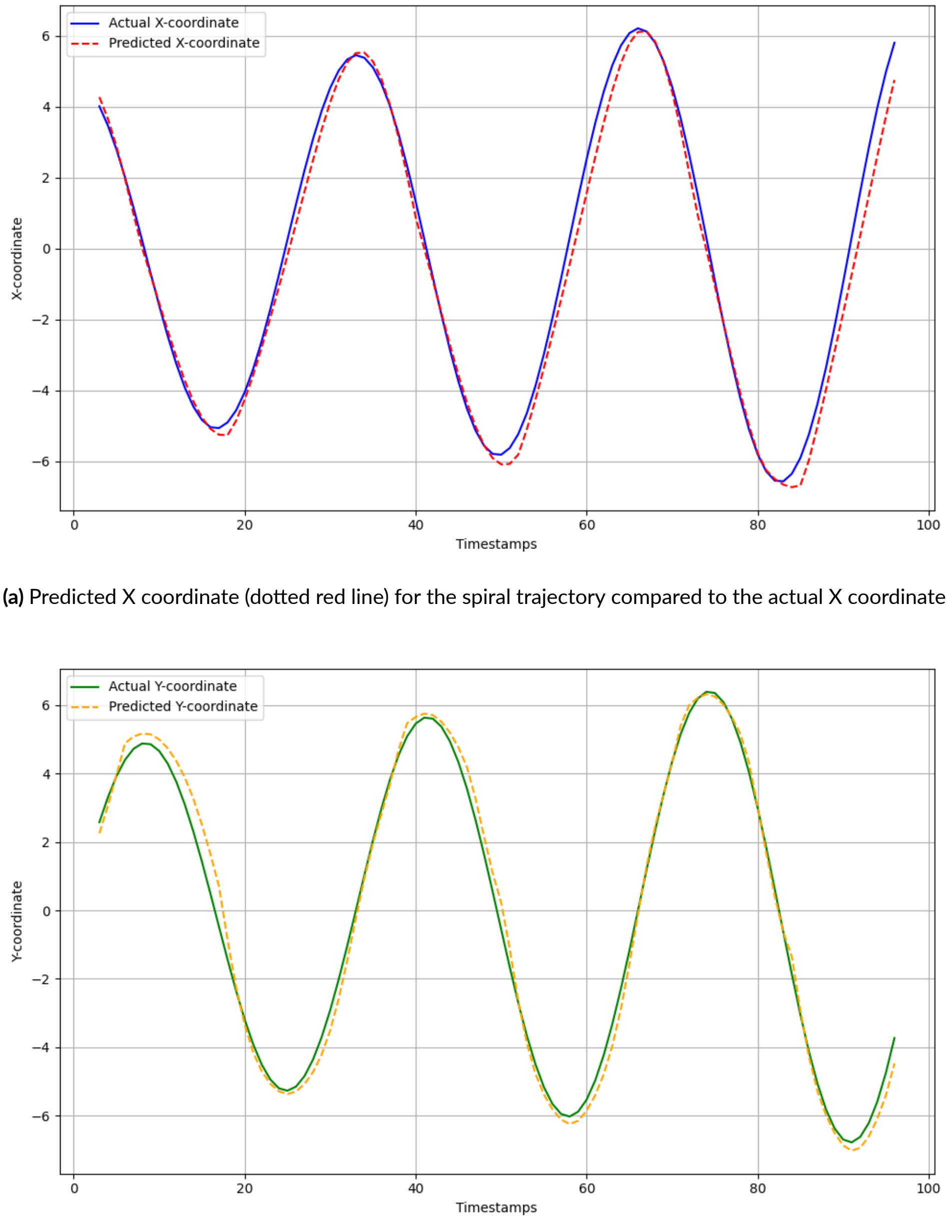
Examining the attack trajectory, it becomes apparent that the model's predictions exhibit slightly sparser results. This characteristic arises from the labeling scheme, where the predicted positions consist of being the positions 3 timestamps further from the current one. Consequently, the model, while providing accurate global predictions, may exhibit occasional local-



(a) Predicted X coordinate (dotted red line) for the straight trajectory compared to the actual X coordinate.

(b) Predicted Y coordinate (dotted yellow line) for the straight trajectory compared to the actual Y coordinate.

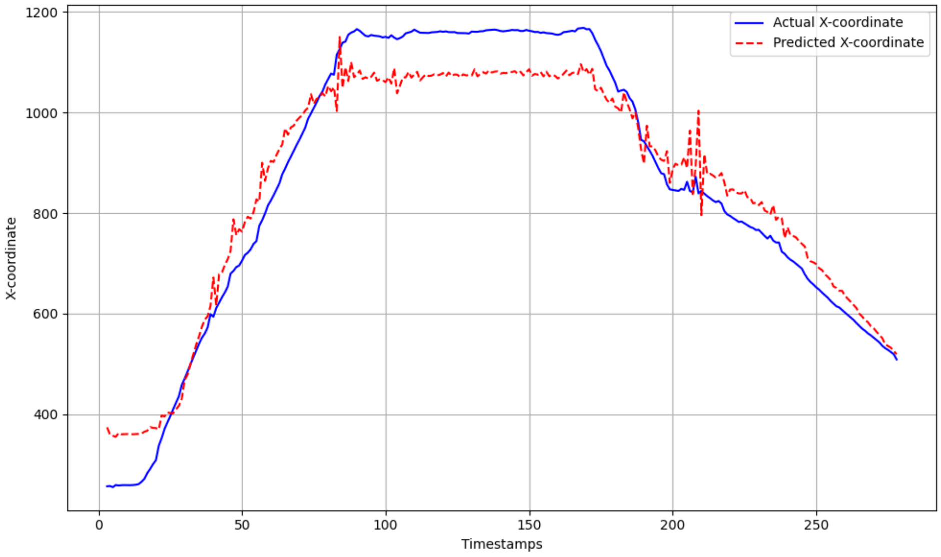
Figure 6.1: Simulated Linear Attack Trajectory Forecasting



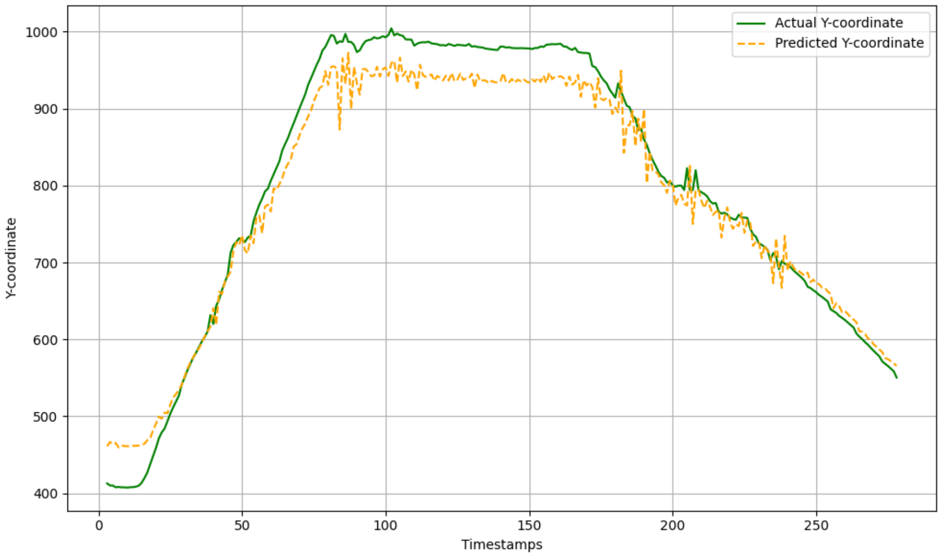
(a) Predicted X coordinate (dotted red line) for the spiral trajectory compared to the actual X coordinate.

(b) Predicted Y coordinate (dotted yellow line) for the spiral trajectory compared to the actual Y coordinate.

Figure 6.2: Simulated Spiral Attack Trajectory Forecasting

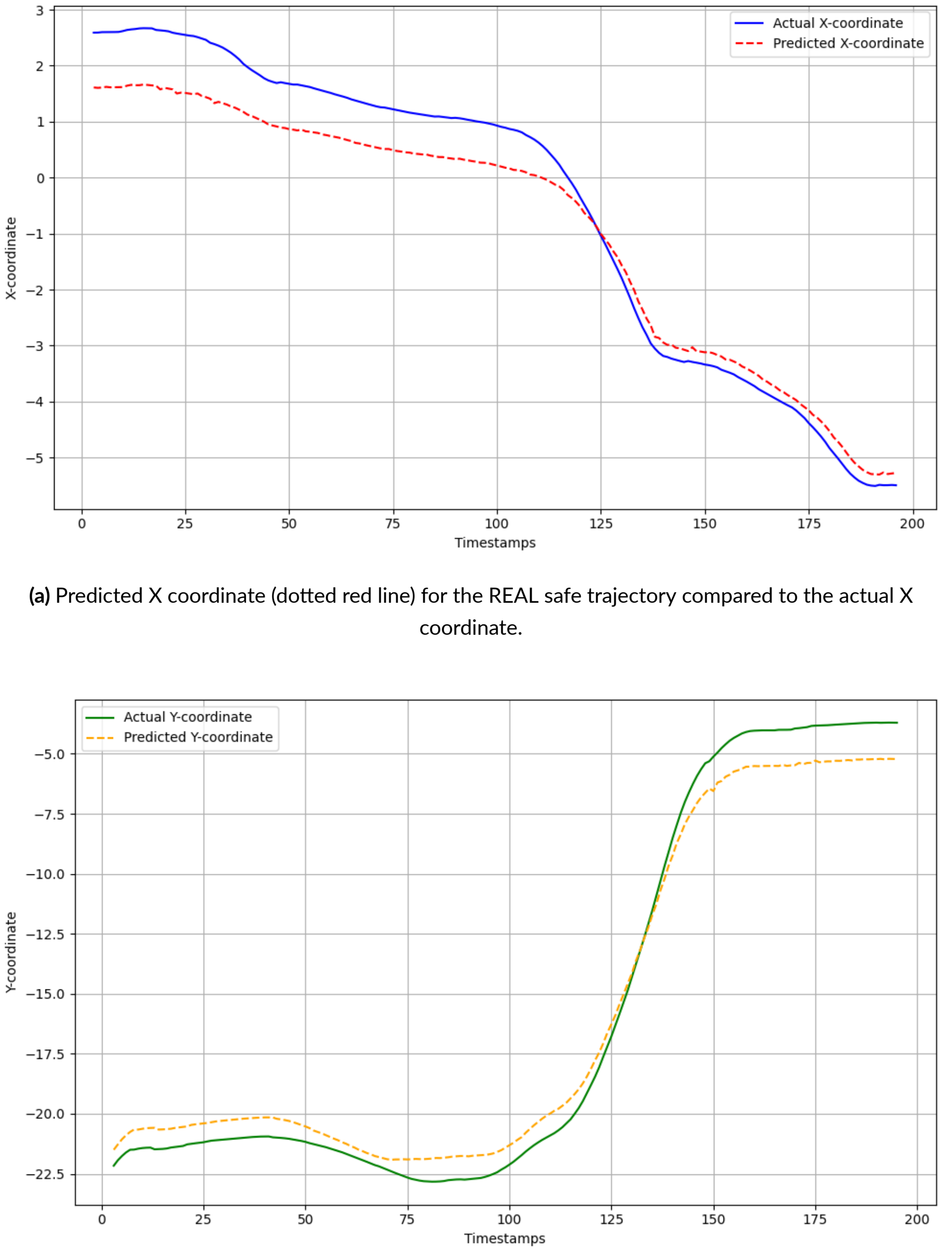


(a) Predicted X coordinate (dotted red line) for the REAL attack trajectory compared to the actual X coordinate.



(b) Predicted Y coordinate (dotted yellow line) for the REAL attack trajectory compared to the actual Y coordinate.

Figure 6.3: Real Data Attack Trajectory Forecasting



(a) Predicted X coordinate (dotted red line) for the REAL safe trajectory compared to the actual X coordinate.

(b) Predicted Y coordinate (dotted yellow line) for the REAL safe trajectory compared to the actual Y coordinate.

Figure 6.4: Real Data Innocuous Trajectory Forecasting

ized variances, manifested as brief impulsive peaks, particularly when the trajectory experiences subtle smoothness fluctuations over time.

6.4 Real-time forecasting

In response to the promising outcomes witnessed in real-data forecasting, an additional test was conducted to evaluate the model's performance in real-time trajectory classification. Trajectories were incrementally provided to the model, one timestamp at a time, simulating what radar-recorded data would appear as in a realistic scenario. The primary objective was to compute the model's efficiency in real-time classification, specifically in discerning whether trajectories represent attacks or innocuous behavior. The determination of attack or innocuous classification hinged on the Mean Absolute Error (MAE). The classification depends on whether the MAE surpasses a manually set threshold, established based on training results. This evaluation culminated in the generation of Table 6.1. An impressive outcome emerged as every trajectory identified as an attack exhibited classification within a range of 5 to 23 timestamps, with an average of 10 timestamps required. Conversely, the sole innocuous trajectory subjected to LSTM scrutiny consistently maintained its classification as safe from the initial timestamp to the last, further highlighting the robustness of the model's classification capabilities. Figure 6.5 represents how the variation in positioning of the UAV would be classified: the initial 5 positions follow a pattern recognized by the model: this lowers the Mean Average Value until it gets below the chosen threshold, triggering the **attack** scenario (reported in color red).

Track ID	Attack Detection Time
track_251	17 timestamps
track_268	8 timestamps
track_311	5 timestamps
track_404	6 timestamps
track_443	5 timestamps
track_447	6 timestamps
track_454	7 timestamps
track_500	23 timestamps
track_0	not detected

Table 6.1: Number of timestamps required to classify as attack each real data trajectory

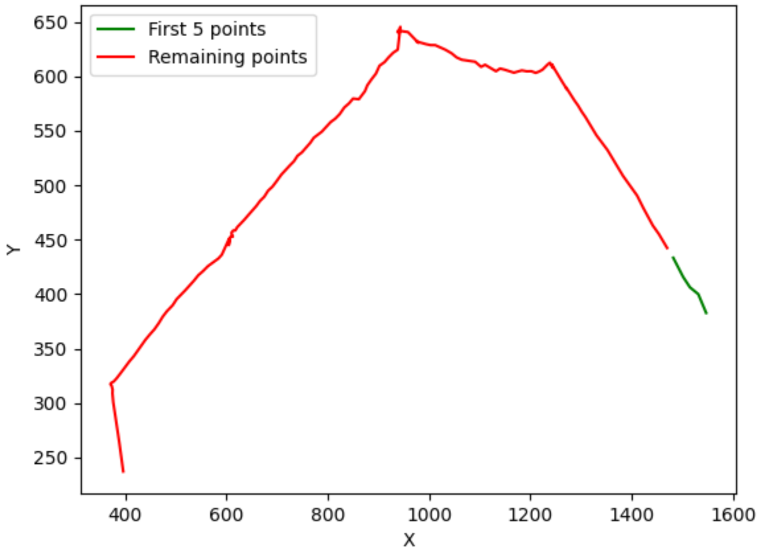


Figure 6.5: A graphic visualization of how the radar would classify this specific trajectory: the first 5 positions lower the MAE error to allow to predict this trajectory as an attack.

7 Conclusion

In this research, the endeavor to address the challenge of UAV behavioral identification based solely on external observation data has been undertaken. A comprehensive exploration of behavioral classification challenges and an examination of the current trajectory forecasting and classification landscape have been presented.

To navigate this intricate problem, a dedicated deep learning model has been meticulously developed. The model undergoes rigorous training and testing phases leveraging synthetically generated data tailored explicitly for this research. These synthetic trajectories are intentionally diverse, aiming to encompass various potential attack scenarios and optimize the model's threat detection capabilities. Encouragingly, the results showcase the feasibility of this analytical approach, revealing minimal errors in trajectory forecasting.

The model's effectiveness is further scrutinized through the evaluation on a set of real-world trajectories sourced from military data. This evaluation encompasses trajectories outlining both attack vectors and benign paths, providing valuable insights into the model's real-world applicability.

The identical trajectories were employed for real-time classification, simulating the live data stream a radar might receive. Impressively, the model adeptly classified all the attack trajectories as intended, accomplishing this task within a matter of seconds.

8 Future Works

This research marks one of the first efforts in classifying UAV behavior exclusively based on radar-captured information, devoid of direct access to the drone itself. The nascent nature of this field underscores its significant unexplored terrain, with anticipated growing importance, particularly in light of current geopolitical developments. Notably, the scarcity of features composing the trajectories generated in this study is attributable to the inherent challenges in crafting accurate attack paths with limited references. The shortage of real data trajectories further compounds this challenge, emphasizing the need for continued exploration in this domain.

Future investigations could leverage Software in the Loop (SITL) simulations, utilizing tools such as Gazebo [51] or Matlab [52], to expedite data collection and ensure the viability of deep learning models. Additionally, a complementary approach involves accessing a comprehensive military dataset containing examples of both attack and innocuous trajectories. Such a dataset could serve as inspiration for research teams to refine simulations, though challenges persist in precisely defining what constitutes an ”attack”. Moreover, platforms like Kaggle host datasets containing drone flight data. This data, derived from readings of individual sensors such as accelerometers, Inertial Measurement Units (IMUs), and others, can be leveraged to augment the dataset with additional innocuous trajectories [53]. Integrating this information into the research dataset offers an opportunity to enhance the diversity and realism of the innocuous trajectory samples, contributing to a more robust analysis and classification model.

Machine learning methodologies, such as Support Vector Machines or Decision Forests, present viable alternatives that circumvent the need for extensive data as shown in [54] and [55] respectively. These models possess a high likelihood of accurate classification. Furthermore, probabilistic approaches could enhance development by accounting for potential variations in pilot behavior [56, 57].

An additional avenue for future work involves an exploration of the field of driver classification, which has seen increased research efforts amid the growing interest and investments in autonomous vehicles. For instance, Bouhoute et al. [36] employ graphical models to represent driver behavior, employing probabilistic hybrid automata and labeled directed graphs. This methodology is notable for utilizing transitions of states to define behavior, providing a structured approach to behavior representation. In the context of drones, these states could serve as a simplistic means of depicting potential threats. The continual tracking of behavior through different states post-detection enables the identification of significant behavioral changes, facilitating the classification of drones as potentially malicious.

References

[1] S. M. Shavarani, M. G. Nejad, F. Rismanchian, and G. Izbirak, "Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of amazon prime air in the city of san francisco," *The International Journal of Advanced Manufacturing Technology*, vol. 95, pp. 3141–3153, 2018.

[2] B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster," *Computer Communications*, vol. 156, pp. 1–10, 2020.

[3] S. K. Gupta and D. P. Shukla, "Application of drone for landslide mapping, dimension estimation and its 3d reconstruction," *Journal of the Indian Society of Remote Sensing*, vol. 46, pp. 903–914, 2018.

[4] S. J. Kim and G. J. Lim, "Drone-aided border surveillance with an electrification line battery charging system," *Journal of Intelligent & Robotic Systems*, vol. 92, pp. 657–670, 2018.

[5] V. Chang, P. Chundury, and M. Chetty, "Spiders in the sky: User perceptions of drones, privacy, and security," in *Proceedings of the 2017 CHI conference on human factors in computing systems*, 2017, pp. 6765–6776.

[6] P. B. Johnston and A. K. Sarbahi, "The impact of us drone strikes on terrorism in pakistan," *International Studies Quarterly*, vol. 60, no. 2, pp. 203–219, 2016.

[7] F. A. Administration, "Remote identification of drones," *https://www.ecfr.gov/current/title-14/chapter-I/subchapter-F/part-89*, 2023.

[8] C. Aker and S. Kalkan, "Using deep networks for drone detection," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[9] S. Singha and B. Aydin, "Automated drone detection using yolov4," *Drones*, vol. 5, no. 3, p. 95, 2021.

[10] A. Coluccia, A. Fascista, A. Schumann, L. Sommer, A. Dimou, D. Zarpalas, F. C. Akyon, O. Eryuksel, K. A. Ozfuttu, S. O. Altinuc *et al.*, "Drone-vs-bird detection challenge at ieee avss2021," in *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS).* IEEE, 2021, pp. 1–8.

[11] J. Wojtanowski, M. Zygmunt, T. Drozd, M. Jakubaszek, M. Życzkowski, and M. Muzal, "Distinguishing drones from birds in a uav searching laser scanner based on echo depolarization measurement," *Sensors*, vol. 21, no. 16, p. 5597, 2021.

[12] I. Kontopoulos, A. Makris, D. Zissis, and K. Tserpes, "A computer vision approach for trajectory classification," in *2021 22nd IEEE International Conference on Mobile Data Management (MDM).* IEEE, 2021, pp. 163–168.

[13] J.-G. Lee, J. Han, X. Li, and H. Gonzalez, "Traclass: trajectory classification using hierarchical region-based and trajectory-based clustering," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 1081–1094, 2008.

[14] C. L. da Silva, L. M. Petry, and V. Bogorny, "A survey and comparison of trajectory classification methods," in *2019 8th Brazilian Conference on Intelligent Systems (BRACIS).* IEEE, 2019, pp. 788–793.

[15] I. Kontopoulos, A. Makris, and K. Tserpes, "A deep learning streaming methodology for trajectory classification," *ISPRS International Journal of Geo-Information*, vol. 10, no. 4, p. 250, 2021.

[16] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16.* Springer, 2020, pp. 683–700.

[17] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," in *2020 25th international conference on pattern recognition (ICPR).* IEEE, 2021, pp. 10 335–10 342.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[19] P. Kothari, S. Kreiss, and A. Alahi, "Human trajectory forecasting in crowds: A deep learning perspective," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7386–7400, 2021.

[20] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *The International Journal of Robotics Research*, vol. 39, no. 8, pp. 895–935, 2020.

[21] X. Zhang, X. Fu, Z. Xiao, H. Xu, and Z. Qin, "Vessel trajectory prediction in maritime transportation: Current approaches and beyond," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[22] K. A. Sørensen, P. Heiselberg, and H. Heiselberg, "Probabilistic maritime trajectory prediction in complex scenarios using deep learning," *Sensors*, vol. 22, no. 5, p. 2058, 2022.

[23] Z. Xiao, L. Ponnambalam, X. Fu, and W. Zhang, "Maritime traffic probabilistic forecasting based on vessels' waterway patterns and motion behaviors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 3122–3134, 2017.

[24] F. Altché and A. de La Fortelle, "An lstm network for highway trajectory prediction," in *2017 IEEE 20th international conference on intelligent transportation systems (ITSC)*. IEEE, 2017, pp. 353–359.

[25] D. Gutierrez-Galan, J. P. Dominguez-Morales, E. Cerezuela-Escudero, A. Rios-Navarro, R. Tapiador-Morales, M. Rivas-Perez, M. Dominguez-Morales, A. Jimenez-Fernandez, and A. Linares-Barranco, "Embedded neural network for real-time animal behavior classification," *Neurocomputing*, vol. 272, pp. 17–26, 2018.

[26] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Driver behavior classification at intersections and validation on large naturalistic data set," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 724–736, 2012.

[27] P. Riley and M. Veloso, "On behavior classification in adversarial environments," in *Distributed autonomous robotic systems 4*. Springer, 2000, pp. 371–380.

[28] Y. Zhou, W. Daamen, T. Vellinga, and S. P. Hoogendoorn, "Ship classification based on ship behavior clustering from ais data," *Ocean Engineering*, vol. 175, pp. 176–187, 2019.

[29] X. Chang, C. Yang, J. Wu, X. Shi, and Z. Shi, "A surveillance system for drone localization and tracking using acoustic arrays," in *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. IEEE, 2018, pp. 573–577.

[30] A. Sedunov, A. Sutin, N. Sedunov, H. Salloum, A. Yakubovskiy, and D. Masters, "Passive acoustic system for tracking low-flying aircraft," *IET Radar, Sonar & Navigation*, vol. 10, no. 9, pp. 1561–1568, 2016.

[31] G. Lykou, D. Moustakas, and D. Gritzalis, "Defending airports from uas: A survey on cyber-attacks and counter-drone sensing technologies," *Sensors*, vol. 20, no. 12, p. 3537, 2020.

[32] M. M. Azari, H. Sallouha, A. Chiumento, S. Rajendran, E. Vinogradov, and S. Pollin, "Key technologies and system trade-offs for detection and localization of amateur drones," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 51–57, 2018.

[33] Y. Chang, "A drone iff and tracking algorithm with the relay drone and the beacon system," *The Eurasia Proceedings of Science Technology Engineering and Mathematics*, vol. 7, pp. 199–203, 2019.

[34] T. Multerer, A. Ganis, U. Prechtel, E. Miralles, A. Meusling, J. Mietzner, M. Vossiek, M. Loghi, and V. Ziegler, "Low-cost jamming system against small drones using a 3d mimo radar based tracking," in *2017 European radar conference (EURAD)*. IEEE, 2017, pp. 299–302.

[35] 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, 2016.

[36] A. Bouhoute, R. Oucheikh, K. Boubouh, and I. Berrada, "Advanced driving behavior analytics for an improved safety assessment and driver fingerprinting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2171–2184, 2018.

[37] D. Solomitckii, M. Gapeyenko, V. Semkin, S. Andreev, and Y. Koucheryavy, "Technologies for efficient amateur drone detection in 5g millimeter-wave cellular infrastructure," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 43–50, 2018.

[38] S. J. Fox, "The 'risk'of disruptive technology today (a case study of aviation–enter the drone)," *Technology in Society*, vol. 62, p. 101304, 2020.

[39] K. Thomasen, "Beyond airspace safety: A feminist perspective on drone privacy regulation," 2017.

[40] B. Fiegel, "Narco-drones: a new way to transport drugs," *Criminal Drone Evolution: Cartel Weaponization of Aerial IEDS*, p. 7, 2021.

[41] N. Poljak, M. Ševo, and I. Livaja, "Security and privacy in an it context—a low-cost wids employed against mitm attacks (concept)," in *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 2016, pp. 1614–1617.

[42] T. Reed, J. Geis, and S. Dietrich, "{SkyNET}: A {3G-Enabled} mobile attack drone and stealth botmaster," in *5th USENIX Workshop on Offensive Technologies (WOOT 11)*, 2011.

[43] L. Bodell, "Frankfurt airport halts flights over unidentified drone," *https://simpleflying.com/frankfurt-airport-halts-flights-unidentified-drone/*, 2023.

[44] B. Systems, "Autonomous real-time ground ubiquitous surveillance imaging system (argus-is)," *https://www.baesystems.com/en-us/product/autonomous-realtime-ground-ubiquitous-surveillance-imaging-system-argusis*, 2013.

[45] J. Bian, D. Tian, Y. Tang, and D. Tao, "Trajectory data classification: A review," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 4, pp. 1–34, 2019.

[46] X. Shi and D.-Y. Yeung, "Machine learning for spatiotemporal sequence forecasting: A survey," *arXiv preprint arXiv:1808.06865*, 2018.

[47] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[48] K. Smagulova and A. P. James, "A survey on lstm memristive neural network architectures and applications," *The European Physical Journal Special Topics*, vol. 228, no. 10, pp. 2313–2324, 2019.

[49] Z. Shi, M. Xu, Q. Pan, B. Yan, and H. Zhang, "Lstm-based flight trajectory prediction," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–8.

[50] Thales, "Quire ground surveillance radar," *https://www.thalesgroup.com/en/squire-ground-surveillance-radar*, 2023.

[51] O. Robotics, "Gazebo," *https://gazebosim.org/home*, 2023.

[52] MathWorks, "What is drone simulation?" *https://www.mathworks.com/discovery/drone-simulation.html*, 2023.

[53] M. Street, "Drone identification and tracking," 2021. [Online]. Available: https://kaggle.com/competitions/icmcis-drone-tracking

[54] R. Saini, P. Kumar, P. P. Roy, and D. P. Dogra, "An efficient approach for trajectory classification using fcm and svm," in *2017 IEEE Region 10 Symposium (TENSYMP)*. IEEE, 2017, pp. 1–4.

[55] J. Tao and R. Klette, "Integrated pedestrian and direction classification using a random decision forest," in *Proceedings of the IEEE international conference on computer vision workshops*, 2013, pp. 230–237.

[56] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction with gaussian mixture models," in *2012 IEEE Intelligent vehicles symposium*. IEEE, 2012, pp. 141–146.

[57] S. Klingelschmitt and J. Eggert, "Using context information and probabilistic classification for making extended long-term trajectory predictions," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015, pp. 705–711.

Acknowledgments

This thesis has been the result of a work that lasted 9 months, from the first time I addressed this issue until now, when I'm writing this last part. These 270 days have been spent either in the LabTA room in Torre Archimede while also preparing projects and studying for the exams, at home, or in a city 235km away from where the courses and lectures took place. It has been an incredible journey, and I can't feel but lucky for having lived it.

I would like to thank my family first, without whom I wouldn't have been able to pursue this Master's Degree and meet all the amazing people who have marked this wonderful experience. A huge ”Thank You” also goes to my supervisor, Professor Alessandro Brighente, for his constant availability and attentive commitment to helping me during this study with new ideas and a never-ending curiosity to pursue something new.

Thank you to my old roommates, for their patience while watching me cook French toasts for the eleventh time that week only, and for having given me a flash of what it would look like to live with someone who loves plants, and a risotto-making god.

A heartfelt thanks must be addressed to ”all my fellas” who followed this path with me: I won't name you all, we've seen that I tend to leave people out, but shoutout to the OSINT group for keeping me entertained for the last 24 months, and thank you ”Cyber-Boys”. We really have to change that name.

Thank you to everyone who has been supporting me from the very beginning, like 2019 beginning. You mean a lot.

Finally, thanks to everyone who crossed my path in these years. If I'm here, it's also because of you.