TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Anna Mandrenko 223691IVCM

Analyzing Drone Flight Logs for Forensic Behaviour Reconstruction

Master's Thesis

Supervisor: Matthew James Sorell
Adjunct Professor

Tallinn 2024

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Anna Mandrenko 223691IVCM

DROONI LENNUPÄEVIKUTE LOGIANALÜÜS DIGITAALSE KÄITUMISEKSPERTIIS REKONSTRUEERIMISEKS

Magistritöö

Juhendaja: Matthew James Sorell
Adjunct Professor

Tallinn 2024

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Anna Mandrenko

11.05.2024

Abstract

Drones, or Unmanned Aerial Vehicles (UAVs), are increasingly used in illegal activities. Consequently, law enforcement agencies face the challenge of investigating and prosecuting crimes involving drones, necessitating the advancement of drone digital forensic science. Often, drones are utilized for the illicit transportation of payloads. When a crashed drone is discovered, with suspicions of its involvement in criminal activity, investigators require insights into its past operations. This study aims to develop analytical techniques enabling forensic experts to reconstruct drone behavior and gather operational data for investigative purposes. We propose three hypotheses regarding drone behavior and test them empirically. Firstly, we hypothesize that motor-related metrics in flight logs exhibit a linear relationship with payload weight, facilitating payload weight estimation. Secondly, we suggest that abrupt changes in metrics can indicate the moment of payload release during flight. Lastly, in the absence of GPS data, alternative metrics within flight logs may enable estimation of the drone's trajectory and the point of payload release. We conduct a series of experiments to test these hypotheses, collect and analyze data, and propose proof-of-concept analytical methods.

The thesis is written in English and is 42 pages long, including 6 chapters, 11 figures and 6 tables.

List of Abbreviations and Terms

UAV	Unmanned Aerial Vehicle
DVS	Downward Vision System
CNN	Convolutional Neural Network
ML	Machine Learning
GPS	Global Positioning System

Table of Contents

List of Figures

List of Tables

1. Introduction

Unmanned aerial vehicles (UAVs), commonly known as drones, are aerial devices that establish a connection with a ground controller through radio frequency and are controlled by a human operator. Initially, drones were primarily developed for military applications alone [1]. However, their usage has experienced significant growth in the commercial and private sectors. In the United States alone, sales of small drones have exceeded 600,000 devices [2]. The commercial drone industry is projected to grow from 30.6 billion US dollars in 2022 to 58.6 billion US dollars in 2030 [2], and some estimates are even more optimistic.

Commercial drones are finding an increasing number of applications in various domains. They have been utilized for purposes such as delivering goods, conducting firefighting and rescue operations, facilitating video production, supporting investigations, and warfare, among many other uses. UAVs have also been increasingly used for illegal purposes, such as transporting drugs or weapons over the border and into prisons, breaching privacy, etc. The most notable of such events happened at Gatwick airport when unauthorized flights disrupted airport activity for 3 days, affecting 1000 flights [3].

When a criminal act involving a drone occurs, investigators might be interested in answering questions related to the device. In such cases, a digital forensic investigation may be performed by an expert [4]. A drone forensic expert is expected to answer the investigator's questions and acquire evidence. To facilitate this, a corpus of knowledge about the evidence that might be recovered from a drone, as well as a standardized framework for the forensics process, must be developed. This need has given rise to an area of drone forensics research. Being a sub-fraction of device forensics, it is still a broad area and fairly under-researched [5].

This thesis is centered around the hypothesis regarding the possibility of restoring some of the drone's past behavior based on the data available in the flight logs. First, in the Section 1.4 the research hypothesis are stated, then to confirm or to reject the hypothesis, a series of experiments has been conducted, and the resulting data has been analyzed using various methods, including Machine Learning models. The results obtained in this work show the possibility of obtaining valuable operational evidence that might assist investigators in responding to the crime.

This thesis is organized as follows: The Introduction presents relevant background and other information about the design of this study. Next, the relevant research is discussed in Chapter 2, followed by the experiment design and methodology in Chapter 3. Chapters 4 and 5 contain data analysis and results. Chapter 6 lists findings, conclusions and outlines directions for the future analysis.

1.1 Background

1.1.1 Drones

Drones come in various shapes and sizes, but they typically consist of two distinct parts: the ground controller and the drone body. The ground controller is used by a human operator to send control commands to the drone body via radio frequency. The drone body, in turn, includes motors, rotors/propellers/wings, speed controllers, a protective case, and a radio receiver. These are essential components, but most drones also carry additional payloads depending on their intended use. Such payloads commonly include a camera, GPS receiver, onboard computer, various sensors, lights, payload-dropping systems, and more.

Every drone includes some form of software system onboard that interprets user input received through the transmitter to control the motors. However, most contemporary commercial drones contain far more sophisticated solutions capable of storing flight data in memory, transmitting video streams to the ground, and assisting pilots in controlling UAVs. Depending on the functionality of the flight-controlling software, it is common to divide it into two groups: flight management software and ground control software [4]. This distinction is important because flight control systems mainly assist the human operator in stabilizing, landing, etc., and produce minimal logging during the flight [6]. In contrast, ground control systems are used to control predetermined navigation and plan flight schedules. The latter types of systems have enhanced logging capabilities to the ground station, which is especially useful for forensic experts in extracting evidence [6].

1.1.2 Forensic examination of a drone

In cases where a UAV or drone is involved in illegal activity, a forensic investigation of the device may be necessary. Most forensic frameworks divide the forensic examination of the drone into two subcategories: hardware or physical forensics and software or digital forensics [1]. Hardware drone forensics may involve collecting wet evidence, examining payloads, and establishing serial numbers, among other essential steps. These actions are often a prerequisite for a digital examination of the drone but are beyond the scope of this work.

The digital software of the drone includes network traffic analysis, which encompasses the analysis of drone-to-controller communication messages, system log analysis, file systems analysis, and camera recordings [1]. Evidence obtained from such analyses can assist investigators in answering questions, such as determining the time of the incident or identifying the device owner.

In cases where investigators need to find physical evidence related to the incident, reconstructing the drone's activities may help narrow the location of the search. In this work, we will attempt to contribute to the development of the data analysis method to reconstruct drone behavior. Such analysis may provide operational information to the investigator.

1.1.3 Drone as the source of evidence

As with any digital device, a UAV produces a digital footprint as a result of its normal operation. This footprint includes data intentionally generated by users, such as photos and videos stored on external memory, as well as unintentionally generated data, such as operational logs and service files. Data on drones that have the potential to serve as evidence can be categorized by storage medium and type.

According to the INTERPOL Drone Incident Response Framework [4], several data storage mediums can be found on a drone:

- On-board Data Storage;
- Removable Storage Devices;
- Mobile Devices and Applications;
- Remote Controllers;
- Ground Stations;
- Cloud-Based Data Platforms;
- Network Packet Data.

It's important to note that not all of these data storage mediums will be present in every model of the drone. Additionally, some data may be inaccessible due to encryption or may require sophisticated and potentially destructive techniques, such as "chip-off" acquisition [7]. All of the listed data storage mediums may contain some of the following types of data [4]:

- Audio Visual Content;
- Flight Schedules;
- Other Payload/Sensor-Created Content;
- Automated Usage Logs.

All these types of data can play a role in reconstructing drone behavior. Yet the most critical source is the automated system and flight logs generated by essential drone components, the operating system, and other available sensors. Modern drones capable of automated flights record every action of the drone and different telemetry. In this thesis, only the flight logs containing various metrics recorded during the flight are analyzed. These flight logs are stored on the drone body's storage in DAT files. No other types of logs or sources of logs are used in this work.

1.2 Motivation

UAVs have been increasingly used for illegal purposes, such as transporting drugs or weapons over the border and into prisons, breaching privacy, etc. Drone-related crime is increasing, as highlighted in a yearly report by DroneSec, a threat intelligence company, summarizing globally notable events. In 2022, over two and a half thousand drone-related incidents were recorded, marking a 60% increase from the previous year [8]. A notable 50% increase has happened in incidents related to smuggling illegal goods to the prisons or over the border [8]. This surge in incidents underscores the relevance of digital forensic investigations in the realm of drones.

Drones are increasingly used to smuggle weapons into prisons and other guarded areas, to transport drugs across borders, or to carry explosives or supplies on the battlefield. In scenarios where such a drone is discovered crushed or taken down with a rifle, it might be crucial to gain operational information from it to take action. In guarded areas like prisons or on the battlefield, GPS readings might often be unavailable or jammed; therefore, recovering the drone's path using other data might be crucial to begin the search for an operator. If the drone carried a dangerous or illegal payload, it might also be necessary to estimate a drop place or payload weight to provide the search team with an estimate of its size.

This research will concentrate on analyzing various variables recorded by drones during flights and their potential to help investigators in gaining operational information to carry on the investigation. The findings aim to contribute to the development of new analytical methods for the investigation of drone-related crimes.

1.3 Scope and limitations

Drone forensics, despite being a relatively new area of research, has many sub-directions. To enhance results, this work will only focus on a subset of problems, assuming that other issues are solved within the scope of this study, which might not always be the case.

Firstly, this work will only explore the topic of drone behavior reconstruction based on data collected from the flight logs. No other evidence, such as CCTVs, physical evidence, hardware, or operational memory data, logs from other sources like controller, will be used for the reconstruction. Secondly, the study heavily relies on the assumption that drones collect enough telemetry in flight logs, and that this data is available for analysis. This might not always be the case, as malicious users, especially cross-border smugglers, may use drones crafted illegally from various components. It cannot be predicted what kind of data these UAVs store during the flight. Additionally, as stated in [6], commercial drones without autopilot functionality usually do not produce detailed telemetry during the flight. Fortunately, most popular drone models do support autopilot functionality in various forms. Finally, there is a current market trend for producers to implement encryption of flight logs. While there are decryptors created by the community for some models [9], for others, data might not be available for analysis at the moment [7].

As the data acquisition process is not the main goal of this study, a drone for experiments has been chosen carefully to ensure flight log data can be decrypted from it. All experiments in this work will be conducted using the DJI Phantom 4 drone, as it is both possible to analyze and is second the most widely used drone for malicious purposes [8].

Another key assumption of the study is that results obtained by analyzing flight logs from the DJI Phantom 4 will be somewhat reproducible on other drone models. This assumes that other DJI models and drones from different producers record comparable telemetry variables, which is not known at this point. Generalizing findings and testing approaches on other drone models are outside the scope and will be a matter of future research.

The experiment design outlined in Chapter 3 includes the data acquisition process. This process might not be forensically sound enough to produce evidence admissible in court. As of the moment of writing, there are no standard principles and legislation for the drone forensic acquisition process [5]. Ensuring that the acquisition methods used preserve the integrity of the data lies outside the scope of this research.

Some methods used in Chapters 5 or 4 may lack robustness and might not perform well on data collected in various environments. As this work is centered around experiments to either prove or dismiss a hypothesis, they serve as a proof of concept to demonstrate the principal possibility of solving the task. Mentioned Chapters include an analysis of the future direction of the research to improve robustness of the proposed methods.

1.4 Hypothesis

This work will explore the possibility of restoring drone behavior from the information stored in the flight logs. Particularly, it will focus on payload detection and path restoration without available or reliable GPS readings. The underlying rationale for this research lies in the fact that drones continuously record numerous metrics, including motor performance, sensor data, and controller inputs. Since payloads alter the weight of the drone, they are likely to alter motor-related parameters. Additionally, drone movement requires adjustments in motor performance, which in turn impact sensor readings. Therefore, it is theoretically possible to reconstruct certain flight events based on these recorded data points.

Since this is a wide field barely researched, several hypotheses will be stated in this section. In subsequent chapters 4 and 5, the analysis will be conducted to either provide evidence to prove the hypotheses or dismiss them. Chapter 6 reviews these hypotheses again and draws a final conclusion based on the results received.

The hypotheses this work centers around are the following:

1. There is a linear relationship between the attached payload weight and the values of the motor-related metrics in the flight logs.
2. If the payload was dropped mid-flight, it is possible to estimate a drop point from the motor related metrics in the flight logs.
3. The metrics recorded in flight logs, excluding GPS metrics, contain sufficient information to enable recreation of drone path.

To test the listed hypotheses, a set of experiments and data analysis will be conducted. Based on the results, a method to demonstrate the principal solvability of the problem will be proposed and tested.

2. Related works

The research on the forensics of small UAVs is in the emerging state [5]. The main challenge of this relatively young topic is the variety of different hardware, software components, and protocols that comprise a UAV [10]. There are still no standard baselines, principles, or legislation that cover the process of the forensic examination of the drone [5].

As stated in Section 1.3, this study is solely focused on analyzing flight logs stored in the memory of the drone's body. There are various types of logs recording different aspects of the drone's behavior, many of which may hold operational or evidential value. However, at the time of writing, there are only a small number of works centered on the analysis of drone flight logs. Most of those works focus on the acquisition and interpretation of the data retrieved from the flight logs and use them as evidence material as is. Such data may be used to establish the time of the incident, connect the controller to the drone or to the suspect, restore the drone path from the fully available GPS, and so on. This work attempts a slightly different approach.

In this work, we try to use available data to infer something that is not recorded on purpose by the drone software. There are two such things as stated in Section 1.4: detection of the payload and path restoration without GPS readings available. We discuss related works separatly in the following sections.

2.1 Drone path restoration

As mentioned, several studies have focused on drone path reconstruction, as documented in works such as [11, 12, 13, 14, 15]. Each of these studies delves into various aspects of GPS log analysis. Here, we will highlight a few examples to illustrate this point. Viswanathan and Baig provided an overview of available open-source drone forensics tools to retrieve GPS readings [15]. Al-Room et al. test the acquisition process for a wide variety of drones, including non-DJI models [13]. Their work is similar to [11], which also focuses on acquisition and GPS path reconstruction for several different drone families. For each drone they studied, both teams reconstructed the drone path using GPS logs. Salamh et al., in their work, proposed a framework and a tool to detect restricted area violations [12]. None of the works mentioned so far challenges the correctness and reliability of the available GPS readings. Cases where the GPS signal might be poor or unavailable are outside the scope of the listed research.

Finally, Prastya et al. is the only study centered around GPS logs that explores different scenarios of GPS availability [14]. They studied how the DJI Phantom 3 drone stores flight logs depending on the flying mode that is used. They also proposed a classification for GPS data quality: the signal is strong and available, the signal is unreliable, and there is no GPS signal.

To our knowledge, there is no existing work that attempts to reconstruct a drone's path from flight logs when GPS readings are unavailable. In this study, we will test if this is possible in principle.

2.2 Payload detection from the drone's flight log

There is only one work, to our knowledge, that attempts to infer a payload presence from the flight log readings. It is "Framework for Payload Detection Through Flight Log Analysis of DJI Mavic 2 Zoom" [16]. In this master's thesis, the author conducts a series of experiments by flying a DJI drone in a controlled environment (indoors) with or without a payload attached to the drone. Flight log records are then partially analyzed to detect the presence of the payload during the flight. The variables that have been analyzed include Revolutions Per Minute and Motor Amperage for each motor out of the 349 available variables. The results confirmed the hypothesis that it is possible to detect the presence of a payload through the analysis of motor-related variables [16].

Our planned research will primarily build upon the work by Ryan et al. [16], aiming to recreate their results and then broaden them by establishing a relationship between the drone's weight and metric readings.

3. Experiment Methodology

The central focus of this thesis revolves around a series of experiments involving a small UAV (hereinafter referred to as "the drone"). These experiments involve flying the drone with various payloads attached to it, followed by performing data acquisition on the drone to collect flight logs that describe its performance during flight.

This chapter describes the experimental methodology of those experiments and the materials used. Section 3.1 details the drone chosen for the experiments and the motivation behind the choice. It also addresses the limitations of this work stemming from the drone choice, which are later revisited in Chapter 6. Next, the other equipment used during the experiments is described in Section 3.2. Finally, the experimental methodology and various experiment scenarios are described in Section 3.3.

3.1 Drone Selection and Rationale

All experiments in this work will be conducted using the DJI Phantom 4 drone modified with the payload system. This particular model is chosen due to several reasons. Firstly, the DJI company dominates the market of consumer drones by being responsible for selling more than 72% of drones worldwide in all price groups [17]. This unsurprisingly makes DJI drones the most popular drones for various criminal activities. Particularly, the DJI Phantom 4 drone was the second most popular drone used for criminal activity in 2022 according to DRONESEC[8].

Secondly, the DJI company started to use encryption to protect flight logs from being read by unauthorized parties [18]. Decryption keys and acquisition methods are available for some models like Phantom 4, but still unavailable for many newer models, thus making them unsuitable choices for this work.

Following the considerations listed, the DJI Phantom 4 drone has been chosen to maximize the significance of the findings of this study.

There is no official documentation on how DJI OS records or calculates variables in the flight logs or how to interpret them.

3.2 Other equipment overview

Other equipment used in the experiments included a payload drop system, a set of payloads of different weights, and other items. The description and models of the items used are summarized in the table below.

Table 1. *Materials used for the experiments*

Item	Model	Description
Drone	DJI Phantom 4	Weight:1380g
Payload system	RCGEEK Payload Release Mechanism	Weight:152g
Set of payloads	Self-crafted	Material: PolymerClay, Weights:200g,50g,100g
Kitchen scales	GENOMSNITT	To weight payloads

The payload releasing mechanism was mounted to the drone's legs using a special platform (See Illustration 1). The payload release mechanism reacts to the light coming from the drone's navigation lights through the optic fiber cable. The navigation lights are controlled by the operator with the key on the remote controller. When the payload release mechanism is triggered, it releases the payloads attached to it. In the experiments that required no GPS signal, the drone was covered with a thick layer of foil on the top.

Figure 1. *An image of the Drone with the payload system attached*

3.3 Experiments design and scenarios

The majority of flights in the dataset were conducted at the coordinates 58.390000, 26.724918, situated in an open-air sport field. For each flight, the time and date, duration, weight of the payload, and description of the path were recorded.

There are four types of flights present in the data:

1. Steady flights where the drone ascends, hovers for 10 seconds, drops payload if attached, hovers for another 10 seconds, and lands.
2. Flights with complicated path routes with the payload attached. The payload is dropped somewhere in the middle of the flight.
3. Flights with complicated path routes with no payload attached.

All these experimental scenarios are explained in more detail in the corresponding chapters where results are analyzed. The whole dataset is available publicly in the CSV format. [1]

3.4 The data acquisition process

The data acquisition process of drone flight logs have been done following the process:

1. The drone is connected to a Mac computer using a USB-C adapter.
2. The drone is powered on.
3. The DJI Assistant application was used to connect the drone from the MacOS.
4. After connection is established, the drone's telemetry data are located through the MacOS Finder interface.
5. Using the Finder utility, the DAT files was copied from the drone storage to the computer.
6. Files has ben converted from the DAT format to .CSV format using DatCon software tool.

The list of software and corresponding versions that were used for the acquisition process:

1. DatCon tool, version 4.3.0
2. DJI Assistant 2 for Phantom, version 2.0.10
3. DJI GO App, version 4.3.60 (Android)
4. Aircraft OS, version 02.00.0810

[1] https://github.com/anisaAnya/droneDataset

5. Controller OS, version 1.9.3
6. macOS Ventura, version 13.6.2

Only flight logs that are stored on the drone body in the DAT format have been used. No logs from the controller or other sources are examined in this thesis. The dataset shared along with this work contains raw DAT files along with cropped/edited corresponding .csv files. One DAT file usually contains one flight, but there are exceptions. Sections 4 and 5 contain tables that list files along with their filenames that are used in the analysis. Those filenames can be used to locate corresponding files in the dataset.

4. Detecting the payload

The UAVs have become a means of committing a crime. Some of these crimes involve payloads to be carried and dropped by the drone. Such payloads may consist of drugs, explosives, illegal goods, and so on. In such scenarios, forensic analysis of the flight logs can become a valuable source of evidence. As explored in Chapter 2, there are many studies that analyze GPS records and reconstruct the path of the drone. However, we only found one study [16] that investigates payload detection on the drone.

We begin this section by replicating the results obtained by Ryan, J. in [16]. Then we proceed to explore how payloads of different weights affect motor-related variables in the readings. We study the relationships between different payload weights and the values of the variables in the flight logs. Next, we assess how the movement of the drone with the payload attached affects the flight logs. Lastly, we propose an algorithm to predict the time of the payload drop and estimate payload weight.

4.1 Recreating previous findings

To confirm whether it is possible to detect the presence of a payload through an increase in motor RPM and motor Amperage measurements in the flight logs, we performed a set of experiments resembling those conducted in [16]. The table 4.1 summarizes differences between experiments in our studies and the source.

Table 2. *Comparison of Criteria Between [16] and Our Experiment*

Criteria	**Ryan, J. [16]**	**Our Experiment**
Drone Model	DJI MAVIC 2 ZOOM	DJI Phantom 4
Environment	Controlled, inside	Outside, open-air sport facility
Wind	Absent	Might be present but speeds do not exceed 3m/s
GPS & Mode of operation	GPS is unavailable and DVS positioning	GPS available and GPS positioning used

Continues...

Table 2 – *Continues...*

Criteria	Ryan, J. [16]	Our Experiment
Scenario	Lift and hover until battery is low	Lift and hover for 10 sec, then payload drop, hover for 10 sec, and land

Using the equipment described in Chapter 3 for this test, we conducted 2 types of flights:

- **Lift and flight with payload**: Lift and fly across the field, then drop the payload, continue the flight, and then lower the drone.
- **Lift payload up and drop**: Lift the payload and hover in one place with neutral controller input for 10 seconds, then drop the payload, continue hovering, and then lower the drone.

The Table 4.1 all data samples and corresponding scenarios are summarized. Data samples can be localized in the dataset using the values from the "File name" column of this table.

Table 3. *List of Experiment Files*

#	File Name	Payload Weight	Scenario
1	FLY_05.03_1.csv	0g	Lift payload up and drop
2	FLY_05.03_2.csv	0g	Lift payload up and drop
3	FLY_08.03_2.csv	200g	Lift payload up and drop
4	FLY_08.03_3.csv	200g	Lift payload up and drop
5	FLY_08.03_4.csv	300g	Lift payload up and drop
6	FLY_08.03_5.csv	300g	Lift payload up and drop
7	FLY_08.03_6.csv	350g	Lift payload up and drop
8	FLY_02.05_2.csv	100g	Lift payload up and drop
9	FLY_08.03_7.csv	200g	Lift and flight with payload
10	FLY_08.03_8.csv	200g	Lift and flight with payload
11	FLY_08.03_9.csv	200g	Lift and flight with payload
12	FLY_02.05_1.csv	100g	Lift and flight with payload

To recreate the results obtained by Ryan J., the file #6, FLY_08.03_5.csv has been chosen randomly. In his work, Ryan selected the variable "Motor:RPM" (Revolutions Per Minute) to demonstrate the effect of the payload on the metrics. Since this column is unavailable in the Phantom 4 flight logs, we chose the most similar one: "Motor:Speed". This column, like every other motor-related column, exists in 4 variants: RFront and RBack, as well as LFront and LBack. This denotes 4 motors: left and right front and back motors. Figure 2 and 3 demonstrates how releasing a payload creates a clear step in the flight metrics same in our and Ryan J. results [16].

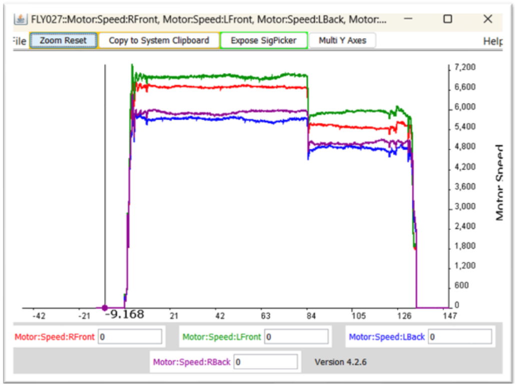

Figure 2. *Motor RPM value plot in [16]*

Figure 3. *Motor:Speed value plot in our data*

The release of payload impacts flight metrics by altering their average values and standard deviation, enabling automatic detection of any significant drops. Our study confirms the findings presented by Ryan J. in the [16].

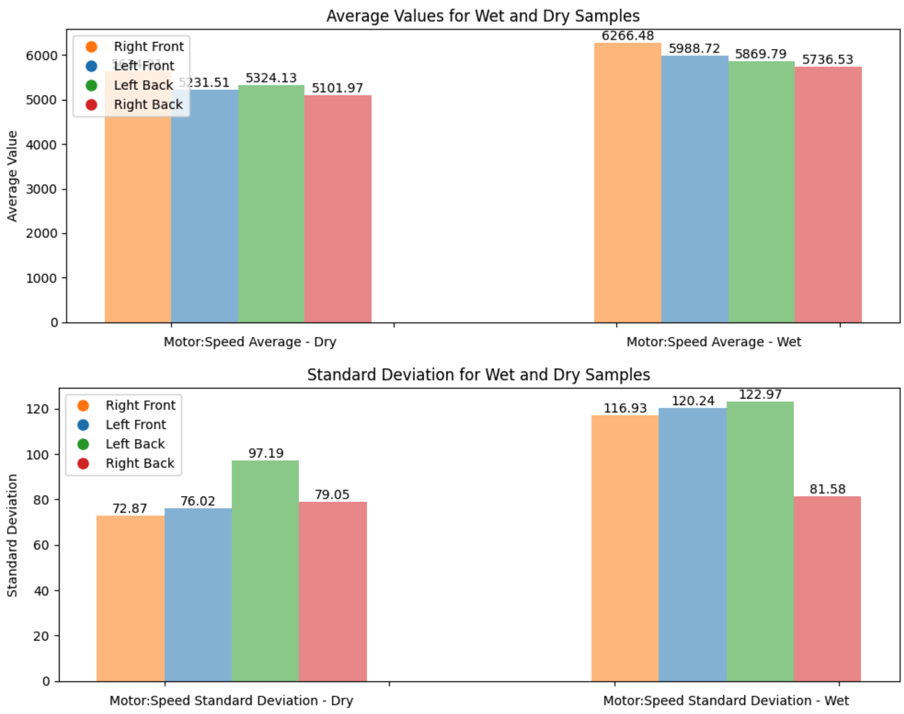

Figure 4. *Average and standard deviation values for the Motor:Speed variable, where "Wet" indicates a flight with the payload and "Dry" without it*

4.2 Payload weights and variables

In this subsection, we will assess which metrics contain information that may allow us to detect the presence of the payload at the moment of the drop. Next, we will analyze the relationships between payload weight and changes in the value of those metrics. There are 9 columns related to the motor performance in the flight logs:

- Motor:Speed
- Motor:EscTemp
- Motor:PPMrecv
- Motor:PPMsend
- Motor:V_out
- Motor:Volts
- Motor:Current
- Motor:Status
- MotorCtrl:PWM

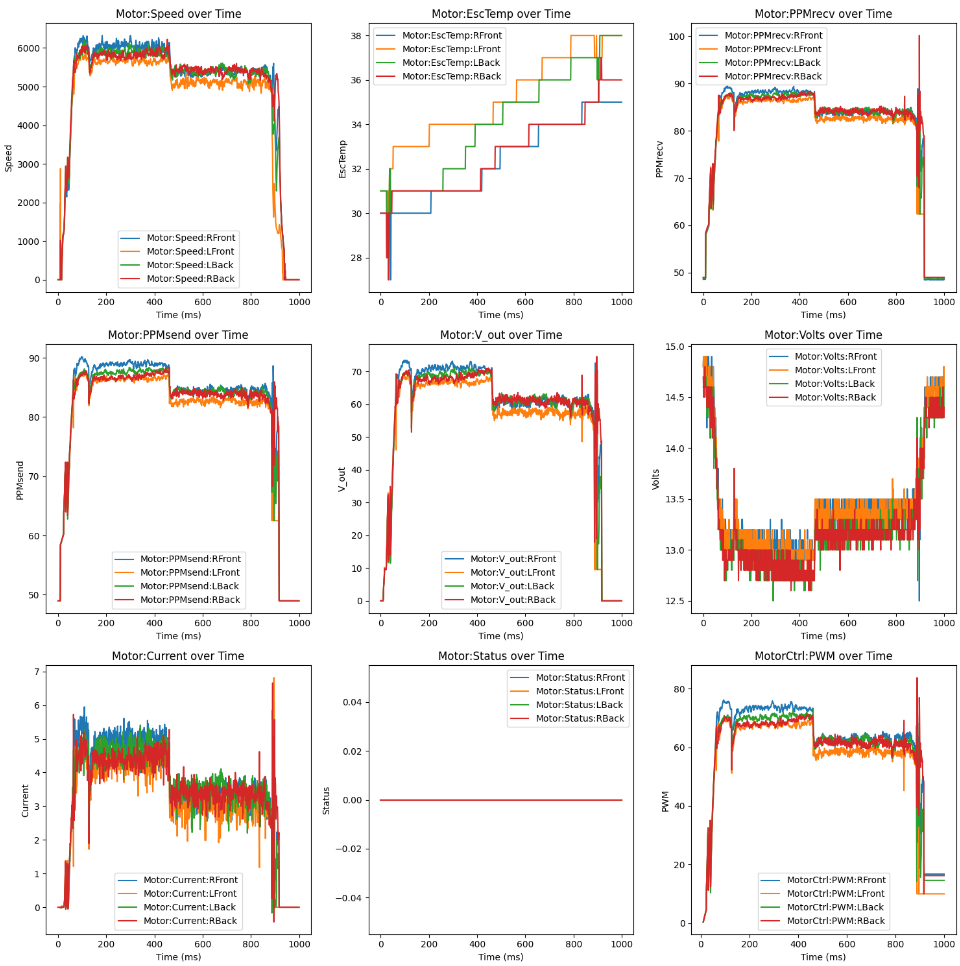

Figure 5. *Variations of motor-related metrics during the flight*

Figure 5 displays variations of each of those 9 metrics during the flight. We can observe that variables Motor:EscTemp and Motor:Status do not have changes that would indicate a drop of the payload, thus we exclude them from the analysis. We also excluded Motor:Volts due to the increased noise in the metric.

For the rest of the metrics we try to establish relationships between the weight of the payload and metric average value. From each of the record 1-8 in the Table 4 we cropped a sample that corresponds to the 10 seconds hovering with the payload of the same weight. Then for each sample we calculated an average value for each motor and between all motors. Figure 6 shows how increase in the payload weight affects average value of the metric. It can be seen that for each variable the relation is linear with different shift along the axis.

After calculating the standard deviation for average values corresponding to the same payload weight, we propose that the metric Motor:Current be utilized for predicting the payload weight based on the flight logs. The fitted equation will be used in later sections to predict the payload weight.

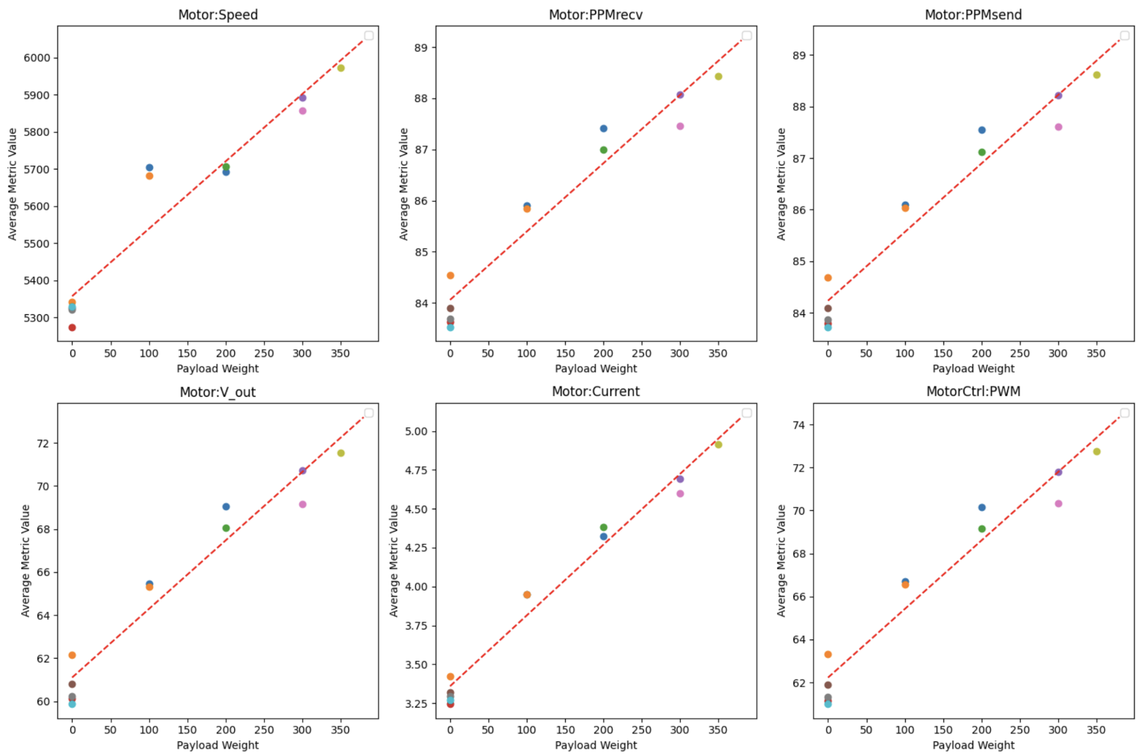

Figure 6. *Weight of the payload to value of the metric relation*

4.3 Moving with the payload

Various motor-related metrics, such as speed of rotation, describe how hard the motor is working at any moment during the flight. The presence of payload affects these readings because lifting a payload and resisting gravity require additional rotation of the propeller, resulting in higher motor speed, electricity consumption, etc. There are other factors that affect motor load as well: movement of the drone and wind.

Consider the drone hovering in one place, with its wings all positioned on one level with the body and horizontal to the ground, as shown in Figure 7. When the drone moves forward, it slows down the two frontal motors, causing the front part of the body to tilt down and the air flow to push the drone, resulting in a change in position. The same would happen in the presence of a small wind; the drone's positioning system would detect the change in position and automatically adjust motor speed to compensate for the wind. Thus, if the drone moves in any direction, it would affect the readings of the motor-related metrics of at least two of the motors. Additionally, if wind is present, it will affect the readings of at least one of the motors.

Figure 7. *Drone hovering in one place. Source: [19]*

In this subsection, we will examine how altering the drone's position during flight affects the motor-related readings and whether payload detection remains possible. Records 8-10 in Table 4.1 correspond to flights with the payload traversing the sports field, making several turns, dropping the payload, continuing the flight, and then landing.

All three records has been done on March 19th with the wind present about 3m/s blowing south-west direction. The location and more details about the experiment design is stated in Chapter 3.

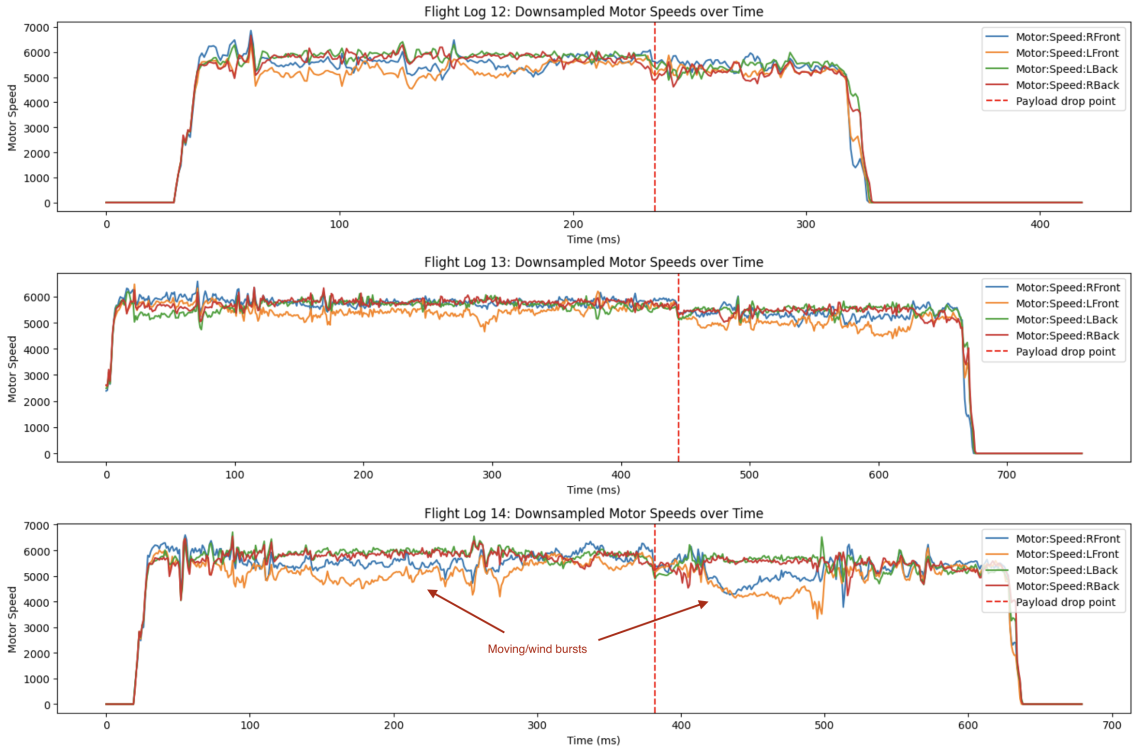

Figure 8. *Records motor flight logs with the payload attached*

In Figure 8, it can be seen that moving forward affects motor-related metrics in the flight logs. For all three records, the speed of the front motors (blue and orange lines) is substantially lower than the other two motors. The faster the drone moves, the bigger this difference will be. Moreover, sudden outbursts of wind, even as slow as 3m/s, affect flight logs visibly.

All these considerations affect the possibility of payload detection and quality of prediction.

4.4 Predicting the payload drop point and weight

In this section, we begin by outlining the scenario in which our method will be applicable. Consider the scenario where a guard discovers a crushed drone with the payload dropping system attached. There is no payload in sight, and the guards suspect that the payload might be dangerous and need to locate it. To initiate the search operation, useful operational information can be extracted from the drone: the probable moment of the payload drop and it's weight. In this scenario, GPS readings and other metrics from the drone are available.

The guard or the officer extracts motor-related metrics from the flight logs. And applies the following algorithm:

1. Isolate records captured after the lift-off and before the landing using log events from the text logs.
2. Apply median aggregation with a sliding window size of 5.
3. Crop samples of length 80 records, varying the interval between them.
4. Calculate the average speed for the two most active motors for each sample.
5. Identify two consecutive samples with the most abrupt drop in the average value between them.
6. The drop point is located between these two samples.
7. Output the GPS coordinates recorded at the moment when the drop occurred.

Additionally, using the fitted expression depicted in Figure 6 for the Motor:Current relationship, as represented by the equation:

$$Y = 3.359801 + x \cdot 0.004549 \tag{4.1}$$

The officer can input the average value of the two back motors' readings during the flight with the payload as Y into this equation. This allows for the calculation of X, which corresponds to the weight of the payload. Note that this formula would work directly only with the Drone and payload system used in this study. But in case when Drone weight is different the expert can use a proportion formula to estimate the weight, using the established knowledge of the relationship between weight and speed of motors.

This algorithm has been tested on four available records of the flight from Table 4.4. The results are listed below. The position of drop error is defined as $MSE = \frac{1}{N}\sum_{i=1}^{N}(y_{\text{real}} - \hat{y}_{\text{predicted}})^2$, where N is the number of records in the recording, and y is an index of the record at the moment the payload was dropped. The weight prediction is defined as percentage error.

Table 4. *Prediction results*

File Name	**Weight Predicted**	**Weight Real**	**Error in Weight Prediction**	**Error in Position Prediction**
FLY_08.03_7.csv	236g	200g	18%	0.003%
FLY_08.03_8.csv	202g	200g	1%	0.05%
FLY_08.03_9.csv	233g	200g	16.5%	0.006%
FLY_05.02_1.csv	112g	100g	12%	0.01%

4.5 Conclusions and findings

In the section 4.3, the evidences has been demonstrated that there is a correlation between the payload weight and the values of the 6 different motor-related variables. The variable Motor:Current values grow linearly as the weight of the payload grows. This variable also demonstrated the lowest error when measurements compared with the fitted curve, thus it is the main candidate for the payload weight estimation algorithm. These constitute the main findings of this chapter.

Findings of the [16] study have been confirmed on a different drone model. While doing that, we also established that the Phantom 4 DJI drone has a different metric set compared to the DJI Zoom 2 drone used in previous work. This underscores challenges and the necessity of generalizing findings across different drone models.

Additionally, a proof of concept algorithm was developed and demonstrated. The algorithm aims to detect the payload drop moment in the flight record and estimate the weight of the payload that was dropped.

The proposed algorithm successfully predicted the exact drop point in all four test records and showed principal ability to estimate the payload weight. However, it's important to note that this algorithm is overly simplistic and designed primarily to illustrate a concept. Various factors such as wind, speed of movement, and direction of movement can significantly affect metric values. To enhance the prediction algorithm and make it robust, the effects of wind and movement on the readings need to be modeled and accounted for. Also, accuracy and boundary requirements for the algorithm need to be established. This suggests a direction for future research.

5. Drone GPS path prediction

Knowing the path of the drone can hold significant operational value for investigators. Even if the exact time when the payload was dropped is known, locating the payload without knowledge of the drone's path presents a challenge. While GPS readings typically provide the drone's path in flight logs, they may become unavailable due to GPS jamming by the targeted facility's defense systems, physical obstacles obstructing the signal, or intentional GPS disabling by the suspect. In such cases, recreating the drone's path becomes necessary.

Chapter 4 focuses on determining the payload size and drop time, while this Chapter attempts to reconstruct the drone's path from flight log metrics excluding GPS.

Two scenarios are considered: first, when an expert knows only the starting or ending point of the path, and second, when both the beginning and ending points of the flight are known. The last point of the flight is typically known, as it is where the drone was found, and the first point may be determined if the suspect left physical evidence at the launch site.

This chapter begins by defining the task and outlining the approach. In Section 5.1.1, the data used to train the model are described, along with the preprocessing steps. It proceeds to discuss the ML model architecture and training algorithm. Section 5.2 demonstrates the model's performance on three randomly selected records not used in training. Section 5.3 explores how errors affect the model's performance, potential explanations for these errors, and suggests future research directions to enhance prediction.

5.1 Objective and approach overview

All metrics recorded in the drone's flight logs have a nature of time series data. Every column contains a series of values that correspond to a value of the same measurement at a particular time. It is natural to approach the prediction of such data with models designed for time series prediction. Yet, such models are usually developed to predict data that have seasonality in them, as they predict naturally occurring processes that often have trends related to the day, month, week, etc.

Data contained in the flight logs, even though they have a time component, do not have any trends or seasonality. Moreover, if we consider that the drone can fly both forwards and backward, we would quickly understand that there is no clear correlation between metrics and the target variable, as the same controller input may correspond to different changes in the target variables, depending on the direction the drone is facing. Because of these considerations, a CNN model was chosen due to its ability to learn features that are a particular combination of the different columns, and because it is possible to build a time series prediction model without reliance on seasonality and trends [20].

To restore a path that the drone has taken, two GPS metrics should be predicted: latitude and longitude coordinates. These metrics have column names 'GPS:Lat' and 'GPS:Long'. Instead of predicting the absolute values of the drone position, the change of the metric between the current moment of time and the previous moment of time was used as a target value. These columns were assigned names: 'delta_lat' and 'delta_long'. To predict each of these two target columns, a separate CNN model was trained with identical architecture. During the training, both target values were excluded from the training data.

5.1.1 Data prepossessing

Table 5. *Flight Dataset Information*

#	File Name	Scenario Description	Dataset
1	FLY_19.03_2.csv	Flight across the field with the payload attached. The payload weight is 200g. The payload is dropped in the middle of the flight. Then the flight is continued. GPS is enabled during the flight.	Training/Validation
2	FLY_19.03_3.csv		Training/Validation
3	FLY_19.03_5.csv		Training/Validation
4	FLY_19.03_7.csv		Training/Validation
5	FLY_19.03_8.csv		Training/Validation
6	FLY_19.03_4.csv		Demonstration
7	FLY_19.03_9.csv		Demonstration
8	FLY_19.03_6.csv		Demonstration

During the acquisition step, as described in Chapter 3, separate flight recordings stored in the DAT file are converted to .CSV files. The list of used files are stated in the Table 5.1.1. Some files are used for demonstration and analysis in the end of this section. They are processed separately and do not used during training and validation of the model.

Before submitting input into the model, the following preprocessing algorithm was applied:

1. **Extract Relevant Columns**: The flight logs originally contain 295 columns. Only 157 + 2 target columns were used for the prediction. All columns that start with 'GPS' were removed regardless of the content, as well as columns that contain the position of the drone calculated from GPS. Some columns have been removed based

on manual analysis of the values, correlation, and variability. A full list of used columns can be found in Appendix 6.

2. **Downsample Data**: Split the data into chunks of 10 rows and calculate the median values for each chunk to downsample the data. The GPS target values were downsampled separately as well. This process also helps to remove outliers.
3. **Replace Missing Values**: Replace missing values in the columns using the `replace_na_with_median` function.
4. **Calculate Differences**: Calculate the differences between consecutive latitude and longitude values to obtain `delta_lat` and `delta_long`. These two columns are kept separately and used as labels.
5. **Normalize Data**: Normalize each column of the training data and target variables to a range between 0 and 1.
6. **Slice Data for Sequence Model**: Slice the processed data into frames of length 15 rows with a step of 2. To each frame, the target latitude and longitude difference (one value) is added.
7. **Split Data**: Split the data into training and validation sets. The training dataset contains 80% of all frames. The frames are not randomized on the split to avoid validation data getting into the training set, since frames overlap.

In the resulting dataset, there are two separate parts: a dataset with target latitude and a dataset with target longitude. Frames that go as input for the model are identical for latitude and longitude. Target variables for the two separate models are: 'delta_lat' and 'delta_long'. The resulting dimensions of the data for each sample are:

1. **Input**: An array of dimensions (157, 15).
2. **Corresponding label**: One value (1) of either 'delta_lat' or 'delta_long' column.

5.1.2 Model Architecture and training

For position change prediction, two separate CNN models have been trained with identical architecture. The models are implemented using the Python Keras framework.

The input with a dimensionality of (157, 15) is fed into the Conv2D layer with the ReLU activation function. This layer consists of 32 separately trained convolutional filters of size (3, 3). The output of the Conv layer is then passed to the MaxPooling2D layer with a pool size of (2,2) for dimensionality reduction. Finally, the resulting set of feature maps is flattened into a single array before being submitted to the fully connected Dense layer with 128 output neurons and ReLU activation function. The last step is submitting the output of the previous step to the Dense fully-connected layer with a single output neuron and linear activation function.

The output of the last Dense layer is trained to be the target delta of the coordinate change. Each of the trained models has 2,208,321 trainable parameters.

Both models are trained separately using the Adam optimizer with a batch size of 30 and for 25 epochs. The loss between the output of the model and the target label is calculated as the Mean Squared Error. Training and validation results for both models are stated below:

Table 6. *CNN Model Training and Validation Results*

Model	**Training Loss**	**Last Validation Loss**
Lat Model	0.0009697	0.0067
Long Model	0.0013	0.0074

Trained model can be found in the code repository with names model_long_4.h5 and model_lat_4.h5 [1].

5.2 GPS path prediction

In this section, the performance of the model trained in Section 5.1.2 is demonstrated and discussed. First, we plot prediction performance using records 6-8 (see Table 5). Then, we describe the process of restoring coordinates from the predicted deltas. Next, we plot predictions and ground truth flights on the map to demonstrate how errors affect the results.

Figure 9 demonstrates the performance of both models in predicting the position change on three different recordings of the flight logs. To build plots with time series demonstration data, we undergo denormalization using the maximum and minimum values that were also used during training. To build a path on the map, the cumulative sum has been calculated for both columns, and the first real GPS coordinate has been added to position the path properly on the map. This GPS coordinate we add is assumed to be known as the drone was discovered somewhere and GPS coordinate of the crush place is known. Also this assumption is discussed in later sections.

It may be noticed by carefully studying the model's performance that they have a peculiar feature. In cases where the model's prediction diverges from the ground truth, the shape of

[1] https://gitlab.cs.ttu.ee/annman/thesiscode

the curve might demonstrate exactly the same fluctuations but be shifted from the ground truth. Another quality of the model is that error, at least in some cases, is synchronized for both models. For example, in the flight at the top, both models diverge from the ground truth in the range [100;230] on the X axis but are precise in every other range. If one studies plotted paths on the map, one can notice that the predicted curves and ground truth share a very similar shape, sometimes even with tiny details, but this shape seems to be shifted in one particular direction on the map.

Our theory, based on observations during the experiments and carefully observing models' performance, is that those shifts are caused by wind. When the drone is blown off by the wind, it corrects its position automatically by adjusting motors' performance. Those adjustments are recorded in metrics and affect a model's prediction. But since those adjustments only cancel out the wind factor and the position of the drone in reality doesn't change, the model's prediction demonstrates a divergence from the ground truth.

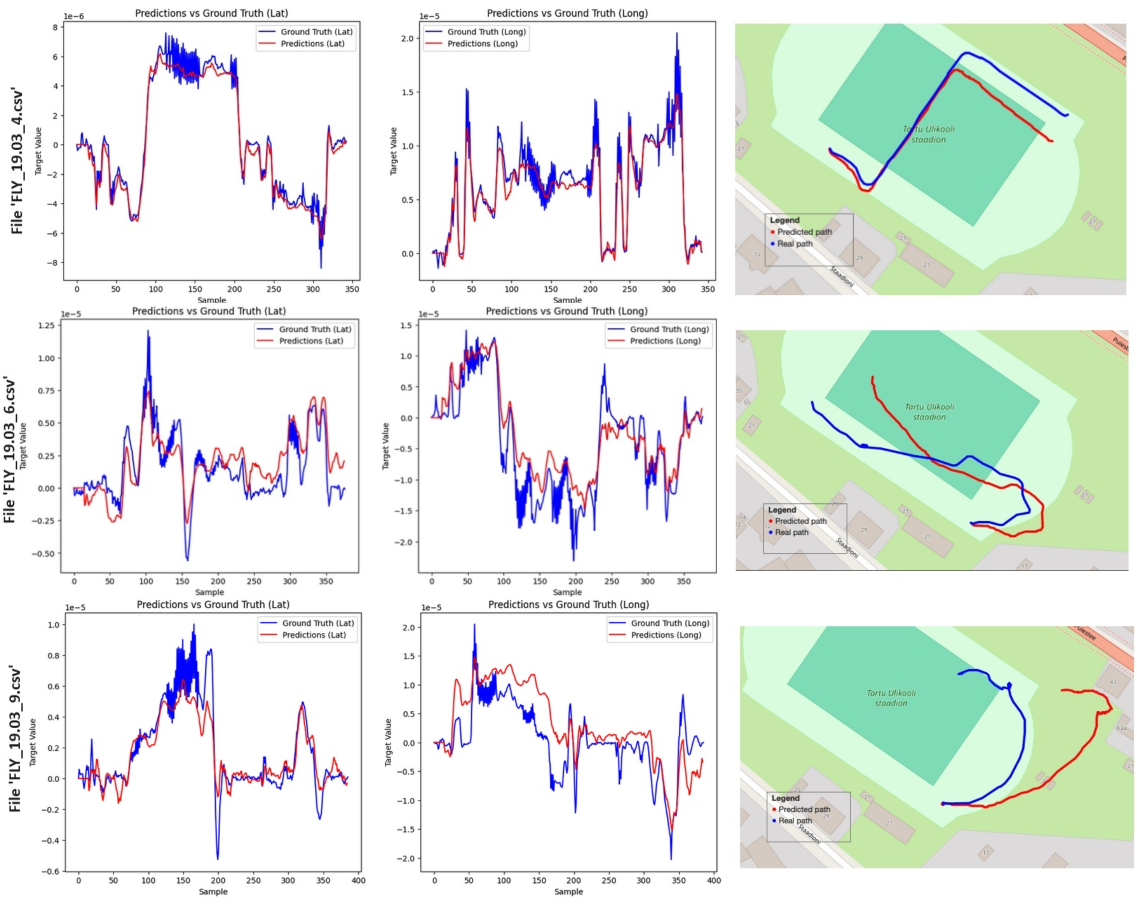

Figure 9. *GPS path prediction model demonstration*

Let's study the error in predictions in more details. Figure 10 shows how error between the real and predicted GPS coordinates grows over the time.

One may notice that the error grows more or less with the constant slope. For the record

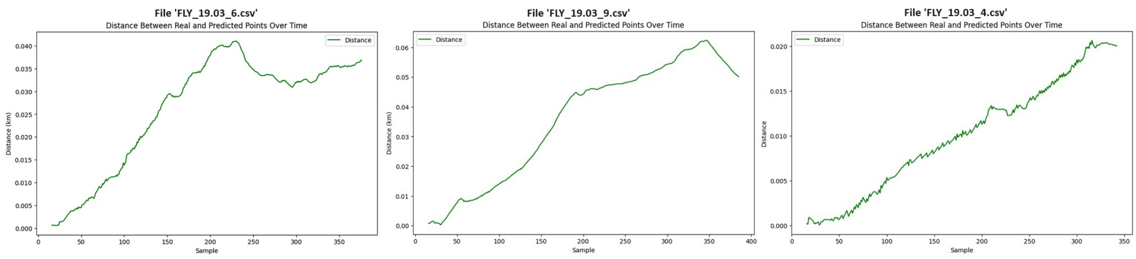

Figure 10. *Error between real and predicted GPS coordinates over the time*

on the left and in the middle, the direction and magnitude of the shift change during the middle of the flight, while for the record on the right, both the direction and magnitude of the shift remain almost perfectly constant.

This characteristic of the error suggests the potential for enhancing prediction accuracy with additional information regarding the drone's position or by modeling the wind to account for and cancel the influence of wind on the measurements.

5.3 Improving prediction

In the scenario explored in the previous section, we assumed that only one GPS coordinate is available for each flight: the starting point. This point may be available for the investigator as the suspect may leave evidence at the point where they controlled the drone from. This point might also have been recorded by the drone at the beginning of the flight even if GPS was later turned off or jammed. Another point that may be available in most cases is the ending point of the flight, as that is where a crashed drone may be discovered by the officers. This consideration presents two distinct scenarios:

1. Only one point is available, either a starting point or the ending point.
2. Both coordinates of the beginning and the end of the flight are available for the investigator.

First case was explored in the previous Section 5.2. The case when only the last point of the flight is available will not be modeled separately as it essentially does not differ from the case when only the starting point is available.

If both points are available for the investigator, we have the ability to improve prediction by attempting to model the wind and cancel the shift resulting from it. For that, we calculate the direction and magnitude of the shift and then rotate the path using the resulting vector. Results of such a shift are displayed in Figure 11. These results can be compared with the results for the first scenario in Figure 9.

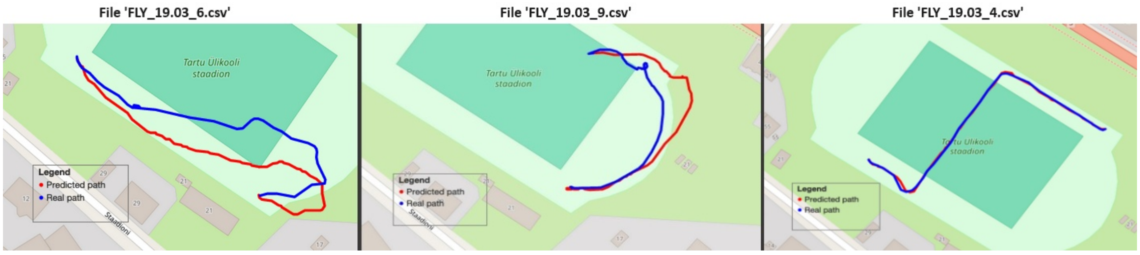

Figure 11. *Predicted GPS path with the wind canceled out. Direction and magnitude of the wind is assumed to be constant.*

This approach works best when applied to the record ’FLY_19.03_4.csv’ when both the direction and magnitude of wind are constant. However, in general and for these records, the assumption that the wind is constant is not true. In cases where wind is not constant an advanced wind modeling is needed.

After receiving these results, research was conducted on the usage of DJI drone flight records for wind prediction. It revealed that over the last 5 years, this research field has been very active. The latest work has achieved precision in measuring both wind direction and magnitude comparable to commercial anemometers traditionally used in meteorology [21]. The models and approaches stated in those papers employ various metrics, including GPS metrics, as there is no reason not to use them for meteorologist. Most of the papers also assumed that the drone is stationary, with only a few using measurements of the moving drones [22]. Thus, those results are not directly transferable for solving forensic path restoring problems. However, adapting the mentioned approaches to modeling wind and canceling it out from the prediction seems to be a promising direction for future research to improve current results. Additionally, some of the papers in the field offer free datasets with detailed wind measurements taken by the anemometer.

5.4 Conclusions and findings

In this chapter, we propose the architecture of the model, discuss data preprocessing, and present the framework for model training. The model showcased its capability to accurately detect drone movements, marking the primary findings of this chapter.

Furthermore, in Section 5.3, we propose a method to enhance the prediction accuracy of the existing model by leveraging additional GPS points accessible to investigators. These points can be inferred through the analysis of physical evidence or by examining other data sources.

Demonstrating the performance of the model in the 5.2 and error characteristics revealed that the model seems to be affected in a specific way that might be caused by the wind moving the drone. Based on this observation, we suggest a potential direction for future research, which involves utilizing existing research in wind modeling from meteorology to improve path recovery techniques for forensic science applications.

6. Conclusions and future research outline

In the Introduction chapter of this thesis, the following hypotheses have been stated:

1. There is a linear relationship between the attached payload weight and the values of the motor-related metrics in the flight logs.
2. If the payload was dropped mid-flight, it is possible to estimate a drop point from the motor related metrics in the flight logs.
3. The metrics recorded in flight logs, excluding GPS metrics, contain sufficient information to enable recreation of drone path.

Now, we summarize the analysis and the results we obtained during testing.

1. There is a linear relationship between the attached payload weight and the values of the motor-related metrics in the flight logs.

To test this, we started in Section 4.1 by successfully recreating previous findings stated in [16]. If a previous study has shown evidence that payload detection is feasible in a controlled environment, we extended those findings to an uncontrolled environment. In Section 4.2, we demonstrated evidences that the values of the metrics that describe motor performance grow linearly as the payload size grows. This gave us the ability to fit an equation to the observed values and use that equation to estimate payload weight in Section 4.4.

These findings suggest several possible directions for future research. Findings need to be generalized for different drone models, not only those from DJI but also from other drone producers. Additionally, more data needs to be collected to determine the accuracy of the method. Finally, different factors such as wind, movement, environmental conditions, and their effects on the possibility to predict the weight of the payload need to be studied and included in the model.

2. If the payload was dropped mid-flight, it is possible to estimate a drop point from the motor related metrics in the flight logs.

Detecting the payload drop mid-flight requires payload detection on the moving drone. First, the analysis of complications this task presents is discussed in Section 4.3. Then, in Section 4.4, an algorithm to detect a drop moment in the flight log is proposed. This algorithm is then employed to predict the exact point of the drop for four different flight records successfully. The results of the prediction are summarized in Table 4.4. This demonstrates possibility of detecting the payload drop under the certain circumstances in the uncontrolled environment.

These findings suffer from a lack of generalization for different environments. Future research might start with modeling the effects wind and movement have on the motor-related variables. Then, those effects should be accounted for in the improved detection algorithm. Additionally, the maximum speed of movement and wind for which detection is still possible need to be determined to define the limits of the method.

3. The metrics recorded in flight logs, excluding GPS metrics, contain sufficient information to enable recreation of drone movement.

To test this, data preprocessing was described in Section 5.1.1. Then, in Section 5.1.2, the model architecture and the training process were described. This model was used to predict drone movement with an error of less than 0.1% in Section 5.2. With this, we have shown the potential for path restoration in an uncontrolled environment through a limited experiment.

Demonstrating the performance of the model and how errors affect the results revealed that the model seems to be affected in a specific way that might be caused by the wind moving the drone. This led to the discovery of existing research on using drone flight logs to model the wind. These findings, as well as the opportunities for future research they reveal, are discussed in Section 5.3. Employing existing approaches in meteorology to model the wind and exclude its effects on the data may be a very fruitful direction for future work.

Additionally, more experiments are needed to study how flight metrics are collected and calculated and how GPS availability affects them.

In conclusion, all three hypotheses previously stated have been tested, and the experiments yield promising results in support of the hypotheses. However, it's important to note that this work was designed as a proof of concept. The results obtained in this work lay down a basis for future work in UAV flight log analysis, which, we hope, would bring the required generalization and soundness to make it an acceptable tool to solve real-life forensic challenges.

References

[1] Ankit Renduchintala et al. “A comprehensive micro unmanned aerial vehicle (UAV/-Drone) forensic framework”. In: *Digital Investigation* 30 (2019), pp. 52–72.

[2] Ed AlvaradoBefore working with drones. *Industry Leading Drone Market Analysis 2022-2030: Droneii*. Sept. 2022. URL: https://droneii.com/drone-market-analysis-2022-2030.

[3] Yonette Joseph. *"damaged" drone found near Gatwick; arrested couple are freed without charge*. Dec. 2018. URL: https://www.nytimes.com/2018/12/23/world/europe/gatwick-airport-drones.html.

[4] INTERPOL. *Framework for responding to a drone incident - INTERPOL*. 2019. URL: https://www.interpol.int/ar/content/download/17307/file/IC_DFL_DroneIncident_Final_EN.pdf?inLanguage=eng-GB.

[5] Evangelos Mantas and Constantinos Patsakis. “Who watches the new watchmen? The challenges for drone digital forensics investigations”. In: *Array* 14 (2022), p. 100135.

[6] Evangelos Mantas and Constantinos Patsakis. “GRYPHON: Drone forensics in dataflash and telemetry logs”. In: *Advances in Information and Computer Security: 14th International Workshop on Security, IWSEC 2019, Tokyo, Japan, August 28–30, 2019, Proceedings 14*. Springer. 2019, pp. 377–390.

[7] James Kin Wah Lan and Frankie Kin Wah Lee. “Drone Forensics: A Case Study on DJI Mavic Air 2”. In: *2022 24th International Conference on Advanced Communication Technology (ICACT)*. IEEE. 2022, pp. 291–296.

[8] “Report on the drone-related incidents”. In: DRONESEC, 2022. URL: https://assets-global.website-files.com/5fb1b8c46daaf8e55874026c/63b77e748179e6d6f40deafc_DroneSec%20Notify%20%23160%20-%2004-01-23-v1.0-PUBLIC.pdf.

[9] Devon R Clark et al. “DROP (DRone Open source Parser) your drone: Forensic analysis of the DJI Phantom III”. In: *Digital Investigation* 22 (2017), S3–S14.

[10] Graeme Horsman. “Unmanned aerial vehicles: A preliminary analysis of forensic challenges”. In: *Digital Investigation* 16 (2016), pp. 1–11.

[11] Ravin Kumar and Animesh Kumar Agrawal. "Drone GPS data analysis for flight path reconstruction: A study on DJI, Parrot & Yuneec make drones". In: *Forensic Science International: Digital Investigation* 38 (2021), p. 301182.

[12] Fahad E Salamh et al. "Drone Disrupted Denial of Service Attack (3DOS): Towards an Incident Response and Forensic Analysis of Remotely Piloted Aerial Systems (RPASs)". In: *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*. 2019, pp. 704–710. DOI: 10.1109/IWCMC.2019.8766538.

[13] Khalifa Al-Room et al. "Drone forensics: A case study of digital forensic investigations conducted on common drone models". In: *International Journal of Digital Crime and Forensics (IJDCF)* 13.1 (2021), pp. 1–25.

[14] Septyan Eka Prastya, Imam Riadi, and Ahmad Luthfi. "Forensic analysis of unmanned aerial vehicle to obtain GPS log data as digital evidence". In: *IJCSIS* 15.3 (2017).

[15] Sowmya Viswanathan and Zubair Baig. "Digital forensics for drones: A study of tools and techniques". In: *Applications and Techniques in Information Security: 11th International Conference, ATIS 2020, Brisbane, QLD, Australia, November 12–13, 2020, Proceedings 11*. Springer. 2020, pp. 29–41.

[16] Justin Ryan. "Drone Forensics: Framework for Payload Detection Through Flight Log Analysis of Dji Mavic 2 Zoom". PhD thesis. University of Colorado at Denver, 2023.

[17] Larisa Kapustina et al. "The global drone market: main development trends". In: *SHS Web of Conferences*. Vol. 129. EDP Sciences. 2021, p. 11004.

[18] James Kin Wah Lan and Frankie Kin Wah Lee. "Drone Forensics: A Case Study on DJI Mavic Air 2". In: *2022 24th International Conference on Advanced Communication Technology (ICACT)*. 2022, pp. 291–296. DOI: 10.23919/ICACT53585.2022.9728844.

[19] ChatGPT. *Drone Image*. https://openai.com/chatgpt. Accessed: 2024-03-30. Mar. 2024.

[20] Keiron O'shea and Ryan Nash. "An introduction to convolutional neural networks". In: *arXiv preprint arXiv:1511.08458* (2015).

[21] Steven Zimmerman et al. "Wind estimation by multirotor dynamic state measurement and machine learning models". In: *Measurement* 198 (2022), p. 111331.

[22] Kilian Meier et al. "Wind estimation with multirotor uavs". In: *Atmosphere* 13.4 (2022), p. 551.

Appendix 1 – Non-Exclusive License for Reproduction and Publication of a Graduation Thesis[1]

I Anna Mandrenko

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Analyzing Drone Flight Logs for Forensic Behaviour Reconstruction”, supervised by Matthew James Sorell
   1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;
   1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

11.05.2024

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.

Appendix 2 – List of the flight log columns included in the analysis

Column name	Column name	Column name
'IMU_ATTI(0):press:D'	'IMU_ATTI(0):alti:D'	'IMU_-ATTI(0):relativeHeight:C'
'IMU_ATTI(0):numSats'	'IMU_ATTI(0):roll:C'	'IMU_ATTI(0):pitch:C'
'IMU_ATTI(0):yaw:C'	'IMU_ATTI(0):accelX'	'IMU_ATTI(0):accelY'
'IMU_ATTI(0):accelZ'	'IMU_-ATTI(0):accelComposite:C'	'IMU_ATTI(0):gyroX'
'IMU_ATTI(0):gyroY'	'IMU_ATTI(0):gyroZ'	'IMU_-ATTI(0):gyroComposite:C'
'IMU_ATTI(0):velN'	'IMU_ATTI(0):velE'	'IMU_ATTI(0):velD'
'IMU_-ATTI(0):velComposite:C'	'IMU_ATTI(0):velH:C'	'IMU_ATTI(0):magX'
'IMU_ATTI(0):magY'	'IMU_ATTI(0):magZ'	'IMU_ATTI(0):magMod:C'
'IMU_ATTI(0):temperature'	'IMU_-ATTI(0):yawUnWrapped:C'	'IMU_-ATTI(0):tiltInclination:C'
'IMU_ATTI(0):tilt Direction-EarthFrame:C'	'IMU_ATTI(0):tilt Direction-BodyFrame:C'	'IMU_ATTI(0):yaw360:C'
'IMU_ATTI(0):magYaw:C'	'IMU_ATTI(0):Yaw-magYaw:C'	'IMUCalcs(0):PosN:C'
'IMUCalcs(0):PosE:C'	'IMUCalcs(0):PosD:C'	'IMUCalcs(0):height:C'
'IMUCalcs(0):totalGyroZ:C'	'IMUCalcs(0):totalGyroX:C'	'IMUCalcs(0):totalGyroY:C'
'IMU_-ATTI(0):distanceTravelled:C'	'IMU_ATTI(0):quatW:D'	'IMU_ATTI(0):quatX:D'
'IMU_ATTI(0):quatY:D'	'IMU_ATTI(0):quatZ:D'	'IMU_ATTI(0):agX:D'
'IMU_ATTI(0):agY:D'	'IMU_ATTI(0):agZ:D'	'IMU_ATTI(0):gbX:D'
'IMU_ATTI(0):gbY:D'	'IMU_ATTI(0):gbZ:D'	'IMU_ATTI(0):tz:D'
'IMU_ATTI(0):atti_cnt:D'	'Controller:ctrl_tick:D'	'Controller:ctrl_pitch:D'
'Controller:ctrl_yaw:D'	'Controller:ctrl_thr:D'	'Controller:motor_state:D'
'Controller:ctrl_level:D'	'Controller:D2H_x:D'	'Controller:D2H_y:D'
'Controller:act_req_id:D'	'Controller:act_act_id:D'	'Controller:cmd_mod:D'
'Controller:mod_req_id:D'	'Controller:mot_sta:D'	'MagCali(0):X:D'

Continues...

Table 7 – *Continues...*

Column name	**Column name**	**Column name**
’MagCali(0):Y:D’	’MagCali(0):Z:D’	’MagCali(0):Mod:C’
’MagCali(0):magYaw:C’	’MagCali(1):X:D’	’MagCali(1):Y:D’
’MagCali(1):Z:D’	’MagCali(1):Mod:C’	’MagCali(1):magYaw:C’
’CompassFilter(0):X:D’	’CompassFilter(0):Y:D’	’CompassFilter(0):Z:D’
’CompassFilter(1):X:D’	’CompassFilter(1):Y:D’	’CompassFilter(1):Z:D’
’MVO:velX’	’MVO:velY’	’MVO:posX’
’MVO:posY’	’MVO:hoverPointUncertainty1’	’MVO:hoverPointUncertainty4’
’MVO:hoverPointUncertainty6’	’MVO:velocityUncertainty1’	’MVO:velocityUncertainty3’
’MVO:velocityUncertainty4’	’MVO:velocityUncertainty6’	’MVO:heightUncertainty’
’Motor:Speed:RFront’	’Motor:Speed:LFront’	’Motor:Speed:LBack’
’Motor:Speed:RBack’	’Motor:EscTemp:RFront’	’Motor:EscTemp:LFront’
’Motor:EscTemp:LBack’	’Motor:EscTemp:RBack’	’Motor:PPMrecv:RFront’
’Motor:PPMrecv:LFront’	’Motor:PPMrecv:LBack’	’Motor:PPMrecv:RBack’
’Motor:PPMsend:RFront’	’Motor:PPMsend:LFront’	’Motor:PPMsend:LBack’
’Motor:PPMsend:RBack’	’Motor:V_out:RFront’	’Motor:V_out:LFront’
’Motor:V_out:LBack’	’Motor:V_out:RBack’	’Motor:Volts:RFront’
’Motor:Volts:LFront’	’Motor:Volts:LBack’	’Motor:Volts:RBack’
’Motor:Current:RFront’	’Motor:Current:LFront’	’Motor:Current:LBack’
’Motor:Current:RBack’	’AirSpeed:air_vbx:D’	’AirSpeed:air_vby:D’
’AirSpeed:comp_alti:D’	’AirSpeed:windSpeed’	’AirSpeed:windDirection’
’AirSpeed:windFromDir’	’AirSpeed:windN’	’AirSpeed:windE’
’AirSpeed:MotorSpd:D’	’osd_data:flightTime’	’osd_data:navHealth’
’osd_data:relativeHeight’	’usonic:usonic_h’	’MotorCtrl:PWM:RFront’
’MotorCtrl:PWM:LFront’	’MotorCtrl:PWM:LBack’	’MotorCtrl:PWM:RBack’
’ATTI_MINI0:s_qw0’	’ATTI_MINI0:s_qx0’	’ATTI_MINI0:s_qy0’
’ATTI_MINI0:s_qz0’	’ATTI_MINI0:roll:C’	’ATTI_MINI0:pitch:C’
’ATTI_MINI0:yaw:C’	’ATTI_MINI0:s_pgz0’	’ATTI_MINI0:s_vgz0’
’ATTI_MINI0:s_agz0’	’ATTI_MINI0:s_cnt0’	’IMUCalcs(0):diffVelE:C’
’IMUCalcs(0):velD:C’	’IMUCalcs(0):velN:C’	’IMU_ATTI(0): directionOf-Travel[true]:C’
’IMU_ATTI(0): directionOf-Travel[mag]:C’	’IMUCalcs(0):diffVelD:C’	’IMUCalcs(0):velE:C’
’IMUCalcs(0):diffVelN:C’		