CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Electrical Engineering

Diploma's Thesis

Bc. Jiří Fiedler

Enhanced racing drone detection using microphones

Department of Telecommunication engineering

Thesis supervisor: **Ing. Lukáš Vojtěch, Ph.D.**

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne............................ ....

ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Fiedler** Jméno: **Jiří** Osobní číslo: **406354**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra řídicí techniky**

Studijní program: **Kybernetika a robotika**

Studijní obor: **Kybernetika a robotika**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Rozšířená detekce závodních dronů s využitím mikrofonů

Název diplomové práce anglicky:

Enhanced racing drone detection using microphones

Pokyny pro vypracování:

Prostudujte současné metody pro detekci a měření času závodních dronů. Prostudujte možnosti využití mikrofonů pro detekci akustické stopy dronu, při jeho průletu bránou. Navrhněte vhodný DSP algoritmus pro detekci dronu využívající signálů snímaných mikrofony. Otestujte navržený algoritmus v reálném experimentu. Dosažené výsledky statisticky vyhodnoťe, určete chybu metody a porovnejte ji s konkurenčními způsoby detekce.

Seznam doporučené literatury:

[1] A. Bateman, I. Paterson-Stephens, The DSP Handbook: Algorithms, Applications and Design Techniques 1st Edition, ISBN-13: 978-0201398519, Prentice Hall, 2002
[2] B. Van Den Broeck, A. Bertrand, P. Karsmakers, B. Vanrumste, H. Van hamme, and M. Moonen. Time-domain generalized cross correlation phase transform sound source localization for small microphone arrays. In 2012 5th European DSP Education and Research Conference (EDERC), pages 76{80, Sept 2012.
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.3764&rep=rep1&type=pdf
[3] University of Cambridge Will Simmons. Churchill compsci talks: Where did that come from? an introduction to sound localisation. http://talks.cam.ac.uk/talk/index/62200,
https://www.youtube.com/watch?v=Z7X7lf6FdYY.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Lukáš Vojtěch, Ph.D., katedra telekomunikační techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **15.02.2018** Termín odevzdání diplomové práce: **25.05.2018**

Platnost zadání diplomové práce: **30.09.2019**

Ing. Lukáš Vojtěch, Ph.D. podpis vedoucí(ho) práce	prof. Ing. Michael Šebek, DrSc. podpis vedoucí(ho) ústavu/katedry	prof. Ing. Pavel Ripka, CSc. podpis děkana(ky)

© ČVUT v Praze, Design: ČVUT v Praze, VIC

MASTER'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Fiedler Jiří** Personal ID number: **406354**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Control Engineering**

Study program: **Cybernetics and Robotics**

Branch of study: **Cybernetics and Robotics**

II. Master's thesis details

Master's thesis title in English:

Enhanced racing drone detection using microphones

Master's thesis title in Czech:

Rozšířená detekce závodních dronů s využitím mikrofonů

Guidelines:

1. Study currently used methods for racing drone detection and time measurement.
2. Study the possibilities of using microphones to detect the drone acoustic traces while passing through the gate.
3. Design an appropriate DSP algorithm for drone detection using microphone signals.
4. Test the proposed algorithm in a real experiment. Statistically evaluate the results, identify the method error and compare it with competing detection methods.

Bibliography / sources:

[1] A. Bateman, I. Paterson-Stephens, The DSP Handbook: Algorithms, Applications and Design Techniques 1st Edition, ISBN-13: 978-0201398519, Prentice Hall, 2002
[2] B. Van Den Broeck, A. Bertrand, P. Karsmakers, B. Vanrumste, H. Van hamme, and M. Moonen. Time-domain generalized cross correlation phase transform sound source localization for small microphone arrays. In 2012 5th European DSP Education and Research Conference (EDERC), pages 76{80, Sept 2012.
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.3764&rep=rep1&type=pdf
[3] University of Cambridge Will Simmons. Churchill compsci talks: Where did that come from? an introduction to sound localisation. http://talks.cam.ac.uk/talk/index/62200,
https://www.youtube.com/watch?v=Z7X7lf6FdYY.

Name and workplace of master's thesis supervisor:

Ing. Lukáš Vojtěch, Ph.D., Department of Telecommunications Engineering, FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **15.02.2018** Deadline for master's thesis submission: **25.05.2018**

Assignment valid until: **30.09.2019**

Ing. Lukáš Vojtěch, Ph.D. Supervisor's signature	prof. Ing. Michael Šebek, DrSc. Head of department's signature	prof. Ing. Pavel Ripka, CSc. Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt — Student's signature

© ČVUT v Praze, Design: ČVUT v Praze, VIC

Acknowledgements

Firstly, I would like to thank my supervisor Ing. Lukáš Vojtěch, Ph.D. for his kindly approach. Furthermore I want to thank my family for the encouragement during my studies.

Abstrakt

Tato práce se zabývá návrhem a implementací algoritmu pro detekci průletu závodního dronu skrz bránu. Oproti standardním algoritmům pro detekci dronu založených na principu měření intenzity video signálu z dronu pomocí video přijímače je tento algoritmus rozšířen o použití mikrofonů. Dvojice mikrofonů je instalována u každého kraje brány. Mezi signály z dvojice mikrofonů je zjištěn časový posun. Z časových posunů mezi signály z obou dvojic mikrofonů lze určit, zda-li dron proletěl skrze bránu nebo ji minul. Je navržen Kalmanův filtr pro filtraci dat z video-přijímače a mikrofonů. Využitím mikrofonů lze přesněji určit okamžik, kdy dron proletěl skrz bránu. Výsledky navrženého algoritmu jsou srovnány s komerčně dostupnými zařízeními. Navržený algoritmus měří časy kol přesněji a spolehlivěji než dostupná zařízení založené pouze na principu měření intenzity video signálu z dronu jak bylo prokázáno v experimentech. Tento algoritmus může být využit v časomírách využívaných při stále populárnějších závodech dronů.

Klíčová slova: Mikrofon, FPV, Závody dronů

Abstract

This thesis deals with design and implementation of an algorithm for detecting the racing drone crossing the gate. Compared to standard algorithms for drone detection based on the principle of measuring the strength of video signal from a drone by video receiver, this algorithm is enhanced by the use of microphones. A pair of microphones is installed at each corner of the gate. Timeshift between signals from a pair of microphones is calculated. It can be determined whether the drone has passed through the gate or missed if from the time shifts between both pairs of microphones. Kalman's filter for data filtration from video receivers and microphones has been designed. It is possible to specify the moment when the drone passed through the gate more precisely by using microphones. The proposed algorithm is compared with other solutions available on the market. Proposed algorithm measures lap times more precisely and reliably than other solutions based purely on the principle or measuring the strength of video signal from a drone as was proven in several experiments. This algorithm can be implemented in timing systems used in increasingly popular drone racing.

Keywords: Microphone, FPV, Drone racing

Contents

List of Figures

1 Introduction

Unmanned aircrafts capable of vertical takeoff are very popular nowadays. Drones became a low-cost easily accessible platform for various tasks in past few years. They can be seen in various shapes and sizes. They cost from dozens to thousands of euro. Smaller and cheaper drones are widely used among the hobbyist. Someone use them just as a toy for fun, others do first-person view (FPV) racing competitions with small, agile and fast quadcopters. Racing drone is shown in figure 1. Bigger drones with stabilized gimbal are used for aerial photography and cinematography. Industrial applications are usually made by professionals with expensive, large and reliable hexacopters or octocopters. Drones are used for aerial observation and exploration, especially in scenarios which could be dangerous for a human and which are too small or too expensive for an aircraft with human crew.

Figure 1: *An example of racing drone [22]*

FPV drone racing is rapidly growing sport, for everyone regardless of the age or gender. It is performed by tens of thousands of hobbyist and professional pilots. The common racing drone size is 210mm in diameter. It weights 500g, flies about 3 minutes and reaches speeds more than $160\frac{km}{h}$. The main parts of a racing drone are brushless motors, propellers, electronic speed controllers, flight controller, battery, RC receiver and FPV system which consists of a camera, 5.8GHz video transmitter and antenna. Each pilot has RC transmitter and special video goggles with build in video receiver. Video goggles are shown in figure 1. Pilots control drones sitting on a chair while seeing what the drone sees. They have truly immersive experience using these goggles. The FPV system brings pilots the adrenaline experience of flying fast near the obstacles and the safety since pilots are still with their feet on the ground.

Figure 2: *Video goggles [22]*

Racing is the main purpose of the FPV racing drones. Up to 8 pilots are racing against each other on a racetrack. The racetrack consists of different types of obstacles like pylons and gates. Example of a racetrack is shown in figure 7. The goal is to fly several laps on the racetrack faster than the others. Drone race is usually great show for spectators. Drone races are organized in interesting places with lots of special effects like lights or fire. Drone racing is divided into several categories depending on the weight and size of the drone. The smallest racing drones weight about 24g and their size is 64mm in diameter. The most popular drones for racing are 210mm in diameter and the largest drones are up to 1200mm in diameter.

The drone racing is considered as a sport of the future. The first race was organized in Australia in late 2014. Since then the drone racing has made a huge step forward and races are all around the world. In 2016, there was a big race in Dubai with 250 000$ prize money and it was the first race that could have been seen on television. Drones are in media quite often nowadays. American drone racing league (DRL) is broadcast on television in USA and Germany [7]. Pilots racing for DRL are full-time professionals. Drone champions league (DCL) is the biggest race serial in Europe in which 9 teams with the best pilots from the world are competing against each other [5]. DCL races take places like Avenue

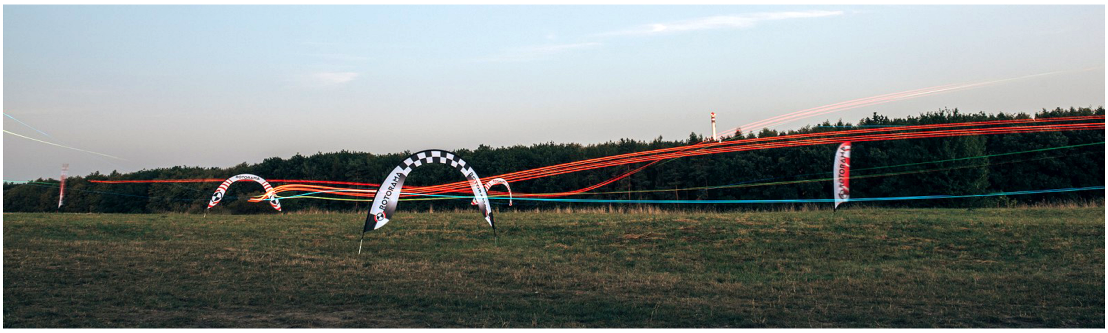


Figure 3: *An example of racetrack [22]*

des Champs-Élysées, Brussels and Vaduz. Czech team Rotorama takes 3rd place in season 2017. MultiGP is the worlds largest drone racing league with more than 22 000 registered pilots [17]. Over 10 races were organized in the Czech Republic in 2017.

Drone races are made for spectators. Hundreds or even thousands of people are watching the race live. Drones with weight around 500g and speeds over a 100$\frac{km}{h}$ are flying on the racetrack and crashes are common.The whole racetrack with obstacles is covered by net in order to provide safety to spectators, pilots, and staff. It is important that spectators have a good overview of the race. For this purpose, each pilot in a group has his own color which helps to identify him and his drone during the race. Large television or LED screen is used to show the spectators the FPV feed from the drone, info-graphics about the pilots and ongoing results of the race. The progress of the race is described by a commentator. The electronic timing system is used for measuring pilots performance. Each pilot is checked by the referee which tracks the drone and measures the pilot time for backup. FPV feed of each pilot is recorded for further analysis if any uncertainties occur. The results provided by timing system are presented to commentator and spectators online in order to provide instant results of the race.

1.1 Problem statement

The timing system is one of the biggest issues in drone racing. Drones fly several laps on the racetrack during one heat. The beginning and end of the lap is usually defined by a finish gate in which is timing system usually placed. Drones can fly through the finish gate with speeds over a 100$\frac{km}{h}$. The measured times are presented to a race director, pilots, and spectators and are used in qualification and finals. Drone racing events are organized all around the world on a different scale. There are large FPV race events, but most of the events are organized by smaller companies and drone enthusiasts who have limited resources.

Timing system must be able to detect drone passing through the gate, distinguish the drone from the others and calculate the time of the lap. The time difference between pilots laps might be lower then one second therefore at least precision in hundredths of a second is required. Timing system must detect the drone only when it flies exactly through the gate, near miss of the gate should not be counted. Robustness and reliability of the timing system are important since using backup timing methods causes a delay in the race and do not provide the results immediately. Timing system should be easy to use, easy to transport and cheap.

1.2 Related work

No research on drone racing timing system has been found. However number of open source project exists. Detection methods used by timing systems are more described in section 2.

Open source timing system based on infrared LEDs is OpenLap [19]. OpenLap is open source hardware and software project. The timing system consists of an infrared emitter controlled by attiny and receiver based on Arduino and ESP12E on later versions. This project might be a good inspiration for other timing systems using infrared diodes. Unfortunately, infrared timing systems are not used nowadays and OpenLap project is without any change for over two years. Easy Race Lap Timer is another example of infrared-based open source timing system [24]. Easy Race Lap Timer is a bit more complicated and wide than OpenLap. This timing system also uses infrared emitter controlled by attiny, but the communication protocol is a bit different. Easy Race Lap Timer uses Raspberry Pi as the brain of the project and supports web access and several racing modes. Easy Race Lap Timer as other infrared based timing systems is no longer under development.

Chorus RF Laptimer is one of open source timing systems based on received signal strength indicator (RSSI) [14]. RSSI based timing systems are alternative to infrared-based timing systems having the advantage that it does not require any additional equipment on the drone except video transmitter. Chorus RF Laptimer uses RX5808 analog video receiver and Arduino. These two parts together create a Chorus node, which is capable of tracking one drone. Each Chorus node measures an RSSI value and compares it with a threshold set up. If the RSSI value is above the threshold, the corresponding drone is considered passing a finish gate. Multiple Chorus nodes can be connected together and create one system capable of measuring more drones. Chorus timing system is controlled by a mobile application using Bluetooth or WiFi. Delta 5 Race Timer is the biggest open source RSSI based timing system. It works on a similar principle as the Chorus RF Laptimer. Each node consists of Arduino Nano and RX5808 video receiver module. Nodes communicate with Raspberry Pi via I2C. Delta 5 Race Timer provides web access and is supported by advanced race management software. Delta 5 Race Times is supported by MultiGP which is the biggest racing league in the world [17].

1.3 Contribution

We present a timing system that utilizes received signal strength indicator (RSSI) from analog video receiver module and two pairs of microphones. It consists of Nucleo-F303RE, analog video receiver module RX5808, and two microphone modules equipped with two microphones and amplifiers. Using the microphones, the timing system is able to determine if the drone passed through the gate or missed it and prevent the false-positive detections. Fusing RSSI data and data from microphones highly improves the accuracy of the timing system since the exact moment when the drone passes through the gate is estimated by Kalman filter more precisely. Our contribution beyond the state-of-the-art approach is summarized in following points:

- We implemented a custom algorithm for estimating time delay between two signals from microphones.
- We present a self-calibrating, precise and robust algorithm for drone detection utilizing RSSI signal and microphones.
- We designed a hardware platform specifically intended for racing drone detection.

2 Drone detection methods

Several approaches can be used in order to detect the drone passing through the finish gate. Each approach has its advantages and disadvantages. The most known commercially available timing systems are mentioned for each approach.

2.1 IR LED

Lap counting using infrared light emitting diodes (IR LEDs) is commonly used method in RC cars community. Many IR based timing systems for drone racing exist, having little modification from the existing RC car timing products [24, 21, 9]. Each drone carries an infrared emitter which constantly flashes a unique sequence. The timing system uses a series of infrared detectors mounted along one side of the finish gate connected to a computer. The detectors translate the flash sequence into an ID code which is paired with a pilot. It is expected that the emitter is crossing perpendicular to the detectors in order to work perfectly. This condition is nearly always true with RC cars bounded to the track surface by gravity, but drones can move freely in all dimensions. Crossing at an angle can advance or reduce the detection point and increase the time error [20].

IR detection sensors, decoder unit, computer and IR emitter are needed to set up an IR based timing system. Usually, special finish gate or stand must be built in order to protect and hold the sensors in place. IR emitters are typically small and light but have specific installation requirements. They must be attached to a power source on the drone, face the direction of the detectors as you cross the line, and nothing can be visually blocking the emitter. Since drones are getting lighter and smaller, it might be a problem to mount the emitter into the frame. The system itself usually cost a few hundred dollars, and individual IR emitters purchased from the system manufacturer often cost around 40$ each. Pilots must have the emitter on each drone or spend time removing and installing to another frame when crash.

2.2 RFID

Radio-frequency identification (RFID) tags are commonly used in timing systems for all kind of sports from running to motorcycle racing and RC racing is no exception. One of the RFID timing systems for drones is Mylaps [18]. Every drone has an RC transponder installed, that sends out a unique signal to the detection loop in the track in order to identify the pilot and record its results. RC Transponders are small and lightweight and have been used in the world of RC racing for many years. The decoder determines the exact time at which each transponder passes the detection loops. The decoder sends this data to the computer. Detection loops are embedded in the finish gate and at intermediate timing points along the track. A detection loop works as the system's antenna. It picks up signals from the transponders and passes them through to the decoder. Since drones are getting lighter and smaller, it might be a problem to mount the transponder into the frame. The decoder itself cost over 2800$, and individual transponder cost around 100$ each. Pilots must have the transponder on each drone or spend time removing and installing to another frame when crash.

2.3 Sound

Drones produce a characteristic sound which can be used for drone detection. Drone sound analysis is used mainly in drone defense systems [16, 8]. Rotorama Laptimer application for Android and iOS is available sound based timing system for drone racing [22]. Sound based timing system use amplitude of the sound as a main source of the date. When the drone flies close to the microphone, the amplitude of the sound raise over a certain threshold and lap is counted. Doppler effect or frequency analysis is not used in currently available sound based timing systems. Simplicity is the main advantage of the sound based timing system. None external components need to be mounted on the drone. The biggest disadvantage is that drones passing through the finish gate are not distinguishable, therefore only one drone can fly on the racetrack. The sound timing system is mainly used for training purposes only.

2.4 Video

Video cameras are used for drone detection even thought drones come in different size and shapes. Drone defense systems are the main industry which uses image recognition for drone detection [3, 6]. The video record of the finish gate might be used as a backup timing method in drone racing. The video record of the finish gate with a high number of frames per second analyzed by a referee is the most reliable and precise source of data. This method is used as a ground truth in experiments. Whoop Laps is available video-based timing system [13]. This timing system is an application for Android and iOS which utilizes phone camera and motion detection in order to detect drone flying through the finish gate. The video-based timing system is not capable of distinguishing drones from each other. Each pilot needs to have its own finish gate when racing together.

FPV video from the drone can be used as a source of information for measuring laps. Video from the drone is one of the key parts of autonomous racing drones [15, 2, 11]. There is no automatic FPV video based timing system available on the market, but measuring the lap times from FPV video by the referee on the stopwatch is commonly used method on smaller and bigger events. Measuring lap times by referee with the stopwatch is a robust and sufficiently reliable method. The main drawback is that it is difficult to process the results immediately by race management software and present them live and online. Also, this method is demanding on human resources.

2.5 RSSI

Timing systems based on the drone video signal strength are mostly used nowadays. The main part of each FPV racing drone regardless of its size is a video transmitter, therefore, none external component needs to be mounted on the drone. Analog video transmitters transmitting on 5.8GHz are used. Each transmitter is transmitting on fixed frequency chosen from 40 commonly used channels. Table of channels is shown in figure 4.

	Frequency [MHz]							
Band	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7	Channel 8
A	5865	5845	5825	5805	5785	5765	5745	5725
B	5733	5752	5771	5790	5809	5828	5847	5866
E	5705	5685	5665	5645	5885	5905	5925	5945
F	5740	5760	5780	5800	5820	5840	5860	5880
R	5658	5695	5732	5769	5806	5843	5880	5917

Figure 4: *Table of channels used for FPV*

Analog video receiver module is used for receiving the video signal. One of the most popular video receiver module widely used in FPV video goggles is RX5808. The receiver module RX5808 is shown in figure 5. The module is powered by 5V and can be controlled by SPI. Video signal, an audio signal, and RSSI signal are the outputs of the module.

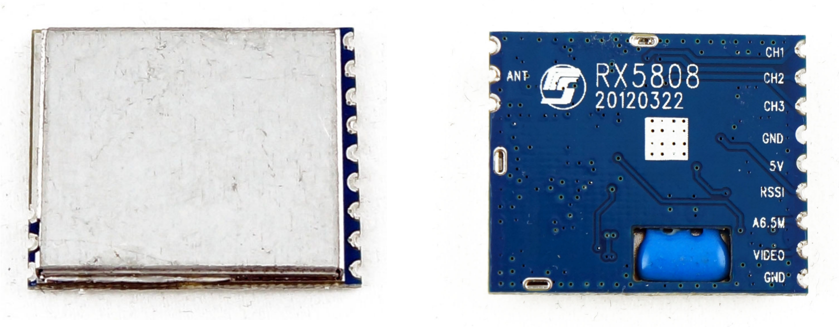


Figure 5: *Receiver module RX5808*

Receiver signal strength indicator (RSSI) signal is used for drone detection in RSSI based timing systems. RSSI correlates with the distance of the drone from the receiver when omnidirectional antennas are used. The timing system is usually a small box with video receivers and microprocessor. The timing system is placed in the middle of the finish gate. Video receiver is tuned to the frequency of the drone. RSSI signal raises when drone passes near the video receiver. The example of a measured RSSI signal is shown in figure

6. The RSSI signal is noisy and flat-top which makes it difficult to find a local extreme and estimate the moment when drone passes through the gate.

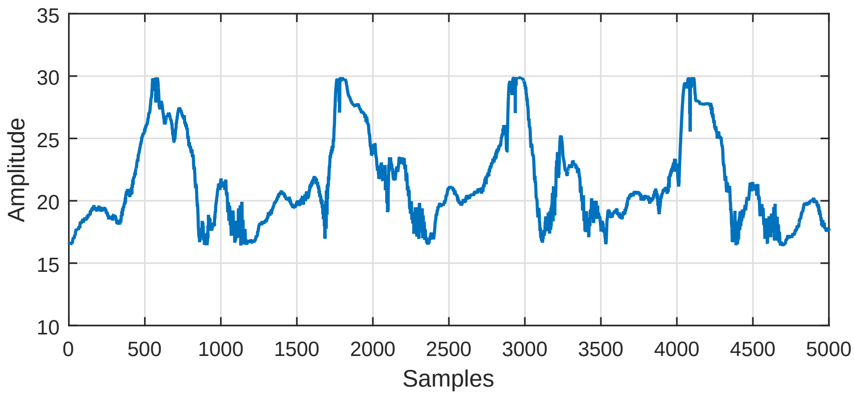


Figure 6: *RSSI signal*

This approach has its drawbacks which have to be considered during designing the race track. The drone can't fly close to the timing system anywhere on the race track except the finish line. None of the obstacles can be near the finish gate because passing the drone near the finish gate can cause false-positive reading of the timing system. This can be a big limitation on tight spaces like indoor tracks or tracks in the city center. Examples of badly designed track and correctly designed track are shown in figure 7.

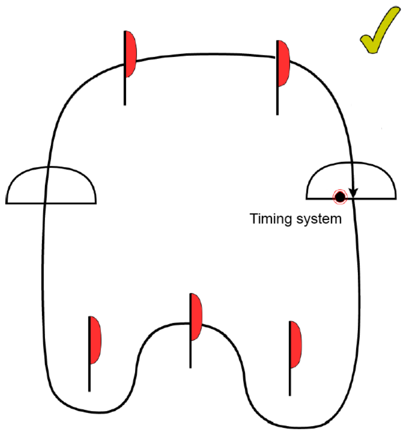


Figure 7: *Race Track*

There are several commercially available RSSI-based timing system solutions for training and professional use. One of the timing systems for personal use is LapRF from ImmersionRC [10]. It is a small standalone device with an integrated battery which connects with phone or tablet using Bluetooth. The results are shown in Android or iOS application. ImmersionRC LapRF has one custom video receiver module and supports up to 8 pilots at once. The retail price is 99$. The ImmersionRC LapRF is shown in figure 8a.

TBS EventTracker is race timing system made by company Team Black Sheep [23]. TBS EventTracker is used by professionals in bigger events mainly because of its complexity and price of 999$. The device needs an external power source and has to be connected to the computer or WiFi router via Ethernet cable. TBS EventTracker uses 8 separate video receiver modules and supports up to 8 pilots at once. The indicated accuracy is +/-10ms. TBS EventTracker is race management system that combines state-of-the-art transponderless lap timing hardware with a full suite of event management software. TBS EventTracker is shown in figure 8b.

(a) *ImmersionRC LapRF*

(b) *TBS Event Tracker*

Figure 8: *Timing Systems*

3 Enhanced drone detection

The goal is to developed timing system which will be accurate, reliable, small, cheap and wouldn't require any additional device to be mounted on the drone. From the mentioned drone detection methods, we choose to use RSSI and sound combined together. RSSI-based timing systems are commonly used in drone racing these days. Enhancing RSSI based timing system by microphones overcome the main drawbacks of pure RSSI based timing systems. Timing system based on RSSI signal and microphones is still small and easy to use.

The RSSI signal is measured by RX5808 analog video receiver module. One RX5808 is needed for each drone to be measured. The sound is measured by two pairs of microphones. Microphone module hardware with analog microphones and amplifiers are developed for microphone pair. Each pair of microphones is placed in the corner of the finish gate. Signals from the microphones are filtered and stored in a buffer. Sample shift between the two signals from a microphone pair is estimated. We can estimate direction to the source of the sound using estimated sample shift. Using two pairs of microphones placed in the corners of the gate, we can determine if the drone flies between the two microphone pairs or next to them therefore if the pilot flies through the gate or missed it. The maximum sound amplitude of the pair of microphones is estimated. Estimated sample shifts, sound amplitude, and RSSI signal are fused together using Kalman filter in order to estimate the moment when the drone passes through the gate more precisely. Estimated state by Kalman filter, estimated sample shifts, and RSSI signal are used by the drone detection algorithm, which decides whether the drone flies through the gate or not.

All calculations are made on Nucleo F303RE development kit. The Nucleo communicates with a computer using USB COM port. All algorithms were developed and tested in Matlab and then re-implemented in C using Mbed and flashed to the Nucleo board. The proposed solution serves as a proof of concept, it is not compatible with any race management software.

Schematic of the timing system based on RSSI and microphones are shown in figure 34 in the appendix. The parts of the system are described further in detail in next chapters.

3.1 Microphones

Racing drones have 4 motors with propellers which can spin up to 30 000RPM and produce characteristic sound. The amplitude of the sound signal change with distance of the drone to the microphone and revolutions of the motors. The record of the sound when the drone is passing through the gate is shown in figure 9. The dominant frequency of the sound is between 800Hz and 1kHz as can be seen in the spectrogram.

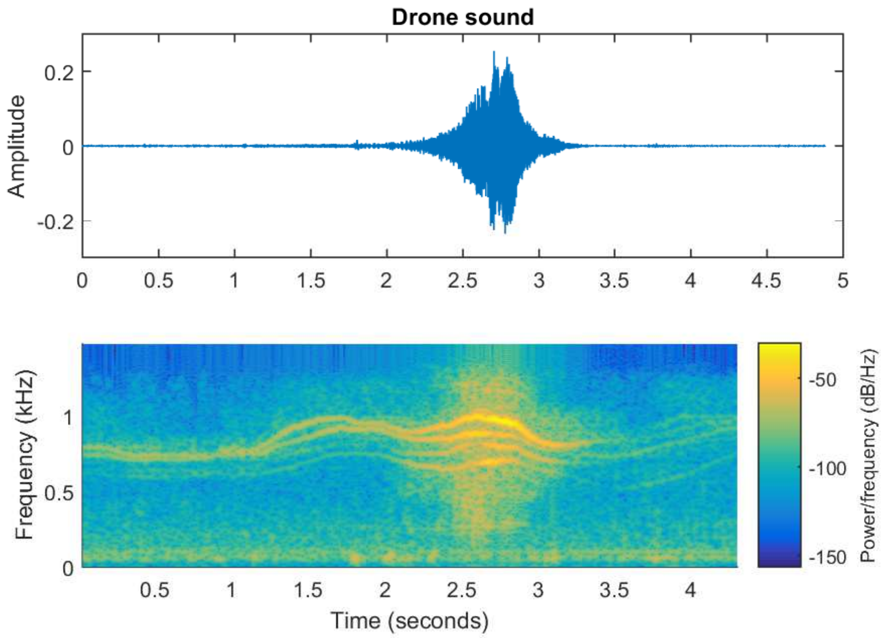


Figure 9: *Sound of the drone*

3.1.1 Filtering

The input signal from microphones has to be filtered in order to suppress noise of the background and maximize the signal to noise ratio. The drone produces sound with frequency from 800Hz to 1kHz during flight. Finite impulse response (FIR) or infinite impulse response (IIR) band-pass filter could be used [1]. FIR filter is defined in equation 1

$$y[n] = b_0x[n] + b_1x[n-1] + \cdots + b_Nx[n-N] = \sum_{i=0}^{N} b_i x[n-i] \tag{1}$$

where $y[n]$ is the output signal, $x[n]$ is the input signal, N is the order of the filter and b_i is the value of the impulse response at the i'th instant. FIR filter is always stable and requires no feedback, therefore, any rounding errors are not compounded by summed iterations. IIR filter is defined in equation 2

$$y[n] = \frac{1}{a_0}(b_0x[n]+b_1x[n-1]+\cdots+b_Px[n-P]-a_1y[n-1]-a_2y[n-2]-\cdots-a_Qy[n-Q]) \tag{2}$$

where $y[n]$ is the output signal, $x[n]$ is the input signal, P is the feedforward order of the filter, Q is the feedback order of the filter, b_i are the feedforward filter coefficients and a_i are the feedback filter coefficients. IIR filter is not always stable and requires feedback. The main advantage over FIR filters is efficiency in implementation. IIR filter can have much lower order than FIR filter to meet the same requirements.

FIR filter is used for sound filtration. FIR filter is easy to implement and always stable. The bigger delay is not an issue since we compare the signals relative to each other as described in section 3.1.2. The sampling frequency of the sound is 20kHz. FIR band-pass filter is designed using fir1 Matlab command. The order of the designed filter is 50. The frequency response of the filter is shown in figure 10.

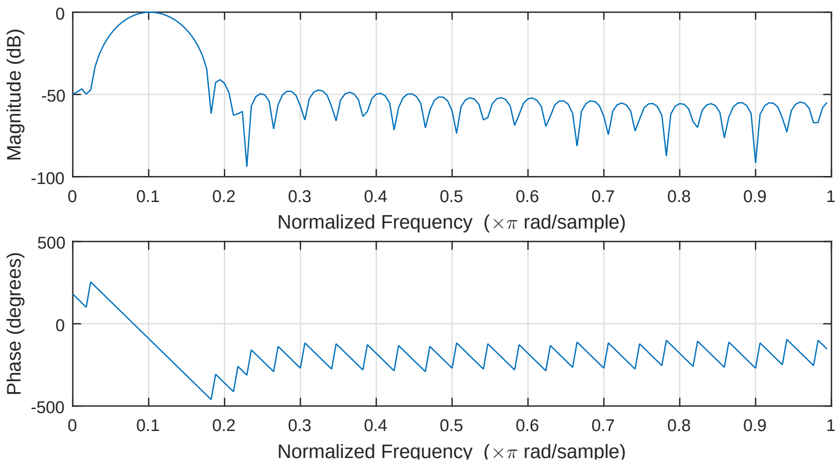
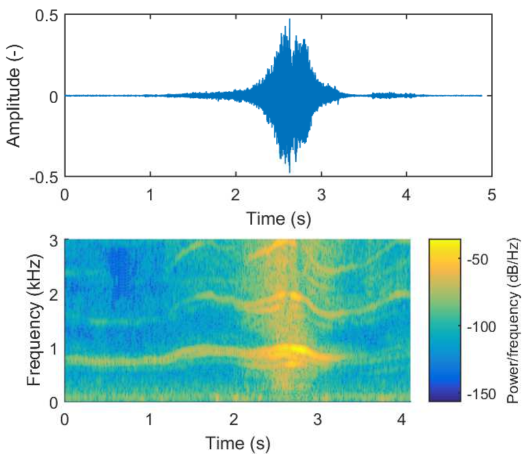
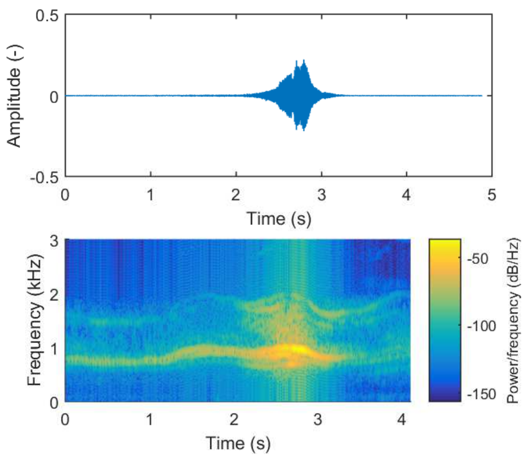


Figure 10: *FIR band-pass*

The performance of the filter is shown in figure 11. Raw data recorded directly by the microphone are shown in figure 11a. The base frequency between 800Hz-1kHz and unwanted higher harmonics can be seen in the spectrogram. The filtered data by proposed FIR filter are shown in figure 11b. The low-frequency noise and higher harmonics are suppressed and the data are more suitable for further processing.



(a) *Raw sound data*



(b) *Filtered sound data*

Figure 11: *Drone sound*

3.1.2 Time delay

It is possible to estimate the direction of the incoming sound from a sound source using at least 2 microphones. We have two microphones at positions M1 and M2. Let's consider racing drone as a source of the sound. Let's assume that the distance between the microphones and the sound source is much greater than the distance between the two microphones, therefore, we can model the sound approaches as a plane wave and thus the angle of which it approaches the microphones is constant. The sound from the drone arrives at the microphones at two different times as shown in figure 12. The signal from the source arrives at different microphones at times proportional to their distance. The time delay $\Delta t = t_1 - t_2$ corresponds to the direction of the sound source. The direction of the sound source can be estimated just in half plane using this microphone configuration. The estimated time delay Δt can be used for determining whether the drone flew through the gate or missed it as described in section 3.2.

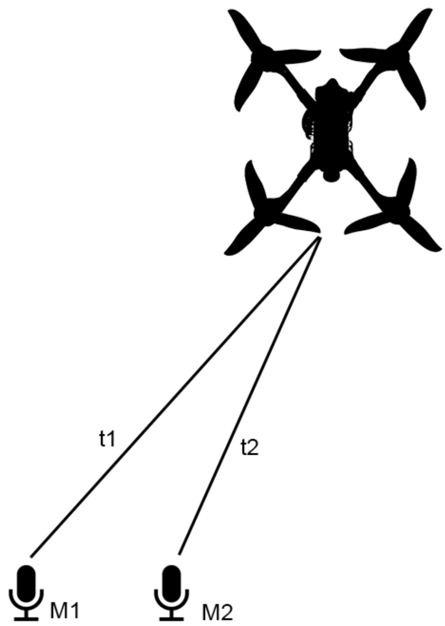


Figure 12: *Drone as a source of sound*

The time delay Δt has to be less than $\frac{1}{2}T_{min}$ of the sound signal, where T_{min} is the minimal period of the sound signal. If we consider the drone as a harmonic sound signal source with the maximum frequency of 1.5kHz and denote the speed of sound equal $345\frac{m}{s}$, the maximum distance between microphones could be 115mm according to the equation 3. The distance between microphones is set to 9cm since it is easier and cheaper to manufacture 10cm wide printed circuit board (PCB). 1.9kHz is the maximal frequency of the signal for which we can estimate ΔT. The sampling frequency is set to 20kHz. The maximum difference between sound signals from microphones is 5 samples according to equation 4.

$$D_{max} = \frac{1}{F_{max}} \cdot \frac{V_s}{2} \tag{3}$$

$$S_d = \frac{D_m \cdot F_s}{V_s} \tag{4}$$

D_{max} is the maximal distance between microphones, F_{max} is the maximal frequency of the sound, V_s is the speed of sound, S_d is the difference in samples, D_m is the distance between microphones and F_s is sampling frequency.

The time delay between two signals from microphones $\Delta t = t_1 - t_2$ needs to be estimated. Lets denote $x_1[n]$ as signal from the first microphone and $x_2[n]$ as signal from the second microphone. The example of received signals with frequency of 1kHz is shown in figure 13. The common method for estimating time delay between two signals is cross-correlation algorithm [4, 26, 12]. Cross-correlation is shown in equation 5. The formula essentially slides the $x_2[n]$ signal along the x-axis, calculating the sum of their product at each position. The value of n that maximizes $Y_{cc}[n]$ leads to approximation $\Delta\tilde{T} = n \cdot F_s$.

We proposed a new approach for estimating time-shift between two signals. The proposed approach is using least mean squares method shown in equation 6. The formula essentially slides the $x_2[n]$ signal along the x-axis, calculating the sum of their squared difference at each position. The value of n that minimizes $Y_{lms}[n]$ leads to approximation $\Delta\tilde{T} = n \cdot F_s$.

$$Y_{cc}[n] = \sum_{m=0}^{M} x_1[m] \cdot x_2[m+n] \tag{5}$$

$$Y_{lms}[n] = \sum_{m=0}^{M} (x_1[m] - x_2[m+n])^2 \tag{6}$$

Buffers of 100 samples are used for time-shift detection. 100 samples, which corresponds to 5ms, are recorded. Shift up to 5 samples is checked by the algorithm because 5 samples are maximum sample shift between the buffers according to the equation 4. Using these parameters we get sample shift estimated by least mean squares by equation 7 and sample shift estimated by cross-correlation by equation 8. Comparison of cross-correlation method and proposed least mean squares method used on a different source signal frequency and amplitude is shown in figure 14. Proposed least mean square method has better results when signals from microphones have lower amplitude or frequency. Estimating the sample shift of the signals using least mean squares will be further referred as sample shift estimator.

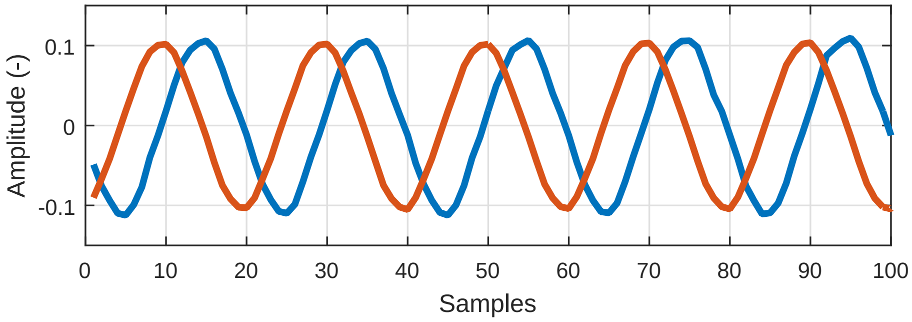
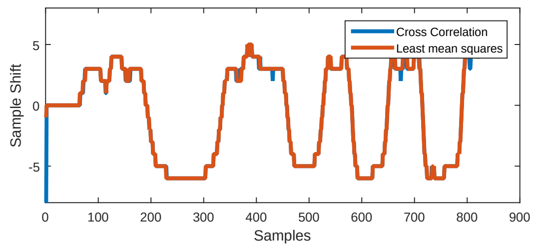


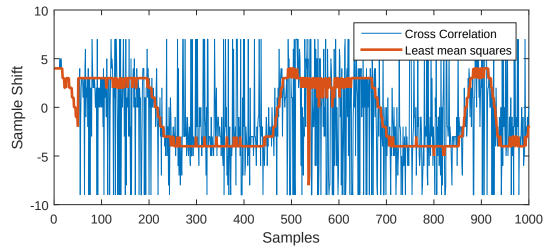
Figure 13: *Microphone signals*

$$S_{lms} = \underset{n \in <0,10>}{\arg\min} \left(\sum_{m=0}^{95} (x_1[m+5] - x_2[m+n])^2) \right) - 5 \tag{7}$$

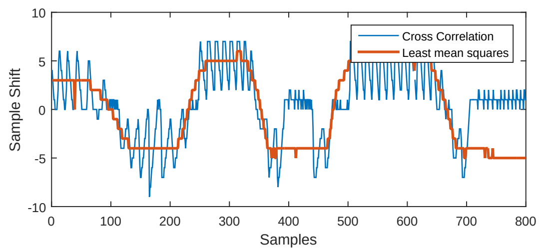
$$S_{cc} = \underset{n \in <0,10>}{\arg\max} \left(\sum_{m=0}^{95} x_1[m+5] x_2[m+n] \right) - 5 \tag{8}$$



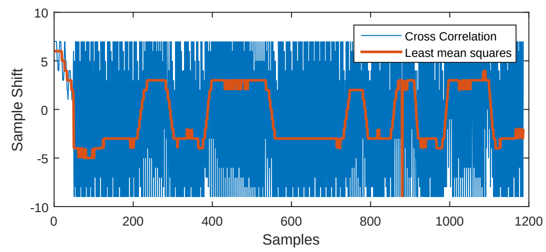
(a) 1000Hz signal



(b) 1000Hz signal with low amplitude



(c) 800Hz signal



(d) 500Hz signal

Figure 14: *Comparison of least mean squares and cross correlation*

The maximal amplitude of the sound signal A_{max} is calculated from the buffers as shown in equation 9.

$$A_{max} = \frac{\max(x_1[n]) + \max(x_2[n])}{2} \tag{9}$$

Performance of sample shift estimator highly depends on the quality of input signal. When signal obtained from the microphones has a low amplitude or is just a noise, the output of sample shift estimator is very noisy. The noise has to be filtered out in for further use of estimated sample shift. The maximal amplitude of the sound signal is used for this purpose. Maximal amplitude of the sound signal is compared with a fixed threshold. The output of the sample shift estimator is set to zero when the maximal amplitude of the sound signal A_{max} is lower than the threshold. The raw output from sample shift estimator and filtered output of sample shift estimator is shown in figure 15.

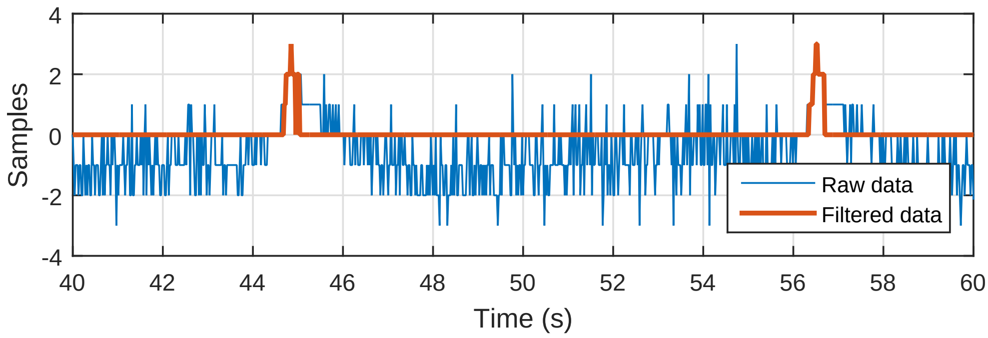


Figure 15: *Filtered sample shift estimator*

3.1.3 Microphone modules

Microphone modules are created as sensors for sample shift estimator. Analog electret microphones are used in microphone modules. They are cheap, easy to use and easy to solder. The signal from electret microphone is amplified by an LM386 amplifier. The LM386 amplifier has the gain fixed gain set to 50. This value has been determined empirically by several experiments. The output signal of the amplifier is lowered in order to be used with A/D converter with input range 0-3.3V. The schematics is shown in figure 16. The first prototype of microphone module is shown in figure 17a. The basic functionality of the microphone modules was tested on the first prototype. When the prototype confirms it is suitable for its purpose, PCB was designed and manufactured. The PCB is 100mm wide, 23mm long and 1.6mm thick. The PCB is designed for two electret microphones with two LM386 SMD amplifiers. The PCB microphone module is shown in figure 17b. The housing for the PCB was designed in Fusion 360 in order to protect the microphone module and improve the overall look. The housing is shown in figure 17b. The housing was 3D printed with ABS plastic. The microphone module is connected by 4-pin microphone XLR connector.

Microphone modules are connected to the Nucleo-F303RE. Nucleo-F303RE is used for sampling the microphone signals and computing the sample-shift estimator.

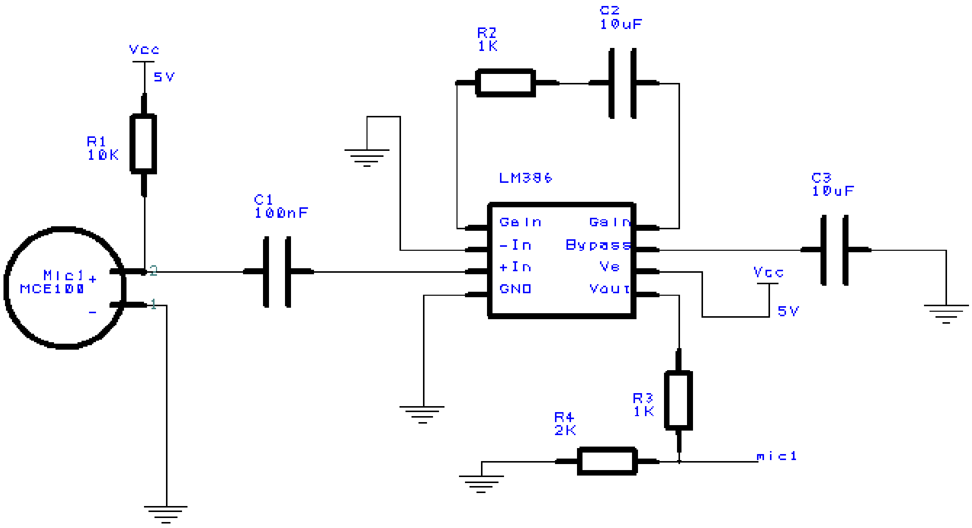
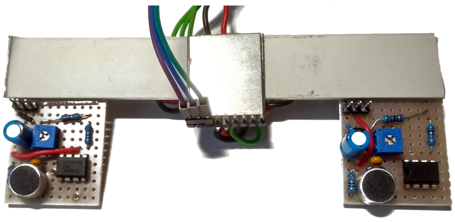
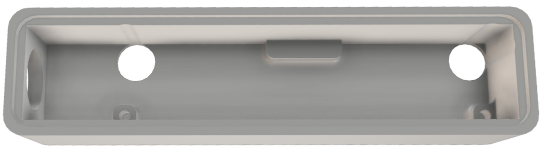


Figure 16: *Schematics of microphone module*



(a) *Microphone module prototype*

(b) *Microphone module PCB*



(c) *Microphone module housing*

(d) *Final microphone module*

Figure 17: *Microphone module*

3.2 Drone Detection

Drones are racing on a racetrack made of several obstacles. Drones fly several laps on the racetrack during one heat. The time of each lap has to be measured. Drones have to fly through the same defined space in order to establish beginning and end of the lap. The beginning and end of the lap is usually defined by a finish gate. The timing system is placed in the finish gate. The timing system has to detect drone and define the moment when the drone flies through the finish gate. The RSSI output of the video receiver module described in section 2.5 and microphone modules used for sample shift estimator described in section 3.1 are used as a source of the data for drone detection algorithm in this thesis.

The video receiver module RX5808 is placed in the middle of the gate. Two microphone modules are placed at both corners of the gate. The microphones are aligned with the plane of the gate. Nucleo-F303RE is used to run the drone detection algorithm. The schematics is shown in figure 18.

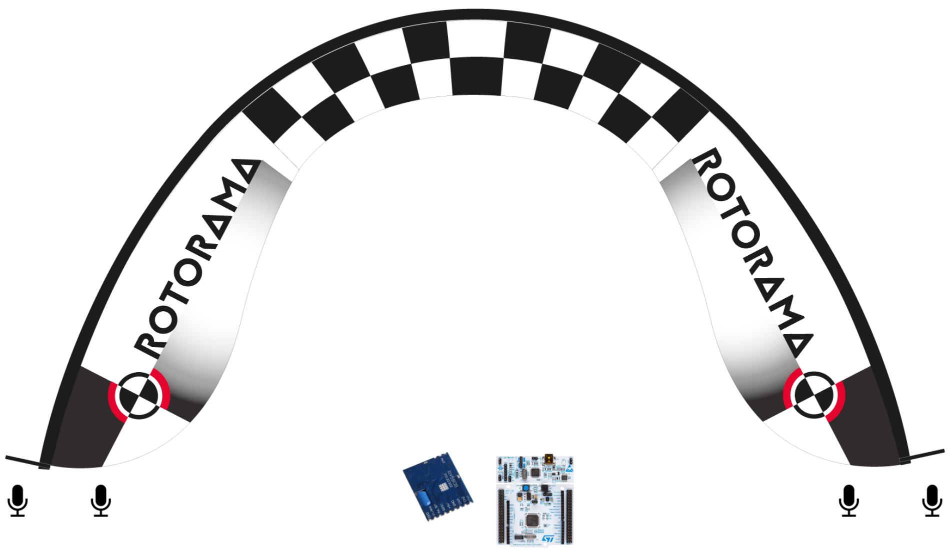


Figure 18: *Gate and sensors arrangement [22]*

When drone flies through the gate, there is a peak in the RSSI signal and peak in the maximal amplitude of microphone signals. The direction to the drone from the microphone module can be estimated using sample shift estimator. Using the estimated sample shift, it can be decided if the drone passes the microphone module from the right side or left side thus it can be decided if the drone has flown through the gate or missed it using two microphone modules with sample shift estimators at the corners of the finish gate. The data from video receiver module's RSSI output, maximal amplitude of the sound signal of both microphone modules and estimated sample shift of both microphone modules are shown in figure 19.

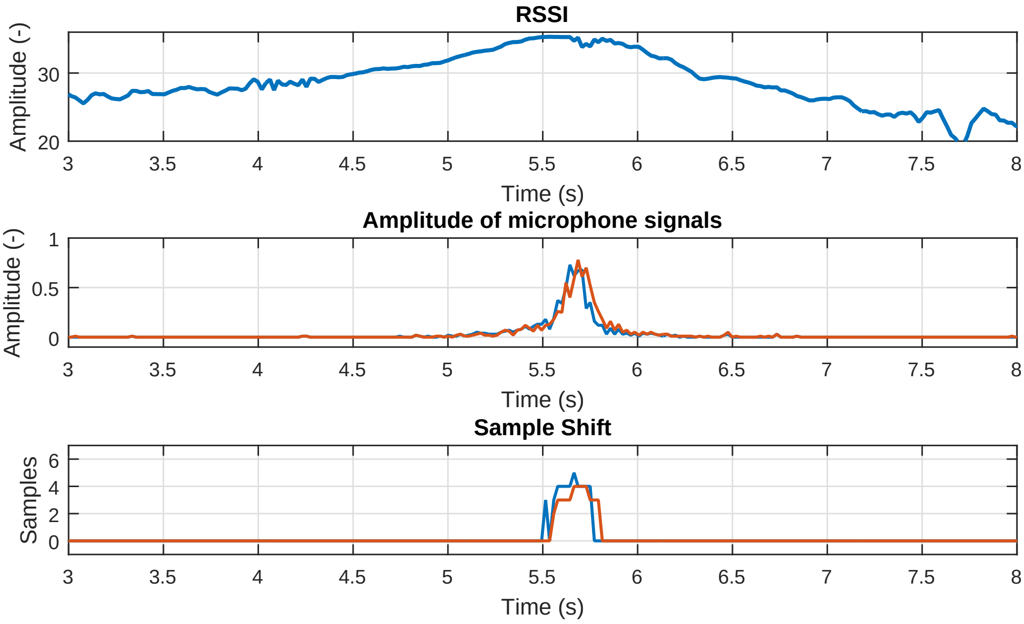


Figure 19: *Gate sensor data*

3.2.1 Kalman's filter

The main sensor for detecting the drone is the video receiver module. The video receiver is tuned to the frequency of the drone and RSSI output is independent of other drones. The RSSI output is noisy and saturated when the drone is passing through the finish gate. The flat-top of the signal complicates determining the moment when the drone flies through the finish gate. The amplitude of the signal from microphone corresponds to the distance between the microphone and the drone. The maximum amplitude of the sound signal from microphone modules has sharp peaks when the drone flies through the finish gate. This peak can increase the precision of estimating the moment when the drone flies through the finish gate. The estimated sample shift can be used to estimate the moment when the drone flies through the finish gate since the sample shift is the biggest when the drone is aligned with the microphones. Kalman's filter is used in order to filter and fuse data from video receiver module, sample shift estimator and maximal amplitude of the sound signal and prepare data suitable for drone detection.

Kalman filter is an algorithm that uses series of measurements observed over time and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone. The algorithm works in a two-step process. In the prediction step, the Kalman filter produces estimates of the current state variables. When the next measurement is observed, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty. It can run in real time, using only the present input measurements and the previously calculated state and its uncertainty matrix [25].

The data from figure 19 filtered by Kalman's filter with raw RSSI data are shown in figure 20. Filtered data are less noisy and have sharp maximum which helps to determine the moment when drone passed through the gate. The Kalman's filter matrices are described by equations 10, 11 and 12.

$$x_k = \Phi x_{k-1} \quad z_k = H x_{k-1} = \begin{pmatrix} R_k & A_{1k} & A_{2k} & S_{1k} & S_{2k} \end{pmatrix}^T \tag{10}$$

$$\Phi = 1 \quad H = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \end{pmatrix}^T \tag{11}$$

$$Q = 0.05 \quad R = \begin{pmatrix} 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{12}$$

where x denotes a state vector, z denotes measurement vector, Φ denotes a state transition matrix, H denotes a measurement matrix, R is signal from video receiver module, A_1 and A_2 are sound amplitudes from two pairs of microphone modules and S_1 and S_2 are signals from two sample shift estimators.

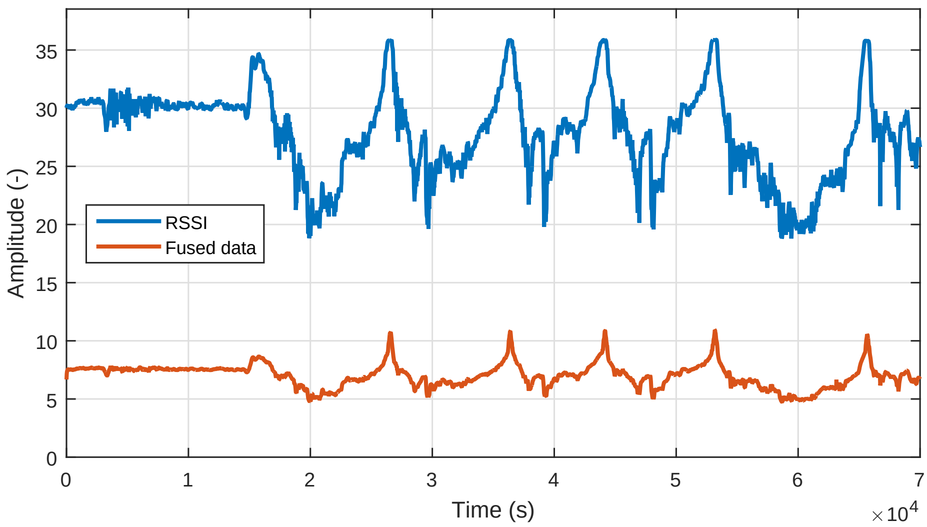


Figure 20: *Kalman's filter output*

3.2.2 Detection Algorithm

The moment when the drone flies through the gate has to be detected and the time of the pass has to be saved. The measured lap time is the difference between times of last two passes. The RSSI data, estimated sample shift, and Kalman's filter output are used for drone detection algorithm. The detection algorithm runs with a frequency of 50Hz.

The main data source for drone detection is the RSSI output of the video receiver module. The RSSI output is independent of other drones on a racetrack tuned to another video frequency. The RSSI output reaches similar local maximum values when the drone flies through the finish gate, therefore it is easier to set the proper threshold and detect the drone. Unfortunately, the value of local maximums varies based on environment, video receiver module, and drone video transmitter power. The RSSI output is filtered by first-order IIR low-pass filter $y(n) = 0.8y(n-1) + 0.2x(n)$ in order to suppress the noise. Performance of the filter is shown in figure 21. Filtered RSSI data by the IIR filter $RSSI_f(n) = 0.8RSSI_f(n-1) + 0.2RSSI(n)$ are used in drone detection algorithm further on.

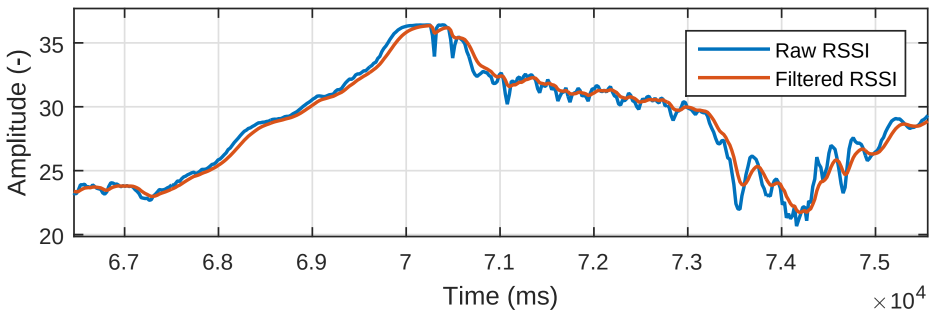


Figure 21: *Performance of RSSI filter*

Drone detection algorithm uses the output of the sample shift estimator to verify that the drone flies through the finish gate. The output of the sample shift estimator is filtered by first-order IIR low-pass filter $y(n) = 0.65y(n-1) + 0.35x(n)$ in order to smooth the signal. The performance of the filter is shown in figure 22. Filtered sample shifts $S_{1f}(n) = 0.65S_{1f}(n-1) + 0.35S_1(n)$ and $S_{2f}(n) = 0.65S_{2f}(n-1) + 0.35S_2(n)$ are used further on. The filtered sample shift are multiplied and gate indicator is introduced as $G(n) = S_{1f}(n) \cdot S_{2f}(n)$. Gate indicator is positive when the drone flies through the finish gate, negative when the drone flies next to the finish gate and change from negative to positive and then back to negative when the drone flies in front of the finish gate. The local maximum and minimum are detected when gate indicator has non zero values as shown in figure 23.

The detection algorithm uses upper threshold Th_{up}, detection threshold Th_{det} and hysteresis threshold Th_{hyst}. Several constants are used for the thresholds calculation. The constants $RSSI_{MaxLimit}$, $RSSI_{DetLimit}$ and $RSSI_{HystLimit}$ are described by equation 13, 14

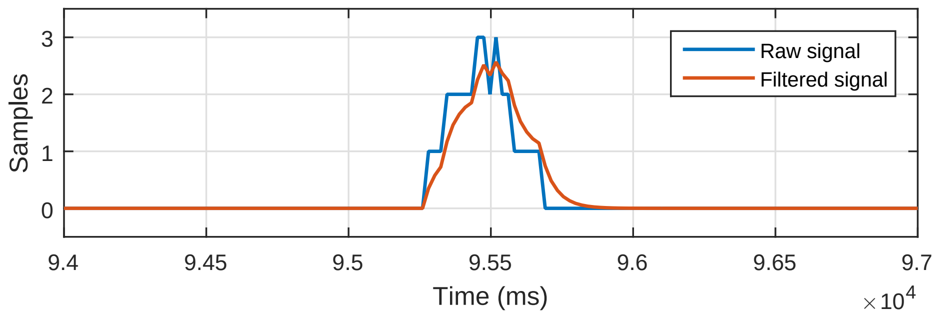


Figure 22: *Performance of sample shift filter*

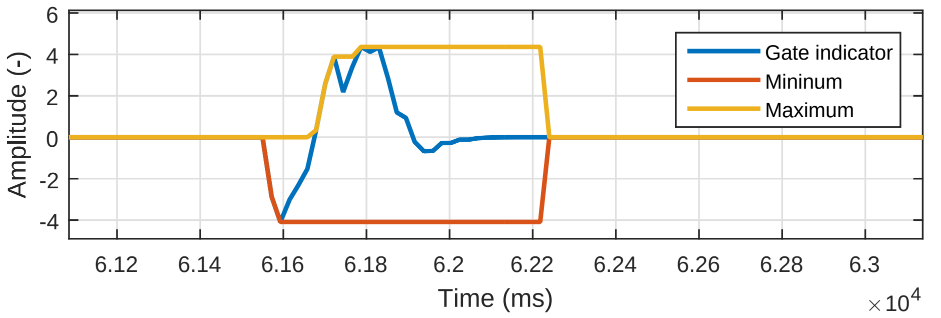


Figure 23: *Gate indicator*

and 15. Parameter S stands for sensitivity and parameter H for hysteresis where $S \in \langle 0; 1\rangle$ and $H \in \langle 0; 1\rangle$. The drone detection algorithm can be tuned by changing these parameters.

$$RSSI_{MaxLimit} = 1.01 + 0.3S \tag{13}$$

$$RSSI_{DetLimit} = 0.99 - 0.3S \tag{14}$$

$$RSSI_{HystLimit} = 0.98 - 0.3S - 0.68H + 0.3S \cdot H \tag{15}$$

Upper threshold Th_{up} is described by equation 16. Detection threshold Th_{det} is described by equation 17. Hysteresis threshold Th_{hyst} is described by equation 18.

$$Th_{up} = RSSI_{MaxLimit} \cdot RSSI_{max} - (RSSI_{MaxLimit} - 1)RSSI_{min} \tag{16}$$

$$Th_{det} = RSSI_{DetLimit} \cdot RSSI_{max} + (1 - RSSI_{DetLimit})RSSI_{min} \tag{17}$$

$$Th_{hyst} = RSSI_{HystLimit} \cdot RSSI_{max} + (1 - RSSI_{HystLimit})RSSI_{min} \tag{18}$$

The drone detection algorithm loop is shown in figure 25. Absolute maximum $RSSI_{max}$, absolute minimum $RSSI_{min}$ and local maximum $RSSI_{lapMax}$ are checked. The local maximum of Kalman's filter output Kal_{max} is checked. When the new local maximum is found, gate indicator maximum, gate indicator minimum and time are saved. The time is counted as the number of milliseconds since the timing system is turned on. The time is saved in variable Kal_{time}. The filtered RSSI value $RSSI_f$ and local maximum $RSSI_{lapMax}$ are compared with thresholds. The loop starts again when the condition $RSSI_f < Th_{hyst} \wedge RSSI_{lapMax} > Th_{det}$ is false. If the condition is true, gate indicator saved maximum and minimum is checked. Both maximum and minimum need to be positive. When this condition is false, the local maximum is set to zero and the loop starts from the beginning. When both maximum and minim are positive, condition $RSSI_{lapMax} < Th_{up}$ is tested. The valid lap is detected and time calculated when the condition is true. Time of the lap is calculated as a saved time of the found Kalman's filter output Kal_{time} minus saved the time of the last detected drone passes referred as lap tip. The valid lap is detected when local RSSI maximum $RSSI_{lapMax}$ is between thresholds Th_{det} and Th_{up} and when $RSSI_f$ is lower than Th_{hyst}. When $RSSI_{lapMax} >= Th_{up}$, the current lap does not count as valid because last detected drone pass has not been a drone pass through a gate but rather just a pass near the gate. This detected drone pass is set as a new lap tip. The local maximum is set to zero and the algorithm starts to measure a new lap. RSSI data with detection thresholds are shown in figure 24.

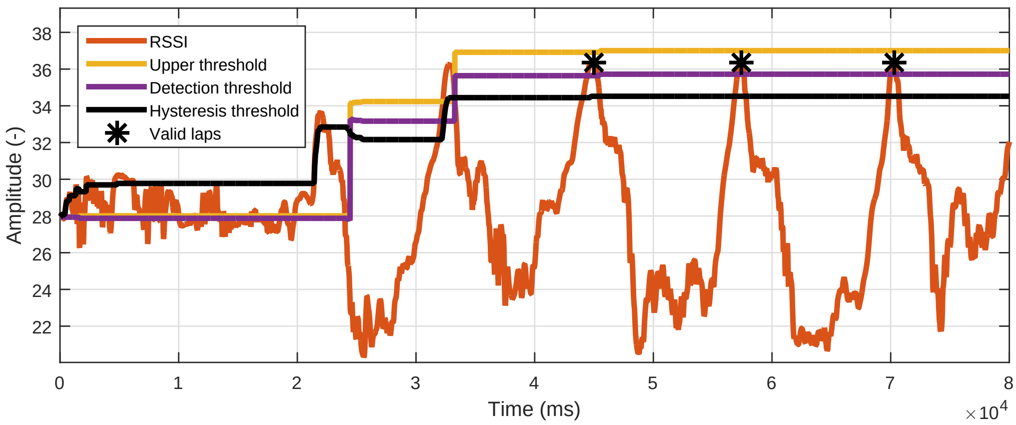


Figure 24: *Drone detection algorithm data*

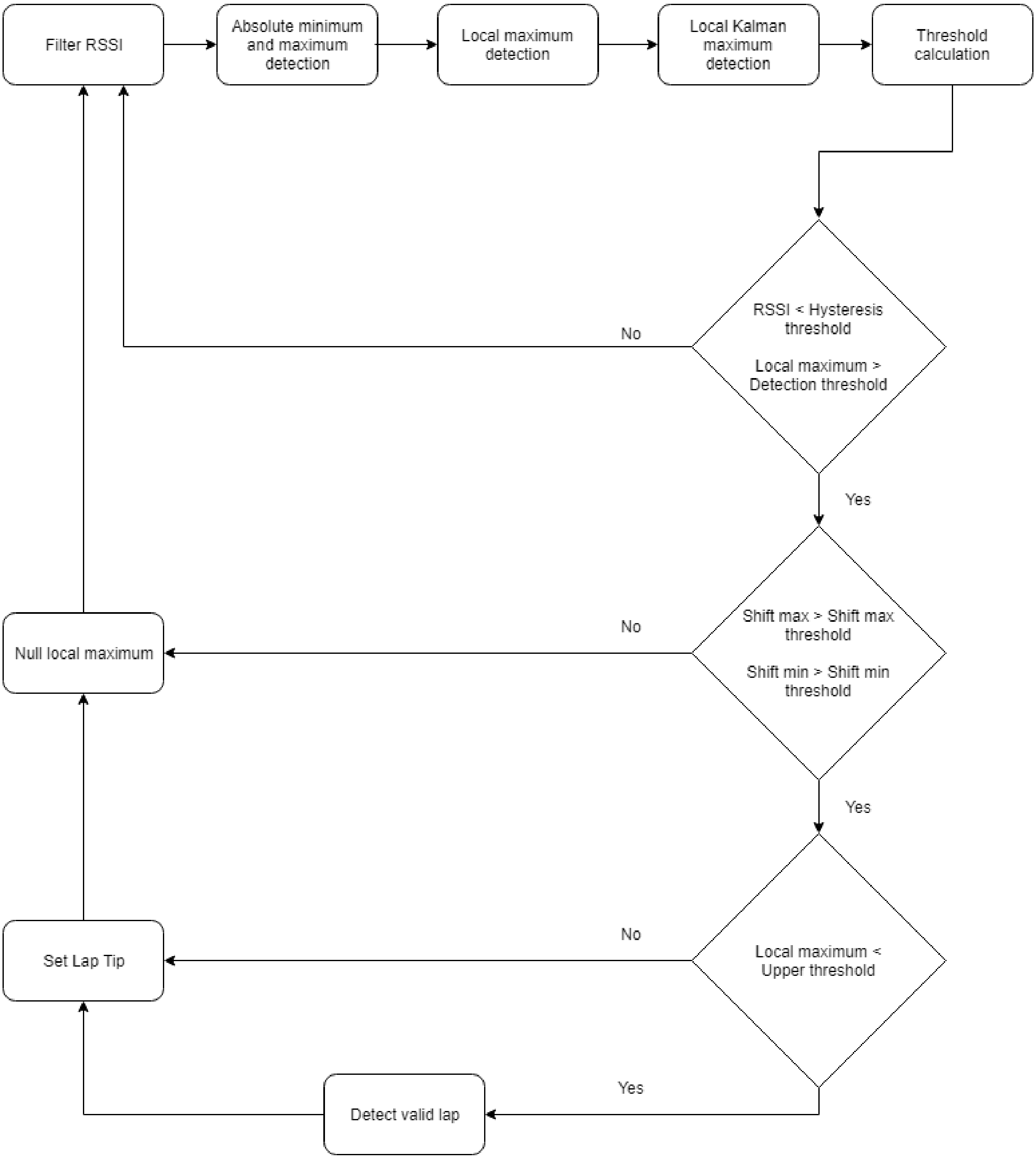


Figure 25: *Drone detection algorithm diagram*

4 Experimental Results

Performance of various timing systems is tested in several experiments and compared with the enhanced method for drone detection proposed in this thesis.

4.1 Time experiment

The capability of measuring the lap precisely is tested in this experiment. Round racetrack with several obstacles and a finish gate was built. Lap times have been measured by TBS EventTracker [23], stopwatch using the FPV feed from the drone and proposed drone detection algorithm using just RSSI data and using RSSI data along with Kalman's filter output. GoPro action camera pointed to finish gate was used for precise measuring the lap times. The video frames used for measuring the lap time is shown in figure 26. The GoPro was set to 240fps. Lap times measured from the video captured by GoPro were used as ground truth. Ground truth and laps measured by other methods are compared and subtracted. The differences between ground truth and measured lap times are statistically evaluated.

Figure 26: *Gopro video frames*

TBS EventTracker is race timing system made by company Team Black Sheep [23]. TBS EventTracker is used by professionals in bigger events. The indicated accuracy is +/-10ms. The retail price is 999$. 276 laps were measured by TBS EventTracker. 7 outliers with errors from 0.54s to 0.84s were measured by the EventTracker which makes 2.5% of all measurements inaccurate. The outliers were filtered out for further statistical evaluation.

The histogram of errors with fitted normal distribution is shown in figure 28a. Median of the errors is -0.005s, mean is 0.0019s and the standard deviation is 0.0675.

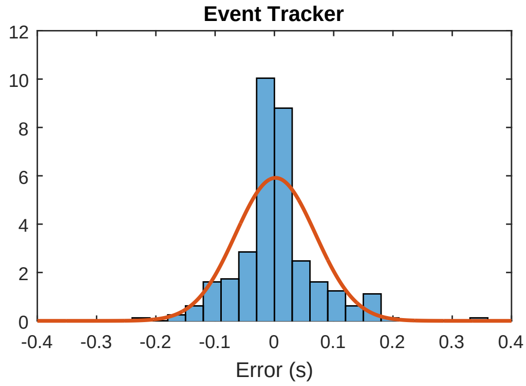
Using a stopwatch to measure lap times is widely used method since humans are more reliable in drone detection than timing systems. Stopwatches are also used on small events when organizers don't have access to any timing system. The referee is measuring the lap times using stopwatch and FPV video from the drone. The video frames from the drone are shown in figure 27.

Figure 27: *DVR video frames*

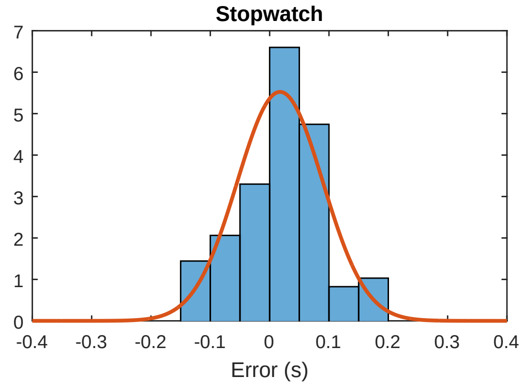
97 laps were measured by stopwatch and used for statistical evaluation. The histogram of errors with fitted normal distribution is shown in figure 28b. Median of the errors is 0.024s, mean is 0.0175s and the standard deviation is 0.0722.

The proposed drone detection algorithm can be used without microphones using only the RSSI signal. 106 laps were measured by proposed drone detection algorithm using only the RSSI data. The histogram of errors with fitted normal distribution is shown in figure 28c. Median of the errors is -0.005s, mean is 0.0005s and the standard deviation is 0.0792.

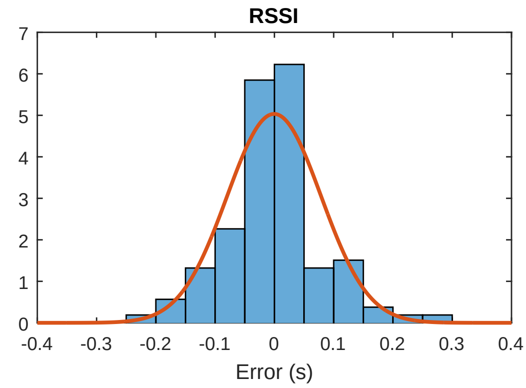
106 laps were measured by proposed enhanced drone detection algorithm using microphones. The histogram of errors with fitted normal distribution is shown in figure 28d. Median of the errors is 0.003s, mean is -0.0012s and the standard deviation is 0.0296.



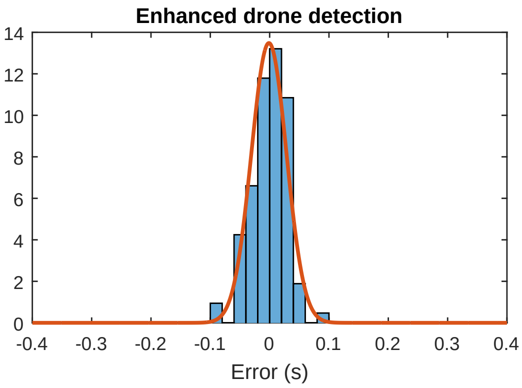
(a) *Time errors of TBS EventTracker*



(b) *Time errors of stopwatch*



(c) *Time errors of proposed algorithm - RSSI*



(d) *Time errors of proposed algorithm*

Figure 28: *Time experiment measured data*

Estimated probability density function of errors are shown in figure 29. Method results are shown in table 1. TBS EventTracker is the biggest disappointment considering it is 999$ timing system designed for professional race organizers. The indicated accuracy of +/-10ms was not confirmed by this test. 2.5% of the measurements were outliers with errors greater than 500ms. Surprisingly good results were obtained by stopwatch. We expected that errors in hundreds of millisecond would be common, since human is not so fast and make mistakes. Measuring lap times using stopwatch seems to be as precise as measuring lap times using RSSI based timing systems. Proposed drone detection algorithm has similar results as TBS EventTracker and stopwatch when using the RSSI output of video receiver module only. Best precision was achieved by enhanced drone detection algorithm with microphones. Finding maximum of Kalman's filter output instead of RSSI signal lowers the standard deviation from 0.0792 to 0.0295. Standard deviation 0.0295 is satisfactory since the drone detection algorithm is running on 50Hz.

	Median (s)	Mean (s)	STD (s)
EventTracker	-0.005	0.0019	0.0675
Stopwatch	0.024	0.0175	0.0722
RSSI algorithm	-0.005	-0.0005	0.0792
Enhanced algorithm	0.003	-0.0012	0.0295

Table 1: Time experiment results

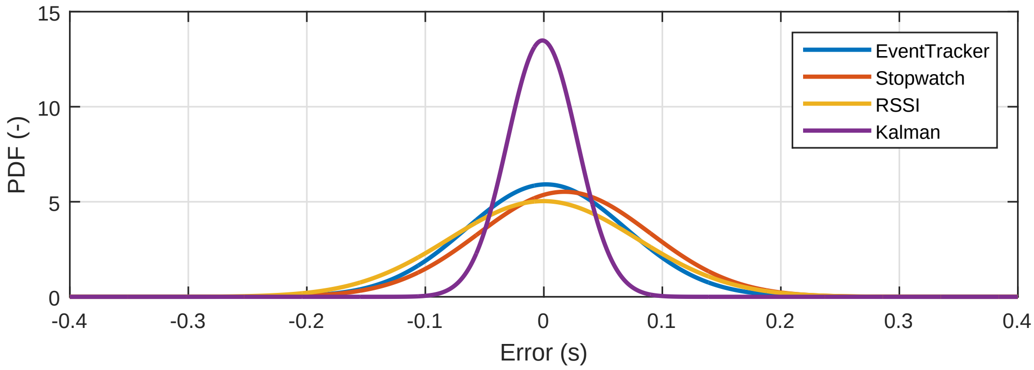


Figure 29: *Time errors of different methods*

4.2 False detection experiments

One of the main drawbacks of pure RSSI-based timing systems is a high number of false-positive detections of the drone. When the drone flies near the finish gate, the RSSI signal raises in amplitude and may trigger the timing system. The ability of the proposed method using microphones to resist false-positive detections is tested and compared with other methods. Counting the number of laps and measuring the lap times by human using stopwatch and the video feed from the drone is the most reliable method. This method is used as a ground truth in these experiments. Lap times PV was measured by TBS EventTracker, proposed drone detection algorithm using just RSSI data and proposed enhanced drone detection algorithm using RSSI data along with Kalman's filter output and output from sample shift estimators.

4.2.1 Parallel miss

Elliptic race track with one gate aligned with the finish gate in distance of five meters was built for this experiment. The RSSI value with marked valid laps and gate indicator value from one of the experimental flights are shown in figure 31. The local maximums of the RSSI signal are similar and difficult to distinguish.

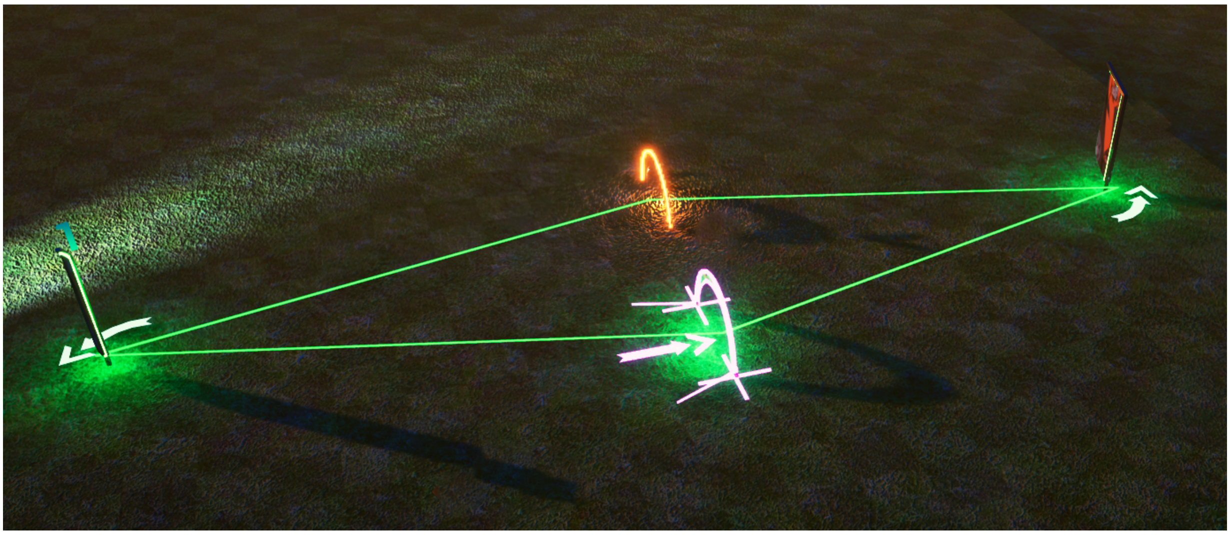


Figure 30: *Simulated experimental track*

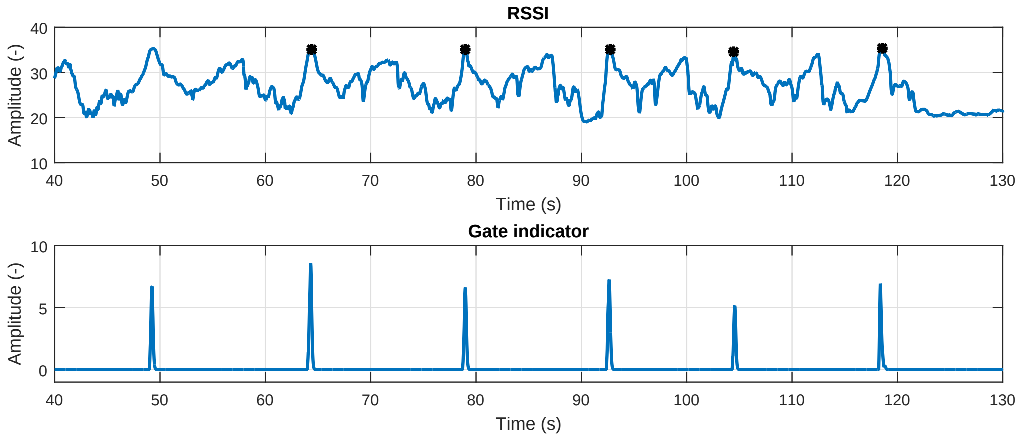


Figure 31: *Parallel miss experiment data*

TBS EventTracker has detected two times more laps than was flown. Every pass through the finish gate and the second gate triggered the timing system which counted lap. Performance of the proposed drone detection algorithm using just RSSI data depends on the value of sensitivity parameter. When sensitivity is low, the algorithm is able to detect laps correctly. For high sensitivity parameter, the algorithm detects when the drone flies through the finish gate and the second gate. The proposed drone detection algorithm detected all laps correctly using the output of the sample shift estimator. When drone flies through the second gate, the maximal amplitude of the sound signal is too low and output of the sample shift estimator is zero, therefore, gate indicator is zero and false-positive detections are eliminated.

4.2.2 Perpendicular miss

Eight-like shaped race track with one gate perpendicular to the finish gate in distance of three meters was built for this experiment. The RSSI value with marked valid laps and gate indicator value from one of the experimental flights are shown in figure 33.

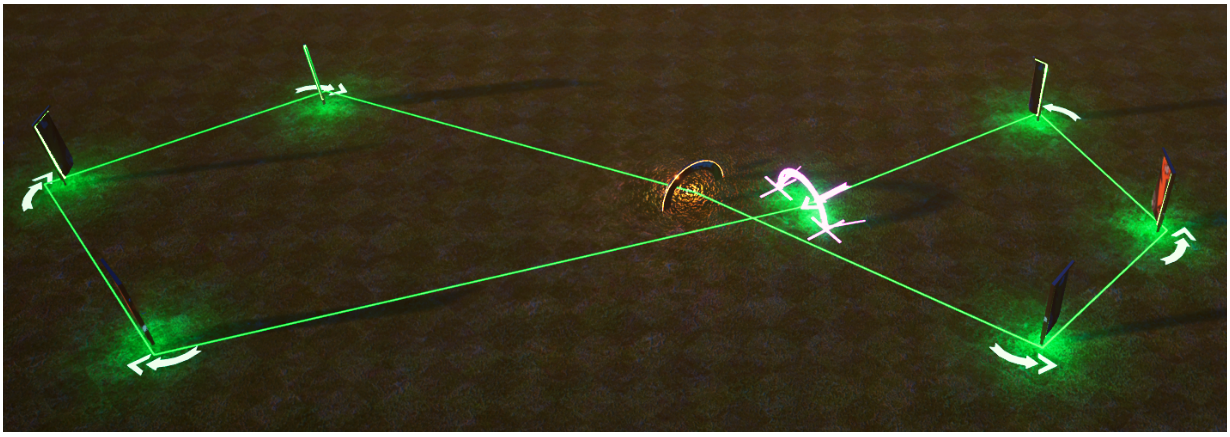


Figure 32: *Simulated experimental track*

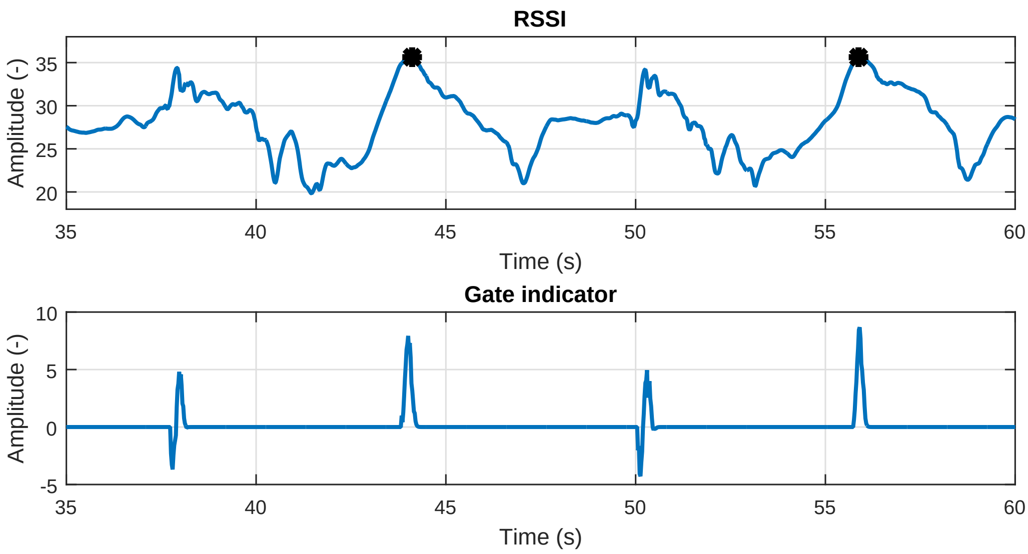


Figure 33: *Perpendicular miss experiment data*

TBS EventTracker has detected two times more laps than was flown. Every pass through the finish gate and the second gate triggered the timing system which counted lap. Performance of the proposed drone detection algorithm using just RSSI data depends on the value of sensitivity parameter. When sensitivity is low, the algorithm is able to detect most laps correctly. The proposed drone detection algorithm detected all laps correctly using the output of the sample shift estimator. When drone flies through the second gate, the gate indicator has a negative peak before it changes to positive. Using the local maximum and minimum of the gate indicator signal, false-positive detections are eliminated.

5 Conclusion

Timing system based on RSSI signal from the analog video receiver and microphones has been developed. Enhancing RSSI based timing system by microphones overcome the main drawbacks of pure RSSI based timing systems. The enhanced timing system is more precise and robust than standard timing system which relies purely on RSSI signal as described in section 4. Both hardware and software of the timing system have been developed. PCB for the microphone module was designed and manufactured. Housing for the microphone modules was designed and printed on a 3D printer from ABS plastic. The proposed timing system is small and easy to use. It consists of two microphone modules, video receiver, and Nucleo kit. It doesn't require any external additional device to be mounted on the drone.

The sound is measured by two pairs of microphones. The signal from microphones is filtered and the time delay between a pair of microphones on the microphone module is estimated. Proposed least mean squares method for sample shift estimation has been tested and compared to cross-correlation method. Results using the least mean squares method are sufficient and suitable for drone detection as described in section 3.1. Proposed least mean squares method provides better results then cross-correlation method when signals from microphones are noisy and have a lower frequency, therefore, fewer periods are sampled in the buffer.

Video receiver module placed in the middle of the gate and two microphone modules placed in the corners of the gate are used for drone detection. RSSI data, maximal amplitude of the sound signal and output of the sample shift estimator are input measurements for Kalman filter. Estimated state by Kalman filter is smoother with sharper local maximums which help to determine the moment when drone passes through the gate more precisely. Drone detection algorithm was designed, tested in real experiments and compared to other methods. The standard deviation of the measured time errors was lowered from 0.0792s using just the RSSI signal to 0.0295s by using the RSSI signal and the state estimated by Kalman filter. Enhanced drone detection algorithm is more precise then EventTracker, which is the state of the art RSSI based timing system, with its measured standard deviation 0.0675s.

The false-positive drone detections are the biggest drawback of the RSSI based timing systems. When drone flies close to the finish gate, the value of the RSSI signal might be higher than certain detection threshold a cause false drone detection. This drawback is eliminated using the output of the sample shift estimator. Using the estimated sample shift, the drone detection algorithm is able to distinguish whether the drone flies through the gate or next to it and filter out these false detections. The robustness of the proposed timing system has been tested by real-world experiments. Unlike EventTracker and detection algorithm using just RSSI signal, enhanced timing system using sample shift estimator was able to detect all laps correctly.

All goals were fulfilled. The benefits of using microphones along with RSSI signal are confirmed. In order to use proposed timing system in a real race, several things have to be done. The proposed timing system should be extended in order to support more drones. This could be done by connecting more video receivers and running more instances of Kalman filter and drone detection algorithm. The Nucleo-F303RE has limited resources and wouldn't be able to process several instances of Kalman filter and drone detection algorithm with sufficient frequency, therefore, another more powerful micro-controller or computer like Raspberry Pi must be used. Timing system should be able to connect to some race management software. Ethernet interface for connecting the timing system to the PC running race management software would be welcome.

The timing system can be enhanced further on. Third microphone module can be added on the top of the gate in order to distinguish whether the drone flies through the gate or above the gate. Measurements from other sensors might be introduced to Kalman filter and make the estimated lap times even more precise.

References

[1] I. Paterson-Stephens A. Bateman. The dsp handbook: Algorithms, applications and design techniques 1st edition. 2002.

[2] Evan Ackerman. https://spectrum.ieee.org/automaton/robotics/drones/jpl-ai-powered-racing-drone.

[3] 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)*, pages 1–6, Aug 2017.

[4] B. Van Den Broeck, A. Bertrand, P. Karsmakers, B. Vanrumste, H. Van hamme, and M. Moonen. Time-domain generalized cross correlation phase transform sound source localization for small microphone arrays. In *2012 5th European DSP Education and Research Conference (EDERC)*, pages 76–80, Sept 2012.

[5] DCL. Drone champions league. http://dcl.aero.

[6] Dedrone. http://www.dedrone.com.

[7] DRL. Drone racing league. http://thedroneracingleague.com.

[8] Louise Hauzenberger and Emma Holmberg Ohlsson. Drone detection using audio analysis. Master's thesis, Department of Electrical and Information Technology, Faculty of Engineering, LTH, Lund University, Sweden, 2015.

[9] I-LAP. http://www.rclapcounter.com.

[10] ImmersionRC. http://www.immersionrc.com/fpv-products/laprf.

[11] S. Jung, S. Hwang, H. Shin, and D. H. Shim. Perception, guidance, and navigation for indoor autonomous drone racing using deep learning. *IEEE Robotics and Automation Letters*, 3(3):2539–2544, July 2018.

[12] C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(4):320–327, Aug 1976.

[13] Whoop laps. http://whooplaps.com.

[14] Chorus RF Laptimer. http://github.com/voroshkov/Chorus-RF-Laptimer.

[15] J. Lu and B. Smith. Autonomous navigation in drone racecourse. In *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*, pages 1–4, Nov 2017.

[16] J. Mezei, V. Fiaska, and A. Molnár. Drone sound detection. In *2015 16th IEEE International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 333–338, Nov 2015.

[17] MultiGP. http://www.multigp.com.

[18] MyLaps. http://www.mylaps.com.

[19] OpenLap. http://github.com/YannikW/Open-Lap.

[20] Propwashed. http://www.propwashed.com/lap-timing-guide-drone-racing.

[21] Track Mate Racing. http://www.trackmateracing.com.

[22] Rotorama. http://rotorama.cz.

[23] TBS. Team black sheep. http://team-blacksheep.com/products/prod:eventtracker.

[24] Easy Race Lap Timer. http://www.easyracelaptimer.com.

[25] Wikipedia. https://en.wikipedia.org/wiki/Kalman_filter.

[26] University of Cambridge Will Simmons. Churchill compsci talks: Where did that come from? http://talks.cam.ac.uk/talk/index/62200, https://www.youtube.com/watch?v=Z7X7lf6FdYY.

Appendix A Edhanced drone detection schematics

This appendix is described in section 3.

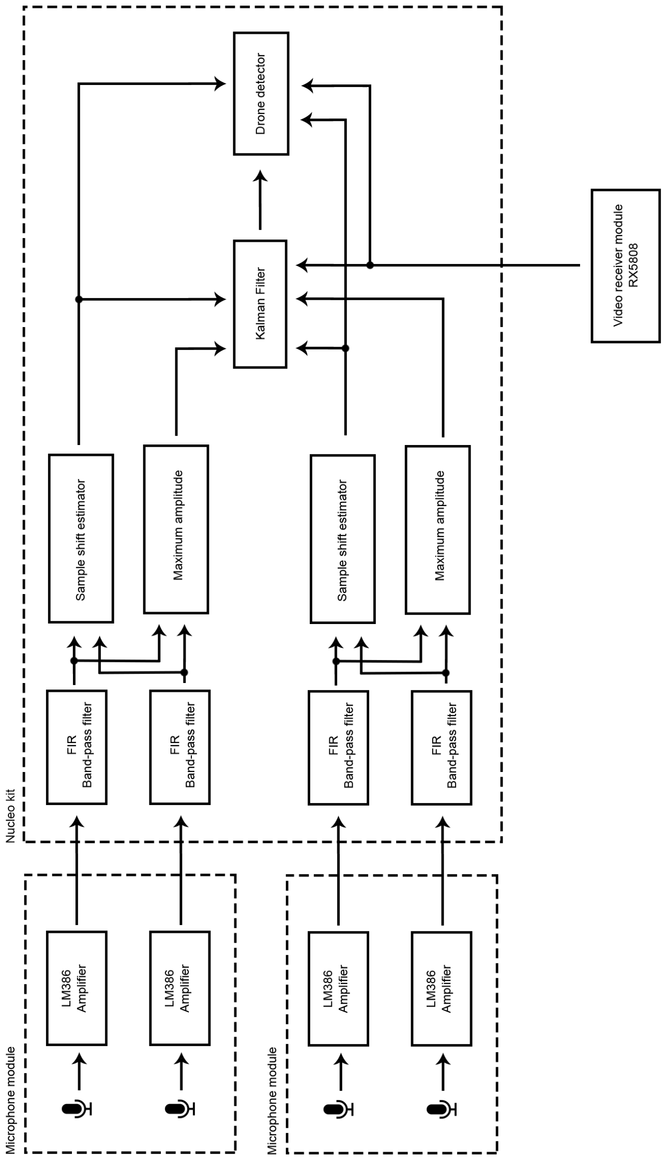


Figure 34: *System schematic*

Appendix B Experimetal data

Laps times in seconds measured in experiment described in secition 4.1 are shown in this appendix in table 2.

Ground Truth	Event Tracker	Stopwatch	RSSI	Enhanced Algorithm
11,681	11,66	11,615	11,724	11,681
12,26	12,37	12,37	12,198	12,219
11,485	11,37	11,355	11,508	11,53
12,012	12,19	12,201	12,005	12,048
10,687	10,56	10,612	10,755	10,669
9,55	9,5	9,591	9,483	9,505
12,059	12,17	12,012	12,069	12,004
12,464	12,36	12,582	12,478	12,5
12,884	12,98	12,881	12,845	12,845
10,812	10,81	10,826	10,819	10,841
9,523	9,54	9,484	9,505	9,44
10,154	9,98	10,228	10,151	10,194
11,692	11,68	11,636	11,767	11,724
14,063	14,9	14,055	13,987	14,008
12,435	12,42	12,453	12,436	12,457
11,246	11,25	11,264	11,25	11,229
12,058	12,1	12,019	12,176	12,047
10,744	10,77	10,828	10,733	10,776
14,584	14,68	14,561	14,547	14,676
11,252	11,14	11,302	11,186	11,229
9,436	9,44	9,478	9,31	9,418
8,774	8,79	8,806	8,901	8,793
8,99	8,98	8,914	8,966	8,988
9,037	9,2	9,05	9,052	8,987
8,785	8,78	8,846	8,772	8,815
8,741	8,68	8,697	8,513	8,751
9,419	9,44	9,568	9,677	9,418
8,546	8,56	8,569	8,535	8,535
7,131	7,11	7,029	7,069	7,134
8,703	8,73	8,747	8,751	8,728
8,844	8,82	8,869	8,685	8,837
7,624	7,67	7,577	7,824	7,629

Ground Truth	Event Tracker	Stopwatch	RSSI	Enhanced Algorithm
7,817	7,81	7,887	7,802	7,846
7,705	7,69	7,775	7,694	7,679
10,439	10,46	10,468	10,409	10,387
9,273	9,17	9,318	9,197	9,283
9,22	9,21	9,251	9,326	9,24
9,257	9,38	9,281	9,284	9,218
8,629	8,54	8,553	8,633	8,634
10,165	10,15	10,359	10,128	10,171
9,17	9,28	9,202	9,175	9,154
8,971	8,93	8,974	8,958	8,98
7,799	7,71	7,864	7,682	7,768
8,176	8,18	8,105	8,179	8,201
8,128	8,11	8,172	8,267	8,137
7,527	7,56	7,531	7,552	7,509
7,959	7,93	7,976	7,941	7,984
6,668	6,7	6,721	6,644	6,665
8,212	8,19	8,077	8,18	8,18
8,95	9,3	8,943	8,937	8,958
9,857	9,79	9,933	9,91	9,824
8,67	8,65	8,672	8,591	8,699
8,977	8,99	8,977	9,024	8,959
10,626	10,64	10,655	10,603	10,646
9,665	9,65	9,724	9,672	9,673
9,978	9,99	9,992	10,019	9,997
9,477	9,55	9,53	9,521	9,456
8,536	8,46	8,48	8,504	8,548
9,339	9,34	9,35	9,262	9,305
8,35	8,35	8,35	8,417	8,374
8,263	8,33	8,25	8,266	8,266
8,862	8,79	8,86	8,894	8,85
9,221	9,24	9,24	9,24	9,197
7,648	7,6	7,62	7,509	7,682
7,564	7,6	7,59	7,682	7,552
9,819	9,77	9,748	9,86	9,802
7,77	7,73	7,823	7,595	7,769
9,02	9,6	9,084	9,197	9,002
12,477	12,45	12,531	12,464	12,486
8,156	8,26	8,155	8,202	8,157
9,898	9,81	9,848	9,759	9,911
9,474	9,46	9,549	9,521	9,456

Ground Truth	**Event Tracker**	**Stopwatch**	**RSSI**	**Enhanced Algorithm**
9,385	9,4	9,275	9,478	9,392
10,245	10,23	10,302	10,214	10,28
8,959	8,91	9,048	8,894	9,002
10,334	10,38	10,243	10,431	10,28
9,939	9,94	10,109	9,912	9,933
9,82	9,83	9,711	9,803	9,824
9,473	9,4	9,635	9,456	9,5
9,926	10,1	9,796	9,932	9,845
8,955	8,89	8,971	8,959	8,981
9,679	9,69	9,751	9,672	9,694
12,276	12,27	12,244	12,313	12,269
11,595	11,67	11,637	11,577	11,556
10,106	10,9	10,127	10,127	10,148
8,409	8,35	8,421	8,374	8,418

Table 2: Measured laps in seconds

Appendix C CD Content

In Table 3 are listed names of all root directories on CD.

Directory name	**Description**
thesis	Diplomas's thesis in pdf format.
thesis_sources	latex source codes
experiments	data measured during experiments
firmware	program for Nucleo F303RE
microphone_module	data for microphone module PCB
sound_analyze	data and matlab codes for sound analysis
sound_correlation	cross-correlation and last mean squares method
drone_detection	Kalman filter and drone detection algorithm

Table 3: CD Content

Appendix D List of abbreviations

In Table 4 are listed abbreviations used in this thesis.

Abbreviation	Meaning
FPV	first person view
RC	radio control
RSSI	reveived signal strength indicator
DRL	drone racing league
DCL	drone champions league
RFID	radio frequency identification
LED	light emitting diode
IR	infra red
SPI	serial peripheral interface
TBS	team black sheep
RPM	revolutions per second
FIR	finite impulse response
IIR	infinite impulse response
PCB	printed circuit board

Table 4: Lists of abbreviations