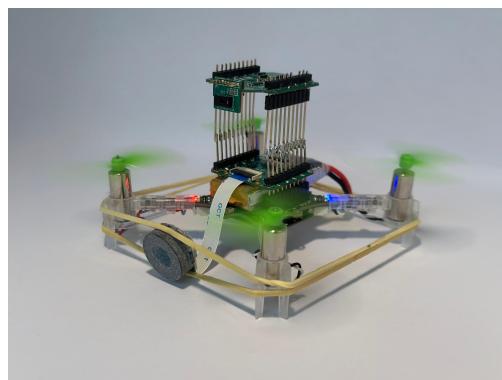


DEPARTMENT OF INFORMATION TECHNOLOGY AND  
ELECTRICAL ENGINEERING

Spring Semester 2024

# Obstacle Avoidance With Ultrasonic Sensors on Nano Drones

Bachelor Thesis

Laurent Schroeder  
[lschroeder@student.ethz.ch](mailto:lschroeder@student.ethz.ch)

May 2024

Supervisors: Hanna Müller, [hanmuell@iis.ee.ethz.ch](mailto:hanmuell@iis.ee.ethz.ch)  
Dr. Tommaso Polonelli, [tommaso.polonelli@pbl.ee.ethz.ch](mailto:tommaso.polonelli@pbl.ee.ethz.ch)  
Professor: Prof. Dr. Luca Benini, [lbenini@ethz.ch](mailto:lbenini@ethz.ch)

# Acknowledgements

First and foremost, I would like to express my sincere gratitude to both Hanna Müller and Dr. Tommaso Polonelli. Their extensive knowledge and innovative approaches to problem-solving were a constant source of inspiration and significantly enhanced my learning experience, which made me a better problem solver. Moreover, throughout the duration of this thesis, they were always available to answer my questions and help me find solutions. I can confidently say that without their continuous support, achieving the same level of quality in my work would have been impossible.

I am also grateful to Prof. Dr. Luca Benini, Dr. Tommaso Polonelli, and Hanna Müller for providing me with the opportunity to work on a thesis that was both enjoyable and intellectually stimulating, offering a visual result of my work.

A huge thanks goes to Dr. Christian Vogt, who patiently explained everything I needed to know to operate the 3D printer independently and always offered his advice and assistance.

Lastly, I would like to thank STMicroelectronics and TDK for providing the materials and support essential to the development of this thesis.

# Abstract

Nanodrones hold significant promise in various applications, such as exploring uncharted environments or aiding in rescue missions by locating injured individuals. Their diminutive size allows them to navigate through tight spaces and fly close to moving objects like humans, making them suitable for nearly any indoor environment. However, their small dimensions pose challenges in terms of payload capacity and onboard computation power, complicating the execution of the sensing and computation necessary for autonomous flight.

Currently, research on nanodrones is still in its early stages, primarily focusing on establishing robust Obstacle Avoidance (OA) algorithms that operate reliably across diverse environments with varying materials. Recent approaches have employed either 1-D sensors like ultrasonic sensors or 2-D sensors such as cameras or Time-of-Flight (ToF) sensors (e.g. *VL53L5CX* sensor). However, these solutions exhibit limitations: ultrasonic sensors have limited reliability, while ToF sensors struggle to detect reflective materials like glass.

In this thesis, I integrate both a *VL53L5CX* and a ultrasonic sensor (ICU-30201) with a gain function to maximize the strengths of both sensors. Moreover, the ICU-30201 sensor gets calibrated to enhance its reliability, particularly for navigating around obstacles like glass. The in-field tests utilized the Crazyflie 2.1 platform (CF), augmented by a Multizone Ranger Deck (for the *VL53L5CX* sensor) and a BatDeck (for the ICU-30201 sensor), resulting in a total flight mass of 38.15 g.

The algorithm, when deployed on the CF, achieves a reliability of 80 % in a novel indoor environment (i.e., meeting room), including glass obstacles, while flying at a maximum speed of 1 m/s. The average distance maintained during flight was 98.39 m, with an average flight duration of 2 minutes and 10 seconds. Additionally, the nanodrone was able to fly through narrow spaces with a width of 75 cm with 100 % reliability. These findings underscore the efficacy of both the ToF sensor and the ultrasonic sensor, even under relatively high-speed flight conditions.

# Declaration of Originality

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor. For a detailed version of the declaration of originality, please refer to Appendix B

Laurent Schroeder,  
Zurich, May 2024

# Contents

<b>List of Acronyms</b>	<b>ix</b>
<b>1. Introduction</b>	<b>1</b>
<b>2. Related Work</b>	<b>4</b>
<b>3. Hardware</b>	<b>7</b>
3.1. <i>Crazyflie 2.1</i> . . . . .	7
3.2. <i>ToF Multizone Sensor</i> . . . . .	8
3.3. <i>Multizone Ranger Deck</i> . . . . .	10
3.4. Ultrasound Horn Decision . . . . .	11
<b>4. Implementation</b>	<b>15</b>
4.1. Hardware . . . . .	15
4.1.1. <i>ICU-30201 sensor</i> . . . . .	15
4.1.2. <i>BatDeck</i> . . . . .	16
4.2. Calibration and Ringdown . . . . .	16
4.3. Sensor Fusion OA algorithm . . . . .	18
<b>5. Results</b>	<b>32</b>
5.1. Power Consumption Due to Payload . . . . .	32
5.2. Narrow Corridor [1] . . . . .	34
5.3. Reliability Test [1] . . . . .	35
5.4. Ultrasound Angle Detection . . . . .	36
<b>6. Conclusion and Future Work</b>	<b>44</b>
<b>A. Task Description</b>	<b>46</b>
<b>B. Declaration of Originality</b>	<b>52</b>

# List of Figures

3.1. Drone facing an obstacle with an angle $\beta$ (with a gap in the middle). $C_x$ is the corresponding column associated with the 8x8 matrix, while $d_x$ is the planar distance to the obstacle. $h_x$ is calculated knowing the corresponding angle $\alpha_x$ and the distance $d_x$ [1]. . . . .	9
3.2. Multizone Ranger Deck compatible with the CF. A forward and a backward facing <i>VL53L5CX</i> sensor can be mounted vertically to the board. The maximum weight is 2.49 g with a size of 9 cm <sup>2</sup> [1]. . . . .	10
3.3. Horizontal Horn on the left and Circular Horn on the right. . . . .	12
3.4. Experiment setup under an angle $\gamma$ , a horizontal distance $d$ and a distance $p$ away from the wall. . . . .	12
3.5. Both measurements were conducted at $\gamma = 45^\circ$ and $d = 1.5$ m, as depicted in Fig. 3.4. In the top row, it is observed that the measurements of the circular Field of View (FoV) fail to detect the wall since there is no amplitude peak at a distance of $p = \frac{1.5\text{ m}}{\cos(45^\circ)} = 2.12$ m. Similarly, the horizontal FoV also fails to detect the wall even at a 45°angle (the first peak does not correspond to the wall). Additionally, the noise level of the horizontal FoV is higher compared to that of the circular FoV. . . . .	13
3.6. Both measurements were taken while flying at a distance of 2.5 m in front of a wall. The circular FoV shows a higher amplitude peak (over 10000). In contrast, the horizontal FoV shows three peaks, with the highest peak corresponding to the wall, but reaching only an amplitude of slightly over 8000. . . . .	14
4.9. Steering velocity computation based on the forward (x) velocity. . . . .	23
4.1. ICU-30201 distance measurement: The sensor records the time $t$ taken for the sound wave to return, while the known speed of sound $v$ is 343 m/s at room temperature. Utilizing the formula $d = v \times t/2$ , the distance to the object can be accurately calculated. . . . .	24

## List of Figures

4.2.	The final setup of the CF with the expansion boards: On top you can see the Multizone Ranger Deck with the <i>VL53L5CX</i> sensor, as the battery placeholder you can see the BatDeck and underneath the drone you can see the <i>Flow-deck v2</i> . . . . .	25
4.3.	Behaviour of motor noise standard deviation with different filter lengths in slow and fast time [2]. . . . .	25
4.4.	Approach on how to set the threshold for the ICU-30201 sensor . . . . .	26
4.5.	Approach on how to calculate the ringdown filter array . . . . .	26
4.6.	Integration of the OA algorithm into the CF firmware. My additions are shown in red, while the blocks from the original firmware are shown in green [1] . . . . .	26
4.7.	Decision on which group gets forwarded to the Decision tree. In this case, the encircled group gets forwarded, since it has a minimal distance of 51 cm which is closer than the minimal distance to any of the other groups. . . . .	27
4.8.	Ultrasound feature extraction: How the distance to the closest object gets extracted out of the ultrasound measurement. . . . .	27
4.10.	Obstacle Avoidance flow diagram. Illustrates how the flight command gets updated every cycle. The x velocity gets computed as explained in 2 and the steering rate as explained in 3. . . . .	28
4.11.	8x8 ToF pixel matrix divided into 4 zones, Ceiling, Ground zone, Caution and Danger zone [1]. . . . .	29
4.12.	The gain_dist function takes values in the range [0,1], with higher values indicating that the sensor detects an obstacle at a greater distance. . . . .	30
4.13.	The gain_dist_tof variable is adjusted based on the zone where the <i>VL53L5CX</i> sensor detected the object (refer to Fig. 4.11). Lower values indicate higher safety due to reduced x velocity. . . . .	30
4.14.	Calculation of the forward velocity based on the introduced gain_dist function 4.12. . . . .	31
5.1.	In the top row, the first setup is shown on the left and the second on the right. In the bottom row, the third setup is on the left and the fourth on the right. . . . .	33
5.2.	Power consumption for the different setups based on their weights. The number on the x axis represents the number of the setup and the y axis represents the power consumption in Watts. The most used setup is the fourth one with a power consumption of 12.53 W 5.1. . . . .	37
5.3.	Flying through a narrow corridor, then turning and flying out again with different widths [1]. . . . .	38
5.4.	Floorplan of the meeting room in which I ran the reliability test. The red crosses indicate the locations of recorded crashes [1]. . . . .	39

*List of Figures*

5.5. In the top row, I present the average time and distance until a crash occurs, excluding battery changes except at the test's conclusion. The middle row illustrates the average flight time and distance. The bottom row depicts the average flight time and distance, exclusively for successful flights. The blue bars indicate the average, while the black lines represent the variance [1]. . . . .	40
5.6. Setup of the Ultrasound Angle Detection experiment with varying approach angle $\alpha$ and constant distance $d = 1\text{ m}$ in both schematics (on the left) and real-world (on the right). . . . .	41
5.7. Plot of the measurement amplitude of the ICU-30201 sensor at a $40^\circ$ angle in front of a plexiglass. . . . .	41
5.8. Two measurements of the ICU-30201 sensor at a $50^\circ$ angle in front of a plexiglass. The first measurement, with a magnitude of around 5500, would likely not trigger detection in my implementation, while the second measurement, with a magnitude above 17500, would clearly recognize the plexiglass at 1 m. . . . .	42
5.9. Plot of the measurement amplitude of the ICU-30201 sensor at a $60^\circ$ angle in front of a plexiglass. The measured amplitude is insufficient to exceed the threshold (see Fig. 4.2), making it impossible to recognize the plexiglass as an object. . . . .	43

# List of Acronyms

CF	.....	Crazyflie 2.1 platform
CNN	.....	Convolutional Neural Network
DFS	.....	Depth-First Search
DNN	.....	Deep Neural Network
DOE	.....	Diffractive Optical Elements
eKF	.....	extended Kalman Filter
FoV	.....	Field of View
GPIO	.....	General Purpose Input/Output
I2C	.....	Inter-Integrated Circuit
IIS	.....	Integrated Systems Laboratory
IMU	.....	Inertial Measurement Unit
IQ	.....	In-phase and Quadrature
LiDAR	.....	Light Detection and Ranging
MCU	.....	Microcontroller
MUTCLK	.....	Multipurpose Timer Clock
OA	.....	Obstacle Avoidance

*List of Acronyms*

ODR	Output Data Rate
PBL	Center for Project-Based Learning
PMUT	Piezoelectric Micromachined Ultrasonic Transducer
PWM	Pulse Width Modulation
RADAR	Radio Detection and Ranging
SLAM	Simultaneous Localization and Mapping
SoA	State of the Art
SPAD	Single-Photon Avalanche Detector
SPI	Serial Peripheral Interface
tinyML	tiny Maschine Learning
ToF	Time-of-Flight
UAV	Unmanned Aerial Vehicles
VCSEL	Vertical-Cavity Surface-Emitting Laser

## Introduction

Drone technology has seen rapid advancements in recent years, with the global market value projected to reach \$39.3 billion next year, marking a \$13.6 billion increase in just one year [3]. This growth is driven by the technology's versatility across various applications, including monitoring, transportation, agriculture, photography, and urban search and rescue [4]. While large drones are capable of performing complex tasks such as 3D modeling, smaller drones offer many advantages. Their compact size enhances agility, enabling navigation through complex environments like narrow passages, and allows for safer operation around humans [5], [1]. Larger drones, especially when used indoors, can pose significant safety risks due to their powerful motors and larger propellers. This makes nanodrones a valuable alternative for such environments.

Ensuring a safe and efficient autonomous Unmanned Aerial Vehicles (UAV) navigation strategy and implementation that is globally supported is essential for the reliable use of these smaller drones [6]. Moreover, implementing navigation strategies on platforms constrained by power, weight, and sensor limitations challenges researchers to optimize every step of the process, from sensor data acquisition to high-level algorithm execution [7]. These advancements could eventually enhance the efficiency of larger UAVs as well.

To implement a reliable navigation system for nano-UAVs, various sensor technologies are necessary, including Light Detection and Ranging (LiDAR), cameras, and laser-based ToF sensors [8]. Ultrasound sensors are also crucial when operating around reflective surfaces [2]. However, developing a dependable obstacle avoidance algorithm with these sensors presents numerous challenges.

Firstly, nano-UAVs have extremely limited power resources, leaving two main options for sensor preprocessing and navigation computation: off-board computation, which allows for running complex algorithms such as Simultaneous Localization and Mapping (SLAM) [9] and advanced navigation models that are computationally demanding; or onboard computation, which relies on restricted navigation strategies used in 84% of cases [10]. Moreover, they mostly rely on a known map of the environment with the

## 1. Introduction

obstacles [11]. Onboard computation allows for independence from wireless connections, enabling operation in various environments and reducing latency since everything is computed in a very compressed area. Authors in [12] demonstrated that even with onboard computation, it is possible to run complex computer vision algorithms like localization, OA, and pattern recognition.

However, reliable autonomous flight at high speeds is still a challenge due to power consumption constraints and limited sensors [1], [2]. Ideally, navigation tasks on nano-UAVs should consume only about 10 % of the available power to reserve resources for motors and other tasks [1]. For instance, the CF has a total power limit of 10 W, with most of that power allocated to the motors, leaving only a few hundred milliwatts for processing to avoid significantly reducing flight time [1]. The ARM Cortex-M4 core Microcontroller (MCU), running at a few hundred megahertz with around 200 kB of RAM, meets these power requirements [1] and is commonly used in nano-UAV applications [1]. However, there are also possibilities to recharge nanodrones autonomously, using renewable energy sources, such as solar energy, to increase flight time and distance [13].

Moreover, the navigation pipeline must be highly robust against disturbances such as sensor noise and varying light conditions because the limited number of sensors on nano-UAVs requires maximizing the efficiency of each sensor.

Currently, OA is mainly performed using vision-based sensors like cameras. However, their high pixel count makes them computationally expensive, therefore only feasible on nanodrones with reduced algorithm precision [14]. Additionally, Radio Detection and Ranging (RADAR) [15], laser range finders [16], and Deep Neural Network (DNN)s with an average detection accuracy of 75.4 % [17] have been used for navigation. However, these models require large datasets to reliably navigate various environments, posing a memory challenge on nano-UAVs, which often necessitates optimization for specific environments.

In this thesis, I use a single *VL53L5CX* sensor and a *ICU-30201* sensor as mentioned as future work in [1] to implement a reliable OA algorithm for safe navigation through various indoor environments, particularly office settings. By utilizing only two sensors and a simple algorithm for navigation, computational resources are conserved for additional tasks.

To achieve an efficient system, I designed the horn for the ICU-30201 sensor to be as light as possible by hollowing it out and using the lightweight material polyamide. Consequently, with both sensors connected, the power required to hover the drone only increased by 21.89 % compared to the setup with only the VL53L5CX sensor connected [1] (refer to 5.1). Moreover, the implemented OA algorithm runs onboard, allowing the drone to navigate in an unknown environment without external positioning or velocity assistance. The algorithm utilizes a scalable gain function that adjusts based on the current ToF sensor measurement and the last five measurements of the ICU-30201 sensor, balancing low memory requirements and decreased latency against accuracy and reliability. The system achieves an 80 % reliability rate in indoor environments (e.g., meeting rooms) with various materials (e.g., tables, chairs, plexiglass) 5.3. Additionally, it can successfully navigate through narrow passages of 75 cm width 100 % of the time 5.2.

### *1. Introduction*

Future researchers could build upon this system to develop more complex OA algorithms by integrating additional sensors, such as a backward-pointing ultrasound sensor, to further enhance reliability and expand the range of environments in which nano-UAVs can operate. At the same time, they should aim to keep the system as efficient as possible in terms of power consumption and memory usage.

# Chapter 2

## Related Work

In general, UAVs are equipped with various sensors, enabling them to perform a wide range of tasks such as OA [18], mapping [19] or object tracking [20]. In particular, indoor environments enable various research areas, because the drones can interact with static and dynamic objects like humans. Nanodrones, which are known for their compact size and lightweight materials, have great potential in these environments [6], because agility, maneuverability and access to confined spaces is key to safely navigate through them. This is why many researchers currently focus on enabling autonomous nano-UAV mapping [19], navigation [1], [5], [1] and swarm formations [21], [22].

However, the first step towards those goals always includes a reliable OA algorithm in order to achieve a robust autonomous flight. OA involves the detection and navigation around obstacles in the UAV's path, using available sensors onboard to ensure a safe operation through complex environments. Currently, there exist 3 main approaches to achieve OA:

First, vision-based methods often using Convolutional Neural Network (CNN)s [23], [8], [24], which analyze visual data to identify and avoid obstacles.

Loquercio et al. [25], [26] conducted a study to demonstrate the feasibility of high-speed flight in complex environments using a stereo camera and a compact CNN trained exclusively in simulation. The study involved conducting controlled experiments in both simulation and real-world environments to evaluate the performance of an autonomous drone navigation system. The experiments included testing the system's ability to follow a reference trajectory, avoid obstacles, and adapt to dynamically changing environments. The results showed that the proposed approach reduced the failure rate compared to State of the Art (SoA) methods in simulation and achieved a higher success rate in both static and dynamic real-world environments than a professional pilot. Additionally, the system demonstrated robustness against external factors such as illumination changes and visual distractors. The authors train the algorithms on randomly generated obstacle

## 2. Related Work

courses composed of simple off-the-shelf objects, such as schematic trees and a small set of convex shapes such as cylinders to work in generalized scenarios. This approach enabled the drones to achieve a maximum flight speed of 10 m/s in real scenarios, with a success rate of 60 % in various and unknown environments. However, it is important to note that their methodology cannot be applied to nano-UAV platforms due to its high memory and computational requirements, as well as reliance on high-resolution sensing, which is not yet possible for nano-UAVs [5], leaving autonomous exploration with nano-UAVs as an open research project. However, authors in [23] showed that it is possible to employ a lightweight depth estimation framework DDND for OA on the nanodrone CF. But considering the limited storage and computing capacity of such a small drone platform, it is only possible to deploy a tiny network on it for monocular depth estimation. In order to improve the learning ability of this tiny network, they integrate knowledge distribution and proposes the CADiT module for better knowledge transfer from a teacher model. Additionally, they present an application pipeline for the reconstruction of the environment using offline metric depth estimation and keyframe pose estimation. However, the limitations of such vision-based methods is that it is unable to avoid transparent objects such as glass [1].

Second, ToF sensors, which measure the time taken for emitted light [27] or sound waves [2] to reflect off objects, enable precise distance measurements and thus obstacle detection [1], [28].

Authors in [27] address the challenge of real-time localization for resource-constrained nano-UAVs such as the CF by proposing a novel approach using a miniature 8x8 pixels ToF sensor from STMicroelectronics (*VL53L5CX*) and a RISC-V-based processor. This allows them to achieve localization on a 31.2 m<sup>2</sup> map with 0.15 m accuracy and a success rate above 95 % in common indoor environments. The algorithm runs at a frequency of 15 Hz and the drone's power consumption only increased by 7 % due to the additional sensing and processing. However their solution does not work with materials that do not reflect light such as glass.

In [2] they overcome this issue by utilizing a forward-facing ICU-30201 sensor operating at approximately 50 kHz, which also has the capability to detect glass. To accommodate this sensor, they developed BatDeck, a platform enabling the integration of two ICU-30201 sensors—one facing forward and one backward. With this configuration, they implemented an OA algorithm operating at 33 Hz and evaluated its performance on the CF. Achieving an average flight distance of 136 m and flight duration of 8'43" until crash in a complex indoor environment, they proved the effectiveness of ultrasound sensors for OA on nanodrones. However, it's worth noting that relying solely on an ultrasonic sensor may still pose challenges in terms of robustness, particularly against disturbances such as motor noise.

Lastly, sensor fusion techniques, which integrate data from multiple sensors such as cameras, LiDAR, and ultrasound [2], to enhance obstacle detection accuracy and reliability

## 2. Related Work

[18], [8].

In the work of [8], the authors present Stargate, an OA algorithm designed for the CF, utilizing only 11 % of its computational capabilities. Stargate is capable of detecting gates and navigating through them. This task is accomplished by integrating a *VL53L5CX* sensor and a low-resolution grayscale camera in order to increase robustness. Additionally, the nano-UAV learns to recognize gates using two tiny Maschine Learning (tinyML) CNN, trained exclusively on simulation-generated data and executed entirely onboard in just 7.6 ms, achieving an accuracy above 91 %. During field tests, the solution only crashed with a probability of  $1.2 \times 10^{-3}$  crash/meter, resulting in only two crashes in a cumulative flight distance of 1.7 km. However, limitations of the sensor combination include the grayscale camera's dependency on lighting conditions and the ToF sensor's restricted range [1].

Authors in [29] propose an approach for reliable obstacle avoidance for a wheeled robot using data fused from multiple sensors including depth cameras and LiDAR [30]. Additionally, they introduced a calibration method based on an external camera and a projection technique to convert the 3D data to its 2D correspondence. Field tests demonstrate that the resulting solution effectively addresses static and dynamic obstacles. Yet, this solution remains unfeasible for nano-UAVs due to the limited power and computational capabilities.

In conclusion, we note that every sensor comes with specific limitations in certain scenarios when attempting to implement a reliable and robust OA algorithm. To optimize the conditions under which the OA algorithm operates effectively, I decided to choose the sensor fusion approach. Additionally, as demonstrated in [31], depth cameras and ultrasound sensors complement each other very well. Hence, I opted to integrate a *VL53L5CX* sensor with an *ICU-30201* sensor.

# Chapter 3

## Hardware

In this study, I implement an OA algorithm using sensor fusion on nano-UAVs. Design optimizations such as weight and area minimization are crucial for this application to reduce power consumption and thus increase the flight time. I utilized the open-source CF from Bitcraze and expanded its functionality by integrating both the ICU-30201 sensor from TDK and the VL53L5CX sensor from STMicroelectronics onto the platform. Hereafter, I provide details regarding the CF platform, the VL53L5CX sensor with its expansion board and the decision on the ultrasound horn.

### 3.1. *Crazyflie 2.1*

The CF is an open-source software/hardware nano-UAV platform from Bitcraze that is widely used in research. Its base board comes equipped with various sensors including an Inertial Measurement Unit (IMU), a barometer and radio communication capabilities. Additionally, it features two MCUs, the nRF51822 from Nordic Semiconductors (16 MHz, 16 kB RAM), responsible for the radio communication and power management, communicating via UART with the *STM32F405* processor (168 MHz, 196 kB), responsible for the sensor readout and processing, motor and flight control, and state estimation. Notably, the CF supports expansion through a variety of extension headers, facilitating the enhancement of state estimation and navigation based on sensor data. In this work, I have incorporated a downward-facing *Flow-deck v2*, integrating an optical flow sensor and a 1-D ToF sensor, which measures the height to improve the position estimation computed by the extended Kalman Filter (eKF). Additionally, I utilize 7×16 mm 19 000 KV motors from betafpv and 47-17 propeller from Bitcraze. The original CF weighs 27.2 g and can fly for up to 7 min with its 250 mAh battery. The maximum payload that still enables takeoff is 14.8 g. However, it is advisable to avoid flying at maximum payload

### 3. Hardware

capacity as it significantly impacts agility, maneuverability, and flight time. The *Flow-deck v2* weighs 1.7 g and to accommodate two additional sensors with their respective expansion decks, I decided to upgrade to a 350 mAh LiPo battery from Tattu to support increased motor power, adding an additional payload of 1.1 g. Consequently, a maximum payload of 12 g remains available for further decks/sensors.

#### 3.2. ToF Multizone Sensor

The *VL53L5CX*<sup>1</sup> sensor is a SoA, ToF sensor that has been designed to achieve the best ranging performance in various ambient lightning and temperature conditions, based on a Vertical-Cavity Surface-Emitting Laser (VCSEL), a Single-Photon Avalanche Detector (SPAD) array, physical infrared filters, and Diffractive Optical Elements (DOE). The *VL53L5CX* sensor can provide either a 4x4 or a 8x8 pixels matrix, which is configurable by software. Each of the 64 zones provides a distance measurement to the closest object in that particular zone and in case the measurement was unsuccessful due to misscalculation or interference, an error flag gets set for that pixel [1]. In that way, by overlapping the validity matrix over the distance matrix, I can filter out wrong or noisy measurements. The sensor itself has dimensions of 6.4 x 3.0 x 1.5 mm and is characterized by STMicroelectronics on having an accuracy of 15 mm in absolute value from 2 cm to 2 m and above the accuracy decreases up to 11 % of the absolute distance, working until 4 m [1].

The ToF sensor offers two configurable ranging modes with varying resolutions, ranging frequencies, and integration times: continuous or autonomous ranging. In autonomous ranging, the VCSEL is deactivated when not in use by adjusting the integration time to conserve energy. For my purposes, I utilize the continuous mode, where the VCSEL remains active continuously, allowing for maximum integration time. Both modes support resolutions of either 4x4 pixels, operating at a maximum frequency of 60 Hz, or 8x8 pixels, as employed in this study, with a maximum frequency of 15 Hz and a wide 65° diagonal FoV, which can be reduced by software.

Moreover, the returned signal can be processed for example by the sharpener value to remove some of the signal caused by veiling glare<sup>2</sup>. Additionally, the FoV of the sensor is influenced by various parameters such as the target distance, reflectance, ambient light level, sensor resolution, sharpener, ranging mode and integration time. Furthermore, the FoV of a pixel is divided into two zones: the Collector Exclusion Zone and the System FoV. To ensure proper functionality of the sensor, the cover glass opening has to be equal or wider than the exclusion zone (55.5° horizontally and 61° vertically). But the

---

<sup>1</sup>Online. Available: <https://www.st.com/en/imaging-and-photonics-solutions/vl53l5cx.html>

<sup>2</sup>Veiling glare is an imperfection of performance in optical instruments arising from incoming light that strays from the normal image-forming paths, and reaches the focal plane

### 3. Hardware

System FoV or detection volume<sup>3</sup> is narrower than the extrusion zone ( $45^\circ$  vertically and horizontally). Fig. 3.1 illustrates the operation of the ToF sensor, showing the drone facing an obstacle at an angle of  $\beta$  with a gap in the middle. Given the ToF sensor's FoV ( $45^\circ$ ),  $\alpha_x$  is known, and  $d_x$  is measured; thus,  $h_x$  can be calculated. In order for the STM32F405 MCU to communicate with the VL53L5CX sensor, it uses its 400 kHz Inter-Integrated Circuit (I2C) digital bus line and two General Purpose Input/Output (GPIO) pins. Additionally, the sensor requires a power supply ranging between 2.8 V and 3.3 V, also compatible with the STM32F405 MCU.

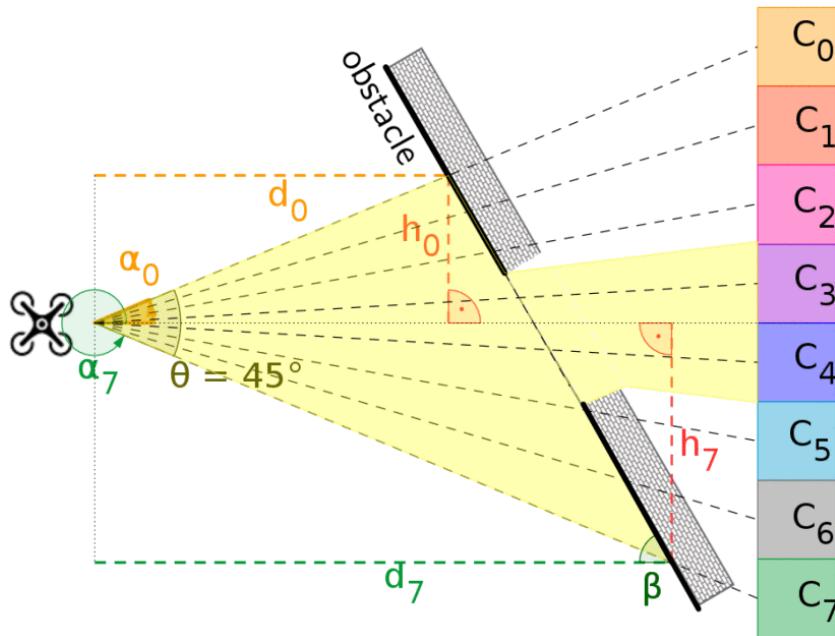


Figure 3.1.: Drone facing an obstacle with an angle  $\beta$  (with a gap in the middle).  $C_x$  is the corresponding column associated with the  $8 \times 8$  matrix, while  $d_x$  is the planar distance to the obstacle.  $h_x$  is calculated knowing the corresponding angle  $\alpha_x$  and the distance  $d_x$  [1].

---

<sup>3</sup>The detection volume represents the applicative or system FoV in which a target is detected, and a distance measured

### 3. Hardware

#### 3.3. Multizone Ranger Deck

To efficiently connect the *VL53L5CX* sensor, I utilize the Multizone Ranger Deck (29.4 mm x 30 mm x 9.5 mm) developed by [1], specifically optimized for this sensor. This deck accommodates two sensors, one facing forward and one backward. Each sensor consumes 286 mW of power when operated in continuous ranging mode. The configuration of the Multizone Ranger Deck is depicted in Fig. 3.2 as presented by [1], with a maximum weight of 2.49 g when both sensors are connected. In my setup, only the forward-facing *VL53L5CX* sensor is connected, resulting in an additional weight of 2.28 g, leaving us with a maximum payload capacity of 9.72 g.

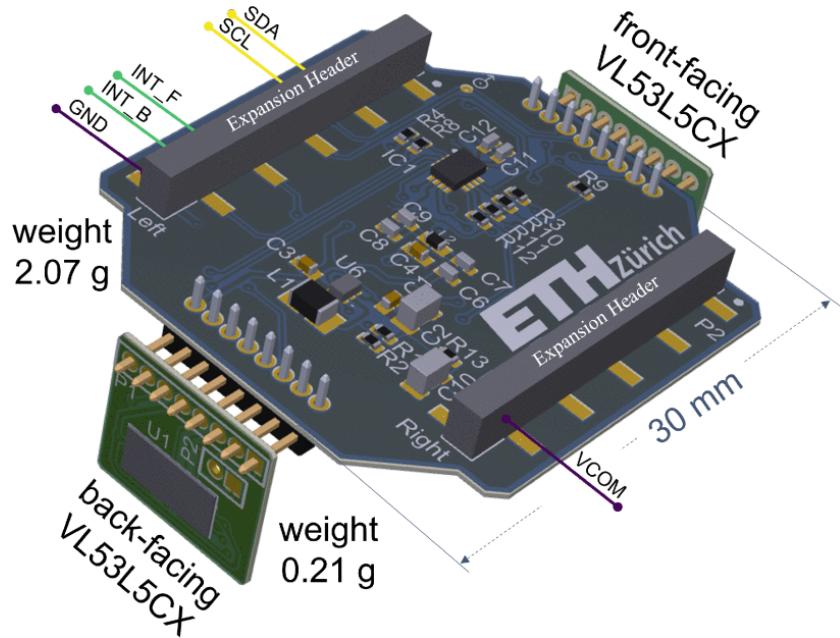


Figure 3.2.: Multizone Ranger Deck compatible with the CF. A forward and a backward facing *VL53L5CX* sensor can be mounted vertically to the board. The maximum weight is 2.49 g with a size of 9 cm<sup>2</sup> [1].

### 3. Hardware

#### 3.4. Ultrasound Horn Decision

I tested different horns with various FoVs and materials to find the best fit for my application. Since we work with nano-UAVs, the main goal was to design a very light horn to minimize power consumption. To achieve this, I decided to hollow out the horn, leaving a 1 mm thickness for the surrounding structure, as recommended by the 3D printing manufacturer i.materialise<sup>4</sup>. I chose polyamide (MJF<sup>5</sup>) as the material due to its printing accuracy of 0.3 mm and low density of  $\rho = 1.01 \text{ g/cm}^3$ , resulting in a final horn weight of 0.69 g compared to the 1.22 g of the original horn used in [2]. So this modification led to a weight reduction of 43.4 %.

Additionally, I tested two different horns with the described configuration: one with a horizontal FoV and one with a circular FoV, as shown in Fig.3.3. I then took measurements to determine which horn is best suited for my application. A good measurement is indicated by a high amplitude peak at the distance of the object compared to the noise floor. Ideally, the noise floor should be around 4000 or lower. Specifically, the difference should exceed  $1.5 \times \text{noise\_floor} + 1500$ , as explained in 4.2. For example, the measurement with the circular FoV shown in Fig. 3.5 is considered good because it has a high amplitude peak at the distance of the object, and the noise floor is low, around 4000. Conversely, the measurement with the Horizontal FoV in Fig. 3.5 is less favorable because the noise floor is slightly higher, around 5000, and the amplitude detected at the distance of the wall is lower, around 8000, not surpassing the  $1.5 \times \text{noise\_floor} + 1500$  threshold as described in 4.2.

Although the horizontal horn was expected to recognize obstacles at a higher angle  $\gamma$  due to its wider FoV, my measurements (refer to Fig.3.5) in the experiment setup 3.4 did not support this expectation. This is due to two main reasons: first, the amplitude of the reflected signal is lower because the signal energy is spread more widely. Second, in my specific application, more noise from the motors was detected, as shown in Fig.3.5, making it harder to differentiate between obstacles and noise.

In this work, I opted for the circular FoV horn because, in this application, the drone mostly moves straight forward or makes slight turns, and this configuration proved to be more reliable for detecting obstacles directly in front of the drone (refer to 3.6).

---

<sup>4</sup>Online. Available: <https://i.materialise.com/en>

<sup>5</sup>Multi Jet Fusion is a powder bed-based additive manufacturing process

### 3. Hardware

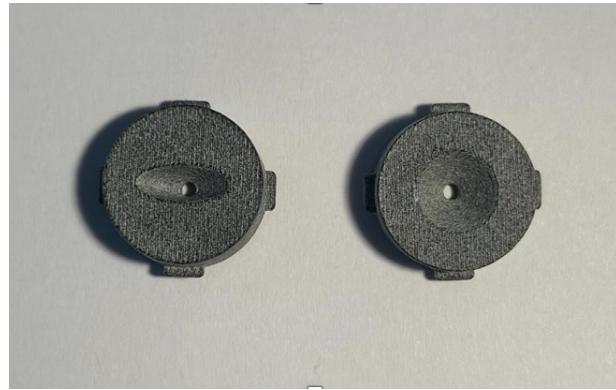


Figure 3.3.: Horizontal Horn on the left and Circular Horn on the right.

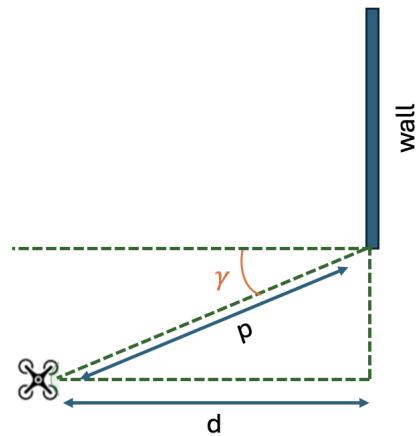


Figure 3.4.: Experiment setup under an angle  $\gamma$ , a horizontal distance  $d$  and a distance  $p$  away from the wall.

### 3. Hardware

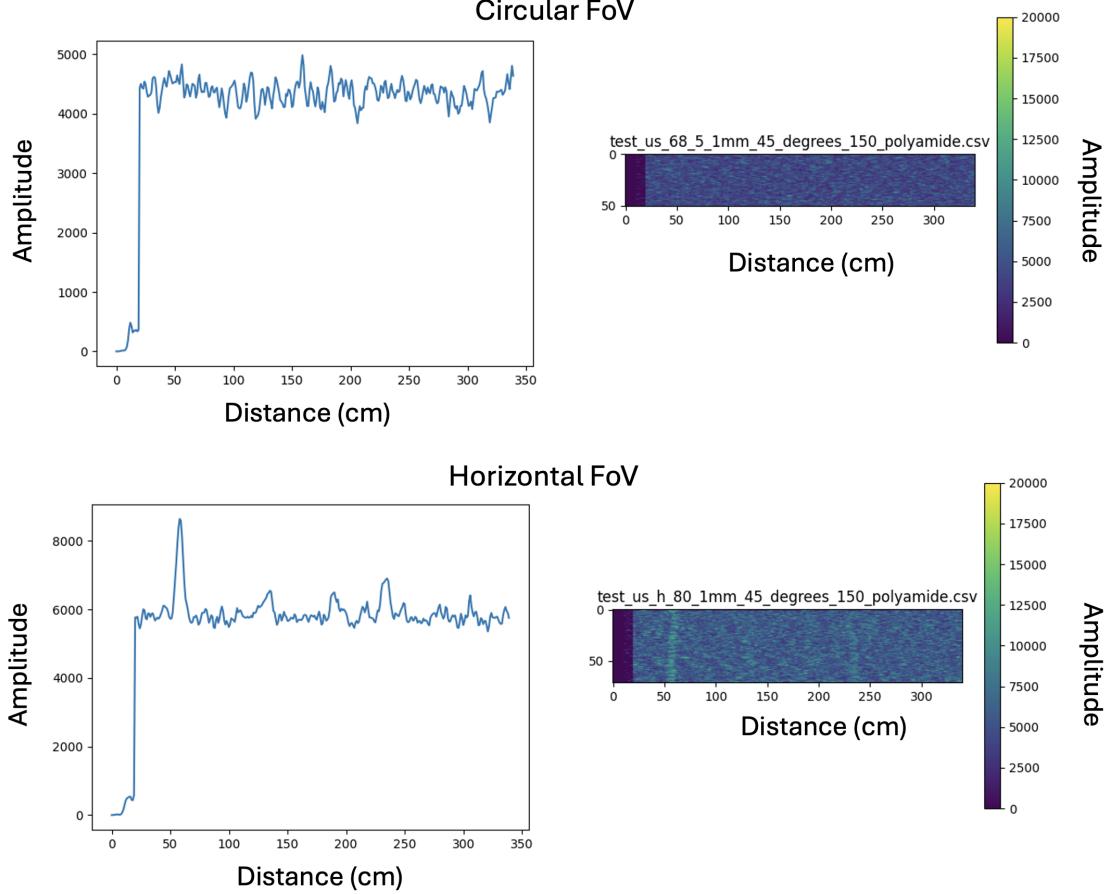


Figure 3.5.: Both measurements were conducted at  $\gamma = 45^\circ$  and  $d = 1.5$  m, as depicted in Fig. 3.4. In the top row, it is observed that the measurements of the circular FoV fail to detect the wall since there is no amplitude peak at a distance of  $p = \frac{1.5 \text{ m}}{\cos(45^\circ)} = 2.12$  m. Similarly, the horizontal FoV also fails to detect the wall even at a  $45^\circ$  angle (the first peak does not correspond to the wall). Additionally, the noise level of the horizontal FoV is higher compared to that of the circular FoV.

### 3. Hardware

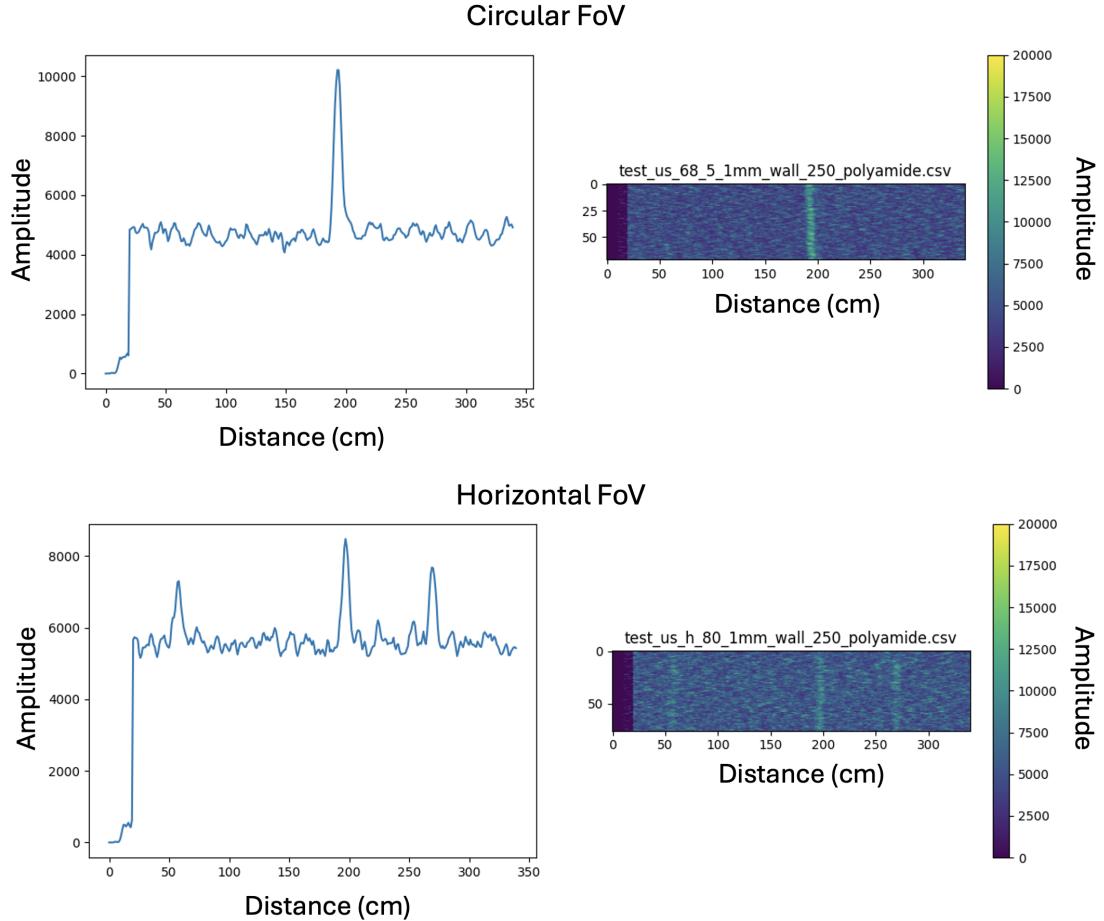


Figure 3.6.: Both measurements were taken while flying at a distance of 2.5 m in front of a wall. The circular FoV shows a higher amplitude peak (over 10000). In contrast, the horizontal FoV shows three peaks, with the highest peak corresponding to the wall, but reaching only an amplitude of slightly over 8000.

# Chapter 4

## Implementation

In the following chapter, I present how I integrated the *ICU-30201* sensor with its corresponding expansion deck in section 4.1. Moreover, I explain how I increased the reliability of the ultrasound sensor by calibrating it and computing the ringdown filter (see section 4.2). Finally, in section 4.3, I explain in detail how the OA algorithm, utilizing sensor fusion of the *VL53L5CX* and the *ICU-30201* sensor, is implemented to maximize the benefits of each sensor.

### 4.1. Hardware

This section introduces the *ICU-30201* sensor, its expansion deck (BatDeck), and how they are connected to the CF.

#### 4.1.1. *ICU-30201* sensor

The TDK ICU-30201 sensor is a ultra-low power ultrasonic transceiver rangefinder utilizing a Piezoelectric Micromachined Ultrasonic Transducer (PMUT), operating at a frequency of  $f_{op} = 50\text{ kHz}$ . It emits short pulses of sound waves into the air, which subsequently hit an object and are reflected. By measuring the ToF of these sound waves and utilizing the known sound velocity in air (343 m/s at room temperature), the system can accurately determine the distance to the object, as depicted in Fig. 4.1. Furthermore, the sensor is equipped with a 40 MHz CPU for sampling and preprocessing, capable of handling a maximum of 340 samples per measurement. These data samples can be stored either as In-phase and Quadrature (IQ) components or as phase and magnitude. Additionally, the Output Data Rate (ODR) is configurable and can be set to the following values:  $\text{ODR} = f_{op}/N$  with  $N = 2, 4, 8$ .

## 4. Implementation

### 4.1.2. *BatDeck*

BatDeck is an extension deck developed by [2] specifically for the CF, enabling the attachment of up to four TDK ICU-x0201 sensors via Serial Peripheral Interface (SPI) lines. In my setup, I opted to utilize a single forward-facing ICU-30201 sensor equipped with a 55° FoV Horn crafted from polyamide with a thickness of only 1 mm. This sensor is attached to the drone using rubber bands for vibration dampening, see Fig. 4.2. The chosen FoV was selected as a balanced compromise between accuracy, maximum range, and FoV coverage (refer to section 3.4), while the thickness and polyamide material for the horn was chosen to make it light as possible, minimizing power consumption of the nano-UAV while ensuring robustness. The total weight of the BatDeck with one sensor, the rubber band and the pin headers is 5.87 g, with the BatDeck itself contributing 1.37 g and the sensor horn 0.69 g, thus still allowing for a maximum payload capacity of 3.85 g (refer to chapter 3).

It's worth noting that both the BatDeck and the Multizone Ranger Deck can be simultaneously mounted on the CF. However, it's crucial to disconnect the R1 pin (Pin7 on the left side) exclusively from the Multizone Ranger Deck (refer to Fig. 4.2). Failure to do so results in the interrupt signal from the ICU-30201 sensor consistently being set to 0. Typically, I position the BatDeck as the battery holder and mount the Multizone Ranger Deck slightly higher to prevent the ICU-30201 sensor of being in the *VL53L5CX* sensor's FoV. The final setup can be seen in Fig. 4.2.

## 4.2. Calibration and Ringdown

This section presents my calibration and implementation process for a ringdown filter utilized during flight with the ICU-30201 sensor, aimed at increasing its object detection reliability. Commands for Transmission/Reception can be configured in cycles of the Multipurpose Timer Clock (MUTCLK), which operates at  $16 \times f_{op}$  (resulting in 800kHz in my case, where  $f_{op} = 50\text{ kHz}$ ). The standard configuration includes transmitting and receiving signals for 512 cycles, followed by the damping of ringdown artifacts for an additional 45 cycles. Additionally, I selected  $N = 4$  (such that  $\text{ODR} = f_{op}/N = 12.5\text{ kHz}$ ), providing a range of 4.6 m and a resolution of 1.35 cm (given  $f_{op} = 50\text{ kHz}$  and 340 samples per measurement). However, it's important to note that the operational frequency ( $f_{op}$ ) may deviate from 50 kHz due to variations in sensors and environmental conditions, thereby affecting both resolution and maximum range.

The measurements from the sensor arrive at a frequency of 33 Hz. In this study, decisions are based solely on the magnitude of the measurement, leading us to exclusively stream out the magnitude data rather than the IQ parameters. Additionally, for highly reliable measurements, I implement filtering techniques both in slow-time (i.e. over consecutive ultrasound measurements) and in fast-time (i.e. over the consecutive 340 ultrasound

#### 4. Implementation

samples from the same measurement). Following the approach proposed in [2], I use an exponentially moving average filter in slow-time:

$$y_{in} = \frac{K_s - 1}{K_s} y_{in-1} + \frac{1}{K_s} x_{in}$$

and an average filter in fast-time:

$$y_{in} = \sum_{j=i-K_f+1}^i \frac{x_{jn}}{K_f}$$

Here  $n$  denotes the measurement number,  $i$  the sample number,  $y$  the filtered magnitude and  $x$  the raw measurement. To best filter the measurements, I chose  $K_s = 5$  and  $K_f = 1$  as depicted in 4.3. To further increase the robustness of the ICU-30201 measurements, I introduce a calibration routine in paragraph 4.2 and a ringdown filter in paragraph 4.2.

##### A. ICU-30201 Calibration

To enable the OA algorithm, I must set a threshold on the signal magnitude returning from the sensor. This threshold distinguishes signals originating from objects (magnitude  $>$  threshold) from those stemming from motor noise or other sources (magnitude  $<$  threshold). The approach to determining this threshold is to set it with a certain security margin above the noise level. I achieve this by configuring the ICU-30201 sensor to only receive without transmitting, effectively capturing the environmental noise. Then the drone flies up to a height of 0.6 m and hovers, while recording the noise. For every measurement, the average noise (`avg_noise`) of the 340 samples is taken:

$$\text{avg\_noise} = \frac{\sum_{i=0}^{339} x_i}{340}$$

where  $x_i$  represents the  $i$ -th sample of the measurement. Subsequently, I compute the global average noise (`final_noise_avg`) over multiple measurements (`nb_meas`) using a sliding window in the code to reduce memory requirements:

$$\text{final\_noise\_avg} = \frac{\sum_{j=0}^{\text{nb\_meas}-1} \text{avg\_noise}_j}{\text{nb\_meas}}$$

Finally, I compute the threshold based on the `final_noise_avg`, incorporating both a linear safety margin of 50 % and an offset of 1500, determined heuristically:

$$\text{threshold} = 1.5 \times \text{final\_noise\_avg} + 1500$$

This approach, contrary to one based on previous motor noise measurements [2], en-

#### 4. Implementation

sures reliable and accurate measurements even in scenarios where motor noise increases significantly due to usage (see Fig. 4.4).

##### B. Ringdown filter

After sensor calibration, I address ringdown artifacts, which occur during the transmission of the early samples due to vibration. These artifacts are identifiable by their notably higher amplitude compared to subsequent samples, even in the absence of nearby obstacles. This presents a measurement challenge, as an elevated sample amplitude may surpass the threshold, falsely indicating the presence of an obstacle near the drone. To overcome this issue, I implement a ringdown filter as follows: following calibration, I re-configure the sensor for transmission and reception. I then measure the first 20 samples across multiple measurements, computing the average of each sample and storing it in a ringdown filter array at the index of the sample:

$$ringdown\_filt_i = \frac{\sum_{j=0}^{nb\_meas-1} x_{ij}}{nb\_meas}$$

Here,  $x_{ij}$  represents the i-th sample of the j-th measurement and  $ringdown\_filt_i$  is the i-th entry of the ringdown filter array. However, it's essential to ensure there are no obstacles within the first 40 cm in front of the drone to avoid filtering out real objects. During the OA algorithm, the amplitude of the first 20 samples with their corresponding entry in the ringdown filter is consistently reduced:

$$amp_i = x_i - ringdown\_filt_i$$

before comparing them to the threshold. This method (see Fig. 4.5) effectively eliminates ringdown artifacts.

### 4.3. Sensor Fusion OA algorithm

This section outlines the complete pipeline of the OA algorithm implementation onboard the CF. Fig. 4.6 illustrates the integration of my implementation into the CF firmware. In contrast to the approach described in [1], I now employ two sensors to gather data for the algorithm, thereby increasing both its complexity and robustness. In Fig. 4.6, inspired by [1], the green blocks represent the original CF firmware, while the red blocks denote my OA algorithm implementation. The base firmware already conducts state estimation based on the onboard IMU for acceleration and orientation estimation, as well as the sensors located on the *Flow-deck v2*: the optical flow sensor used for horizontal velocity estimation and the downward-facing 1-D ToF sensor used for height estimation.

#### 4. Implementation

This information is then utilized by the eKF to generate state estimations for position, velocity, acceleration, and attitude. My implementation leverages data from both the *VL53L5CX* sensor and the *ICU-30201* sensor to compute optimized forward and upward velocity commands and steering inputs, enabling the drone to safely navigate around obstacles within its FoV. Following computation, these commands are transmitted to the onboard controller of the CF firmware, which subsequently adjusts the drone's behavior by modulating Pulse Width Modulation (PWM) signals to its four motors.

In the OA algorithm implementation, I initially conduct feature extraction from both the *VL53L5CX* and *ICU-30201* sensors separately. Whenever one sensor provides available data, I employ a decision tree using this new data together with the most recent data from the other sensor to determine the forward/upward velocity and steering rate.

##### A. Feature extraction

###### 1. ToF sensor

For the feature extraction of the ToF sensor, I adopt the methodology outlined in [1]. Initially, all pixels associated with distances exceeding 2 m are eliminated to enhance reliability, as measurements beyond this threshold exhibit a validity decrease below 90 %. Consequently, an occupancy binary matrix is established, describing whether each pixel has detected an obstacle. Subsequently, neighboring pixels detecting obstacles are grouped into clusters through a Depth-First Search (DFS) algorithm (referred to as grouping), thereby treating overlapping objects within the sensor's FoV as one object. Furthermore, to enhance robustness against outliers or noise, groups are required to meet a minimal size criterion of 2, thus ignoring occupied pixels without occupied neighbours, likely indicative of false positives. Each group has the following attributes:

- minimal and maximal X/Y coordinates (borders);
- number of occupied pixels;
- position (averaged position of all pixels belonging to the group);
- minimum distance to the object;

Upon computing all groups, the closest group to the drone (determined by the minimum distance attribute) is identified, and only this group is forwarded to the Decision Tree as illustrated in Fig. 4.7.

###### 2. Ultrasound sensor

For the feature extraction of the ultrasound sensor, the initial step involves reading

#### 4. Implementation

out the 340 samples of the measurement, which are stored as IQ values. Once these values are acquired, they are converted to amplitude values using the formula:

$$amp_i = \sqrt{I_i^2 + Q_i^2}$$

Subsequently, the amplitude of the first 20 samples get reduced by their corresponding amplitude stored in the ringdown filter (refer to paragraph 4.2):

$$amp_i = amp_i - ringdown\_filt_i$$

Following this adjustment, I start with the first modified sample ( $amp_i$  with  $i = 0$ ) and compare it with the threshold computed in 4.2. There are 2 possible outcomes:

- $amp_i > \text{threshold}$ : then I have identified the closest object and cease further comparisons
- $amp_i \leq \text{threshold}$ : then I have not identified any object yet and thus I repeat the same process with the next sample (incrementing  $i$  by one)

Upon executing this algorithm, I either detect the closest object or fail to identify any object. In the case of obstacle detection, I know the distance to the object based on the frequency of the ultrasound and the index of the sample, which is then transmitted to the Decision tree. Conversely, if no object is detected, I simply forward the maximum range of the ultrasound sensor as the distance to the closest object to the Decision tree. The entire process is illustrated in Fig. 4.8.

#### B. Decision Tree

Fig. 4.10 depicts the flow diagram detailing the functionality of my OA algorithm. This flow operates within an infinite loop and is engineered to minimize latency and complexity. As a safety measure, the system continually monitors the battery level and initiates landing when the battery level is low. The "New sensor data available?" trigger, described in the Feature Extraction paragraph 4.3 triggers the decision tree whenever new data is acquired from either sensor. Subsequently, a new flight command is computed based on the updated sensor data and the most recent information from the other sensor, or retains the last flight command if no new data is available. To simplify the computation of the new flight command, I partition the 8x8 ToF pixel matrix into four distinct zones (i.e., Ground, Ceiling, Caution, and Drone) and two sides (i.e., left and right), as illustrated in 4.11[1]. Moreover, given the focus on indoor environments where the floor and ceiling are predominantly flat, I command the drone to maintain a fixed height (0.6 m) from the floor. The new flight command is then computed based on the object's distance and zone within the *VL53L5CX* sensor's FoV and the distance detected by the *ICU-30201* sensor. In the following, I divide the computation of the new flight command into three parts: Ground/Ceiling detection, Forward velocity computation, and Steering rate computation.

#### 4. Implementation

##### 1. Ground/Ceiling detection

Detection of an object on the Ground or Ceiling relies solely on the *VL53L5CX* data, since the 8x8 ToF pixel matrix (refer to Fig. 4.7) provides insights into the obstacle's location in the room, while the ICU-30201 sensor offers information solely regarding the object's distance. Initially, the OA algorithm verifies if the object detected by the ToF sensor is within the Ground area and closer than DIS\_GROUND\_MIN (40 cm); if so, the drone ascends. Alternatively, if the detected object is within the Ceiling area and closer than DIS\_CEILING\_MIN (60 cm), the drone descends. Both DIS\_GROUND\_MIN and DIS\_CEILING\_MIN were empirically determined by [1]. If both checks fail, the algorithm proceeds with computing the forward (x) velocity and the steering rate.

##### 2. Forward velocity computation

To determine the forward (x) velocity, I use the information of the last five ultrasound measurements to enhance robustness against outliers, along with the most recent input of the ToF sensor. The main idea is to establish two gain functions, each for one sensor, which diminish as an object approaches, and then merging them using dynamically calculated weights to get the best out of both sensors. To achieve this, I first define a piecewise linear function (see Fig. 4.12) to calculate a gain\_dist function for both sensors, ranging within [0,1].

Subsequently, I scale the last five ultrasound distance measurements with this function 4.12 and compute a weighted average of these adjusted values:

$$gain\_dist\_us = 0.45 \times us[0] + 0.3 \times us[1] + 0.15 \times us[2] + 0.075 \times us[3] + 0.025 \times us[4]$$

where  $us[0]$  represents the most recent ultrasound measurement. Intuitively, the most recent measurement holds the highest weight (0.45), gradually diminishing as we move backward in the sequence, yielding a  $gain\_dist\_us$  value ranging between 0 and 1 (the weights add up to 1). Following a similar process, I obtain a  $gain\_dist\_tof$  value from the minimal distance detected by the ToF sensor.

Furthermore, leveraging the ToF sensor's insights into the object's location, I refine the  $gain\_dist\_tof$  variable, particularly when the object is within the predefined minimal reaction distance of 1.4 m 4.12. I distinguish between three cases: either the object is in the caution zone, the danger zone, or neither of them. I adjust the  $gain\_dist\_tof$  variable accordingly based on 4.13, decreasing it if the object is in a more dangerous zone (i.e., the drone zone), thus enhancing safety.

The next step involves combining both gain functions through weighted averages, where I assign increasing weights to the ultrasound sensor as objects approach. This is because I aim to avoid obstacles like glass, which are detected solely by the ultrasound sensor. Thus, I chose the following weight for the ICU-30201 sensor:

$$weight\_us = 1 - gain\_dist\_us$$

#### *4. Implementation*

The weight for the gain\_dist\_tof function is then computed as follows:

$$weight\_tof = 1 - weight\_us$$

Subsequently, I calculate the overall gain\_dist function, incorporating inputs from both sensors:

$$gain\_dist = weight\_us \times gain\_dist\_us + weight\_tof \times gain\_dist\_tof$$

Finally, I constrain the velocity within the range of  $-0.15\text{ m/s}$  to  $1\text{ m/s}$ :

$$vel\_x = gain\_dist \times 1.15 - 0.15$$

with velocity increments limited to  $0.05\text{ m/s}$  and decrements to  $0.25\text{ m/s}$  from the previous velocity to ensure a smooth and stable flight behavior. Referencing Fig. 4.14 provides a comprehensive overview of the forward velocity computation process.

### 3. Steering rate computation

The steering rate has two parameters: the velocity at which it turns and the direction in which it turns. First, the velocity at which it turns is calculated based on the previously computed forward velocity, as shown in Fig. 4.9.

#### 4. Implementation

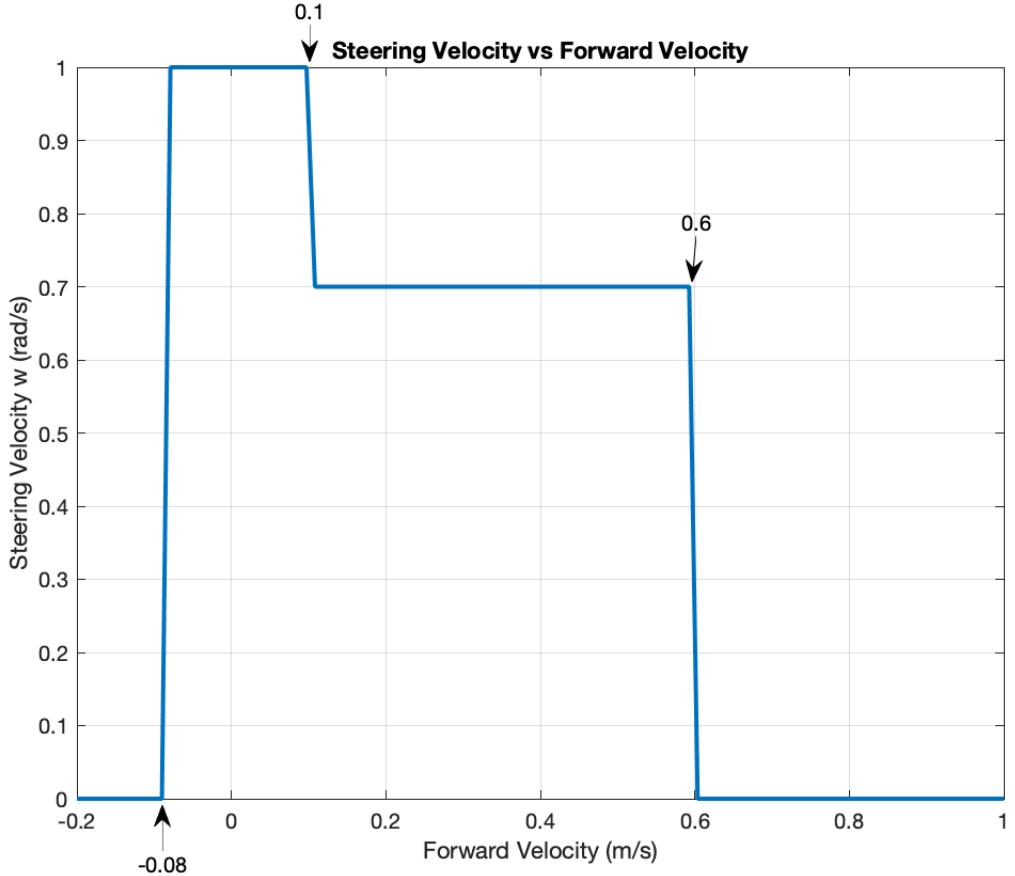


Figure 4.9.: Steering velocity computation based on the forward (x) velocity.

Second, the direction in which it turns is determined based on the position of the object within the ToF sensor's FoV. If the object is detected on the right side, the nano-UAV turns to the left, and vice versa. Moreover, in situations where the ToF sensor fails to detect any object, but the ultrasound sensor does (such as when only a glass obstacle is in front of the drone), the default action is to turn to the right. Additionally, the algorithm also checks for dead ends. For instance, if the nano-UAV finds itself in a narrow corridor with a dead end, it may begin oscillating between right and left turns in an attempt to find an exit. To address this scenario, a record of the recent steering commands is maintained. If the algorithm detects a pattern of repeated oscillations between left and right commands surpassing a predefined threshold, it triggers the drone to execute a  $180^\circ$  turn to escape the corner.

Note that I allow autonomous flight solely with either the connected ToF sensor, utilizing the algorithm outlined in [1], or with only the ultrasound sensor, employing the algorithm detailed in [2].

#### 4. Implementation

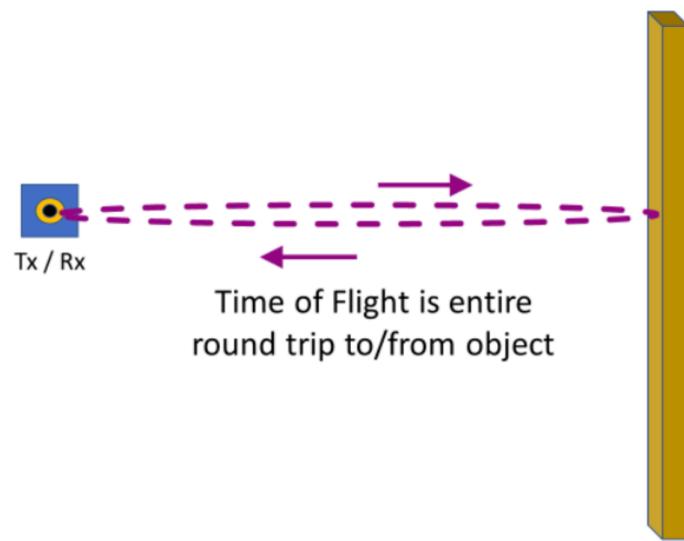


Figure 4.1.: ICU-30201 distance measurement: The sensor records the time  $t$  taken for the sound wave to return, while the known speed of sound  $v$  is 343 m/s at room temperature. Utilizing the formula  $d = v \times t/2$ , the distance to the object can be accurately calculated.

#### 4. Implementation

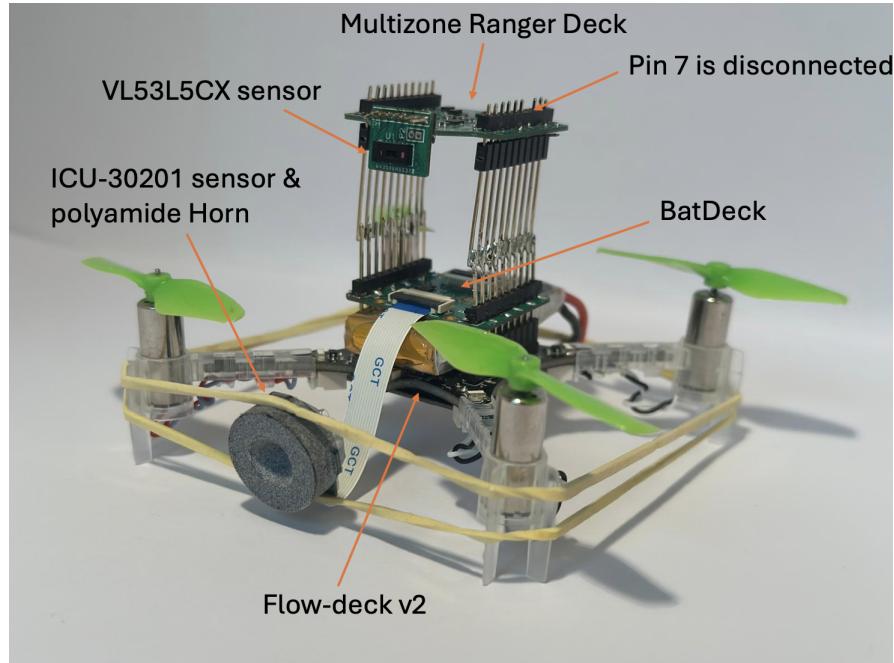


Figure 4.2.: The final setup of the CF with the expansion boards: On top you can see the Multizone Ranger Deck with the *VL53L5CX* sensor, as the battery placeholder you can see the BatDeck and underneath the drone you can see the *Flow-deck v2*.

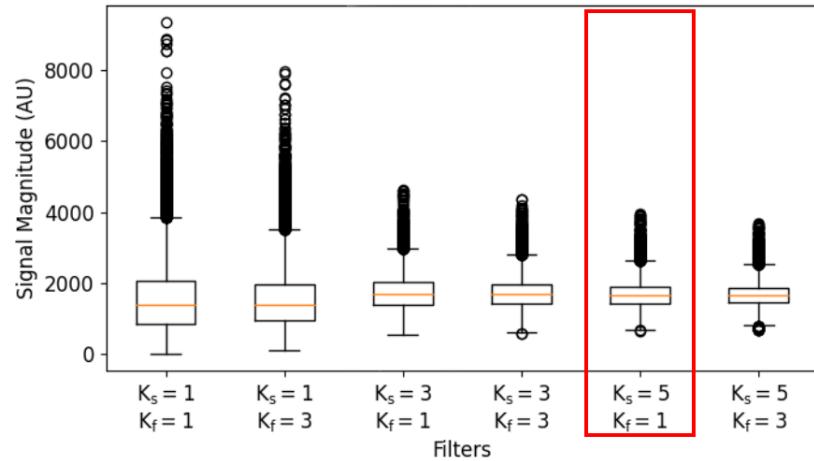


Figure 4.3.: Behaviour of motor noise standard deviation with different filter lengths in slow and fast time [2].

#### 4. Implementation

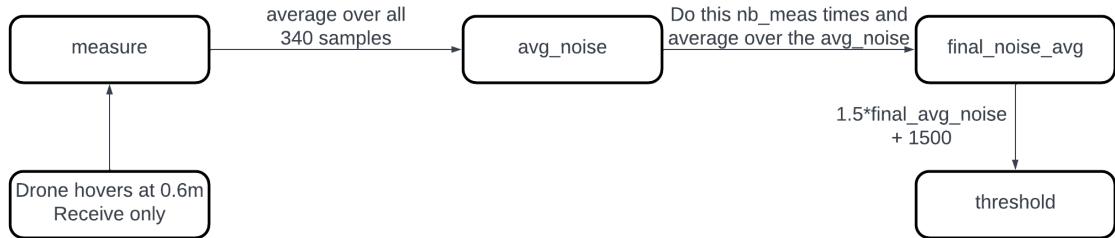


Figure 4.4.: Approach on how to set the threshold for the ICU-30201 sensor

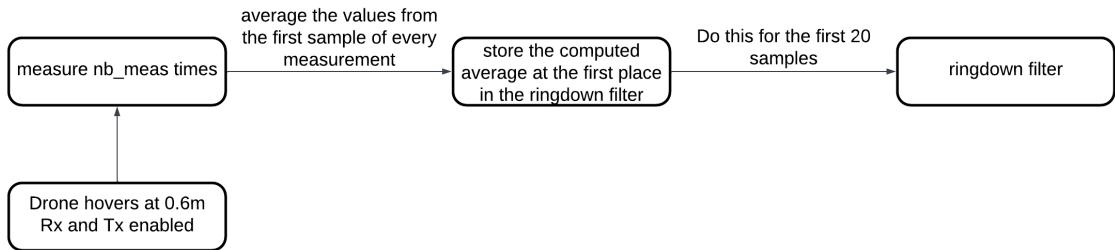


Figure 4.5.: Approach on how to calculate the ringdown filter array

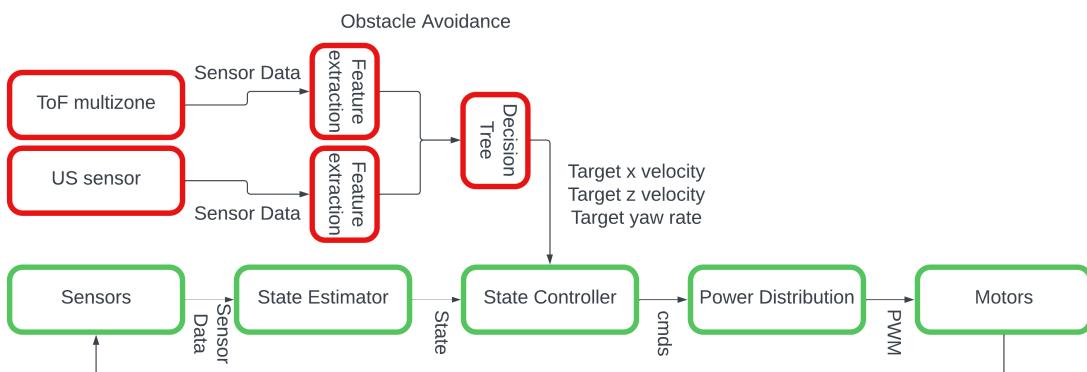


Figure 4.6.: Integration of the OA algorithm into the CF firmware. My additions are shown in red, while the blocks from the original firmware are shown in green [1].

#### 4. Implementation

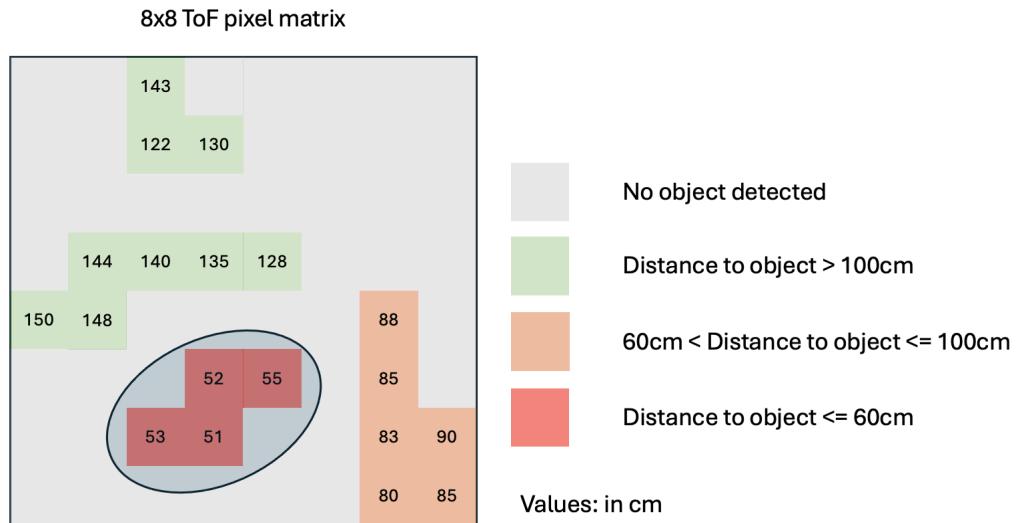


Figure 4.7.: Decision on which group gets forwarded to the Decision tree. In this case, the encircled group gets forwarded, since it has a minimal distance of 51 cm which is closer than the minimal distance to any of the other groups.

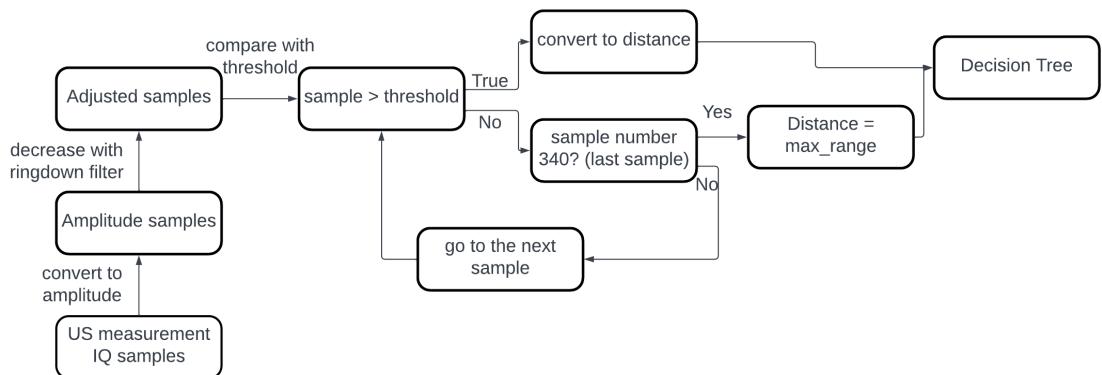


Figure 4.8.: Ultrasound feature extraction: How the distance to the closest object gets extracted out of the ultrasound measurement.

#### 4. Implementation

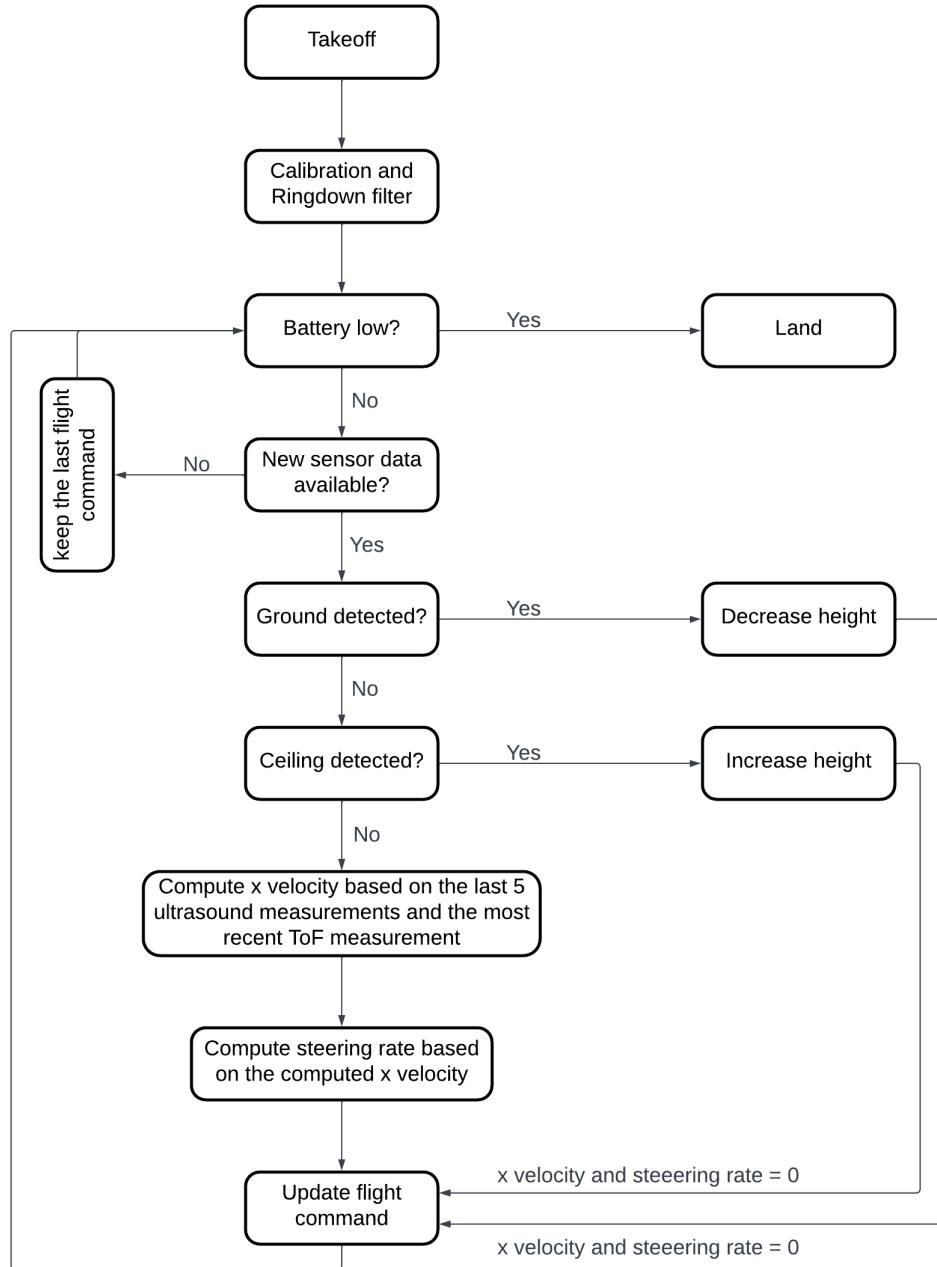


Figure 4.10.: Obstacle Avoidance flow diagram. Illustrates how the flight command gets updated every cycle. The x velocity gets computed as explained in 2 and the steering rate as explained in 3.

4. Implementation

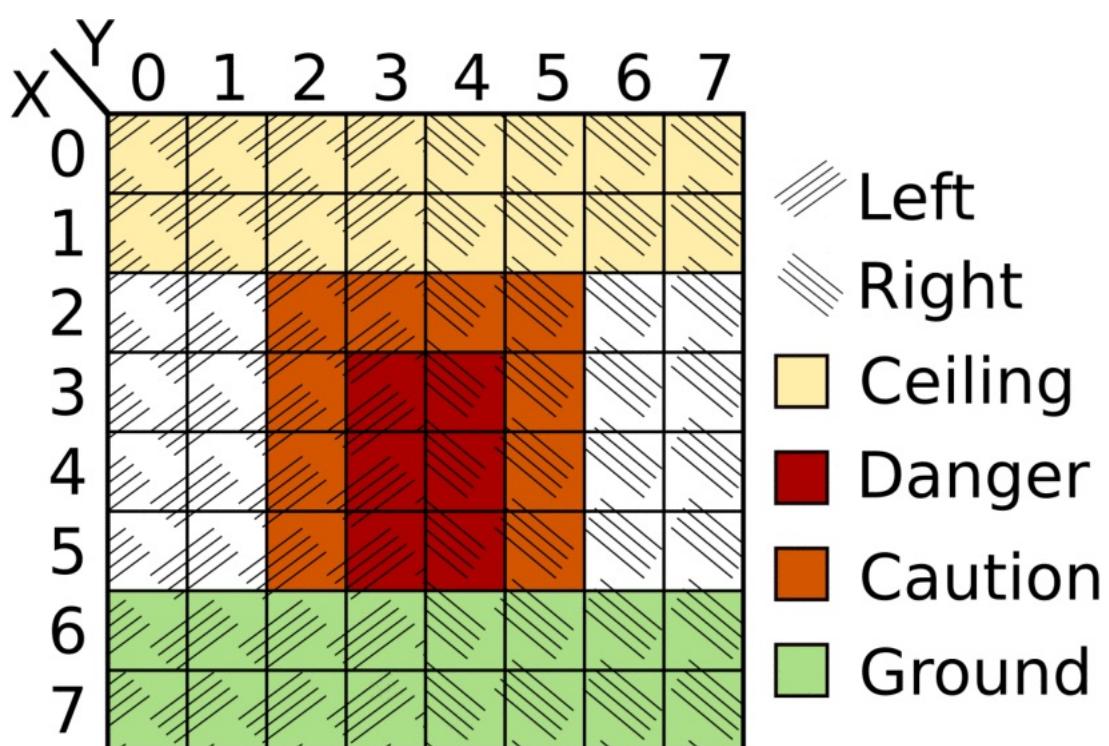


Figure 4.11.: 8x8 ToF pixel matrix divided into 4 zones, Ceiling, Ground zone, Caution and Danger zone [1].

#### 4. Implementation

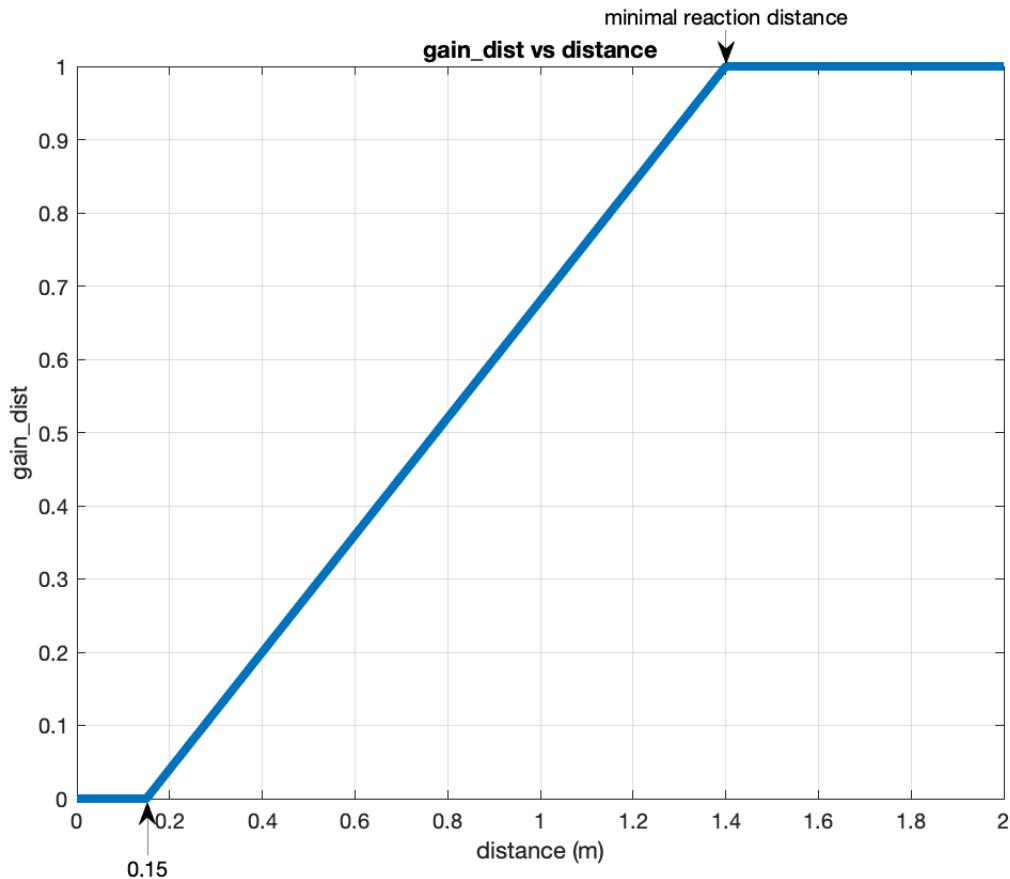


Figure 4.12.: The gain\_dist function takes values in the range [0,1], with higher values indicating that the sensor detects an obstacle at a greater distance.

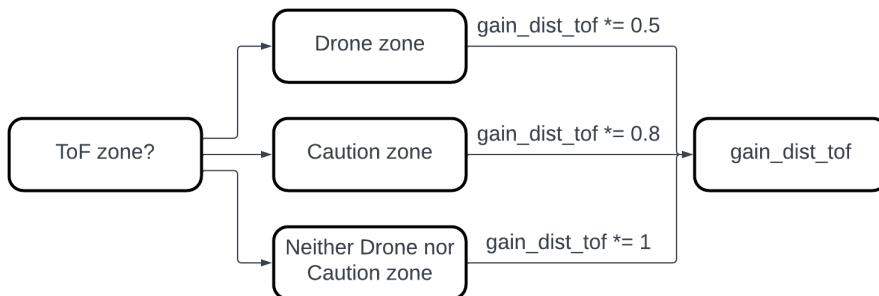


Figure 4.13.: The gain\_dist\_tof variable is adjusted based on the zone where the VL53L5CX sensor detected the object (refer to Fig. 4.11). Lower values indicate higher safety due to reduced x velocity.

#### 4. Implementation

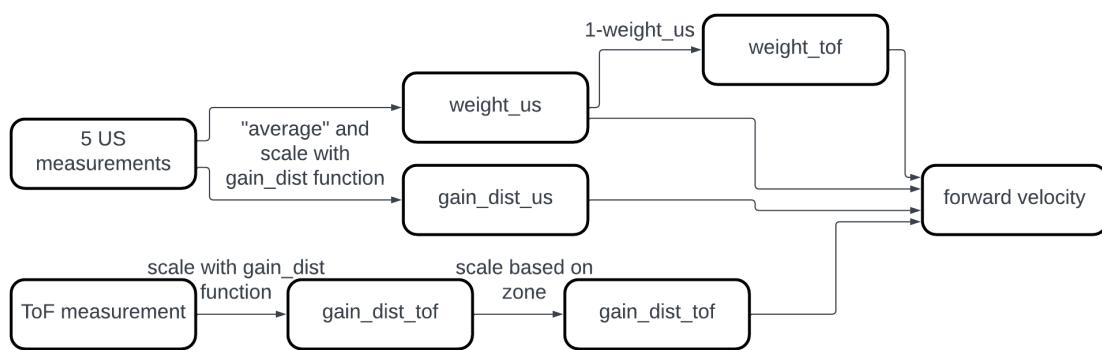


Figure 4.14.: Calculation of the forward velocity based on the introduced gain\_dist function 4.12.

# Chapter 5

## Results

This section evaluates my system, showcasing its OA and exploration capabilities through real-world experiments, discussing the limitations of the ICU-30201 sensor, and analyzing the power consumption due to the increased payload. Additionally, I compare my results with the current SoA.

### 5.1. Power Consumption Due to Payload

I measured the weight of the different components used in my setup and estimated the power consumption based on these weights and the estimations from [1].

#### A. Results

I evaluated my system in four different configurations:

1. The first setup, where neither the ultrasound nor the ToF sensor is connected, has a final weight of 34.3 g (27.2 g initial drone, 1.7 g flow-deck, 1.1 g heavier battery, 1.37 g BatDeck, 2.93 g pin headers [1]).
2. In the second setup, only the ultrasound sensor is connected, resulting in a final weight of 35.87 g (34.3 g first setup, 0.4 g rubber band, 0.69 g horn, 0.48 g ultrasound sensor + connection cable).
3. The third setup includes only the ToF sensor, leading to a final weight of 36.58 g (34.3 g first setup, 2.28 g Multizone Ranger Deck + forward-facing sensor [1]).
4. The fourth and most commonly used setup connects both sensors, resulting in a final weight of 38.15 g (36.58 g second setup, 2.28 g Multizone Ranger Deck + forward-facing sensor).

## 5. Results

All four setups are shown in Fig.5.1. The maximum payload is 42 g, but a decrease in maneuverability and agility can already be observed with my setup [1]. Based on the estimations from [1], I estimated the required motor power as 11.16 W, 12.02 W, 12.08 W, and 12.53 W for the first, second, third, and fourth setups, respectively, to carry the extra payload, as seen in Fig.5.2.

### B. Comparison with SoA

My setup is based on the work of [1], so I also compare my power consumption with theirs. The final weight is 38.15 g compared to their 35 g, representing only a 9% increase. Additionally, the required motor power estimation increases by just 21.89% (12.53 W compared to 10.28 W) for the fourth setup. Overall, I successfully connected an additional sensor without significantly increasing the required power consumption.

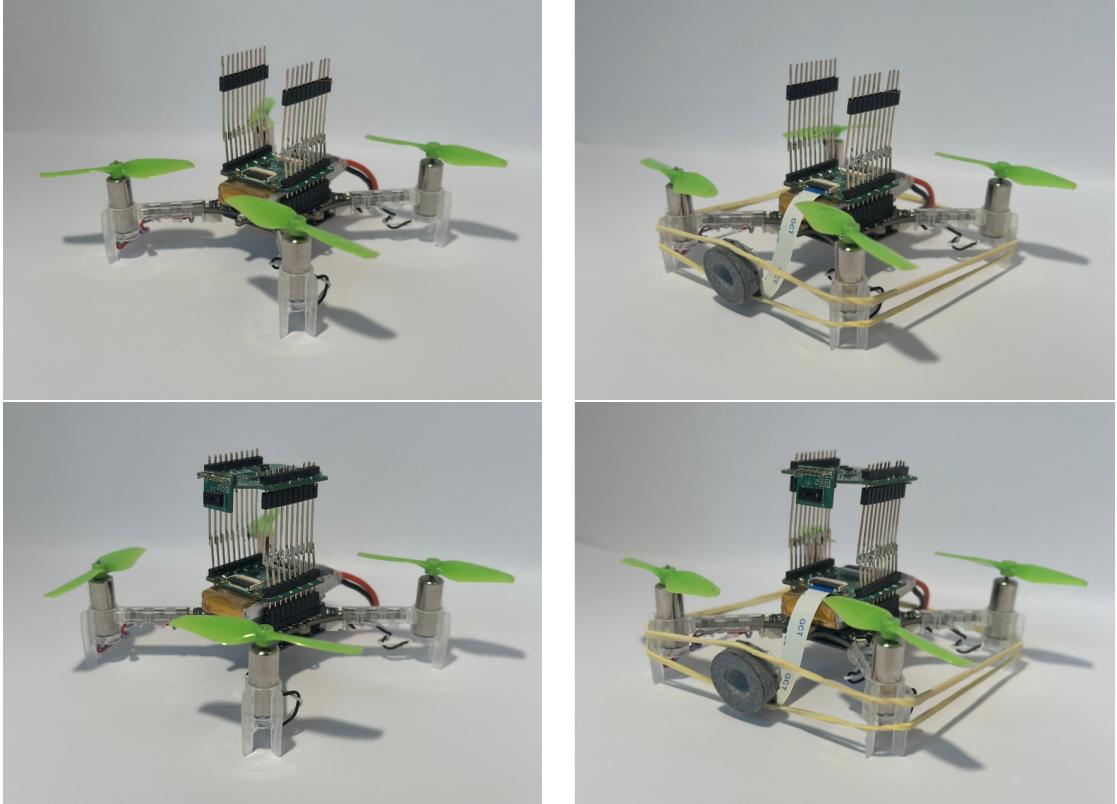


Figure 5.1.: In the top row, the first setup is shown on the left and the second on the right. In the bottom row, the third setup is on the left and the fourth on the right.

## 5. Results

### 5.2. Narrow Corridor [1]

I constructed a 3 m long narrow corridor with varying widths, placing a plexiglass barrier at the end to assess the drone's ability to navigate through tight spaces, identify glass obstacles, and execute turns. The experiment aimed for the drone to navigate 3 m into the narrow corridor, detect the glass barrier, execute a 180° turn within the corridor, and then exit the 3 m corridor. Initially, the drone was positioned 1 m from the corridor entrance, oriented to fly straight through it.

#### A. Results

I conducted 5 trials for each width: at 85 cm, the maneuver succeeded 3 times with an average escape time of 33 s; it crashed once during the turn and once became stuck in front of the glass until the battery depleted, and it landed (after 60 s). For the 75 cm width, the maneuver succeeded 3 times with an average escape time of 33.3 s; it crashed twice while attempting to turn in front of the glass. At a width of 65 cm, the maneuver never succeeded; three times, the drone immediately turned around or retreated, once it crashed while attempting to turn in front of the glass, and once it successfully turned but landed due to low battery. However, it is worth noting that the drone could navigate through smaller openings if obstacles were not pipes, but shorter in length. Additionally, most crashes during turns occurred because the drone turned around but remained too close to the corridor, resulting in backward collision with the glass. Integrating an ultrasound sensor facing backward to detect glass obstacles could overcome this issue. Fig. 5.3 shows the experimental setup and the results.

#### B. Comparison with SoA

Authors in [1] conducted a similar experiment involving flying the same CF through a narrow corridor of varying widths. However, my experiment differs in several aspects: my corridor length is 3 m (compared to their 4 m), I maintain a flight height of 0.6 m (compared to their 0.4 m), and I conducted 5 trials for each width (compared to their 3). However, one has to note that my setup is more complex as it involves turning 180° inside the corridor in front of a glass obstacle and maneuvering back out. It's also worth mentioning that up to a width of 75 cm, we both achieved a 100 % success rate when traversing the corridor, with crashes occurring only during turning maneuvers. At a width of 65 cm, my algorithm successfully navigated the corridor in 2 out of 5 attempts (40 %), compared to their 1 out of 3 attempts (33 %). It's important to consider that my corridor was shorter with 3 m, yet both experiments yielded similar results when navigating narrow spaces. Additionally, my experiment introduced the challenge of turning in front of the glass obstacle, significantly impacting success rates. Notably, the implementation of [1] would likely have a 0 % success rate in my experiment, as it lacks the capability to detect glass with only a ToF sensor connected. Overall, my implementation enables the CF to navigate narrow corridors with similar reliability while extending its capabilities to detect and respond to reflective materials like glass.

## 5. Results

### 5.3. Reliability Test [1]

I evaluated the reliability of my system in a real-world experiment through 20 flights conducted in an office environment (meeting room) measuring 11 m x 6 m, as depicted in Fig. 5.4. The meeting room is equipped with six tables, numerous chairs, and various other utilities, including a projector, jacket rack, trash can, plexiglass, and four closed doors (in my setup). I set the drone to fly at a height of 0.6 m above the ground and tested it at a maximum speed of 1 m/s as suggested by [1], because the greatest distance was covered at that maximum velocity in the same meeting room.

#### A. Results

My testing resulted in an 80 % reliability rate, with only four crashes occurring in 20 flights. The average flight time and distance were 130.3 s and 98.39 m, respectively. Notably, even when a crash occurred, the drone often completed several OA scenarios, given that each trial encompassed several minutes of fully autonomous flight. Furthermore, in 75 % (3 out of 4) of the crashes, the drone collided with the plexiglass at a steep angle, because of unreliable detection by the ultrasound sensor, as discussed in Section 5.4. This issue is primarily hardware-related. Addressing it may involve increasing the FoV of the ultrasound sensor. Furthermore, I calculated the flight time and distance covered for each flight, and I depict the corresponding plots, akin to those in [1], in Fig. 5.5, facilitating a comparison between the two studies as done in the following paragraph 5.3.

#### B. Comparison with SoA

Given the similarities between my experiment and the Reliability Test conducted in [1] with a maximum velocity of  $v_{max} = 1$  m/s, I compare my findings with theirs. The primary difference in the setup is the inclusion of a plexiglass obstacle in the meeting room configuration, necessitating the removal of one table to create a passage for the drone 5.4. Additionally, I configured the drone to fly at a height of 0.6 m instead of 0.4 m to explore higher areas. My results indicate a comparable reliability of 80 %. However, it's worth noting that 75 % of the crashes occurred as a result of encounters with the new plexiglass obstacle, highlighting the effectiveness of my OA system in detecting and navigating around obstacles identified by both sensors. Nonetheless, my average flight time of 130 seconds contrasts with their 360 seconds. This reduction can be attributed to the added payload of the BatDeck and the ICU-30201 sensor 3 and the usage of older batteries. Despite this, my average flight distance remains almost identical, suggesting that my implementation corrects its flight path early to maintain a high velocity. But one also has to point out that the drone did not fly under tables during my tests (contrary to their implementation), likely due to the increased flight height and the ultrasound sensor's recognition of the table as an obstacle, leading to avoidance maneuvers. This phenomenon also partly explains my implementation's ability to sustain higher velocities over extended periods of time. In conclusion, both implementations demonstrate high reliability, with my setup enhancing theirs by effectively avoiding reflective obstacles, provided the approach angle is not excessively steep 5.4.

## 5. Results

### 5.4. Ultrasound Angle Detection

To measure the maximum angle of approach at which the ultrasound sensor can still detect reflective obstacles (e.g., plexiglass), I positioned the drone 1 m away from the glass. I then varied the angle  $\alpha$  while keeping the distance  $d = 1 \text{ m}$  constant [5.6], and measured the sensor readings using only the ICU-30201 sensor while the drone was hovering. The real-world setup is also depicted in Fig. 5.6.

#### A. Results

My experiment demonstrates that the ICU-30201 sensor can reliably detect reflective material (e.g., plexiglass) up to an angle of  $40^\circ$ , as shown in Fig. 5.7. At  $50^\circ$ , the ICU-30201 sensor occasionally detects the obstacle, but not consistently. Sometimes, it shows a clear and high amplitude peak at 1 m, while other times it only shows a small peak (see Fig. 5.8), likely not recognizable by my algorithm due to the safety margin in the threshold (see section 4.2), which is probably higher than the measured peak. At  $60^\circ$ , the combination of the ICU-30201 sensor and the safety margin fails to reliably detect reflective objects (Fig. 5.9), due to the minimal difference in magnitude between the motor noise and the object's signal. In conclusion, the ICU-30201 sensor with the chosen horn reliably detects reflective objects up to an approach angle of  $40^\circ$ . At  $50^\circ$ , it occasionally detects the object, but at  $60^\circ$  or higher, my implementation is no longer reliable. To increase the reliable detection angle, one can either widen the FoV of the horn and thus the sensor or lower the detection threshold, though the latter would reduce robustness to outliers.

#### B. Comparison with SoA

Comparing the maximum angle of approach with other studies is challenging, as each used different ultrasound sensors, horns, and materials. However, authors in [32] found that reliable detection is possible up to a tilt angle  $\alpha$  of  $25^\circ$  but which can be increased to  $40^\circ$ - $45^\circ$  by operating with a higher gain on the receiver circuit. This matches the maximum tilt angle I measured. Additionally, authors in [2] found that a smooth obstacle is detectable from 1 m distance at a tilt angle of  $45^\circ$ , with a peak magnitude of about 9000 (where the noise level is around 3500). Although they used the same FoV of  $55^\circ$ , they employed a different material for the Horn and different filter parameters ( $K_s = 1$ ,  $K_f = 3$  compared to my  $K_s = 1$ ,  $K_f = 5$ , see section 4.2). They did not, however, reproduce the experiment at higher tilt angles. Overall, I can conclude that my implementation performs comparably to the SoA. To increase the tilt angle at which the sensor can detect objects, the FoV would need to be increased.

## 5. Results

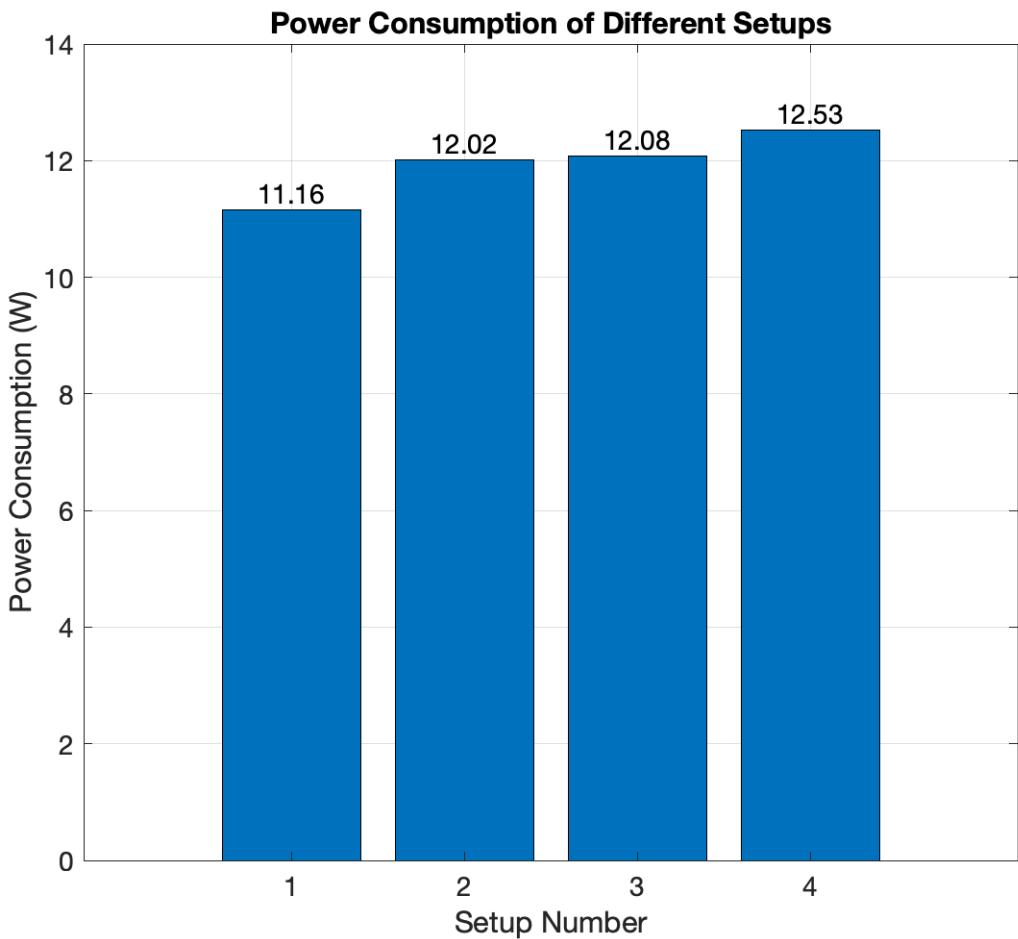


Figure 5.2.: Power consumption for the different setups based on their weights. The number on the x axis represents the number of the setup and the y axis represents the power consumption in Watts. The most used setup is the fourth one with a power consumption of 12.53 W 5.1.

## 5. Results

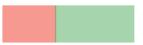
	Picture (75cm)	Width	Outcome	#failed manoeuvres	Time to escape
Narrow corridor		85cm	<b>3x escaped</b> 1x stuck when turning 1x crash when turning	 3/5	33s
		75cm	<b>3x escaped</b> 2x crash when turning	 3/5	33.3s
		65cm	<b>1x flew in and turned but then battery was low</b> <b>3x turn around</b> <b>1x crash when turning</b>	 0/5	/

Figure 5.3.: Flying through a narrow corridor, then turning and flying out again with different widths [1].

## 5. Results

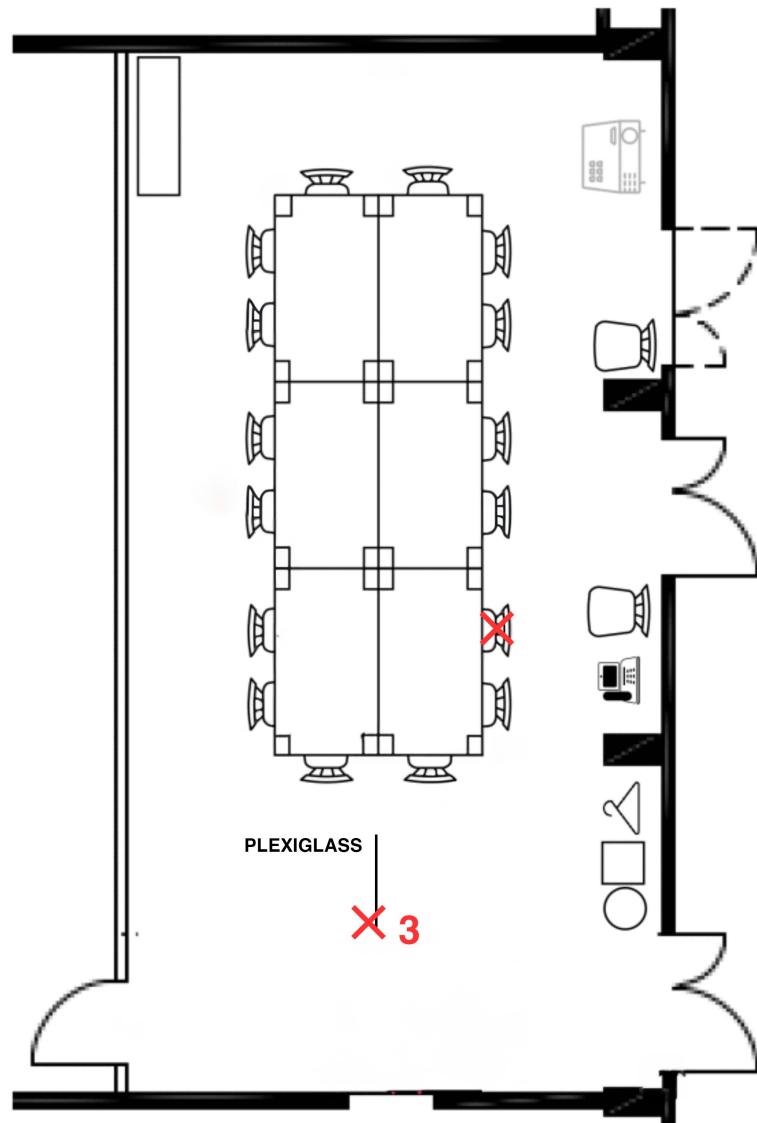


Figure 5.4.: Floorplan of the meeting room in which I ran the reliability test. The red crosses indicate the locations of recorded crashes [1].

## 5. Results

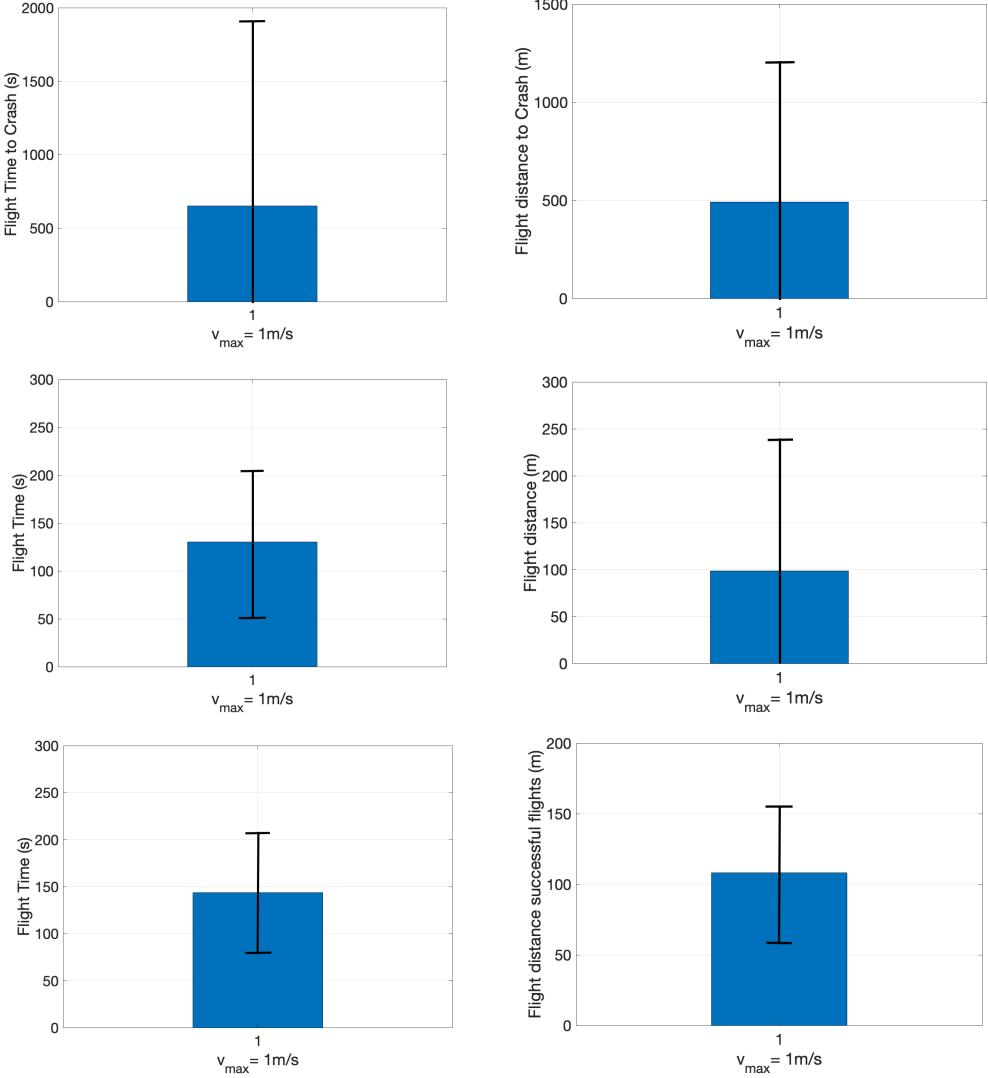


Figure 5.5.: In the top row, I present the average time and distance until a crash occurs, excluding battery changes except at the test's conclusion. The middle row illustrates the average flight time and distance. The bottom row depicts the average flight time and distance, exclusively for successful flights. The blue bars indicate the average, while the black lines represent the variance [1].

## 5. Results

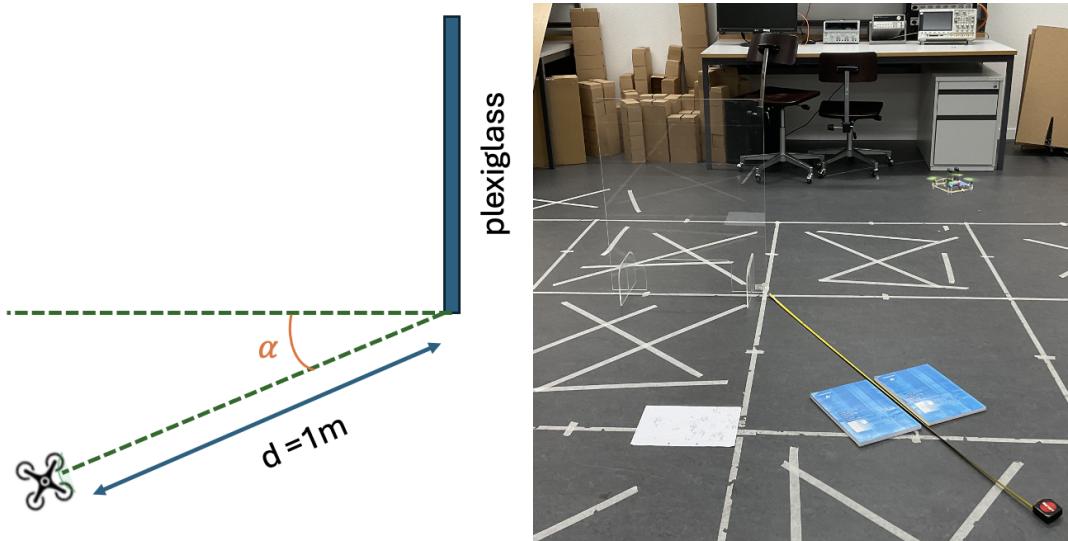


Figure 5.6.: Setup of the Ultrasound Angle Detection experiment with varying approach angle  $\alpha$  and constant distance  $d = 1\text{ m}$  in both schematics (on the left) and real-world (on the right).

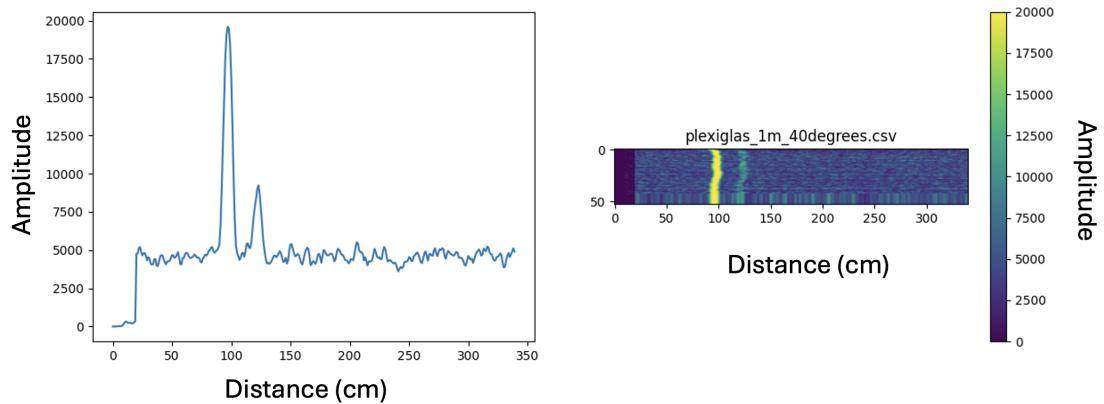


Figure 5.7.: Plot of the measurement amplitude of the ICU-30201 sensor at a  $40^\circ$  angle in front of a plexiglass.

## 5. Results

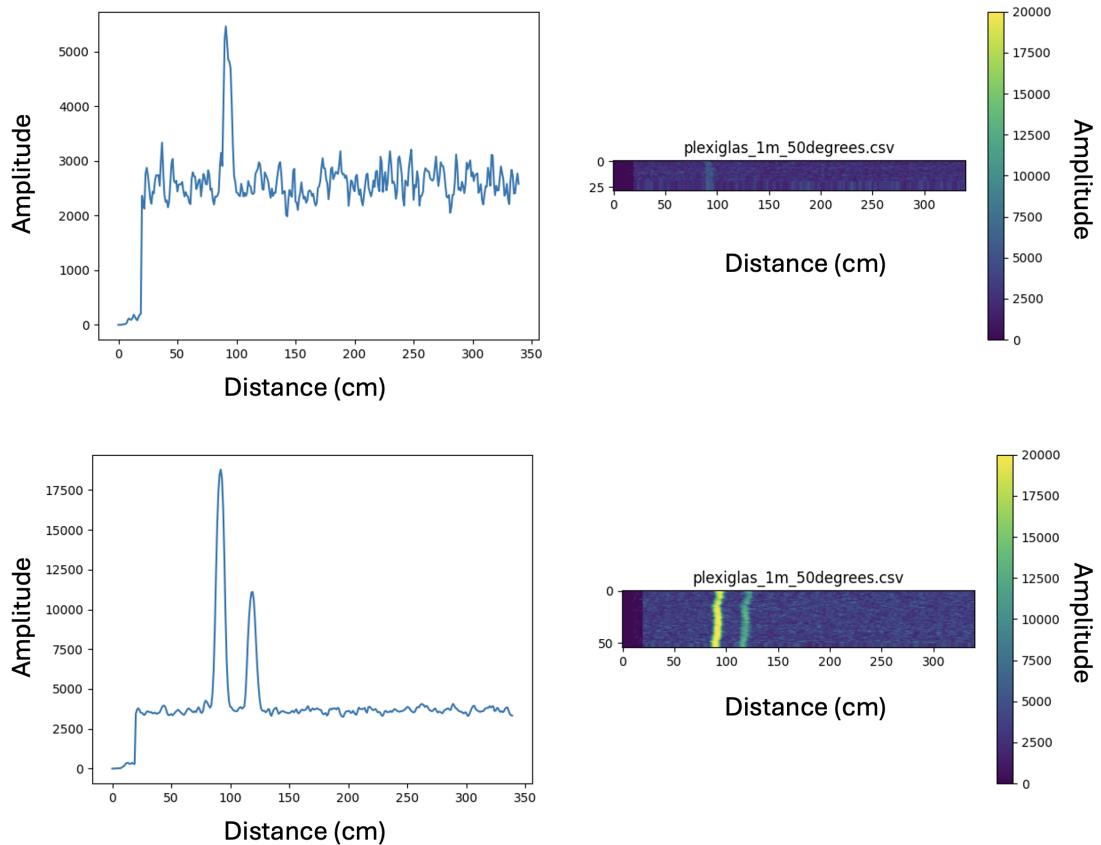


Figure 5.8.: Two measurements of the ICU-30201 sensor at a  $50^\circ$  angle in front of a plexiglass. The first measurement, with a magnitude of around 5500, would likely not trigger detection in my implementation, while the second measurement, with a magnitude above 17500, would clearly recognize the plexiglass at 1 m.

## 5. Results

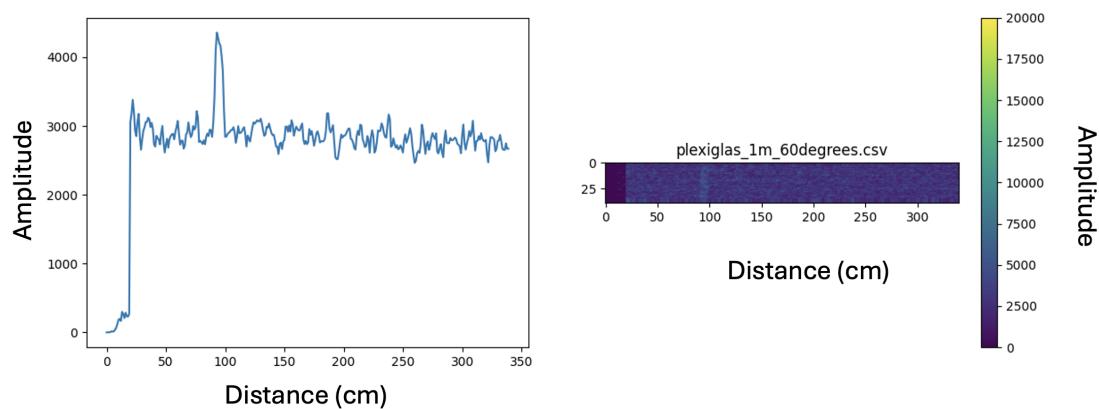


Figure 5.9.: Plot of the measurement amplitude of the ICU-30201 sensor at a  $60^\circ$  angle in front of a plexiglass. The measured amplitude is insufficient to exceed the threshold (see Fig. 4.2), making it impossible to recognize the plexiglass as an object.

## Conclusion and Future Work

This work, based on [1], presents an onboard OA algorithm for nano-UAVs using sensor fusion. It was tested in indoor environments on the open-source CF, utilizing only the *STM32F405* MCU for onboard computation. In addition to the originally installed sensors, such as the IMU and the downward-pointing *Flow-deck v2*, I integrated a 64-pixel forward-pointing ToF sensor (*VL53L5CX*) from STMicroelectronics and a forward-pointing ultrasound sensor (*ICU-30201*) from TDK. The additional payload of 3.15 g results in an increase of 21.89 % in the necessary power to lift the drone compared to [1].

Furthermore, the author demonstrated the limitations of the ultrasound sensor, identifying a maximum tilt angle of 40° while using a polyamide horn with a FoV of 55°. Despite this constraint, he achieved high reliability with his implementation across various environments, such as narrow corridors and office environments, which included transparent, reflective materials like plexiglass. This demonstrated the system's capability to navigate through narrow spaces and cover distances of up to 186 m in a single autonomous flight.

This work demonstrated that integrating additional sensors, such as an ultrasound sensor, significantly enhances the drone's ability to detect and avoid obstacles of various forms and materials.

The presented study could serve as a foundation for developing more complex obstacle avoidance algorithms. For instance, integrating a backward-facing ultrasound sensor could detect obstacles while flying backwards, thereby increasing the system's robustness and reliability (see paragraph 5.2). Additionally, exploring a wider FoV for the ultrasound sensor might allow for the detection of obstacles at higher tilt angles, thus recognizing obstacles more reliably. However, this approach requires caution, as a wider FoV can reduce the peak amplitude detected due to the spreading of the transmitted wave's energy. This generally results in objects reflecting a lower amplitude, making them more

## *6. Conclusion and Future Work*

difficult to detect, and at some point, the sensor will significantly pick up motor noise. Finally, the presented obstacle avoidance algorithm can be fine-tuned for various speeds and environments by adjusting the gain function introduced in Section 4.3. To conclude, this work lays a strong foundation for future studies in autonomous flight technology, offering a scalable solution for more advanced OA systems on nano-UAVs.

Appendix A

## Task Description



Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich



Task Description for a Bachelor Thesis on

### Obstacle Avoidance With Ultrasonic Sensors on Nano Drones

at the Department of Information Technology and  
Electrical Engineering

for

**Laurent Schroeder**  
lschroeder@student.ethz.ch

**Advisors:** Hanna Müller, hanmuell@iis.ee.ethz.ch  
Dr. Tommaso Polonelli, tommaso.polonelli@pbl.ee.ethz.ch  
**Professor:** Prof. Dr. Luca Benini, lbenini@ethz.ch  
**Handout Date:** 19.02.2024  
**Due Date:** 31.05.2024

## 1 Project Goals

A critical aspect of ensuring a drone platform's safe and efficient functionality is obstacle avoidance (OA). OA requires appropriate sensing technologies that typically include LIDAR, RADAR, cameras, and Time of Flight (ToF) sensors [1, 2]. However, these sensors do not perform well when dealing with reflective surfaces or transparent barriers such as glass walls. While sensor fusion allows larger drones to overcome this issue, nano-drones, typically below 50 grams, must resort to a different approach due to their payload limitations. In light of these limitations, ultrasound technology emerges as a promising alternative. Ultrasound sensors, which measure the reflection of sound waves, have shown better performance in representing the physical world across different materials. This technology has been successfully utilized by various animals in nature, like bats, for navigation and obstacle detection. Recent research highlights the potential of ultrasound sensors to overcome the shortcomings of conventional sensors [3]. Ultrasound has been used for decades [4], however, the transmitters and receivers used to be too bulky for nano-drones which fit in the palm of your hand. Recently, ultrasound sensors have been miniaturized and with this enable the use on lightweight vehicles. In this thesis the student will investigate a novel approach using a lightweight and low-power ultrasound sensor for onboard obstacle avoidance in a Crazyflie 2.1 nano-drone. The student will explore different horn shapes to find the ideal trade-off between a wide field of view and noise reduction, investigate OA algorithms and evaluate performance with in-field tests.

## 2 Tasks

The project will be split into three phases, as described below:

### Phase 1 (Week 1-3)

1. Investigate the state-of-the-art of the single blocks (ultrasonic sensing, obstacle avoidance)
2. Study and get used to the hardware and the tools to program, i.e. microcontroller programming, the Crazyflie, the TDK-ICU30201, sensor data acquisition, 3d-printing.
3. Evaluation of different, already existing (180 degree, 55 degree, combined) horn shapes.

### Phase 2 (Week 4-12)

1. Print and test/characterize different horns for ultrasonic sensors (with different obstacle materials, sizes, distances,...)
2. Implement different OA/random exploration algorithms with 1 or more ultrasonic sensors

3. In-field test and evaluation of the OA
4. Optimization (if any)
5. (optional) Sensor fusion with a multizone laser ranger sensor as the VL53L5CX

### **Phase 3 (Week 13-14)**

1. Power and latency measurements.
2. Report writing and presentation preparation.

### **Milestones**

The following milestones need to be reached during the thesis:

- Testing and characterization of different horns for an ultrasonic sensor
- Implementation of at least 1 obstacle avoidance approach
- In-field tests
- Power and latency measurements
- Final report and presentation

## **3 Project Organization and Grading**

During the thesis, students will gain experience in the independent solution of a technical-scientific problem by applying the acquired specialist and social skills.

The grade is based on the following: Student effort; thoroughness and learning curve; achieving qualitative and quantitative results with a scientific approach; supporting practical findings with theoretical background and literature investigations; final presentation and report; documentation and reproducibility. All theses include an oral presentation, a written report and are graded. The report and presentation need to have publication grade quality to achieve a good grade. Students are graded based on the official ITET grading form<sup>1</sup>.

For students of IIS (Prof. Benini) a special grading scheme exists, please contact your supervisor for details there. Before starting, the project must be registered in myStudies and all required documents need to be handed in for archiving by PBL.

---

<sup>1</sup><https://ethz.ch/content/dam/ethz/special-interest/itet/department/Studies/Forms/Grading%20Form.xlsx>

### **3.1 Laboratory Rules**

The students agree to follow the lab rules set by PBL staff, for detail please contact us. The most important points are:

- All ETH safety regulations need to be followed<sup>2</sup>, in addition to ones given by PBL staff
- No device in the lab is used without introduction by your supervisor or PBL staff
- No device leaves the lab without being officially borrowed, this is done by PBL staff and needs your Legi.
- Any damage to devices or tools needs to be reported immediately to PBL staff.
- The Lab-desk is clean and free for others after you finished your task, or when you take longer breaks. All tools are correctly sorted into their drawers/cupboards when you leave

### **3.2 Weekly Report**

There will be a weekly report/meeting held between the student and the assistants. The exact time and location of these meetings will be determined within the first week of the project in order to fit the students and the assistants schedule. These meetings will be used to evaluate the status and document the progress of the project (required to be done by the student). Beside these regular meetings, additional meetings can be organized to address urgent issues as well. The weekly report, along with all other relevant documents (source code, datasheets, papers, etc), should be uploaded to a clouding service, such as Polybox and shared with the assistants.

### **3.3 Project Plan**

Within the first month of the project, you will be asked to prepare a project plan. This plan should identify the tasks to be performed during the project and sets deadlines for those tasks. The prepared plan will be a topic of discussion of the first week's meeting between you and your assistants. Note that the project plan should be updated constantly depending on the project's status.

### **3.4 Final Report and Paper**

PDF copies of the final report written in English are to be turned in. Basic references will be provided by the supervisors by mail and at the meetings during the whole project, but the students are expected to add a considerable amount of their own literature research to the project ("state of the art").

---

<sup>2</sup><https://ethz.ch/staffnet/en/service/safety-security-health-environment/sicherheit-in-laboren-und-werkstaetten/laborsicherheit.html>

### **3.5 Final Presentation**

There will be a presentation (15 min presentation and 5 min Q&A for BT/ST and 20 min presentation and 10 min Q&A for MT) at the end of this project in order to present your results to a wider audience. The exact date will be determined towards the end of the work.

### **References**

Will be provided by the supervisors by mail and at the meetings during the whole project.

Place and Date Zurich, 08/02/2024

Signature Student 

# Bibliography

- [1] N. Zhang, F. Nex, G. Vosselman, and N. Kerle, “End-to-end nano-drone obstacle avoidance for indoor exploration,” *Drones*, vol. 8, no. 2, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/2/33>
- [2] H. Müller, V. Niculescu, T. Polonelli, M. Magno, and L. Benini, “Robust and efficient depth-based obstacle avoidance for autonomous miniaturized uavs,” *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4935–4951, 2023.
- [3] N. Seckel and A. Singh, “Physics of 3d ultrasonic sensors,” 07 2019. [Online]. Available: [https://toposens.com/wp-content/uploads/2020/02/Physics\\_3D\\_Ultrasonic\\_Sensors\\_Topolosens.pdf](https://toposens.com/wp-content/uploads/2020/02/Physics_3D_Ultrasonic_Sensors_Topolosens.pdf)
- [4] J. Borenstein and Y. Koren, “Obstacle avoidance with ultrasonic sensors,” *IEEE Journal on Robotics and Automation*, vol. 4, no. 2, pp. 213–218, 1988.

# Appendix B

## Declaration of Originality



Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

### Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. In consultation with the supervisor, one of the following three options must be selected:

- I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are accepted. I used no generative artificial intelligence technologies<sup>1</sup>.
- I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are accepted. I used and cited generative artificial intelligence technologies<sup>2</sup>.
- I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are accepted. I used generative artificial intelligence technologies<sup>3</sup>. In consultation with the supervisor, I did not cite them.

Title of paper or thesis:

Obstacle Avoidance With Ultrasonic Sensors on Nano Drones

Authored by:

If the work was compiled in a group, the names of all authors are required.

Last name(s):

Schroeder

First name(s):

Laurent

With my signature I confirm the following:

- I have adhered to the rules set out in the Citation Guide.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date

Zurich, 26/05/2024

Signature(s)


If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

<sup>1</sup> E.g. ChatGPT, DALL E 2, Google Bard

<sup>2</sup> E.g. ChatGPT, DALL E 2, Google Bard

<sup>3</sup> E.g. ChatGPT, DALL E 2, Google Bard

# Bibliography

- [1] H. Müller, V. Niculescu, T. Polonelli, M. Magno, and L. Benini, “Robust and efficient depth-based obstacle avoidance for autonomous miniaturized uavs,” *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4935–4951, 2023.
- [2] H. Müller, V. Kartsch, M. Magno, and L. Benini, “Batdeck: Advancing nano-drone navigation with low-power ultrasound-based obstacle avoidance,” 2024.
- [3] Statista. (2024, January) Drone market size worldwide in selected years from 2021 to 2030. [Online]. Available: <https://www.statista.com/statistics/1234521/worldwide-drone-market/>
- [4] R. E.-P. A. Braverman, BSc, “Unmanned aerial systems (uas) in urban search and rescue-methodology, capacity development, and integration,” *Journal of emergency management (Weston, Mass.)*, vol. 19, no. 1, pp. 33–38, 2021.
- [5] H. Müller, D. Palossi, S. Mach, F. Conti, and L. Benini, “Fünfiber-drone: A modular open-platform 18-grams autonomous nano-drone,” in *2021 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2021, pp. 1610–1615.
- [6] S. Rezwan and W. Choi, “Artificial intelligence approaches for uav navigation: Recent advances and future challenges,” *IEEE Access*, vol. 10, pp. 26 320–26 339, 01 2022.
- [7] M. Coppola, K. N. McGuire, C. De Wagter, and G. C. De Croon, “A survey on swarming with micro air vehicles: Fundamental challenges and constraints,” *Frontiers in Robotics and AI*, vol. 7, p. 18, 2020.
- [8] K. Kalenberg, H. Müller, T. Polonelli, A. Schiaffino, V. Niculescu, C. Cioflan, M. Magno, and L. Benini, “Stargate: Multimodal sensor fusion for autonomous navigation on miniaturized uavs,” *IEEE Internet of Things Journal*, pp. 1–1, 2024.

## Bibliography

- [9] Y. Song, M. Steinweg, E. Kaufmann, and D. Scaramuzza, “Autonomous drone racing with deep reinforcement learning,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 1205–1212.
- [10] N. Gyagenda, J. V. Hatilima, H. Roth, and V. Zhmud, “A review of gnss-independent uav navigation techniques,” *Robotics and Autonomous Systems*, vol. 152, p. 104069, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889022000343>
- [11] C. Wang, J. Wang, Y. Shen, and X. Zhang, “Autonomous navigation of uavs in large-scale complex environments: A deep reinforcement learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2124–2136, 2019.
- [12] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, “Pixhawk: A system for autonomous flight using onboard computer vision,” in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2992–2997.
- [13] N. Elkunchwar, S. Chandrasekaran, V. Iyer, and S. B. Fuller, “Toward battery-free flight: Duty cycled recharging of small drones,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5234–5241.
- [14] S. Bahnam, S. Pfeiffer, and G. C. de Croon, “Stereo visual inertial odometry for robots with limited computational resources,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 9154–9159.
- [15] G. Schouten and J. Steckel, “A biomimetic radar system for autonomous navigation,” *IEEE Transactions on Robotics*, vol. 35, no. 3, pp. 539–548, 2019.
- [16] J.-H. Tzou, K. Su, J.-H. Tzou, and K. L. Su, “High-speed laser localization for a restaurant service mobile robot,” *Artificial life and robotics*, vol. 14, no. 2, pp. 252–256, 2009-11.
- [17] D. Wang, W. Li, X. Liu, N. Li, and C. Zhang, “Uav environmental perception and autonomous obstacle avoidance: A deep learning and depth camera combined solution,” *Computers and Electronics in Agriculture*, vol. 175, p. 105523, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169920303379>
- [18] J. N. Yasin, S. A. S. Mohamed, M.-H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, “Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches,” *IEEE Access*, vol. 8, pp. 105 139–105 155, 2020.
- [19] T. N. Canh, A. Elibol, N. Y. Chong, and X. HoangVan, “Object-oriented semantic mapping for reliable uavs navigation,” in *2023 12th International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2023, pp. 139–144.
- [20] M. Mueller, N. Smith, and B. Ghanem, “A benchmark and simulator for uav tracking,” in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 445–461.

## Bibliography

- [21] E. Tsykunov, L. Labazanova, A. Tleugazy, and D. Tsetserukou, “Swarmtouch: Tactile interaction of human with impedance controlled swarm of nano-quadrotors,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4204–4209.
- [22] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreichah, and M. Guizani, “Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges,” *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019.
- [23] N. Zhang, F. Nex, G. Vosselman, and N. Kerle, “End-to-end nano-drone obstacle avoidance for indoor exploration,” *Drones*, vol. 8, no. 2, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/2/33>
- [24] M. H. Rahman and S. Madria, “An augmented dataset for vision-based unmanned aerial vehicles detection and tracking,” in *2023 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, 2023, pp. 1–8.
- [25] A. Loquercio, E. Kaufmann, R. Ranftl, A. Dosovitskiy, V. Koltun, and D. Scaramuzza, “Deep drone racing: From simulation to reality with domain randomization,” *CoRR*, vol. abs/1905.09727, 2019. [Online]. Available: <http://arxiv.org/abs/1905.09727>
- [26] A. Loquercio, E. Kaufmann, R. Ranftl, M. Møller, V. Koltun, and D. Scaramuzza, “Learning high-speed flight in the wild,” *Science Robotics*, vol. 6, no. 59, p. eabg5810, 2021. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.abg5810>
- [27] H. Müller, N. Zimmerman, T. Polonelli, M. Magno, J. Behley, C. Stachniss, and L. Benini, “Fully on-board low-power localization with multizone time-of-flight sensors on nano-uavs,” in *2023 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2023, pp. 1–6.
- [28] I. Ostovar, V. Niculescu, H. Müller, T. Polonelli, M. Magno, and L. Benini, “Demo abstract: Towards reliable obstacle avoidance for nano-uavs,” in *2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2022, pp. 501–502.
- [29] T. Nguyen Canh, T. Son Nguyen, C. Hoang Quach, X. HoangVan, and M. Duong Phung, “Multisensor data fusion for reliable obstacle avoidance,” in *2022 11th International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2022, pp. 385–390.
- [30] T.-T.-N. Nguyen, T.-D. Phan, M.-T. Duong, C.-T. Nguyen, H.-P. Ly, and M.-H. Le, “Sensor fusion of camera and 2d lidar for self-driving automobile in obstacle avoidance scenarios,” in *2022 International Workshop on Intelligent Systems (IWIS)*, 2022, pp. 1–7.

## Bibliography

- [31] D. Forouher, M. Grosse Besselmann, and E. Maehle, “Sensor fusion of depth camera and ultrasound data for obstacle detection and robot navigation,” *11* 2016, pp. 1–6.
- [32] J. Borenstein and Y. Koren, “Obstacle avoidance with ultrasonic sensors,” *IEEE Journal on Robotics and Automation*, vol. 4, no. 2, pp. 213–218, 1988.
- [33] V. A. Zhmud, N. O. Kondratiev, K. A. Kuznetsov, V. G. Trubin, and L. V. Dimitrov, “Application of ultrasonic sensor for measuring distances in robotics,” *Journal of Physics: Conference Series*, vol. 1015, no. 3, p. 032189, may 2018. [Online]. Available: <https://dx.doi.org/10.1088/1742-6596/1015/3/032189>
- [34] Q. Zhou, D. Zou, and P. Liu, “Hybrid obstacle avoidance system with vision and ultrasonic sensors for multi-rotor mavs,” *Industrial Robot*, vol. 45, 02 2018.
- [35] D. G. Davies, R. C. Bolam, Y. Vagapov, and P. Excell, “Ultrasonic sensor for uav flight navigation,” in *2018 25th International Workshop on Electric Drives: Optimization in Control of Electric Drives (IWED)*, 2018, pp. 1–7.