Time of Flight Sensors on Wombats

Thomas Gruber
HTL St. Pölten
St. Pölten, Austria
thomas.gruber@htlstp.at

Abstract—This paper explores the integration of a Time-of-Flight (ToF) sensor into the Wombat robot to enhance its navigation and object detection capabilities. The project involves both an onboard ToF sensor directly integrated into the robot and an external ToF sensor connected via an ESP module for additional data processing. Various experiments were conducted to compare the performance of these setups in terms of accuracy, response time, and computational efficiency. Additionally, challenges such as ambient light interference and measurement noise were analyzed. The results indicate that ToF sensors significantly improve the robot's environmental awareness, with the onboard integration offering faster real-time processing, while the external ESP setup provides additional flexibility in data handling. These findings highlight the potential of ToF sensors in optimizing autonomous robotic systems.

Keywords— TOF, Robotics, Wombat

I. INTRODUCTION

In the field of robotics, precise environmental perception is crucial for navigation, object detection, and interaction. To enhance the capabilities of the Wombat robot, this project integrates a Time-of-Flight (ToF) sensor both directly onto the robot and externally via an ESP module. By utilizing ToF technology, the robot can achieve improved depth perception and obstacle detection, making it more efficient in autonomous tasks. This paper explores the technical aspects of ToF sensors, compares different integration approaches, and evaluates their impact on robotic performance. Through experimentation and analysis, the goal is to optimize the Wombat robot's perception system and demonstrate the benefits of ToF sensors in modern robotics applications.

II. PROCESS OF PROJECT

A. Selecting the Testing Environment

To evaluate the integration of Time-of-Flight (ToF) sensors, suitable environments were identified. The goal was to improve the perception capabilities of the Wombat robot by adding a ToF sensor both directly on the robot and externally via an ESP module. The testing setup was designed to ensure reliable data collection under different conditions.

B. Setting Up the Testing Framework

Appointments were scheduled to conduct experiments in controlled environments. The test conditions were defined to assess depth measurement accuracy, latency, and the impact of environmental factors such as lighting and surface reflectivity. Each test session was structured to allow sufficient time for setup, data collection, and analysis.

C. Preparing the Hardware and Software

Before conducting tests, the ToF sensors were checked for proper functionality. Any faulty components, such as cables, connectors, or microcontrollers, were replaced. The Wombat robot and ESP module were programmed and configured to process sensor data efficiently. Prior to testing, all necessary software tools were installed and calibrated to ensure accurate measurements.

III. WOMBAT AND TOF

A. Womat and Time of Flight sensors

The Wombat Robot is a compact and adaptable robotic system based on a Raspberry Pi, which is extended with a custom adapter board. This board enables the control of servos, motors, and sensors, allowing the robot to interact with its environment and execute programmed tasks such as autonomous navigation, object detection, and environmental interaction. To enhance its capabilities, a Time-of-Flight (ToF) sensor is integrated, providing precise distance measurement and improving obstacle detection.

Time-of-Flight sensors measure the time it takes for a light signal to travel to an object and back, allowing for accurate depth perception. In this project, a Direct Time-of-Flight (dToF) sensor is used. Unlike indirect ToF sensors, which rely on phase shifts, dToF sensors emit short laser pulses and measure the time delay of the reflected signal. This approach ensures high accuracy even over longer distances, fast response times for real-time applications, and improved performance in varying lighting conditions compared to traditional vision-based sensors.

Figure 1: Time of Flight Sensor (ToF)

IV. Structure of the First Method (Onboard ToF Sensor)

A. Hardware Setup

The first method involves the direct integration of a Time-of-Flight (ToF) sensor onto the Wombat robot to measure distances in real time. The ToF sensor was connected directly to the Raspberry Pi's I2C interface. The following steps were taken for the hardware setup:

1. Wiring:
   - The ToF sensor's SCL (SerialClock) and SDA (Serial Data) pins were connected to the corresponding I2C pins on the Raspberry Pi.
   - The VCC pin of the sensor was connected to the 5V power supply, and the GND pin was connected to the ground of the Raspberry Pi.
   - The GPIO and XSHUT pins of the ToF sensor were not utilized in this setup, as they were not required for basic functionality.
2. Placement:
   - The ToF sensor was mounted at the frontmost point of the Wombat robot to ensure optimal obstacle detection and distance measurement during navigation.
   - Care was taken to position the sensor in a way that minimizes interference from the robot's own structure or moving parts.

B. Software Implementation

1. Programming the ToF Sensor:
   - A Python script was developed to interface with the ToF sensor using the Raspberry Pi's I2C communication protocol.
   - Libraries the VL53L0X Library was utilized to read distance measurements from the sensor.

V. Structhture and Hardware for the secon Method(External TOF Sensor via ESP module)

A. Hardware Setup

1. Conection between Time of Flight Sensor and ESP:
   - The ToF sensor is connected to the ESP module via the I2C protocol (using SCL and SDA pins). This connection allows the ESP to read distance measurements from the sensor efficiently.
   - The Both the ToF sensor and the ESP module are powered by an external battery.
   - The GPIO and XSHUT pins of the ToF sensor were not utilized in this setup, as they were not required for basic functionality.
2. Placement:
   - The use of an external battery enabled flexible positioning of the ESP module and ToF sensor. This flexibility facilitated multiple test configurations, allowing for comprehensive evaluation of the system's performance under varying conditions. By testing different placements, the system's ability to adapt to diverse environments and maintain accurate measurements was validated.

B. Software Implementation

I. Programming the ESP Module

- The ESP32-S3-DevKit1 was programmed using Visual Studio Code with the PlatformIO extension. The code was written entirely in C++ .
- Library Used:

  Key libraries were integrated into the project, including:

  - The arduino framework for basic functions.
  - The PubSubClient library for implementing MQTT communication.
  - The VL53L0X library to interface with the Time-of-Flight sensor, enabling precise distance measurements.
- Communication Protocoll:

  The ESP module communicates with the RaspberryPi using the MQTT protocol, ensuring lightweight and efficient data exchange over Wi-Fi.

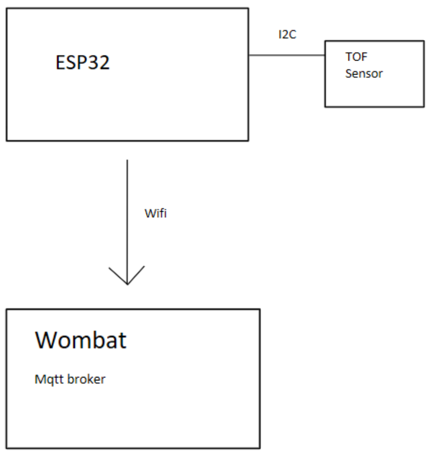

Figure 2: Blocdiagram of setup

VI. Results

To evaluate the effectiveness of the ToF sensor in improving the Wombat robot's navigation and obstacle detection, a series of experiments were conducted. The results are divided into two scenarios:

A. *ToF Sensor Mounted on the Wombat*

In this scenario, the ToF sensor was directly integrated into the Wombat robot to measure distances in real-time during navigation. The results show the expected distances versus the actually measured distances across one trial, demonstrating the sensor's accuracy and reliability when mounted on the robot.

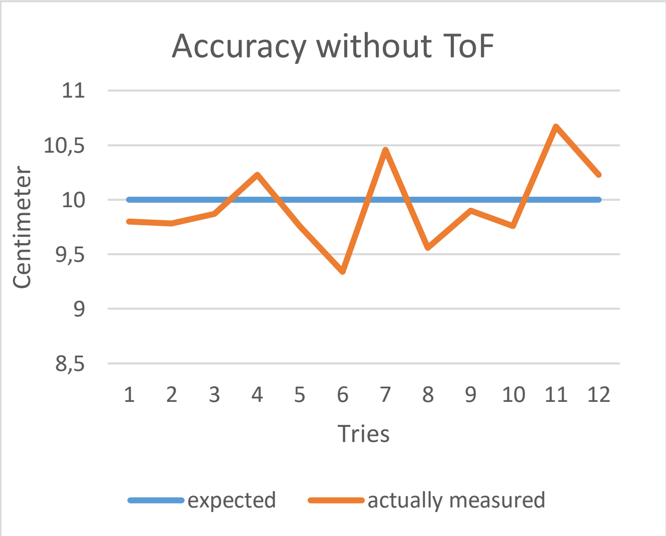

Figure 3 Comparison of measured and expected distances without the ToF sensor on the Wombat.

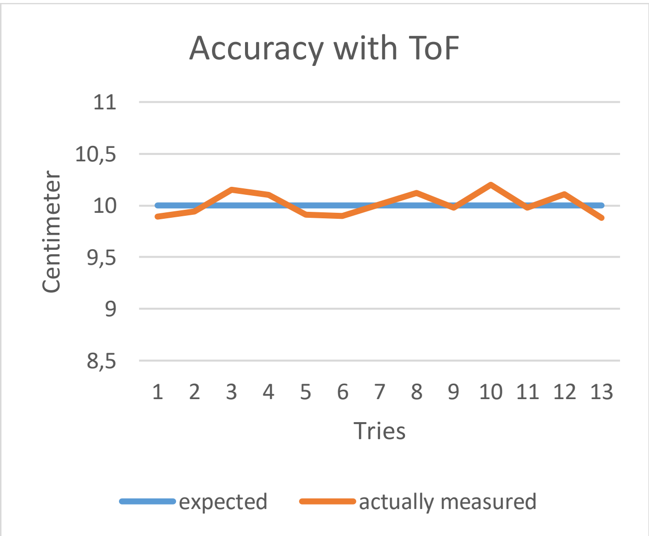

Figure 4: Comparison of measured and expected distances with the ToF sensor on the Wombat.

B. *ToF Sensor Positioned Beside the Wombat*

In this scenario, the ToF sensor was placed externally beside the Wombat robot, allowing for additional flexibility in data collection and environmental perception. The results compare the expected distances with the actually measured distances, highlighting the sensor's performance in an external configuration.

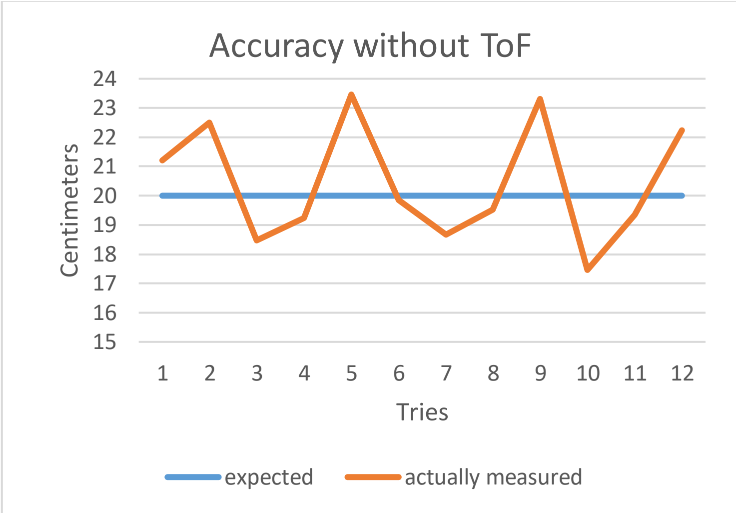

Figure 5: Comparison of measured and expected distances without the ToF sensor.

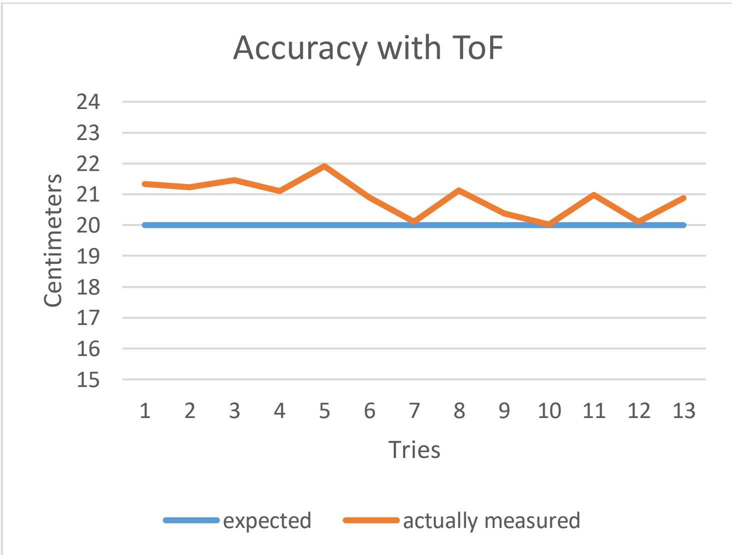

Figure 6: Comparison of measured and expected distances with the ToF sensor.

VII. Dicussion

A. *Tolance Difference*

The results indicate that the Wombat robot achieves significantly higher accuracy in its movements when using the ToF sensor. However, due to the sensor's 15° field of view (FOV), the system is not extremely precise. This limitation affects the robot's ability to detect objects outside this angular range, leading to minor inaccuracies. While the ToF sensor improves performance within its FOV, external factors such as ambient light and reflective surfaces occasionally impact its measurements, suggesting areas for further refinement.

B. Driving to far Due to Delay in external ToF Sensor Setup

In the external ToF sensor setup, the robot consistently drove too far before stopping. This occurred because the ESP module sent a stop command only when the measured distance reached exactly 20 cm. Due to the delay in data processing and transmission, the robot overshot the target position, reducing navigation accuracy. In contrast, the onboard ToF sensor allowed for real-time adjustments, enabling more precise stopping. Future improvements could include dynamic thresholds, predictive algorithms, or faster communication protocols.

VIII. CONCLUSION

This project successfully demonstrated the integration of a Time-of-Flight (ToF) sensor into the Wombat robot, significantly improving its navigation and object detection capabilities. Two configurations were tested: an onboard ToF sensor directly connected to the robot and an external ToF sensor connected via an ESP module. The results showed that the onboard setup provided real-time data processing and faster response times, enabling more precise navigation. In contrast, the external setup offered greater flexibility in sensor placement but introduced delays due to wireless communication, causing the robot to overshoot its target position.

A. Key finding inlcude:

- The ToF sensor enhanced the robot's accuracy in distance measurement and obstacle detection, particularly in the onboard configuration.
- The external setup, while versatile, faced challenges such as communication delays and a fixed stopping threshold, which reduced its precision.
- The 15° field of view (FOV) of the ToF sensor limited its detection range, highlighting the need for additional sensors or a wider FOV in future implementations.

B. Future Work:

To further improve the system, the following steps coud be considered:

- Optimizing Communication: Reducing latency in the external setup by using faster protocols or predictive algorithms.
- Expanding Sensor Coverage: Integrating additional sensors or using a wider FOV sensor to improve detection range.
- Dynamic Thresholds: Implementing adaptive stopping distances to account for delays and improve precision.
- Hybrid Approach: Combining the strengths of both onboard and external setups for more robust performance.

C. Comparison with Gyro-Based Navigation

For reference, a test group was created in which the Wombat robot navigated using a gyro-based code without ToF sensor input. The gyro-based approach relied solely on rotational data for distance estimation, which introduced drift errors over time. Compared to the ToF-enhanced trials, the gyro-only tests exhibited:

- Higher cumulative error in position tracking, especially over longer distances.
- No obstacle detection capability, requiring manual intervention in cluttered environments.
- Less consistent stopping accuracy, as the gyro could not account for external obstacles or terrain variations.

  This underscores the ToF sensor’s advantage in providing absolute distance feedback, compensating for the limitations of inertial navigation.