Università degli Studi della Campania "Luigi Vanvitelli"

DRIM

Ph.D. Program of National Interest in Robotics and Intelligent Machines

Administrative Headquarters: Università di Genova

Design and Integration of Multi-Modal Sensor Systems for Enhanced Robotic Manipulation: Applications in Grasping & Control Across Adaptive Grippers and Robotic Platforms

by

Tanzeel Ahmad Fazal

Thesis submitted for the degree of *Doctor of Philosophy* (38° cycle)
October 2025

Salvatore Pirozzi	Supervisor
Antonio Sgorbissa	Head of the PhD program

Thesis Jury:

Ciro Natale, *University of Campania, Luigi Vanvitelli*	Internal examiner
Cristiano Maria Verrelli, *University of Rome Tor Vergata*	External examiner
Filippo Arrichiello, *University of Cassino and Southern Lazio*	External examiner

Borsa di dottorato cofinanziata con risorse dell'Unione europea-NextGeneration EU
Piano Nazionale di Ripresa e Resilienza Missione 4, componente 1 "Potenziamento
dell'offerta dei servizi di istruzione: dagli asili nido all'Università

To my **honorable parents**, I dedicate this research with profound love, gratitude, and respect. Your endless prayers, unwavering support, and lifelong sacrifices have been the true foundation of my life and this greatest achievement. Your strength, resilience, and faith in me have been a constant source of inspiration, guiding me through every challenge with courage and grace. I am forever indebted to you for being my greatest pillars of strength.

To my **beloved wife**, whose patience, encouragement, and unconditional support have sustained me throughout this demanding journey. Your love and belief in me have been a beacon of strength in times of hardship, and I owe much of this accomplishment to your unwavering presence by my side.

To my **precious daughter**, whose arrival brought immense joy, hope, and inspiration during the final stages of this journey. You are a reminder of new beginnings, and this achievement is as much for your future as it is for my present.

I also dedicate this research as a reflection of fortitude, steadfastness, and unwavering commitment. This journey has tested the depths of endurance and self-belief, yet through every challenge and moment of doubt, I remained resolute in pursuing my goal. May this accomplishment stand as a reminder that with faith, determination, and inner strength, even the most difficult paths can be transformed into milestones of success.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Tanzeel Ahmad Fazal
December 2025

Acknowledgements

All praise and boundless gratitude are due to Allah Almighty, the Most Merciful and the Source of infinite wisdom and guidance, who blessed me with knowledge, resilience, and perseverance to accomplish this research journey. Without His countless favors and blessings, none of this would have been possible.

I wish to express my deepest and most respectful gratitude to my esteemed supervisor, **Prof. Salvatore Pirozzi**, for his invaluable mentorship, inspiring guidance, and ever-encouraging attitude throughout the course of my doctoral research. His trust in my abilities and his exceptional mentorship have been a constant source of motivation. His consistent support, insightful feedback, and ever-gracious attitude have been a beacon of motivation at every stage. This dissertation is a testament to his exceptional supervision. Working under his guidance has been both a privilege and an immense honor, and the wisdom I have gained from him will remain a lifelong asset.

I am also sincerely grateful to my dear friends and colleagues, whose encouragement, companionship, and prayers have been a constant source of strength. I am especially thankful to **Dr. Gianluca Laudante, Dr. Rizwan Zahoor, Dr. Muhammad Ishaq** and **Dr. Zulfiqar** whose steadfast support, kind words, and moments of joy have made this endeavor both bearable and memorable.

Finally, I would like to express sincere appreciation to all my fellows for their generous cooperation, cheerful companionship, and readiness to extend help whenever needed. Their presence has not only enriched my academic experience but has also made this long journey rewarding and unforgettable.

Abstract

This thesis presents the design, development, and application of three new variants of multimodal sensor systems, each created to address key challenges in robotic manipulation, perception, and human-robot interaction. Modern robots are increasingly expected to operate in unstructured, sensitive, and shared environments. To perform effectively in such contexts, they require compact, high-resolution, and versatile sensing platforms that can combine tactile, proximity, and orientation or posture-based data. This research responds to that need through a step-by-step development process, where each sensor builds on the previous design to progressively expand capability and application.

The first prototype integrates eight optoelectronic components, a time-of-flight (ToF) sensor for depth perception, and a 6-axis Inertial Measurement Unit (IMU). It provided the foundation for multimodal sensing, enabling the collection of proximity, motion, and acceleration data. This early platform served as a testbed for calibration, signal integration, and basic interaction tasks, establishing the principles for sensor fusion in robotic systems.

The second variant extended these capabilities through a 4×3 matrix array of optoelectronic elements having enhanced spatial resolution combined with a 6-axis IMU and a modular interface. Its adaptable design allows mounting on a wide range of robotic grippers, including both commercial devices and custom in-lab prototypes. This version was validated in three distinct domains:

1. Delicate object handling, such as fresh food, where sensitive force control was required.
2. Human-Robot Interaction (HRI), where the system enabled safer, more intuitive collaboration between humans and robots.
3. In satellite robotics, Only IMU for feedback to enhance attitude control.
   These applications confirmed the sensor's robustness, real-time performance, and flexibility.

The third and most advanced sensor is a fingertip-integrated module developed for the SEEDS anthropomorphic robotic gripper, very similar to the human fingertip. It incorporates

PWM-controlled optoelectronic elements, a 9-axis Bosch IMU with onboard processing, and a high-resolution ToF sensor, all embedded directly within the fingertip. This miniaturized design delivers high-frequency, low-latency data, supporting precise force estimation, object shape recognition, and compliant grasping. The result is a sensing solution that enables dexterous, human-like manipulation.

Alongside these multi-modal sensor developments, this thesis also addresses vision-based perception challenges through the integration of a novel stereo vision approach for detecting thin wire features. Standard 3D cameras are often bulky and costly, making them unsuitable for fine robotic manipulation. To overcome this, a stereo vision system using two low-cost, off-the-shelf endoscopic cameras was developed, allowing integration directly into the robot's end-effector. By applying state-of-the-art vision algorithms, this system achieved a diameter estimation error below 10% and a location error below 3% for wires of only a few millimeters or less. This contribution demonstrates how compact, affordable vision systems can support fine-grained robotic perception tasks, expanding the scope of multimodal sensing in robotics.

The contributions of this thesis can be summarized as:

- A progressive methodology for the scalable and modular design of robotic sensor systems.
- The integration of optoelectronic, inertial, and depth sensing technologies into compact, customizable formats.
- Development of low-cost stereo vision system for robotic manipulation.
- Experimental validation of the sensors in diverse application domains including space robotics, food handling, and HRI.
- Establishing a foundation for future work in soft robotics, teleoperation, and assistive technologies.

Overall, this thesis advances the field of tactile and perceptual sensing in robotics by introducing adaptable, high-performance, and application-specific sensory systems. In addition to tactile and proprioceptive sensing, it establishes a pathway toward vision-augmented multimodal perception, enabling richer environmental awareness and more intelligent robot behavior. These contributions open new opportunities for sensor-driven control strategies, machine learning-based perception, and enhanced robot cognition, while also fostering safer human-robot collaboration and paving the way for next-generation intelligent machine applications.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

ADC	Analog to Digital Converter
CAD	Computer Aided Drawing
CoG	Center of Gravity
DAC	Digital to Analog Converter
DLO	Deformable Linear Object
DOF	Degrees of Freedom
DPS	Degrees per Second
ESO	Extended State Observer
HMI	Human Machine Interface
HRI	Human Robot Interaction
I/O’s	Inputs/Outputs
IMU	Inertial Measurement Unit
LED	Light Emitting Diode
LiPo	Lithium Polymer
MCU	Micro Controller Unit
PD	Proportional Derivative
PID	Proportional Integral Derivation

PI	Proportional Integral
PWM	Pulse Width Modulation
RMS	Root Mean Square
ROS	Robot Operating System
RPM	Revolutions Per Minute
ToF	Time of Flight

Part I

Background & Context

Chapter 1

Introduction

Robots are increasingly being deployed in environments that are dynamic, unstructured, and shared with humans. From industrial automation and healthcare to space robotics and food handling, the ability of a robot to sense and adapt to its surroundings is critical for safe and effective operation. Unlike traditional machines that rely on rigid programming and limited sensory inputs, intelligent robotic systems require rich, multimodal perception that integrates tactile, proprioceptive, and visual information.

Particularly tasks involving delicate manipulation, such as handling fragile objects, operating in microgravity, or interacting directly with humans demand high-resolution, compact, and adaptive sensors. Traditional robotic systems relied heavily on vision sensors to interpret the environment. However, conventional vision or tactile alone faces challenges such as precision, poor lighting, or difficulties in reconstructing fine features. Also, vision systems capable of small-scale 3D reconstruction are typically bulky and expensive. These gaps motivate the development of novel sensing architectures that combine multiple modalities into compact, scalable, and versatile platforms, allowing robots to measure forces, recognize textures, detect slip, and adjust their actions in real time.

A critical example is the manipulation of delicate or deformable objects, such as food items, wires, or fabrics, where excessive force can damage the object or the robot may fail to achieve stable grasping. Similarly, in space robotics, precision in microgravity requires compact and reliable sensing architectures, while in human-robot interaction (HRI), safety demands context-aware, adaptive feedback. These application scenarios highlight the limitations of existing sensors and motivate the development of compact, high-resolution, modular, and versatile sensing platforms.

1.1 Motivation

As robots move from controlled factory floors into messy, unpredictable, and human-centered environments, the way they perceive and interact with the world becomes incredibly important. Whether a robot is sorting delicate fruit, assisting a human in a shared workspace, or performing precise manipulation in space or industry, it must understand far more than just where an object is, it must sense how it feels, how it moves, and how it changes during interaction. These are abilities that humans take for granted, but remain extremely challenging for robots to achieve.

Traditional robotic sensing technologies tend to be specialized but narrow in their capabilities. Vision systems struggle when objects are occluded or too small for conventional cameras. Force sensors provide overall magnitude, but cannot describe where on a fingertip contact occurs. Proximity sensors alone cannot reveal the dynamics of motion or orientation. And most importantly, these sensors are often too large, too expensive, or too difficult to integrate directly into compact, dexterous robotic hands.

For robots to handle fragile items gently, collaborate safely with people, or manipulate tiny or deformable objects, they need sensing capabilities that more closely resemble those of the human fingertip-rich, multi-modal, and seamlessly embedded into the body of the manipulator. This realization drives the core motivation of this research: to create sensing systems that allow robots to feel, understand, and adapt in ways that are more natural and intuitive.

At the same time, many emerging applications require robots to perceive very fine details that vision alone cannot easily capture. For example, identifying and handling thin wires or micro-scale objects demands a level of visual precision that standard depth cameras simply cannot provide. This gap inspires the development of compact, low-cost stereo vision systems that bring high-resolution perception directly to the robot's end-effector.

Altogether, this thesis is motivated by a simple but powerful idea, to give robots sensory abilities that bring them closer to how humans naturally perceive and interact with the world. By designing progressively more capable multi-modal sensors—and extending these capabilities with embedded stereo vision—this work aims to build robotic systems that are more dexterous, safer around humans, and better suited for real-world tasks in industries ranging from food handling to space robotics.

1.2 Research Problem

Despite significant progress, several key challenges limit the effectiveness of current robotic sensing systems:

- Bulky or application-specific sensors: Many existing solutions are difficult to integrate across diverse robotic platforms.

- Latency and resolution issues: Real-time high-frequency feedback remains difficult to achieve in compact form factors.

- Fragmented multimodal integration: Vision, tactile, and inertial systems are often developed separately, leading to poor data fusion.

- Fine-scale perception gaps: Vision-based 3D reconstruction of thin, deformable objects (e.g., wires) is particularly difficult with affordable, lightweight systems.

These challenges restrict robots from achieving human-like dexterity, safe HRI, and precision in specialized domains such as satellite assembly or cable manipulation.

1.3 Problem Formulation

The primary objective of this research is to design, develop, and integrate multi-modal sensing systems and endoscopic stereo vision technologies to address the challenges of manipulating delicate objects, with a primary focus on fresh food handling and human-robot interaction. From a scientific perspective, the goal is to create a robust framework for accurate perception and reliable interaction with objects that are fragile, variable in shape, and sensitive to force. From a technological standpoint, the aim is to develop sensor-equipped systems that can be seamlessly integrated into existing robotic platforms and industrial infrastructures, thereby enhancing automation in domains where precision and care are critical. The specific objectives of the research include:

1. Development of Multi-Modal Sensors: To design and fabricate a compact sensor suite that integrates tactile sensing, inertial measurement units (IMUs), and proximity detection. This system will be capable of capturing multimodal data to support robust perception and manipulation in dynamic and complex environments, such as robotic grippers used for handling fresh produce.

2. Fresh Food Handling Applications: To apply the developed sensor systems in the automation of food handling, sorting, and processing tasks. The goal is to enable robots to mimic the sensitivity and adaptability of human handling by leveraging multimodal feedback particularly texture, mass, and proximity data to minimize damage and maintain quality in delicate items such as fruits and vegetables.
3. Satellite Attitude Control: To extend the applicability of the second sensor variant beyond terrestrial applications by exploring its role in satellite attitude determination and control using a reaction wheel configuration. This objective highlights the cross-disciplinary potential of the sensor system and its adaptability to space robotics.
4. Human-Robot Interaction (HRI): To investigate the role of multimodal sensing in enabling safe, intuitive, and context-aware collaboration between humans and robots, particularly in environments where robots must adapt to unpredictable human behavior and operate with a high degree of safety and reliability.
5. Endoscopic Vision Integration: To incorporate endoscopic stereo vision for Robotic 3D Detection of Thin Wire Features, to provide precise 3D perception of texture and geometry. This capability can be adapted to extended applications related to tiny objects, where accurate surface and shape detection is vital for delicate manipulation and classification.
6. Design of a Compact Fingertip-Integrated Multi-Modal Sensor: To develop the most advanced and compact version of the multimodal sensor, designed as a replica of the human fingertip for integration into the SEEDs anthropomorphic robotic gripper. This version incorporates PWM-driven optoelectronic elements for efficient optical sensing, a high-degree-of-freedom IMU for faster and more accurate inertial feedback, and an enhanced time-of-flight (ToF) sensor with greater precision and range. Importantly, the system is designed with wireless communication capability, enabling seamless integration into robotic hands without bulky cabling. Its compact form factor—similar in size to a human fingertip—allows for naturalistic, human-like manipulation in anthropomorphic robotic applications.
7. Technological Innovation: Achieve technological innovation by combining these sensors to address challenges in precision automation, robotic manipulation, and control across different sectors.

The outcomes will contribute to advancing automated food processing technologies, minimally invasive surgical robotics, and satellite control systems, demonstrating the versatility of these multi-modal sensors. Moreover, for clear understanding, a summary of different variants is presented below.

Table 1.1 Comparative summary of sensor variants and their key specifications

Sensor Variants	Sensor Technology	Relevant Applications
First Variant	Optoelectronic + ToF + 6-axis IMU	Basic interaction and object detection
Second Variant	4×3 Optoelectronic Array + 6-axis IMU	Food handling, HRI & satellite robot control
Third Variant (SEEDs Gripper)	PWM-controlled Optoelectronics + 9-axis IMU + ToF	High-resolution fingertip, HRI, Compliant grasping of delicate objects

1.4 Research Innovation

This thesis introduces several novel contributions to the field of tactile and multi-modal robotic sensing, particularly in terms of sensor miniaturization, modularity, and application-specific adaptability. The key innovations of our research are summarized as follows:

- **Modular and Scalable Sensor Architecture:** Three distinct sensor variants were developed, each tailored to a different level of robotic integration and task complexity. These designs introduce a scalable hardware architecture that accommodates a range of sensing needs, from basic proximity detection to high-precision manipulation in constrained environments.
- **High-Resolution Optoelectronic Arrays:** The second sensor variant features a 4×3 array of optoelectronic elements, enabling fine-grained tactile feedback and surface interaction mapping. This compact yet powerful layout enhances the resolution of tactile sensing, supporting more accurate grasping strategies.
- **Plug-and-Play Integration for Robotic Grippers:** A flexible adapter system was developed to allow the second variant to be mounted on a wide range of commercial and custom-built robotic grippers. This significantly reduces the integration overhead and promotes widespread applicability across robotic platforms.

- **Advanced Fingertip-Integrated Sensor Module:** The third variant marks a breakthrough in compact sensor design, embedding a PWM-based optical system, a 9-axis Bosch IMU with onboard computation, and a state-of-the-art time-of-flight sensor directly into the fingertip of an anthropomorphic gripper. This innovation offers human-like sensing capabilities in a form factor that closely replicates the dimensions and functionality of a biological fingertip.
- **Endoscopic Vision for thin wires** The application of endoscopic stereo vision technology to small objects is a novel concept. The high precision offered by endoscopic stereo vision, traditionally used in the detection of thin wires in medical robotics, can now be applied to detecting small features in tiny objects, offering more accurate and delicate handling solutions.
- **Cross-Domain Applications and Validation:** All three sensor variants were experimentally validated across a diverse set of use cases, including fresh food and delicate object manipulation, physical interaction in human-robot collaboration, and attitude control in satellite robots. The ability of a single sensor platform to generalize across such varied applications demonstrates its robustness and versatility.

Collectively, these innovations contribute to a new generation of robotic sensors that are not only compact and multi-functional but also highly adaptable to the evolving requirements of intelligent manipulation and perception in robotics.

This thesis is organized into three main sections and nine chapters, covering the complete research pathway from the conceptual foundation to the design, implementation, and application of multi-modal sensing systems for robotic perception, manipulation, and interaction. Each section contributes to building a coherent framework that advances the integration of tactile, inertial, and proximity sensing technologies in intelligent robotic platforms.

Section I establishes the research background and theoretical framework. Chapter 1 introduces the overall motivation, objectives, and scientific relevance of the work, emphasizing the growing need for integrated sensing in robotics. Chapter 2 provides a detailed literature review, outlining the state of the art in tactile, proximity, and inertial sensing technologies, and analyzing key advancements in multi-modal sensor fusion and human-robot interaction (HRI). Together, these chapters define the problem space and highlight the research gaps addressed in the subsequent sections.

Section II presents the design, development, and application of the proposed multi-modal sensing systems across three generations of design. Chapter 3 focuses on the first sensor variant, describing its architecture, optoelectronic design, signal processing methodology, and preliminary validation through basic manipulation experiments. Chapter 4 introduces the second sensor variant and its integration into two robotic grippers, the commercial Robotiq 3-Finger Adaptive Gripper and a custom four-finger gripper developed in collaboration with the University of Genoa (UniGe). This chapter also discusses the implementation of a Proportional-Integral (PI) control strategy, an HRI-based release mechanism, and application-oriented experiments in agri-food manipulation, demonstrating the effectiveness of sensor in handling delicate and variable objects. Chapter 5 presents the third-generation compact multi-modal fingertip sensor, developed with PWM-driven optoelectronic elements, an embedded 9-axis IMU, and a Time-of-Flight (ToF) proximity module. The chapter details its mechanical and electronic integration with the SEEDs anthropomorphic robotic hand, along with its data acquisition pipeline for synchronized tactile, proximity, and inertial feedback.

Section III expands the scope of the research to vision and inertial sensing applications beyond tactile-based manipulation. Chapter 6 introduces an endoscopic stereo vision system designed for 3D reconstruction and diameter estimation of thin deformable linear objects (DLOs), highlighting its relevance in precision robotic perception. Chapter 7 explores the application of the IMU subsystem for attitude estimation and control in satellite robotics, demonstrating how the developed sensing architecture can be adapted for space-oriented systems requiring high stability and precision.

The thesis concludes with Chapter 8, which summarizes the key scientific and technological contributions, and outlines future research directions. These include the development of model-based slippage control algorithms, improved multi-finger coordination, and learning-based perception for advanced collaborative and assistive robotics.

Chapter 2

Literature Review

2.1 Multi-Modal Sensing

Multi-modal sensor systems currently available focus primarily on individual sensing aspects, such as force/torque or vision, with limited integration between tactile, inertial, and proximity sensing for cohesive control in robotic systems. In fields such as food handling, where delicacy and precision are paramount, existing solutions primarily rely on visual or force feedback, but the combination of multiple sensing modalities is still underdeveloped.

Robotic systems for food handling are often based on either purely visual feedback or force feedback, and their ability to process delicate items without causing damage remains limited. Current solutions struggle with deformable, variable, and fragile food items, making manual handling or low-level automation common.

Manipulation and grasping require detailed information of the object and it can be achieved by using vision system, but due to uncertainties of environment including dust, obstruction, and lighting conditions, the alone vision system may cause failure in obtaining accurate information Cirillo et al. (2021a). Also, in minimally invasive surgeries, endoscopic systems have been designed to navigate and operate in constrained environments. However, these systems are limited in their ability to sense detailed surface geometry or texture, which are essential for tasks like food handling that require fine-grained manipulation. These challenges can be addressed using optoelectronic sensors along vision system to get robust information of the targeted objects / tasks.

Tactile sensing embedded in the gripper's fingertips enables the acquisition of high-resolution contact information. This tactile feedback facilitates robust grasping, and object recognition, contributing to improved manipulation performance in challenging scenarios. The integration of IMUs further enhances the sensor system's versatility by capturing the gripper's motion, slip detection, and orientation data.

Literature reflects many solutions based on different transduction technologies, (e.g., capacitive, resistive, and magnetic). In the past, a huge number of relevant papers have been reported about tactile sensing. Suen and Chen (2018) introduced concept of capacitive tactile sensing of three axial forces through five concentric shape independent sensing electrodes. Makihata et al. (2018) designed a more precious capacitive distributed sensor that uses custom designed ASIC matrix inside the silicon wafer structure. Wang et al. (2019) developed a tactile sensor with 3x3 array of resistive sensing elements enclosed by conductive rubber bubbles. During manipulation task, those bubbles in contact with object surface enable the detection of object textures using slipping or sliding operations. Liu et al. (2020) and Klimaszewski et al. (2019) proposed a solution based on distributed tactile sensors using the matrix of piezoelectric or resistive sensing elements covered by flexible material for the detection of contact pressure.

Moreover, multiple commercial solutions have also been presented, e.g., the Xela Robotics (2023) utilizes the technology of magnetic field changes induced by the skin deformation, Contactile Khamis et al. (2021) utilizes Optical transduction, Syntouch Loeb et al. (2012) uses changes in impedance as the fluid deforms are detected by an array of electrodes and GelSight (2022) patented elastomeric sensor technology conforms to the topography of any surface.

Figure 2.1 State of the art commercial solutions of tactile sensing

Further cameras can be employed by robots to sense objects from relatively high distances, tactile sensing can be used only in strict contact with an object. However, tactile sensors are typically installed on the robot end-effector, e.g., robot grippers, robotic hands, and could potentially sense areas closer to objects that could be not accessible to cameras.

By exploiting the strategic position of tactile sensors, the integration of proximity sensors on them could enable robots to rely on pre-touch sensing, easily allowing the estimation of the object pose and shape when the robot is in the approaching and grasping phase. Differently from vision-based and tactile sensors, pre-touch sensors operate at an intermediate range, providing benefits of both the mentioned class of sensors: mounted to the robot end-effector, they are more robust against occlusion than cameras; mounted at a closer range, they may potentially provide more precise measurements; like camera/depth sensors, they do not require to get in contact with objects.

The literature also highlights the limitations of vision-only pipelines for deformable linear object (DLO) manipulation. Thin electrical wires, for instance, are difficult to localize precisely with depth cameras, even after calibration, because of their small cross-section, transparency, and background contrast issues. Tactile sensing has been proposed to recover local features once contact occurs, enabling recognition of properties such as diameter and curvature. However, tactile sensing alone cannot support pre-grasp planning since it requires physical interaction, which risks failure if initial pose estimates are inaccurate.

This gap has motivated increasing research into proximity and pre-touch sensing, which operates at intermediate distances between vision and touch. Early approaches exploited electric-field and acoustic techniques, but their applicability was limited to specific materials or conditions. More recently, optical proximity sensors, and in particular compact Time-of-Flight (ToF) modules, have been shown to provide robust, calibration-free, and material-independent depth measurements. By integrating ToF sensors with tactile arrays in the same robotic finger, researchers have demonstrated hybrid systems capable of first estimating wire pose during approach and then validating grasp success during contact, even under challenging scenarios such as wires lying flat on the surface.

Together, these studies underscore a clear trend: multimodal sensor integration is essential for dexterous robotic manipulation. Tactile sensing provides high-resolution contact data for fine force control and object feature recognition; proximity sensing bridges the pre-contact gap with robust short-range depth cues; and vision systems extend perception to larger-scale scene understanding. The literature establishes that combining these modalities—particularly tactile and ToF-based proximity—enables reliable grasping of small, fragile, or deformable objects, which are otherwise intractable using vision alone. This provides a strong foundation for the present research on multi-modal sensors incorporating optoelectronic tactile elements, ToF modules, and inertial measurement units (IMUs), aimed at delivering compact, versatile, and application-ready solutions for domains such as food handling, satellite robotics, and human–robot collaboration.

Modern manipulation in unstructured or human-shared environments relies on combining local contact information (tactile/proprioceptive) with near-field perception (pre-touch/proximity) and far-field vision to overcome occlusions, lighting variability, and fine-feature ambiguity. In tasks involving deformable linear objects (DLOs) i.e., wires, cables, and filaments, fine 3D pose/shape estimation is especially challenging for stand-alone vision; multimodal sensing therefore plays a central role in achieving reliable grasping, in-hand regulation, and safe HRI. The reviewed works collectively motivate a design space where compact, high-resolution tactile arrays are paired with embedded ToF proximity modules to deliver robust pose and shape cues during approach and during contact, enabling sensor-driven control and closed-loop refinement.

Cirillo et al. (2021a) present an optoelectronic tactile finger specifically engineered for parallel grippers, describing a reproducible hardware or firmware stack and a 5×5 taxel array ($21 \times 21\,\mathrm{mm}^2$, 3.55 mm spatial resolution) with improvements in current-driven LED stability, buffered readout, on-board 12-bit ADC, and ROS integration. The design emphasizes low noise, repeatability, response time, and mechanical robustness (rigid grid + silicone pad) to furnish a high-quality tactile map suitable for object shape inference and force/torque estimation via learning-based calibration. This research positions optoelectronic tactile skins as a practical, application-oriented foundation for manipulation in cluttered settings and highlights why flat vs domed pads matter for grasping small/thin objects.

Building on that hardware, Cirillo et al. (2021c) focus on tactile signal processing and learning-based estimation of wire diameter and in-contact shape. They introduce normalization with linearization (per-taxel LUTs) to mitigate taxel variability, plus selection of informative taxels to reduce bias during curve fitting for wire shape reconstruction. A supervised pipeline classifies wire diameters with very high recognition rates (99%+), and a pick-and-place demo shows how tactile-only classification can drive autonomous sorting by diameter. The developed research demonstrates that, once carefully calibrated and normalized, optoelectronic tactile arrays can deliver fine-scale geometric features even for thin, deformable objects precisely the regime where vision is brittle.

To bridge the gap before contact, Cirillo et al. (2021b) integrate Time-of-Flight (ToF) modules (e.g., VL6180X) around the tactile finger, proposing a compact interface/processing board and multi-axis module placement for pre-touch scanning. This hybrid finger allows pose/shape estimation during approach, offering robustness to occlusion and improved near-field depth compared with cameras alone. The paper analyzes measurement behavior at close range, reports SNR/repeatability/hysteresis, and devises object scanning strategies tailored to wires, highlighting that ToF pre-touch can recover geometric information where low-cost depth cameras underperform (e.g., <4 mm wires and material-dependent reflectance).

Extending the above, Cirillo et al. (2022) present a combined proximity–tactile grasping strategy for thin electrical wires in two scenarios: (i) wire on a plane and (ii) wire lifted off the plane using fingernail-like adapters designed into the pre-touch housing. The study

characterizes close-range ToF behavior (including non-linear “jumps” below 15 mm that can be exploited as a sensitive wire detector) and proposes optimized scanning routines to reduce time-to-detect. After approach and alignment via ToF, tactile feedback verifies and refines desired pose at contact, supporting accurate, repeatable wire grasping in conditions that defeat standard calibrated 2D/3D vision pipelines.
Taken together, these works articulate a cohesive multimodal stack:

- Tactile (optoelectronic) for high-resolution contact maps, slip/force cues, and in-contact feature estimation (diameter/shape).
- ToF pre-touch for near-field 3D cues and pose/shape estimation during approach, robust to occlusion and capable of detecting sub-centimeter features beyond typical commodity depth cameras.

For a multi-modal sensor with tactile, ToF and IMU, this literature supports:

1. Optical taxel arrays as the tactile backbone.
2. Embedded ToF to close the perception gap between vision and touch.
3. Inertial sensing (IMU) to stabilize estimation and support sensor-driven control (e.g., attitude/pose compensation during approach and in-hand manipulation). The integration pathway suggested by these papers aligns with your thesis goals: compact, modular, and application-ready sensing that generalizes from industrial DLO manipulation to fresh-food handling, HRI, and space robotics.

Hence, this research endeavors the advancement of robotic grasping by introducing a sensorized fingertip including different technologies i.e., tactile sensing, inertial measurement unit and time of flight proximity sensing to enable the robots for complex manipulations.

2.2 Applications of Multi-Modal Sensor

2.2.1 Human Robot Interaction

Human-Robot Interaction (HRI) aims to enable robots to collaborate safely, intuitively, and efficiently with humans in shared environments. For effective HRI, robots must be equipped with perceptual capabilities that allow them to sense human intent, adapt their behavior in real time, and ensure safety during physical interactions. Multi-modal sensors integrating tactile, proprioceptive, inertial, and proximity information plays a pivotal role in enhancing these perceptual capabilities.

Tactile sensing is foundational to physical HRI. Dahiya et al. (2010) emphasized the importance of high-resolution tactile feedback in humanoid robots, highlighting how distributed pressure sensors across the body or fingers can provide robots with human-like touch perception. This enables the detection of contact forces, surface textures, and physical gestures, which are crucial for responsive and safe interaction. Building on this, Kaboli et al. (2020) demonstrated that multi-modal tactile feedback, when combined with learning algorithms, significantly improves a robot's ability to recognize objects and respond to unstructured interactions with humans.

In addition to tactile feedback, inertial sensors and proximity sensors are increasingly used in HRI scenarios. For example, few authors introduced a fingertip-integrated multi-modal sensor system that fuses 6-axis inertial data with tactile feedback for precise gesture tracking and interaction. The sensor's small form factor allowed it to be worn on human fingertips or integrated into robotic grippers, demonstrating improved recognition of motion patterns and touch-based commands. Such configurations are particularly useful in applications where human gestures or contact forces are used to guide or teach robots through physical interaction, a technique commonly referred to as physical human-robot interaction (pHRI).

Biomimetic design further enhances the intuitiveness of HRI. Wettels et al. (2008) developed a fingertip sensor mimicking the mechanical and sensory properties of human skin, enabling robots to replicate nuanced human touch behavior. Similarly, the GelSight system by Yuan et al. (2017) provides high-resolution surface geometry and force distribution maps, supporting HRI tasks that require subtle touch perception, such as handshakes or guided manipulation.

Numerous studies have highlighted the role of multi-modal sensors in facilitating responsive and adaptive robotic behavior. Dahiya et al. (2010) emphasized the importance of tactile sensing in robotic skin for safe human-robot collaboration, proposing that high spatial and temporal resolution tactile data are essential for replicating human-like perception. Also, Kim et al. (2018) introduced a bioinspired soft sensor combining piezoresistive and capacitive layers, demonstrating its effectiveness in human touch recognition and grip force modulation, both essential for handling delicate objects.

2.2.2 Delicate Object Handling

In the context of delicate object manipulation, such as handling soft fruits, surgical tools, or laboratory specimens, force feedback and contact detection are critical. Traditional single-modal sensors often fall short due to limited resolution or lack of redundancy. Multi-modal sensors, by contrast, provide cross-validated data streams that improve reliability and responsiveness. For example, the GelSight sensor developed by Yuan et al. (2017) combines optical and tactile feedback to reconstruct surface geometries with micrometer accuracy, enabling precise control during soft object manipulation. Similarly, Zhou et al. (2024) developed a flexible tactile sensor array with embedded deep learning algorithms for predicting object deformation and slippage during grasping tasks.

Despite these advances, integration and miniaturization remain ongoing challenges. Many HRI systems use distributed sensing on large surfaces (e.g., full-body robotic skins), which are not always suitable for precision tasks like object manipulation or fingertip-level contact. Moreover, most systems are platform-specific and lack modularity. The present research addresses these gaps by designing compact, fingertip-integrated multi-modal sensors that combine PWM-controlled optoelectronic arrays, IMUs, and ToF sensors into a form factor closely resembling the human fingertip. These sensors enable robots to perceive contact pressure, orientation, and proximity in real time, facilitating safe, adaptive, and context-aware interaction with humans.

In summary, multi-modal sensing systems have significantly advanced the field of HRI by enabling fine-grained perception, gesture recognition, and adaptive control during physical interaction. By further miniaturizing these systems and integrating them at the fingertip level, where most human-robot contact occurs, this research contributes to making HRI more natural, safer, and functionally versatile across applications such as collaborative assembly, assistive robotics, and social robotics.

Current research builds on these foundational works by addressing limitations in size, integration flexibility, and sensory fusion. It presents three novel multi-modal sensor variants optimized for both delicate object handling and human-robot interaction, including a fingertip-integrated solution with high-performance PWM-based optoelectronics, a $9-$axis IMU, and advanced ToF sensing. This body of work aims to bridge the performance gap between biological and robotic touch, allowing safe, precise, and intuitive interaction in diverse robotic applications.

2.2.3 Attitude Control of Satellite

Effective control of attitude or orientation in robotic and satellite systems is essential for achieving stable and precise behavior, especially under external disturbances or environmental uncertainties. In robotic systems, particularly maintaining orientation, and compensate for unexpected interactions. A widely adopted approach for such systems is the Proportional-Integral (PI) or Proportional-Derivative (PD) control architecture, which provides a balance between responsiveness and long-term stability.

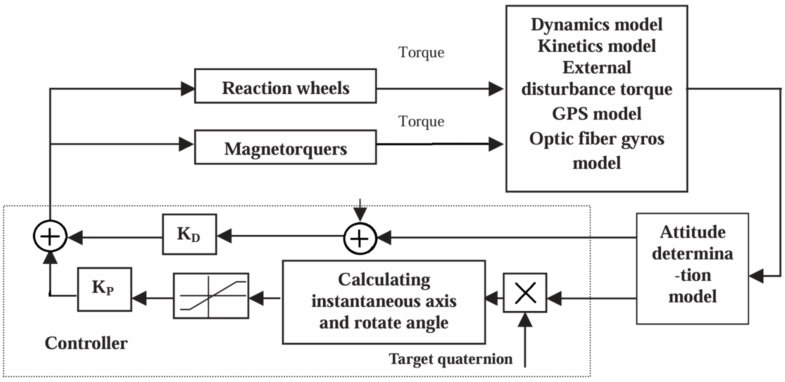


Figure 2.2 The combined control diagram of HITSAT-1 Zhaowei et al. (2004)

In the spacecraft attitude control literature, PD control has been extensively applied due to its simplicity, reliability, and real-time compatibility. As noted by Ismail and Varatharajoo (2010), quaternion-based PD controllers are effective in stabilizing three-axis satellite rotations by computing control torques directly from quaternion and angular velocity errors.

Zhaowei et al. (2004) further demonstrated this in the HITSAT-1 satellite, where a combined magnetorquer-reaction wheel control system used a PD controller to achieve reliable performance during large-angle maneuvers.

In the presence of external disturbances such as gravitational torques or unpredictable environmental forces, PD controllers alone may not suffice. Eddine et al. (2015) proposed a hybrid control strategy involving an extended state observer (ESO) coupled with a PD controller to estimate and reject disturbances in flexible spacecraft. This architecture significantly improved robustness without increasing the complexity of the controller, making it suitable for lightweight or embedded systems.

For more energy-efficient and constrained platforms, similar to robotic fingertips or end effectors used in delicate manipulation, reaction wheel systems have been configured in redundant or pyramidal arrangements. These arrangements allow fine torque control across

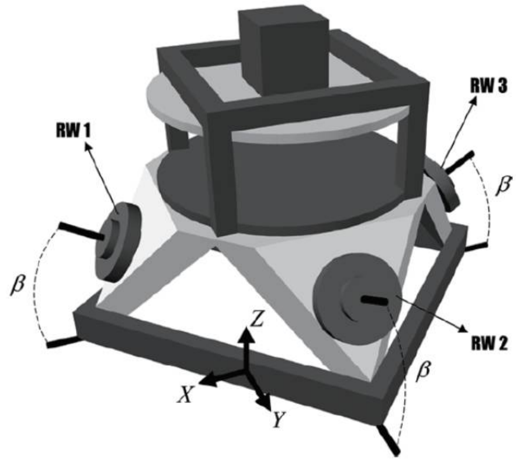


Figure 2.3 Pyramidal configuration of reaction wheel arrangement (Shirazi and Mirshams (2014))

multiple axes, as seen in the ERS-1 satellite Suzuki et al. (1990) and optimized by Shirazi and Mirshams (2014) for minimum power consumption. Though originally intended for satellites, these configurations bear direct relevance to robotics, where fingertip-integrated sensor systems must perform fine adjustments in multiple degrees of freedom with limited actuation power.

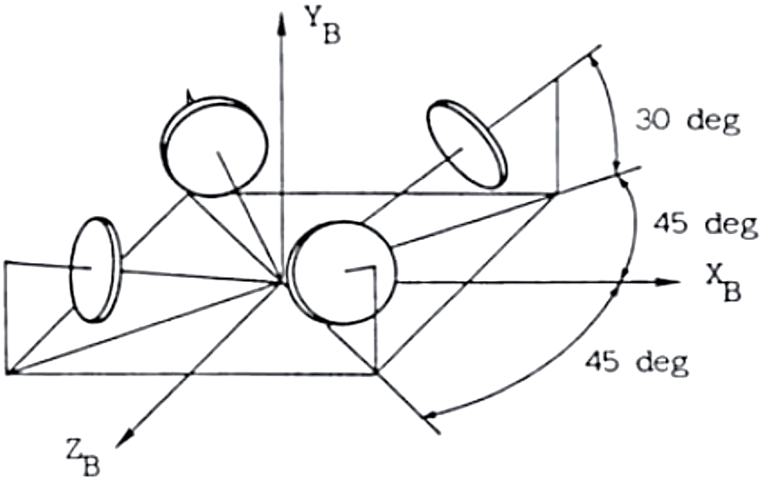


Figure 2.4 ERS-1 reaction wheel configuration Suzuki et al. (1990)

In terms of sensor feedback and fusion, the integration of inertial sensors (gyroscopes, accelerometers), magnetometers, and ToF sensors has become standard for multi-modal systems. A Kalman filter-based sensor fusion algorithm is commonly used to merge noisy IMU data with magnetometer and accelerometer output to obtain reliable orientation estimates presented by Abyarjoo et al. (2014), Kim (2011). This approach supports stable closed-loop control even in the presence of noise and drift, which is crucial when manipulating delicate items or collaborating with humans. Sensor fusion enables continuous estimation of pose and orientation, which serves as a critical input to PI or PD control loops that regulate grip force or robotic arm motion.

In conclusion, prior work on attitude control for small satellite systems and robotic manipulators demonstrates the effectiveness of simple yet robust control strategies like PI/PD when enhanced by multi-modal sensing and fusion algorithms. The novelty of our research lies in the extension of these concepts to highly compact sensors at the fingertip level that support precise, adaptive control in delicate object handling, human-robot interaction, and modular robotic platforms.

2.3 Endoscopic Stereo Vision for 3D detection

The manipulation of deformable linear objects (DLOs) poses significant challenges in robotics. DLOs, such as wires, cables, tubes, and medical hoses, are characterized by having one dimension much larger than the other two. Their often very small diameters, sometimes less than one millimeter, make accurate 3D reconstruction and shape estimation particularly difficult with conventional vision systems. This study addresses the estimation of both the position and diameter of thin DLOs through a stereo vision approach that integrates compact endoscopic cameras.

Standard depth sensors, including Intel RealSense, Microsoft Kinect, and PMD Camboard Pico Flexx, generally do not recognize very thin wires. Other high-performance devices such as Zivid One+, IDS Ensenso X36, and Photoneo PhoXi achieve sub-millimeter accuracy but are too costly and bulky for integration into compact robotic end-effectors, particularly when space for manipulation is limited Cop et al. (2021).To overcome these constraints, we propose a setup based on two commercially available endoscopic macro cameras integrated into the robot's fingertip. These cameras provide high-resolution images at close range, making them suitable for detecting DLOs in confined workspaces.

3D reconstruction of small objects in restricted areas is still an open research problem, and multiple strategies have been explored. De Gregorio et al. (2021) introduced a stereo vision approach using a single 2D camera mounted on a robotic arm to collect multiple views, thereby obtaining multiple baselines while maintaining a compact system. Neto et al.

(2021) developed a low-cost 3D scanning system that employed LiDAR to reconstruct small objects into CAD models. Similarly, Sheng et al. (2020) applied stereo vision for volume measurement of different-sized objects, while Harjoko et al. (2017) used a rotating platform to gather multi-view data of small objects for 3D reconstruction.
Research into DLO manipulation is particularly active. Caporali et al. (2023) utilized a 2D camera on a robotic arm to detect thin wires and reconstruct their shape using B-splines across multiple views. Cirillo et al. (2021d) proposed a vision-based technique that extracted wire pixels from 2D images via machine learning and, through stereo vision, guided wire insertion with controlled orientation. However, many such methods require extensive calibration and involve long acquisition and processing times. Other approaches, such as that of Monguzzi et al. (2023), focused on classifying cable connectors and estimating their state and pose for robotic bin picking. Hybrid methods have also combined tactile sensing with machine learning to estimate wire diameters for manipulation Cirillo et al. (2021a).

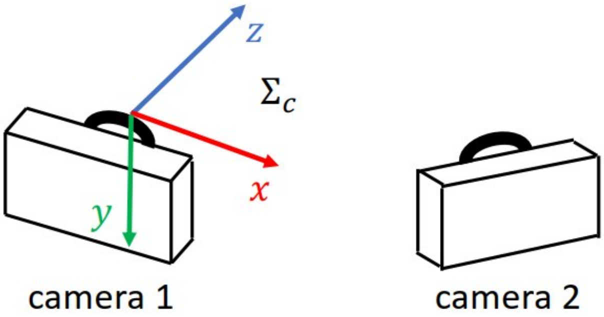


Figure 2.5 Stereo system and $\sum_c$ camera frame

The main contribution of our research is the design and development of a compact stereo vision system using two low-cost endoscopic macro cameras. The system estimates both the position and diameter of thin wires with diameters ranging from 0.5 mm to 4 mm. Furthermore, the reconstructed 3D scene can be segmented into clusters to identify wires, their centers, and orientations. To achieve this, we build upon established calibration and stereo vision techniques presented in Zhang (2002),Heikkila and Silvén (1997), Hirschmuller (2005),and Muja and Lowe (2009) , adapting them for endoscopic imagery. Finally, the reconstructed data is validated by integrating with tactile-sensorized robotic fingers previously developed by the authors Cirillo et al. (2021a).

Part II

Evolution of Multi-Modal Sensor and Robotic Applications

Chapter 3

Design and Development of Multi-Modal Sensor for Adaptive Gripper

The trend of robotic manipulation in complex and unstructured environments necessitates the integration of various sensory modalities to enable accurate and adaptive grasping and control. This phase of the research presents the design, development, and integration of multi-modal sensor for robotic gripper, incorporating optoelectronic based technology, Time of Flight (ToF) pre-touch sensing, and inertial measurement unit (IMU). Tactile sensing is fundamental for perceiving fine-grained surface textures, detecting pressure, and enabling precise object manipulation. Time-ofFlight pre-touch sensors offer proximity information with ultralow latency, enhancing anticipatory Human Machine Interaction (HMI) responses. IMUs provide critical motion and orientation data to track gestures and postures. The developed sensor system aims to enhance the gripper's abilities by providing real-time feedback about object properties, pose estimation and dynamic interactions during grasping and manipulation tasks. The novel multi-modal sensor has very smaller size equivalent to human finger tip and more robust design that yields the higher sensitivity, higher flexibility and less noise.

3.1 Sensor Technology

The working principle is based on the designing of a multilayered PCB, microcontroller based, constituted by an IMU and a discrete number of optoelectronic photo-reflectors (named ”taxels” and consisting of Light Emitting Diode and Photo-transistor couples), arranged in a specified configuration enclosed by a deformable layer. The same microcontroller, through a specific connector available on the PCB, allow also the connection of the ToF sensor module, designed by authors in Cirillo et al. (2021b). It completes the hardware of this multi-modal sensor. The connector is mounted by allowing the integration of ToF on the top side of multi-modal sensor. The target objectives of the whole system are the following.

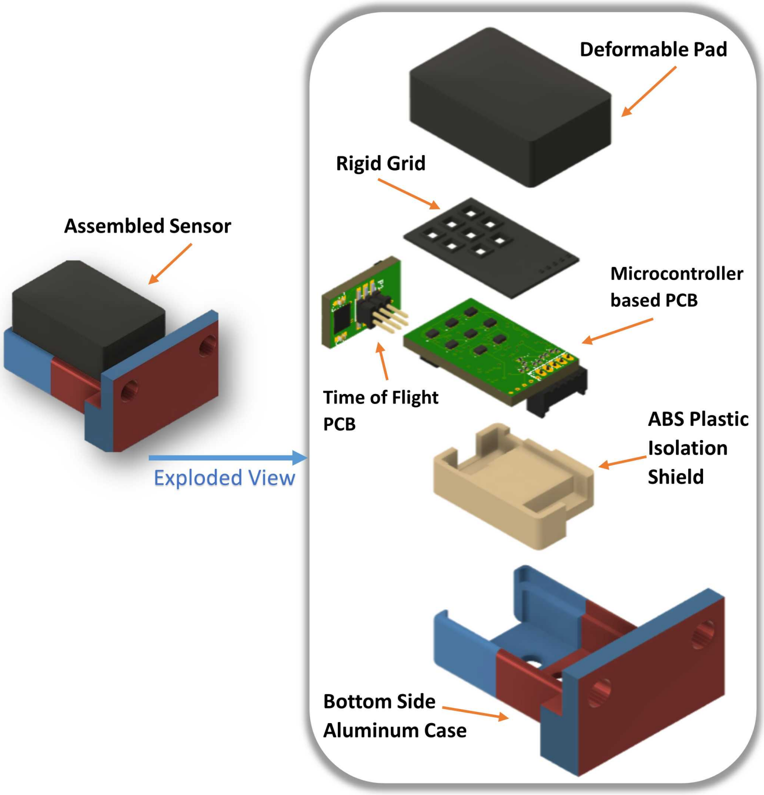


Figure 3.1 CAD of an assembled sensor (left) and exploded view with all components (right).

1. To transduce the contact data into deformations measured by the optical components. The light emitted by the LEDs is reflected by the bottom side of deformable layer and it is measured by the photo-transistors. The measured signals depends from the amount of deformations, related to external contact. The measuring points, spatially distributed, correspond to the tactile map. These data can be used, for example, to evaluate the shape of grasped object or reconstruct the external forces.

2. To continuously monitor the close range by means of the proximity data acquired through the ToF sensor. These data can be used, for example, to avoid obstacles/collision, to interact with the environment/human or reconstruct locally the 3D information from the scene.

3. To continuously monitor IMU data, which can be used, for example, to estimate the pose, to prevent slippage events or to evaluate the object temperature.

The exploded CAD model of multi-modal sensor with all parts aligned is reported in the Fig. 4.2. Some mechanical parts (e.g., grid, PCB support) have been designed to complete the assembly and realized in ABS plastic by means of 3D printing technology. Also the deformable pad, made of silicone, has been realized by using ABS printed molds with high accuracy. The external support for the connection with the gripper has been designed in order to be realized in aluminum by means of CNC machines.

3.1.1 Electronics Section

The electronic components of sensor are integrated on multi-layered PCBs, including opto-electronic based taxels and inertial measurement unit on a first main board with dimensions equal to $12.5 \times 20\,\mathrm{mm}$ and Time of Flight sensor on a second board, connectable to first one, with dimensions equal to $12 \times 8\,\mathrm{mm}$. The main PCB is designed with new current driving channels for LEDs, optoelectronic section, buffering section and an efficient micro-controller with a sufficient number of Analog-to-Digital (A/D) channels for the tactile signals acquisition. The optoelectronic section consists of 8 taxels organized as shown in Fig. 3.2 with a spatial resolution equal to 3.55 mm. Each taxel exploits a photoreflector (manufactured by

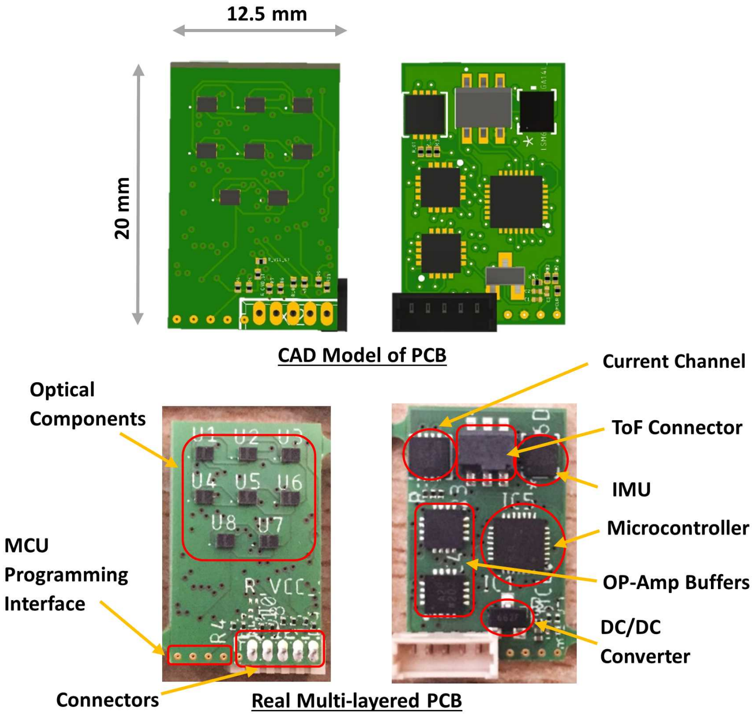
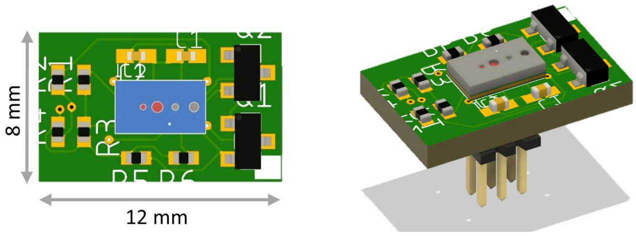
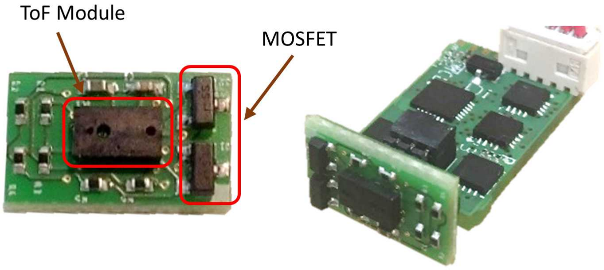


Figure 3.2 Multi-layered PCB with Front View on Left Side and Bottom View on Right Side

New Japan Radio, product code NJL5912R) composed of a LED and a photo-transistor, with the dimensions of $1.06 \times 1.46 \times 0.5$ mm. In previous version De Maria et al. (2012), each LED was powered with voltage supply but in the later version, LEDs are connected in series and they are driven by an adjustable current channel (manufactured by Texas Instrument, product code LM334) with an output current fixed to 4 mA. For the proposed design, by exploiting the high sensitivity of the optoelectronic component (NJL5912R), a new robust current source with output current equal to 1 mA is introduced in the circuit, in order to reduce also the power consumption of the sensor. The current channel implements the standard configuration available from datasheet, with single resistor, which yields the low temperature dependance characteristic, to fix the output current with high precision, from a voltage power supply. Analogue buffers have been introduced in the circuit, in order to decouple the voltage signals from photo-transistor and the A/D converter inputs: the low power operational amplifiers (Manufactured by Analog Devices, product code ADA4691) have been used in the voltage follower configuration. The low power micro-controller PIC16F19155 (Manufactured by Microchip), equipped with a $12-$bit resolution on-board A/D converter, directly digitalized the buffer outputs. These low noise A/D converters enhance the signal to noise ratio and enables the simplification of firmware development.



CAD Model of ToF Proximity PCB



Proximity Module **Integration of ToF PCB with Main PCB**

Figure 3.3 Time of flight sensor module PCB

The IMU (Product code LSM6DSV16X, manufactured by STM Microelectronics) is another useful addition to the PCB. It provides 3−axis gyroscope, 3−axis accelerometer, 3−axis magnetometer and temperature, with also processing capabilities on board. It has multiple protocols of communication (i.e., Serial, I2C, SPI), but the developed sensor utilizes the I2C protocol.

Regarding the ToF sensor module, it was precisely designed to ensure self-consistency. All essential components required for seamless communication with the VL6180X device (manufactured by STMicroelectronics), and its power supply are incorporated on the module's PCB. Particularly, a Texas Instruments TPS76928 LDO Regulator is employed to provide the sensor with a stable 2.8 V power source. In addition, two NXP BSS138 components function as level shifters, adapting logic levels for the I2C bus. This compact module offers accessibility through a 6−way connector (Product code FTSH-103-04-F-DV, manufactured by Samtec) for communication with microcontroller. The details about the interconnection among ToF module and main board are reported in Fig. 3.9. Additional details about the ToF module can be found by the interested readers in Cirillo et al. (2021b). The complete block diagram of all electronic components with connections is summarized in Fig. 4.4

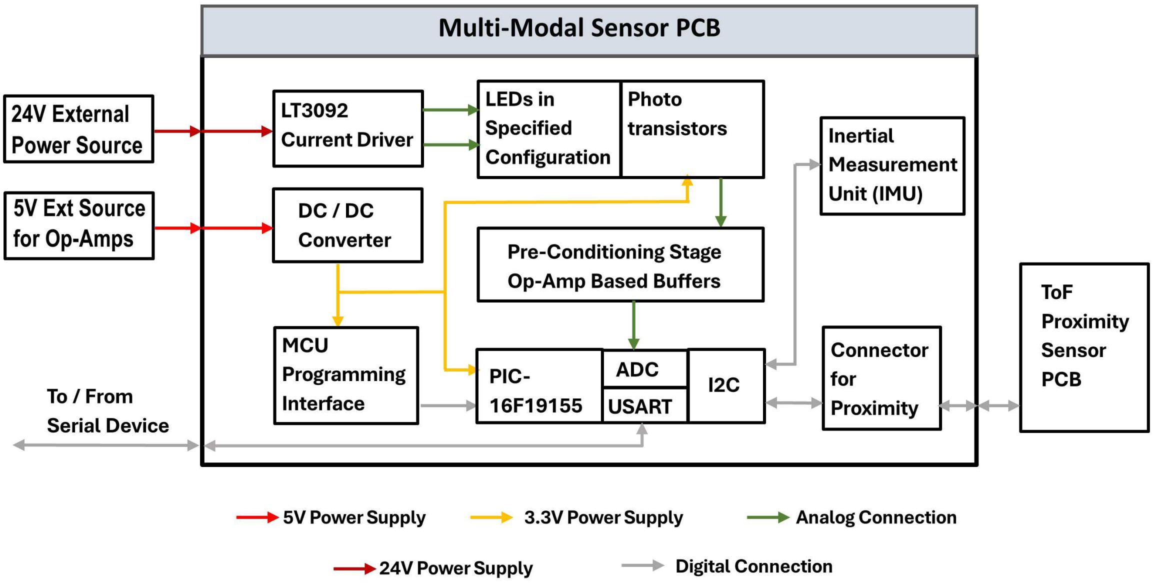


Figure 3.4 Block diagram of electronic components

3.1.2 Mechanical Parts

A fundamental part of this multi-modal sensor is the deformable pad, that can be realized in different shapes depending upon the application scenario. For example, as reported in Costanzo et al. (2019a), if information regarding torques and contact forces for manipulation are needed, a domed shaped deformable pad can be used for their reconstruction, by applying a trained neural network to tactile map signals. Alternatively, if object sizes are smaller with respect to pad and the objective is the recognition of object features for manipulation (e.g., the shape of a grasped wire), a pad with flat surface can be utilized Palli and Pirozzi (2019a). The presented design in this study of research has flat surface, while the bottom side of the pad presents black walls, to optically isolate the taxels, and white cells in front of optical components, in order to maximize reflectivity, as reported in Fig. 3.5. The material used for the manufacturing of pad is silicone, due to its low hysteresis as well as good elastic property, compared to other deformable materials.

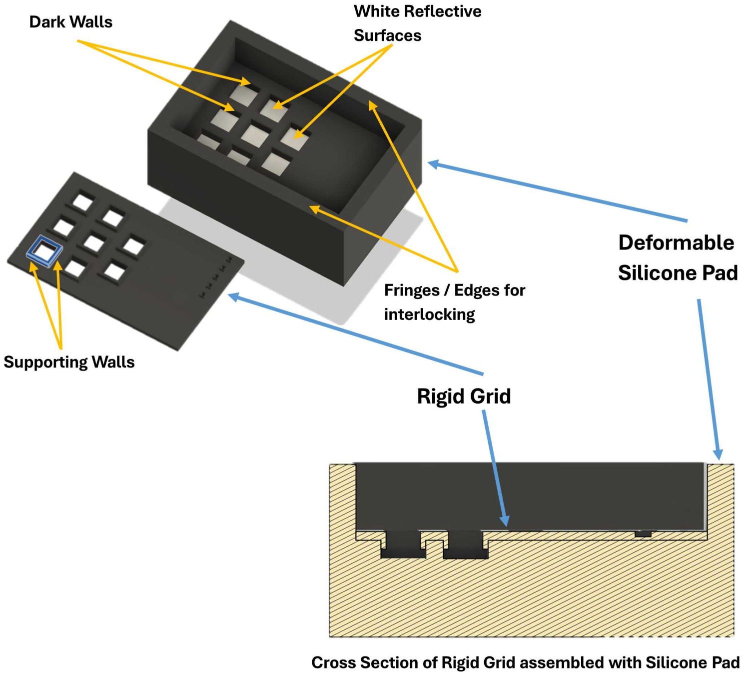


Figure 3.5 Rigid grid and deformable silicone Layer characteristics.

Between the deformable pad and main PCB, there is the rigid grid, which main objective is to ensure for the optoelectronic components to work within a monotonic range, since these components exhibit a non-monotonic behaviour when the reflective surface distance becomes less than 300μm Costanzo et al. (2019b). In particular, the designed ensures that the distance of reflective surface from the photo-reflector cannot reach a value lower than 500μm. The upper side of grid is designed in a way to connected with dark walls of pad, to enhance the interlinking robustness between deformable pad and rigid grid. The grid has been manufactured by 3D printing technology using black ABS plastic, with precision of about $\pm 100\mu$m and the black color is selected to avoid undesired lateral reflections. The grid is bonded to deformable pad on one side and PCB on the other side (see Fig. 4.2), by means of a cyanoacrylate glue.

To complete the design of sensorized fingertip, a case for mounting and housing the multi-modal sensor has been designed. The case constitutes of two parts: a first part, manufactured by ABS plastic via 3D-printing technology, designed for isolating the bottom section of PCB from the mounting case; the second part which is the mounting case, realized in aluminium by means of CNC machining technique. The latter is composed of two sub-sections, that laterally slide to coincide at an intermediate point, and lock the deformable pad. These sub-parts got fixed with screws located at the back side of isolation shield, and the bottom sub-part has two holes for mounting the sensor on the commercial gripper's finger (see Fig. 4.2).

3.2 Software Design

Third section describes the development of software design, consisting of two main components:

1. Microcontroller Firmware

   - Obtain the data from optoelectronic components by microcontroller unit.
   - Get the proximity information from ToF sensor using I2C communication protocol.
   - Acquire the accelerometer, gyroscope and temperature data from IMU on microcontroller side.

2. Software for Elaboration Unit

   - Acquire, process and formulation the information from multi-modal sensor to the elaboration unit (e.g., Personal Computer).

3.2.1 Microcontroller Firmware

The simplest firmware for microcontroller has been developed for the purpose of efficient and robust as well as to make the high acquisition rate for optoelectronic components and inertial measurement unit. To maximize the performances of controller, all the processing and elaboration for taxels take place in PC based elaboration unit while for the IMU, all the processing take place at the builtin computational unit of IMU and transmit data to the microcontroller (PIC16F19155) using I2C communication protocol. Similarly VL6180X ToF sensor utilizes the I2C protocol (a new custom designed low level driver) for communication. The firmware for ToF includes the identification of devices, setup registers and measure the range. Microcontroller just convert the voltages and transmit the digital data through serial port to the elaboration unit. Pseudo-code (Algorithm 1) presents the structure and operations carried by the microcontroller.

Algorithm 1 Microcontroller Firmware

```
procedure MAIN
    Initialize System
    Configure ADC
    Configure I2C
    Configure Serial
    while true do
        $rxData \leftarrow$ read $char$
        if $rxData == $ a then
            for $k \leftarrow 1$ to 8 do
                $tactData \leftarrow$ data from analog channel $k$
                Send $tactData$ over serial
            end for
        else if $rxData == $ b then
            Send $SensorID$
        else if $rxData == $ c then
            $tofData \leftarrow$ read ToF registers from I2C
            Send $tofData$ over serial
        else if $rxData == $ d then
            $imuData \leftarrow$ read imu registers from I2C
            Send $imuData$ over serial
        end if
    end while
end procedure
```

3.2.2 Elaboration Software (PC-Based)

Regarding the software for the elaboration of data, Robot Operating System (ROS) has been selected as the framework for handling the interprocess communication and task scheduling since it is broadly acknowledged within the robotics research community. Interrogation software, developed as a ROS node, is the one responsible for requesting data from the sensor and making it available for all the elaboration nodes on the ROS network.

This software has been designed to offer maximum flexibility, regardless of the sensor types or the number of sensing elements. As for the data elaboration, instead, new ROS nodes can be easily added to the network since they only need to subscribe to the topics where sensor data is published.

Algorithm 2 PC Side ROS Node

```
procedure MAIN
    Initialize ROS node and publishers
    Enumerate available serial ports
    if no port available then
        Terminate program
    end if
    Open serial port
    Send 'b' to request Sensor ID
    if sensor not detected then
        Close port and terminate
    end if
    Send 'c' to request number of sensing elements
    if number of sensing elements = 0 then
        Close port and terminate
    end if
    while ROS is running do
        Send 'a' to request raw sensor data
        Read raw data from serial port
        Publish data on ROS topic
    end while
    Close serial port
end procedure
```

3.3 Integration and Testing

This section reports experimental results about the mechanical integration within robotic gripper and signal acquisition for the proposed sensing solution.

3.3.1 Integration with Three-Finger Adaptive Gripper

The multi-modal sensor designed for this study has been integrated on the three-finger adaptive gripper manufactured by Robotiq (2020). This integration engaged delicately aligning and installing the sensor on the gripper's structure to guarantee optimal performance and minimal interference with the gripper's functionality. Exceptional attention was given to the placement of the sensor to allow for effective data collection during gripping and manipulation tasks. The sensor was firmly attached to the gripper using custom-designed mounting brackets as shown in Fig. 3.6. This ensured stability and alignment, granting the sensor to gain data accurately during various gripping scenarios. The mechanical integration was executed to maintain the overall physical integrity of the gripper.

Figure 3.6 Mechanical integration within three-fingers adaptive gripper

3.3.2 Testing Procedures

To evaluate the performance of the integrated multi-modal sensor, comprehensive testing experiments were conducted. The testing aimed to assess the sensor's capabilities in capturing tactile feedback, measuring time-of-flight proximity, and recording inertial measurements (i.e., accelerations, angular rates and temperature) during dynamic movements.

Tactile Signals

The sensor was subjected to various objects with distinct surface textures, shapes, and materials. hence tactile capabilities were assessed by measuring the pressure distribution exerted on the gripper's fingers during gripping tasks with data acquisition at sampling frequency up to 1000 Hz. The corresponding data for tactile sensing are presented in Fig. 3.7, where raw voltages coming from the 8 taxels are reported.

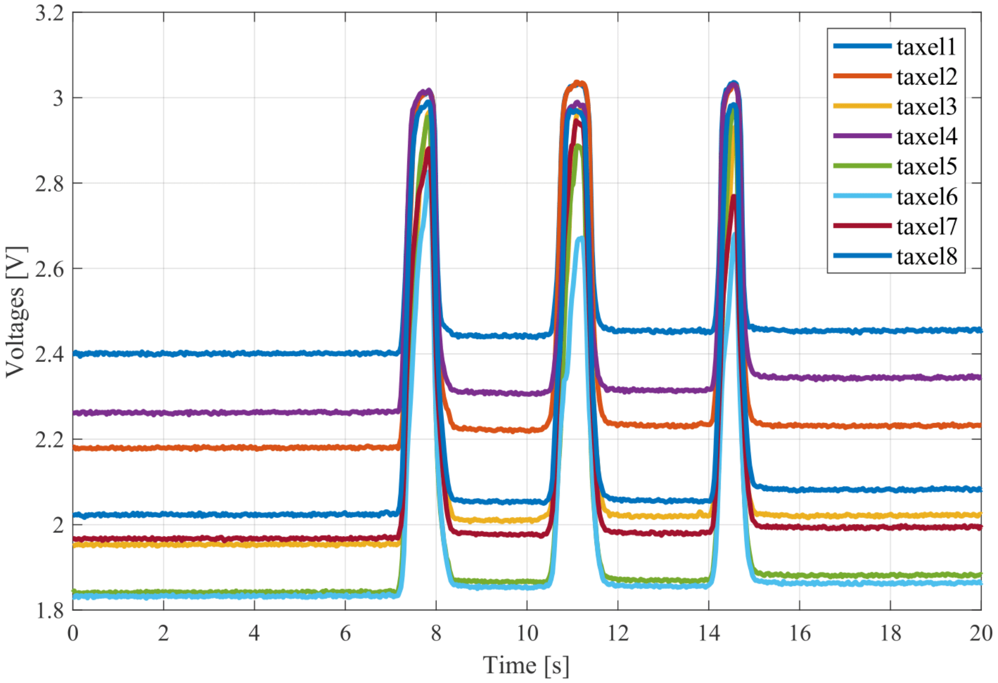


Figure 3.7 Detection peaks for opto-electronic components

Data show how the raw signals present variations already sufficiently high and with low noise, to be easily used for robotic applications, without the needs of additional conditioning electronics and/or pre-processing algorithms.

Time-of-Flight Proximity Signal

The integrated proximity sensor can measure the distance from a target with a sampling frequency equal to 50 Hz. An example of distance measurements with a moving target is reported in Fig. 3.8, where the low-noise feature is clear. Then, the ToF proximity measurement was tested by analyzing its capability to detect the distance between the gripper and surrounding objects, positioned at different distances. In this case, a comparison of measured distances with ground truth data is reported in Fig. 3.9.

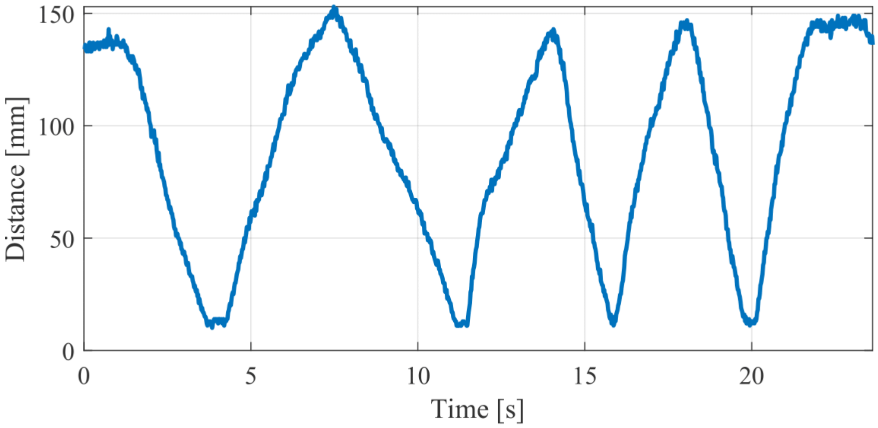


Figure 3.8 Distance measurements through ToF sensor.

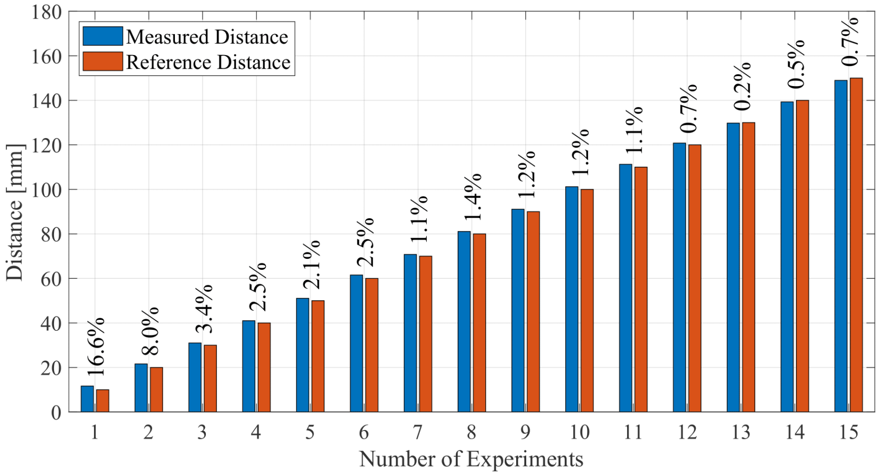


Figure 3.9 Comparison of time of flight measured and reference distances.

The ToF measures have been computed as the mean value of 10 measurements, while for the ground truth, the distances have been acquired by means of a measuring scale. The experiments show a precision of the sensor almost equal to its resolution, i.e., 1 mm. This results in higher percentage errors, i.e., $> 10\%$, when the object is very close to the sensor, and smaller errors, i.e., $< 1\%$, for farther objects.

Inertial Measurement Unit Signals

Dynamic movements of the gripper's fingers were analyzed using the integrated inertial measurement unit. The sensor's ability to gain accelerations, angular rates and temperature signals was evaluated during controlled motion sequences, providing data at a sampling frequency equal to 500 Hz.

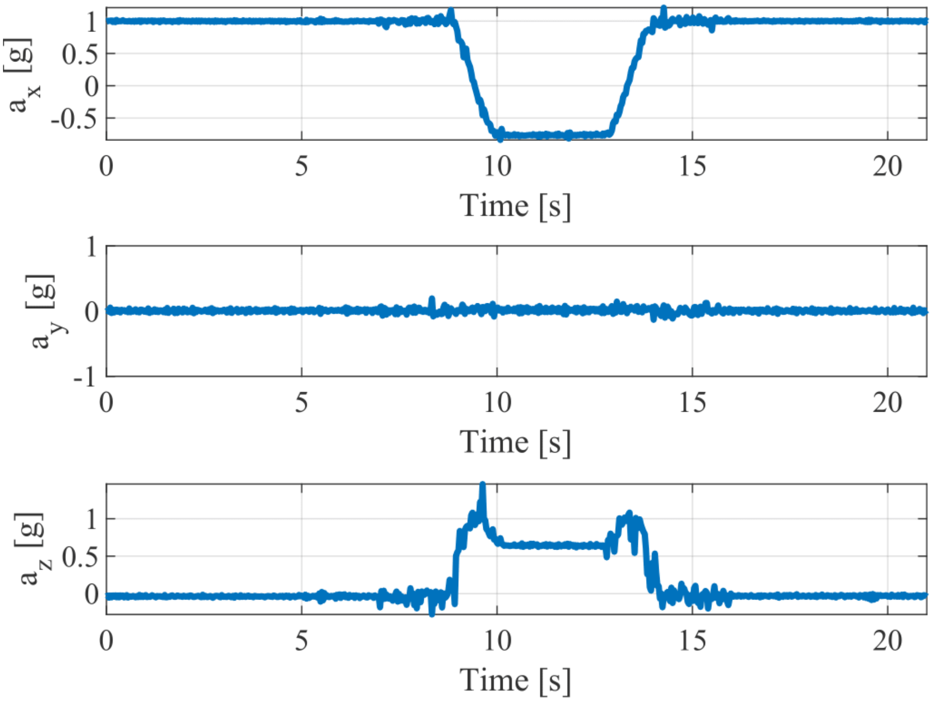


Figure 3.10 Accelerations produced during slippage

The data in Fig. 3.10 presents an example of accelerations measured during a closing, followed by an opening, of the gripper finger. The labels incorporated along the vertical axes, denoted as a_x, a_y, and a_z, serve to articulate the linear accelerations along the respective Cartesian axes, namely the $x-$axis, $y-$axis, and $z-$axis. From data, since a_y is almost zero, it is clear how the IMU have the $y-$axis orthogonal with respect to the plane in which the finger movement happens.

Similarly, Fig. 3.11 reports the angular rates in degrees per second (dps) of the finger tip, measured by the IMU during an adduction-abduction movement, followed by a closing

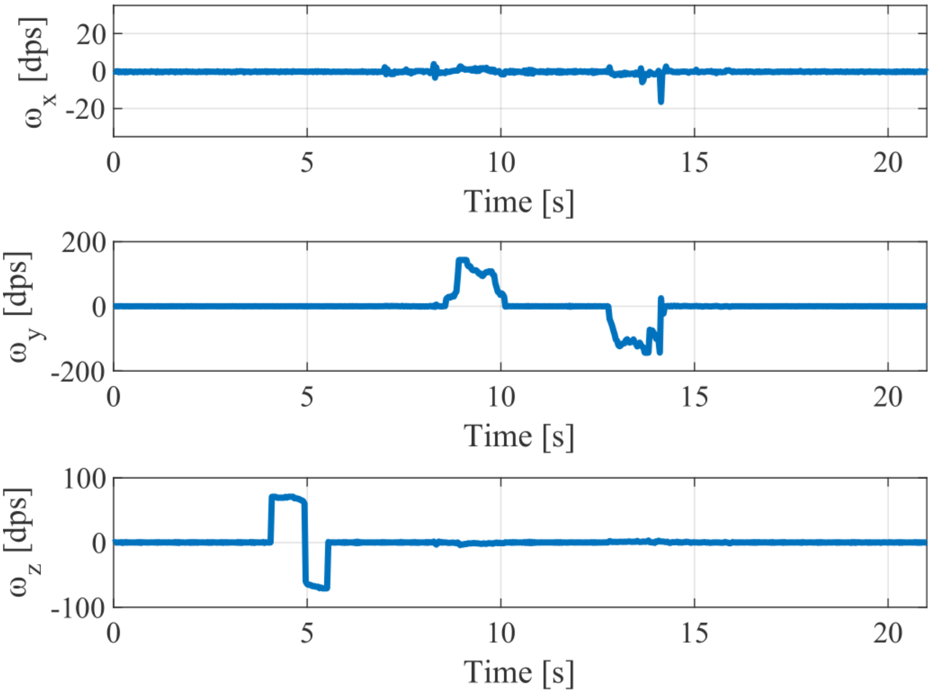


Figure 3.11 Angular rate of the finger-tip measured by gyroscope

and an opening. In this case, the assigned labels denoting the angular rates as ω_x, ω_y, and ω_z around the $x-$axis, $y-$axis, and $z-$axis, respectively. Also in this case it is evident that

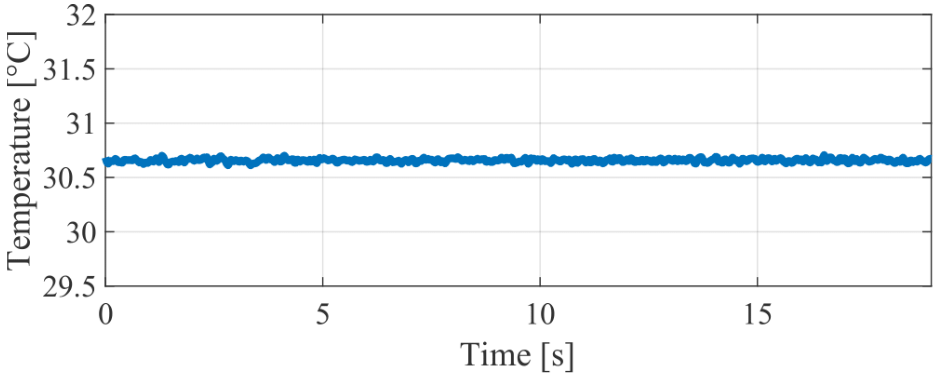


Figure 3.12 Temperature sensed by sensor equipped in IMU

during the adduction/abduction movement only a rotation around $z-$axis is visible (at about $t = 5\,\mathrm{s}$), while during the closing/opening only a rotation around $y-$axis is visible (during the second part of experiment).

Finally, the IMU allows also to obtain a measurement of temperature as reported in Fig. 3.12.

Chapter 4

Multi-Modal Sensing for Human-Robot Interaction and Agri-food

Recent advancements in robotic grasping have increasingly emphasized the importance of intelligent manipulation in industrial and collaborative applications, particularly for handling delicate, deformable, or perishable objects. The ability of robotic grippers to securely and adaptively manipulate both soft and rigid items is critical in domains such as automated food processing, precision assembly, and human-robot collaborative workspaces. Equally important is the integration of human-friendly interaction mechanisms that ensure safety, adaptability, and intuitive control in shared environments.

This research addresses these challenges through the development of a compact multi-modal sensor system that combines a high-resolution optoelectronic tactile matrix with an IMU, enabling enriched object perception and adaptive force feedback. The sensor is integrated into a Robotiq 3-finger adaptive gripper, significantly enhancing its manipulation capabilities across diverse object types. To ensure robust and precise control, the system employs a Proportional-Integral (PI) controller that dynamically regulates finger positioning based on real-time tactile and inertial feedback.

Further innovation of proposed design lies in its HRI-oriented control strategy, which enables the gripper to respond adaptively to external forces applied by humans. Depending on the direction of force, longitudinal or lateral, the gripper either facilitates controlled object release or reinforces object retention, thereby promoting safe and collaborative operation in distributed workstations.

The primary contributions of the proposed system can be summarized as follows.

- Development of an extended multi-modal fingertip sensor with high tactile resolution for enhanced object perception.

- Integration of the sensor within a Robotiq 3-finger adaptive gripper to improve dexterous grasping across varied objects, as presented in Fig 4.1
- Implementation of PI control for precise finger position regulation using tactile-inertial feedback.
- Introduction of an HRI-based control strategy that allows intuitive and safe object release mechanisms.
- Integration of the sensor within a four-finger developed gripper to improve gripping across the agri-food industry.
- Implementation of a proportional controller for the precise regulation of finger position using tactile-inertial feedback.
- Experimental validation of the proposed system, demonstrating its effectiveness in delicate object handling and collaborative scenarios.

Through these contributions, this study advances the state of the art in compact, human-inspired multi-modal sensors for robotic grippers, enabling both enhanced dexterity and human-friendly interaction in industrial and collaborative robotics.

Figure 4.1 Gripper demonstrating its adaptability by securely holding a soft ball, a hard ball, and a water bottle in both horizontal and lateral orientations

4.1 Sensor Design

The second variant of the proposed multi-modal fingertip sensor, represents a deliberate evolution of the initial design with a strong focus on improving spatial resolution and tactile interpret-ability. In contrast to the first prototype, which employed a set of eight optoelectronic tactile elements arranged in a non-uniform configuration, the second variant adopts a structured 4×3 tactile array.

This regular matrix layout enables a more consistent and meaningful tactile mapping of the contact surface, facilitating the reconstruction of contact distribution and improving robustness across different grasping scenarios. The increased spatial resolution and ordered taxel arrangement allow the sensor to capture richer information about object geometry, contact location, and force distribution, which is particularly beneficial for adaptive and multi-contact manipulation tasks.

Furthermore, the matrix configuration naturally supports the computation of centroid-based features, providing valuable information for grasp alignment, contact stability assessment, and interaction control. Overall, this design refinement significantly enhances the sensing capabilities of the system while preserving the compact form factor and multi-modal integration established in the first sensor variant.

The multi-modal sensor incorporates a multilayer printed circuit board (PCB) design that integrates a microcontroller, a 4×3 array of optoelectronic photo-reflectors, referred to as "taxels", comprising LED and phototransistor pairs arranged in a matrix beneath a deformable layer, along with an inertial measurement unit (IMU) to extend sensing capabilities. An exploded CAD model of the sensor, illustrating all internal components, is presented in Fig. 4.2. The mechanical elements, including the grid and PCB adapter for gripper integration, were fabricated from ABS plastic using 3D printing. The deformable sensing pad was produced from silicone using precision ABS-printed molds.

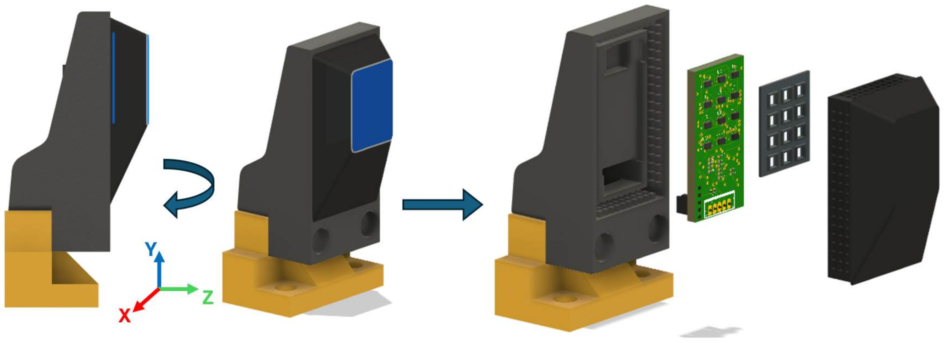


Figure 4.2 CAD of an assembled sensor (left) and exploded view with all components (right)

The sensor's primary function is to transduce contact interactions into quantifiable deformations through its optical components. Light emitted by the LEDs is reflected from the underside of the deformable layer, and the corresponding variations are detected by the photo-transistors. These variations correlate with the magnitude and distribution of surface deformation caused by external contact, enabling the generation of a tactile map for object shape estimation and force reconstruction. Complementing this, the IMU provides continuous data that supports object pose estimation, slip detection, and temperature measurement of grasped items.

The sensor is built on a six-layer PCB measuring (12.65×25.9) mm, featuring an array of 12 taxels (4×3), implemented using NJL5912R photo-reflectors from New Japan Radio, soldered onto the top layer along with supporting resistors, as shown in Fig. 4.3.

The bottom layer consists of the analog buffering circuitry with low-power operational amplifiers (ADA4691), connectors, an adjustable current source (LT3092), an inertial measurement unit (LSM6DSV16X), and a microcontroller (PIC16F19155). The microcontroller provides a sufficient number of 12-bit A/D channels for tactile signal acquisition, enabling digitization of buffered outputs to improve signal-to-noise ratio while also supporting efficient firmware development.

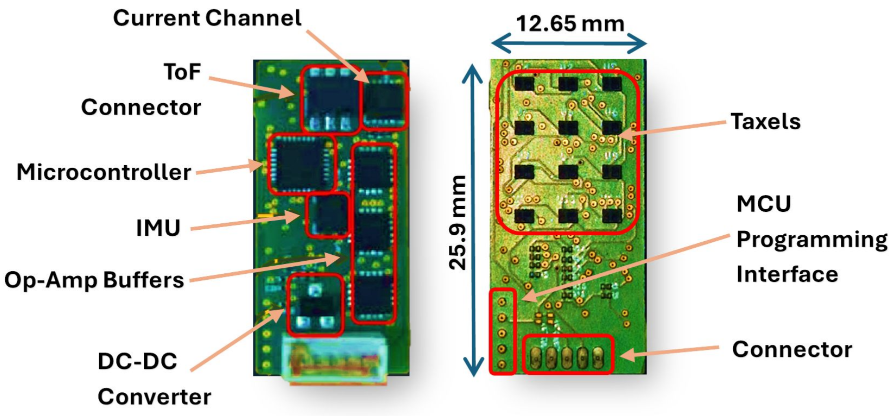


Figure 4.3 Multi-modal sensor PCB: bottom (left) and top (right) views.

The LEDs of the optoelectronic elements connected in series are powered by 1 mA from the adjustable current channel, connected to the 24 V external power supply. The photo-transistors are powered up by the 3.3 V derived from a DC/DC converter, which is connected to a 5 V external power source. The signals from the photo-transistor are transmitted to the OP-Amp based buffers, which subsequently feed into the analog-to-digital (AD) channels of the microcontroller. The IMU integrated on PCB offers a 3−axis gyroscope, accelerometer, and temperature sensor, with on-board processing capabilities, is also powered by a 3.3 V source and communicates with microcontroller using the I2C communication protocol. and

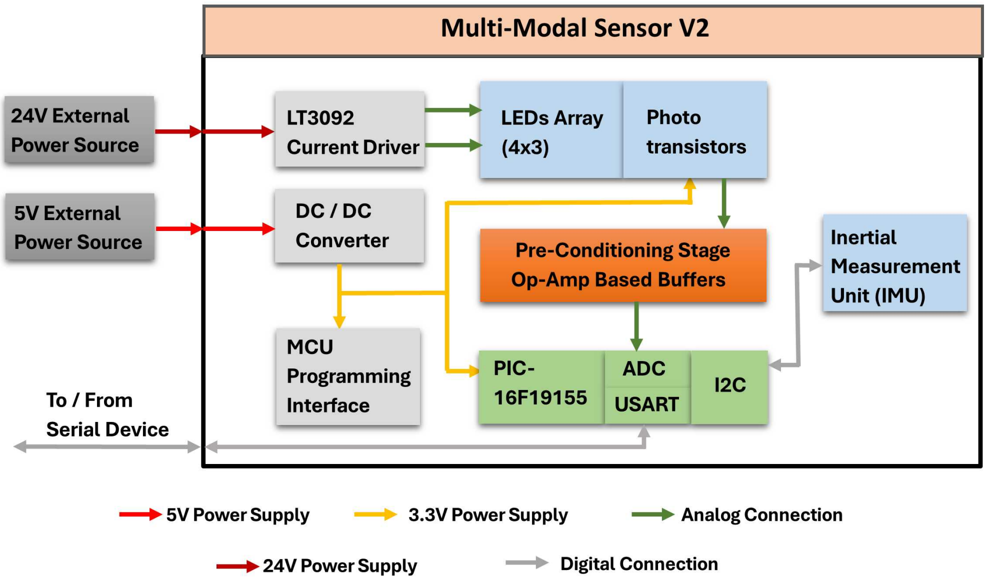


Figure 4.4 Architecture of power and communication flow within sensor

the sensor utilizes the UART serial communication protocol to communicate with the main PC as illustrated in the block diagram shown in Fig. 4.4.

The deformable pad is a fundamental component of the sensor, as its geometry directly influences the type and quality of tactile information acquired. Depending on the target application, the pad can be realized in different shapes. For example, when the objective is to identify fine features or manipulate objects smaller than the pad dimensions, a flat surface pad offers superior contact uniformity and improved resolution of object features Palli and Pirozzi (2019b). Conversely, dome-shaped deformable pads have been shown to enhance the reconstruction of multi-directional contact forces and torques. By exploiting tactile map signals and employing trained neural network models, such geometries can provide robust estimation of external interactions, as reported in Costanzo et al. (2019a).

In the proposed design, the sensor employs a flat pad with a polygonal frustum-like profile, as illustrated in Fig. 4.2. The active region of the pad is carefully designed to enclose only the optoelectronic sensing elements, ensuring that external deformations are accurately transduced while minimizing interference with other components. This targeted design enhances the functional efficiency of the optoelectronic elements by focusing deformation on the active sensing area, thereby improving both sensitivity and accuracy. Additionally, the polygonal frustum shape provides a streamlined structure that contributes to compactness and ease of integration with robotic grippers, without compromising tactile performance. The resulting design not only ensures high-quality tactile perception but also maintains an effective balance between robustness, miniaturization, and application-specific adaptability. The pad is realized from silicone, chosen for its low hysteresis and favorable elastic properties compared to other deformable materials. To ensure accurate signal acquisition, the underside of the pad incorporates black walls that provide optical isolation between taxels, thereby minimizing cross-talk. In contrast, white reflective surfaces are strategically positioned in front of the optical components to maximize light reflection, as illustrated in Fig. 4.5.

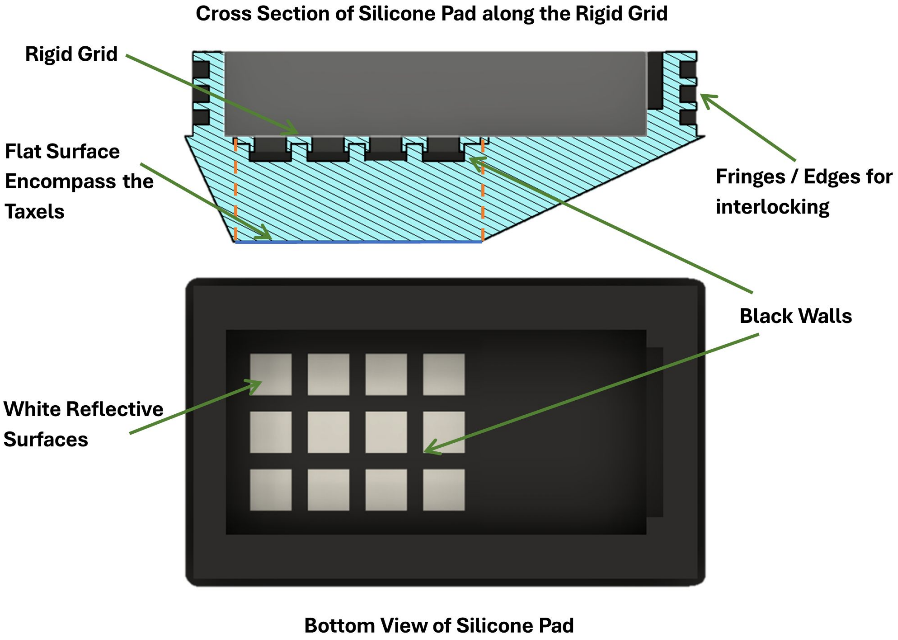


Figure 4.5 Fingertip detailed cross-section (top) and a bottom view of pad (bottom).

A rigid grid is placed between the PCB and the deformable pad to maintain consistent operating conditions for the optoelectronic components, ensuring that their response remains within a monotonic range. The structural design guarantees that the minimum separation between the reflective surface and the photo-reflectors does not fall below $500,\mu$m, thereby preserving measurement accuracy and preventing signal saturation. The grid is fabricated from black ABS plastic using high-precision 3D printing, achieving dimensional tolerances of approximately $\pm 100,\mu$m. The use of black material is deliberate, as it minimizes unwanted lateral light reflections and enhances the optical isolation of each sensing element, contributing to the reliability and accuracy of the tactile measurements.

The tactile sensor acquires raw voltage outputs, as depicted in Fig. 4.6, which are subsequently transformed into force measurements through a calibrated transfer function. Calibration is performed against ground-truth reference data obtained from a load cell (Fig. 4.7), enabling the derivation of an accurate voltage-to-force mapping. This process ensures high-fidelity force estimation during grasping tasks and improves the reliability of tactile sensing in manipulation scenarios.

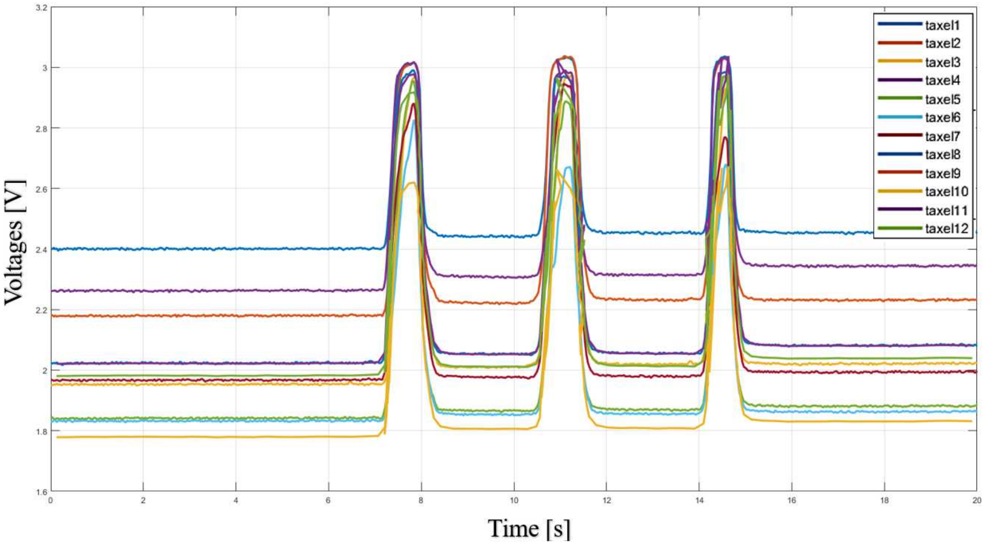


Figure 4.6 Signals from the 12 taxels, with three distinct peaks upon contact.

The experiment was conducted to investigate the correlation between the cumulative voltage outputs of the tactile sensor, integrated into a Robotiq gripper operating at minimal velocity, and the corresponding force measured by a reference load cell. The primary objective was to characterize how variations in voltage readings reflect the applied contact force, a critical aspect for applications in robotic manipulation, force estimation, and closed-loop feedback control.

The proportionality observed in both plots of Fig. 4.7 demonstrates a strong correlation between tactile sensor voltages [V] and applied forces [N] over an interval 5 s. The tactile sensor exhibits a voltage response ranging from 0 V to 6.8 V, corresponding to an applied force range of approximately 0 N to 34 N. This linear trend indicates that increasing contact forces result in proportional increases in voltage output, thereby validating the sensor's capability to reliably transduce tactile stimuli into measurable electrical signals.

4.2 Integration with Robotiq 3-Fingers Adaptive Gripper

4.2.1 Control Architecture

To enable precise and responsive object manipulation, the control architecture has been developed to regulate the position of all three fingers of the adaptive gripper in a coordinated

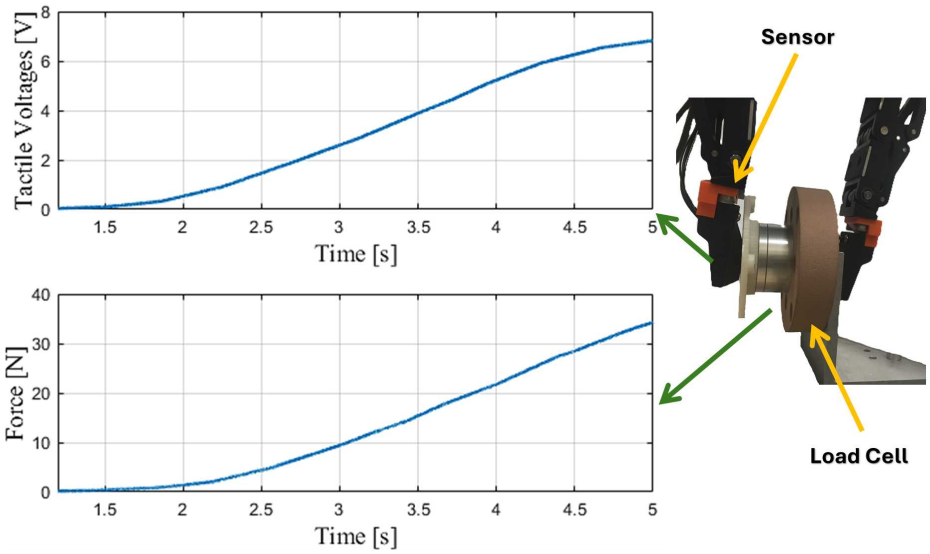


Figure 4.7 Relationship between the sum of voltages measured and the force measured by the load cell

manner. Finger is actuated to follow reference trajectories derived from the grasping task requirements, while tactile feedback from the integrated multi-modal sensor is continuously processed to adjust the finger positions in real time. This closed-loop position control ensures that the gripper can adapt to variations in object geometry, surface properties, and external disturbances, thereby improving grasp stability and reducing the risk of slippage or excessive force application. The architecture, therefore, not only enhances the accuracy of finger positioning but also contributes to the overall dexterity, reliability, and safety of the manipulation process.

The control architecture is based on a Proportional-Integral (PI) control strategy, selected for its effectiveness in minimizing steady-state error while maintaining a fast and stable system response. By compensating for accumulated error through the integral action and correcting instantaneous deviations via the proportional term, the PI controller ensures accurate tracking of reference trajectories without introducing excessive overshoot or sluggish dynamics. The primary motivation for adopting this strategy is to guarantee a consistent and reliable performance of the gripper under varying operational conditions and object interaction scenarios.

The overall control system is organized around a closed-loop feedback structure, in which the positions of the gripper fingers are continuously monitored and compared against desired setpoints. Any deviation generates an error signal that is processed by the PI controller to dynamically adjust the control input, thereby achieving precise regulation of finger motion.

The architecture integrates three key components: (i) the multi-modal sensor, which provides tactile and inertial feedback for accurate perception of object interaction; (ii) the PI controller, which computes corrective actions to minimize position error; and (iii) the actuator interface, which translates the controller output into physical finger movements. Together, these components enable robust, adaptive, and responsive manipulation of objects across diverse scenarios. The multi-modal sensor serves as a critical element of the control architecture, delivering continuous real-time feedback on the position and orientation of the gripper fingers. The raw sensor data is processed to extract relevant positional information, which is subsequently relayed to the PI controller for instantaneous corrective action, as illustrated in Fig. 4.8. Within this closed-loop framework, the PI controller regulates the finger positions by comparing the sensed feedback against predefined reference setpoints. Any deviation between the actual and desired positions generates an error signal, which the controller compensates for through proportional and integral actions, thereby ensuring precise finger positioning and stable grasping performance.

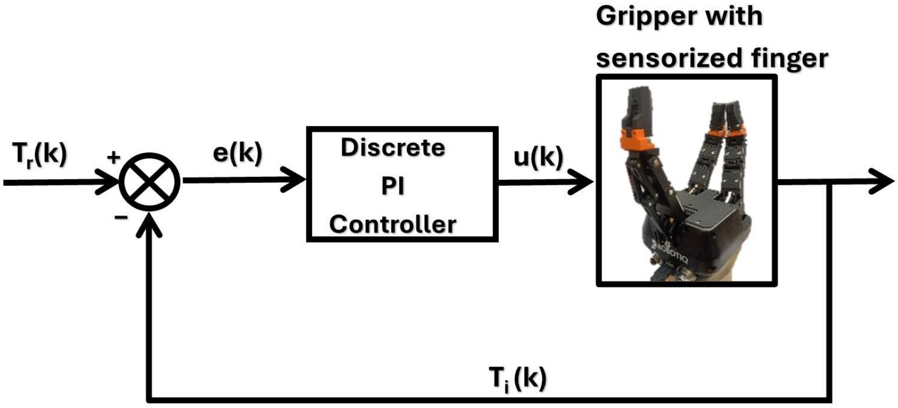


Figure 4.8 Control scheme based on sensorized fingers

The discrete-time implementation of the PI controller can be formulated as:

$$u[k] = u[k-1] + K_p(e[k] - e[k-1]) + K_i e[k]\tau_s \tag{4.1}$$

where the variables and parameters are defined as follows:

- $u[k]$is the control output at discrete time step k (discrete). This is the control signal used to adjust the finger position.

- $e[k]$ is the error signal at discrete time step k. It is defined as the difference between the reference signal $T_r[k]$ and the measured tactile indicator $T_i[k]$:

$$e[k] = T_r[k] - T_i[k]$$

- $T_i[k]$ is the tactile indicator computed as the sum of all voltages acquired from taxel_i, with $i = 1, \ldots, 12$.
- K_p is the proportional gain. This parameter determines the reaction of the controller to the current error.
- K_i is the integral gain. This parameter determines the reaction based on the accumulation of past errors. It helps eliminate steady-state errors.
- τ_s is the sampling time used to discretize the controller.

The PI controller processes the error signal at each sampling step to compute the corresponding control output, which is written into the gripper's actuator register to regulate finger positioning. This operation is iteratively executed in real time, forming a closed-loop feedback system in which the actual finger positions are continuously compared with the desired setpoints. By dynamically adjusting the control input based on the evolving error, the loop ensures that the gripper maintains the commanded position with high accuracy, stability, and responsiveness during manipulation tasks.
Figure 4.9 illustrates the experimental performance of the Proportional-Integral (PI) control system, configured with parameters $K_p = 1.0$, $K_i = 0.06$ and $\tau_s = 0.002\,\text{s}$. The first subplot shows the reference signal, set to $1.5\,\text{V}$, which defines the target tactile indicator value for the control system. The second subplot presents the measured tactile indicator during the experiment, representing the system's actual output in response to the control input and enabling a direct comparison with the reference.

At the start of the experiment, the tactile indicator remains at zero, since the fingertip has not yet made contact with an object. During this phase, the percentage error, displayed in the third subplot, is at its maximum, prompting the gripper to begin closing. Upon contact, the tactile indicator rises sharply to approximately $1.65\,\text{V}$ and subsequently converges to the reference value within about $0.33\,\text{s}$. This transient response demonstrates the controller's ability to quickly reduce the error. The percentage error signal confirms this, approaching zero as the system stabilizes.

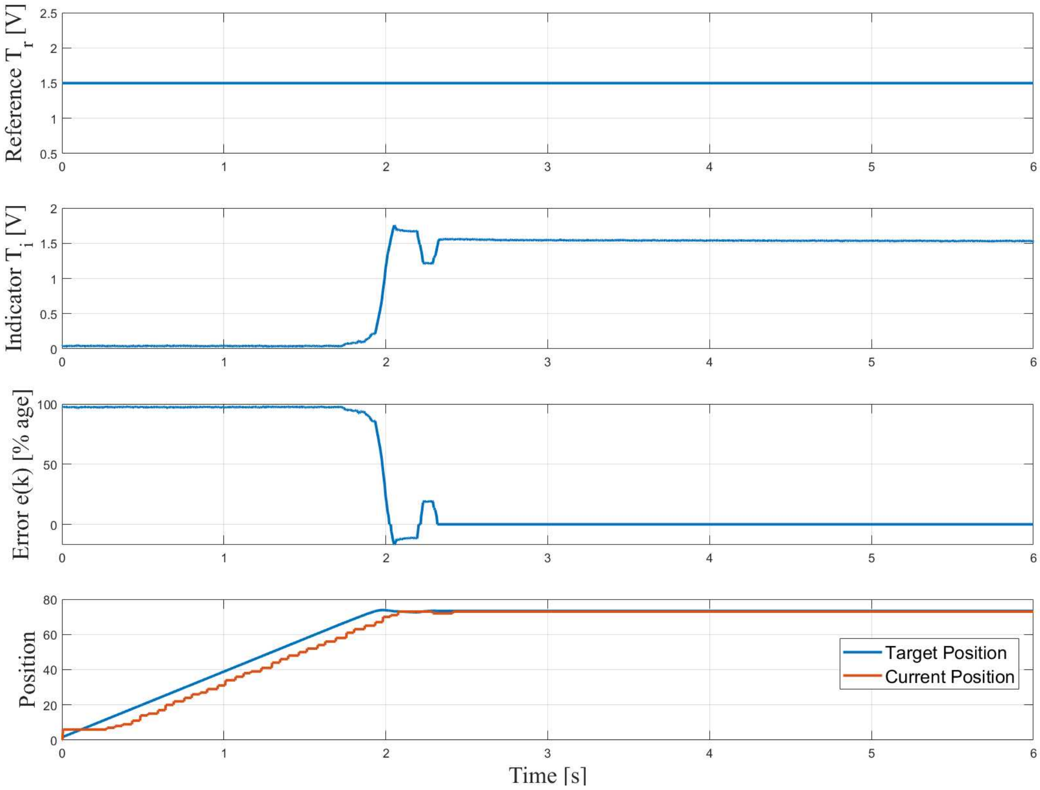


Figure 4.9 Experimental validation for PI control of real gripper fingers

The fourth subplot reports the gripper finger position throughout the experiment. A close alignment is observed between the target position and the measured position, with convergence achieved within approximately 2 s. This alignment indicates that the PI controller effectively regulates the actuators, ensuring accurate finger positioning.

Together, these results confirm the robustness and reliability of the proposed PI control strategy. The system not only stabilizes the tactile indicator at the desired reference value but also achieves consistent finger positioning, demonstrating precise force regulation during grasping. This control mode has been validated across multiple test objects, as shown in Fig. 4.1, confirming its effectiveness for practical robotic manipulation tasks.

4.2.2 Human Robot Interaction

In the field of human-robot interaction (HRI), the incorporation of advanced sensory feedback is critical for enhancing both the operational performance of robotic systems and the overall safety and comfort of human collaborators. Within this context, the tactile sensing functionality of the proposed multi-modal sensor serves as a key enabler, allowing the robotic gripper to perceive and interpret contact forces in real time. This capability empowers the system to adapt its response dynamically to external forces applied by objects or human operators, thereby facilitating safe, intuitive, and reliable manipulation during collaborative tasks.

The centroid estimation of the force distribution represents a key functionality of the tactile sensor integrated into the robotic gripper. By calculating the centroid, the system obtains critical information regarding how contact forces are spatially distributed across the gripper's fingers during manipulation. This information enables the robotic controller to assess whether the applied forces are balanced or uneven, and to adapt the grip strategy accordingly. As a result, the gripper can optimize its contact with the object, ensuring enhanced stability, reducing the likelihood of slippage, and maintaining precise control during interactions with a wide range of objects.

The method proposed in Caccavale et al. (2023) is employed to calculate the **centroid coordinates** of the tactile map, which serve as an estimation of the contact force application point. During a grasping task, the centroid coordinates initially correspond to the equilibrium position established by the contact between the sensorized fingertip and the object. If an additional external force is subsequently applied, the centroid shifts in the direction of the applied force. This displacement provides valuable information regarding the interaction dynamics and can be utilized to implement different control modalities, such as adaptive release or reinforcement of the grip, based on the measured centroid variation. The centroid coordinates are formally estimated as:

$$C_x = \frac{\Sigma_{i=1}^{12} v_i x_i}{\Sigma_{i=1}^{12} v_i}, \qquad C_y = \frac{\Sigma_{i=1}^{12} v_i y_i}{\Sigma_{i=1}^{12} v_i} \tag{4.2}$$

where:

- C_x is the $x-$coordinate of the centroid (in mm)
- C_y is the $y-$coordinate of the centroid (in mm)
- x_i is the physical $x-$position of $i-$th taxel (in mm)
- y_i is the physical $y-$position of $i-$th taxel (in mm)
- v_i is the voltage variation measured by the $i-$th taxel

The centroid coordinates are continuously estimated by combining the physical layout of the sensor, illustrated in Fig. 4.10, with the corresponding voltage measurements acquired

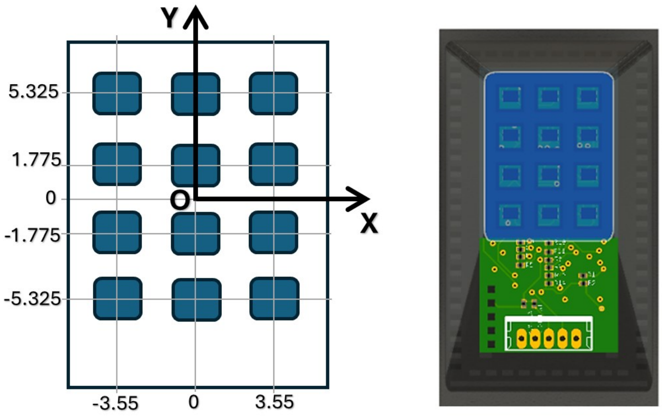


Figure 4.10 Physical positions of taxels with respect to sensor reference frame (left) and with respect to the whole fingertip (right).

from the taxels. As shown in Fig. 4.10, the tactile matrix consists of a 4×3 grid of sensing elements, providing a spatial resolution of 3.55 mm. This configuration enables a precise mapping of the contact area, allowing the centroid calculation to accurately represent the location and distribution of forces across the sensing surface.

The estimated centroid is utilized to implement an intuitive human-robot interaction (HRI) strategy. Once an object has been securely grasped, any external action applied by the user induces a measurable shift in the centroid coordinates relative to the reference frame of gripper. This shift serves as an indicator of the direction and magnitude of the applied force. In the proposed implementation, the system is programmed to release the object if an external force is exerted along the $y-$axis, specifically when the displacement of centroid exceeds 0.3 mm in that direction. Conversely, if the user applies a force along the $x-$axis, resulting in a centroid shift on that axis, the gripper maintains its hold on the object.

Figure 4.11 illustrates a sequence of frames with two different test objects. In the first phase, an external interaction is applied along the $x-$axis, where the gripper continues to retain the object. In the second phase, a force is applied along the $y-$axis, causing the centroid displacement to exceed the defined threshold and triggering the release of the object. This interaction paradigm demonstrates how centroid-based sensing can be leveraged to achieve safe, responsive, and user-friendly collaboration in manipulation tasks.

The plots shown in Fig. 4.12 provide experimental validation of the proposed human–robot interaction strategy. The first plot reports the tactile voltage outputs obtained from the multi-modal sensor while the gripper engages with the object under Proportional-Integral (PI) control. Each curve corresponds to the response of an individual taxel, collectively reflecting

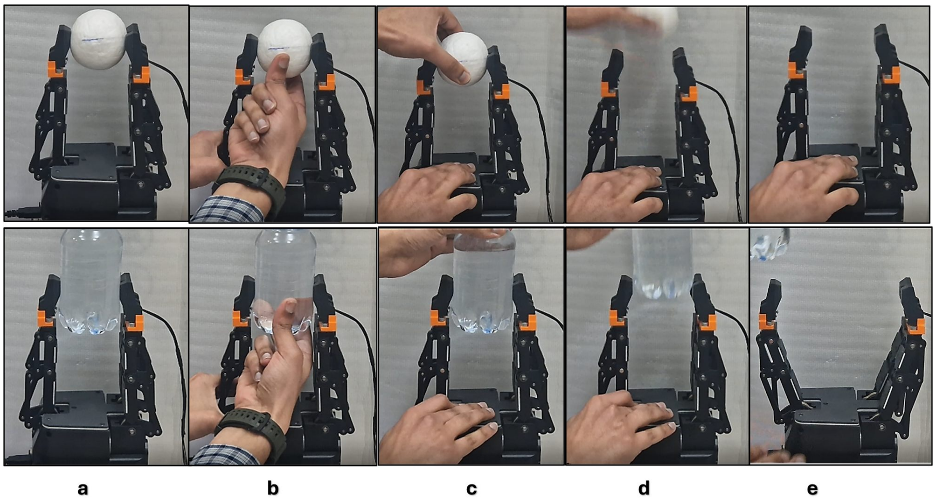


Figure 4.11 From grasp to release: (a) object grasping, (b) pulling along $x-$axis, (c) transitioning to $y-$axis, (d) object releasing along y direction.

the contact distribution across the sensing surface. The second plot illustrates the C_x centroid coordinate, both before and after the object is grasped. Vertical red markers indicate the time intervals when external push or pull forces are applied along the $x-$axis. The centroid trajectory confirms that the gripper maintains a stable hold during these perturbations. The third plot depicts the C_y centroid coordinate, with two red markers denoting the instants at which external forces along the $y-$axis are applied. In this case, a significant lateral displacement of the centroid is observed, which triggers the programmed release of the object. Together, these results demonstrate that centroid-based sensing provides an effective means of enabling intuitive and responsive interaction between the user and the robotic gripper.

Finally, the fourth plot in Fig. 4.12 illustrates the evolution of the gripper finger positions during the experiment. Notably, around 28 s, the position curve exhibits a sharp drop, corresponding to the fingers moving to their fully open state. This behavior reflects the release action triggered by the centroid-based HRI strategy. Taken together, the four plots comprehensively validate the effectiveness of the proposed human–robot interaction algorithm, demonstrating the system's ability to reliably distinguish interaction directions and execute the appropriate response, either maintaining a stable grasp or releasing the object in a controlled manner.

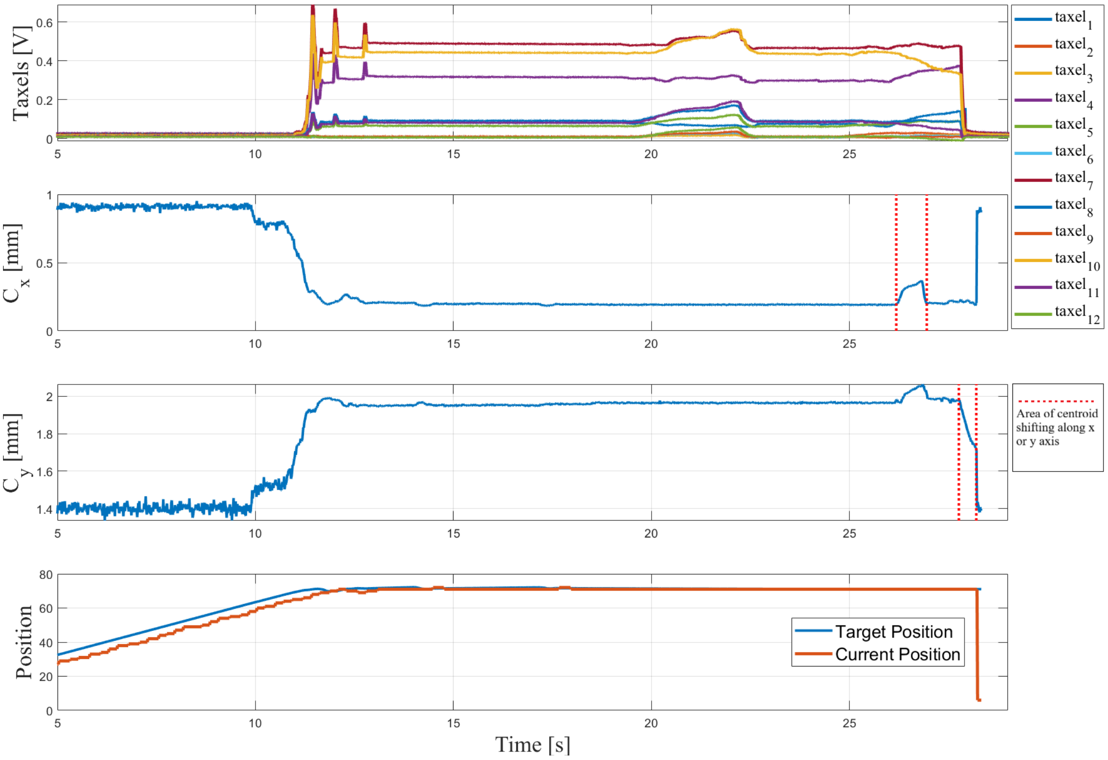


Figure 4.12 Gripper performance during human-robot interaction

4.3 Integration with 4-Finger Developed Gripper

To validate the proposed design, a fully functional physical prototype of the robotic gripper was assembled and experimentally tested. The prototype, illustrated in Fig. 4.13, features a compact chassis that houses two motors and the internal mechanism responsible for dynamically adjusting the grasp configuration. The gripper comprises four fingers, two of which are equipped with multi-modal fingertip sensors that provide real-time tactile, inertial, and proximity feedback. This arrangement ensures that in every grasp configuration, at least one finger from each opposing pair is sensorized, enabling continuous perception and monitoring of the contact conditions during manipulation.

The prototype was mounted on a UR5 robotic manipulator to evaluate both its mechanical integration and its functional performance in practical use cases, particularly within agri-food manipulation scenarios, as proposed in this study. A comprehensive set of grasping experiments was performed, as shown in Fig. 4.15, covering all possible grasp configurations. These experiments demonstrated the gripper's adaptability and stability when handling objects of varying shapes, sizes, and mechanical properties.

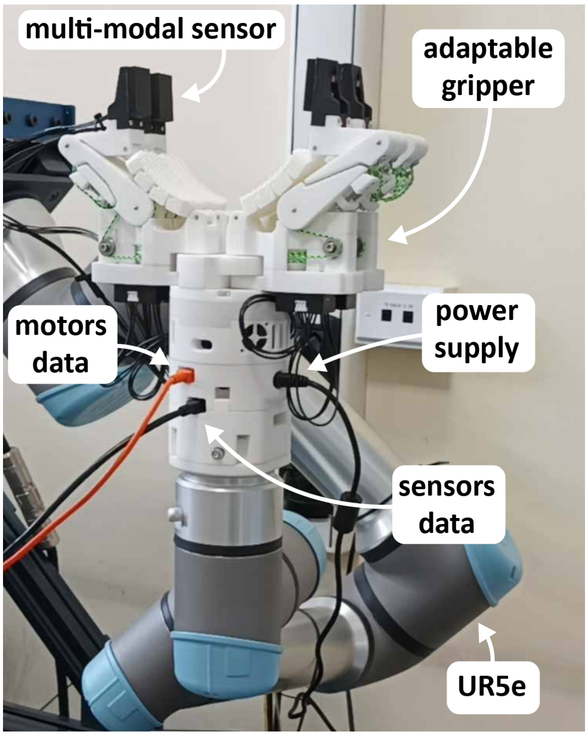


Figure 4.13 Experimental setup to perform agri-food grasping tests

4.3.1 Control Scheme and Experimentation

Consistent with the virtual model, the configuration transitions were executed using position control, while each finger operated under a closed-loop velocity control scheme leveraging feedback from the integrated tactile sensors (Fig. 4.14). This control approach ensured a compliant and damage-free interaction with delicate items. The complete experimental validation, including grasping sequences and interaction tests, is presented in Fig 4.15, which highlights the effectiveness, reliability, and responsiveness of the developed system in real-world manipulation tasks.
To evaluate the gripping capability of the proposed device, a series of grasping experiments were conducted on objects of varying shapes, sizes, and weights, utilizing all possible configurations available in the prototype. The demonstrated research primarily focused on pinch grasps, which emphasize the use of the fingertip-integrated with multi-modal sensors. The grasping process was performed under the closed-loop control architecture illustrated in Fig. 4.14.

The control model considers a single sensorized finger actuated by a Dynamixel servomotor (XC330-T288-T). The desired velocity (v^d) is generated by a proportional controller

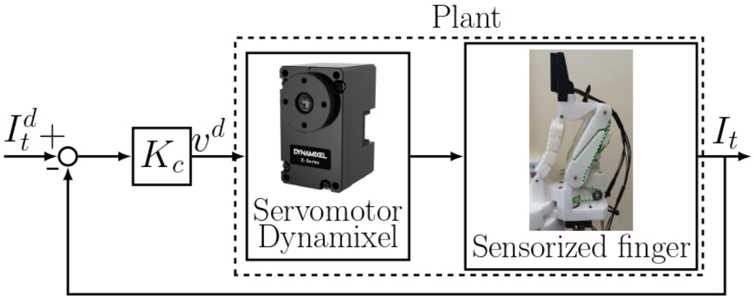


Figure 4.14 Scheme of the closed-loop control based on tactile sensory feedback

with gain K_c, which continuously adjusts the actuator command based on the error between the desired tactile indicator (I_t^d) and the measured tactile indicator (I_t). The tactile indicator I_t represents the aggregate contact response of the fingertip and is computed as the sum of voltages from the tactile array:

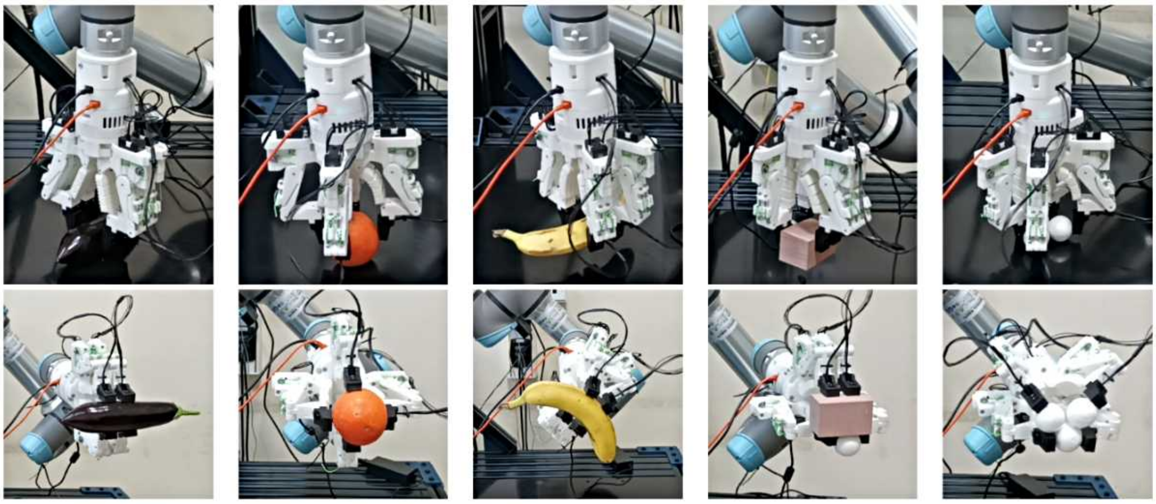
$$I_t = \sum_{i=1}^{N} v_i, \quad \text{where } N = 12 \tag{4.3}$$

The reference signal I_t^d follows a step profile, becoming positive when the robot initiates the grasping sequence. This ensures that control action is only activated once the gripper is positioned for contact. Importantly, I_t serves as a synthetic indicator of contact pressure between the fingertip and the object surface, directly reflecting the delicacy and stability of the grasp. Through this approach, the controller effectively regulates finger velocity to achieve gentle yet firm contact, preventing damage to fragile objects while maintaining a stable hold during manipulation.

The experimental tests were performed with the gripper mounted on the robotic arm to evaluate its grasping performance across multiple configurations. The testing procedure was carried out as follows:

1. The robot approaches the target object.
2. The gripper automatically switches to the most suitable configuration based on the geometry of the object i.e., parallel, spherical, and tripod for eggplant, orange, and banana, respectively; multi-pick for simultaneous grasping of a parallelepiped and a sphere; and scoop for grasping multiple spherical objects.
3. The gripper closes under the closed-loop control scheme to ensure stable grasping while preventing damage to the object.

4. The robot lifts and reorients the object, first positioning it in front of the camera and then rotating it into an upside-down vertical orientation (see supplementary material) to assess grasp stability under different gravitational conditions.



(a) parallel: eggplant (b) spherical: orange (c) tripod: banana (d) multi-pick: box & sphere (e) scoop: 3 spheres

Figure 4.15 Physical prototype of the gripper during the grasping of objects of various shapes and sizes, utilizing the five different possible grasp configurations.

The signals acquired from the integrated sensors were continuously recorded during each grasping operation to analyze the dynamic behavior of the system and assess the sensor–controller interaction. As a representative example, Fig. 4.16 presents the experimental data corresponding to the grasping of an eggplant, chosen due to its irregular shape, soft surface, and moderate weight, which make it ideal for evaluating compliant and adaptive grasping performance.

The figure reports the evolution of the tactile indicator (I_t) with respect to its desired reference value (I_t^d), along with the three components of linear acceleration (a_x, a_y, a_z) and angular velocity (ω_x, ω_y, ω_z) measured by the IMU in its local reference frame, as defined in the manufacturer's datasheet. The control signal (v^d) applied to the actuator is derived from the proportional control law, using a controller gain of $K_c = 27$, which determines the movement of finger based on the instantaneous error between the desired and measured tactile responses.

The experimental sequence is divided into five distinct phases, each characterized by a specific control behavior and corresponding sensor response:

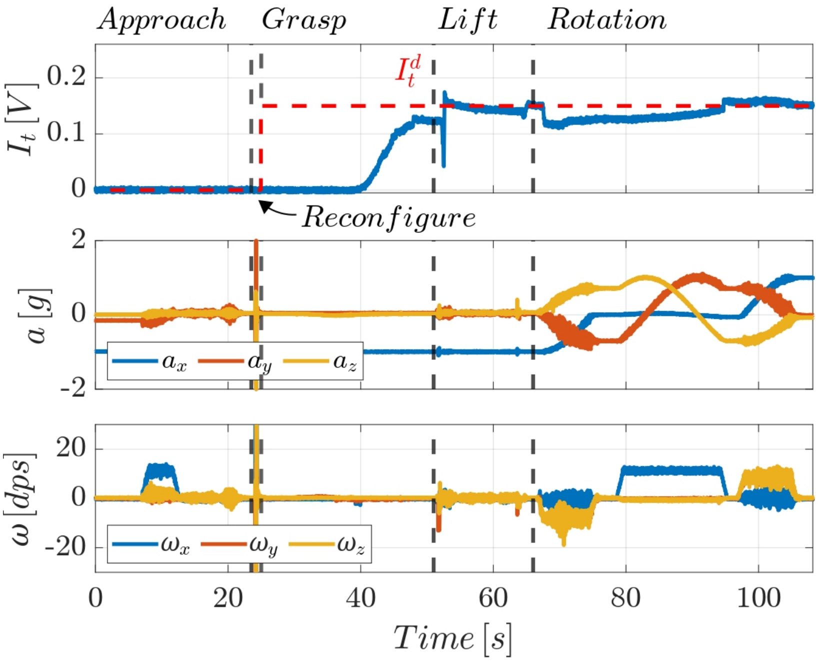


Figure 4.16 Experimental Validation for eggplant grasping test with Kc = 27

- **Approach:** The robot arm moves toward the object, as reflected by fluctuations in both the angular velocity (ω) and the linear acceleration (a). The tactile indicator remains near zero, confirming that no physical contact has yet occurred.

- **Reconfigure:** At approximately $t = 20\,\mathrm{s}$, the palm motors are activated to adjust the grip configuration according to the geometry of the object. This phase is marked by sharp peaks in a and ω, indicating transient mechanical adjustments as the fingers reposition.

- **Grasp:** Once the gripper makes contact with the object, the tactile indicator (I_t) increases rapidly, signaling the start of contact. The controller then regulates the actuator velocity until the error between I_t and I_t^d becomes negligible. The smooth convergence of these signals demonstrates the system's ability to achieve stable and compliant contact without overshoot or oscillation.

- **Lift:** In this phase, the robot lifts the grasped object from the surface. Small fluctuations in I_t indicate the compensatory control action of the system, which counteracts the influence of gravity and inertial forces to maintain a consistent contact pressure.

- **Rotation:** Finally, the robot reorients the object to evaluate grasp stability under changing gravitational loads. The larger movements observed in the IMU signals correspond to the rotational motion of the gripper, while the steady tactile readings confirm that the gripper maintains a firm and stable grasp throughout the maneuver.

The synchronized tactile and inertial data reveal that the multi-modal sensing system effectively supports adaptive control during all stages of manipulation. The average sampling frequency of both the sensors and the control loop is 250 Hz, providing high temporal resolution and ensuring reliable real-time feedback for dynamic grasp regulation.

The selection of the control strategy adopted in two grippers is intrinsically linked to the actuation capabilities and control interfaces of the robotic platforms under consideration. For the Robotiq 3-Finger Adaptive Gripper, the low-level controller does not provide direct access to velocity or torque control; instead, the gripper is operated through position commands with limited exposure to internal motor dynamics. Under these constraints, a Proportional-Integral (PI) controller was employed to regulate finger position while compensating for steady-state errors caused by friction, backlash, and load variability. The integral term proved essential for eliminating residual positioning errors and achieving consistent contact forces during grasping of objects with different mechanical properties. Experimental tuning confirmed that this approach yields stable and repeatable performance within the operational limits of the Robotiq gripper.

In contrast, the four-finger robotic gripper actuated by Dynamixel motors offers extensive access to low-level control modes, including position, velocity, and current (torque) control, as specified in the manufacturer's documentation. These actuators embed high-resolution encoders and internal feedback loops that already ensure accurate position tracking and disturbance rejection. Consequently, introducing an external integral action at the supervisory control level was unnecessary and, in preliminary evaluations, resulted in slower transient responses and increased overshoot. For this reason, a pure Proportional control scheme was adopted, allowing the system to leverage the embedded control features of the motors while maintaining fast, predictable, and stable behavior. This choice reflects a deliberate hardware-aware control design, rather than a simplification of the control architecture.

With respect to the experimental workflow, the choice of the tactile indicator was motivated by considerations of observability, robustness, and control relevance. The selected indicator, derived from the magnitude and spatial distribution of the tactile response, provides a reliable estimate of contact onset and interaction direction. It was expected to exhibit a monotonic relationship with contact intensity and a consistent spatial variation under tangential forces, making it suitable for triggering control actions such as grasp stabilization, release, or compliance modulation. Experimental results confirmed that this indicator remains stable across different object geometries and interaction conditions, thereby justifying its integration within the control framework.

Overall, the adoption of different control strategies across the two gripper platforms is the result of a system-level and methodology-driven approach, in which control design is tailored to the sensing and actuation capabilities of each robotic system. This approach enhances control performance, transparency, and reproducibility, while ensuring that the experimental outcomes accurately reflect the strengths and limitations of the underlying hardware.

Chapter 5

Multi-Modal Sensor Fusion in a Biomimetic Fingertip for Adaptive Robotic Manipulation

The growing need for intelligent and adaptive robotic manipulation has intensified research into multi-modal sensing technologies that can emulate the rich perception capabilities of the human fingertip. The proposed system presents an enhanced bio-mimetic multi-modal fingertip sensor, designed to replicate both the morphology and multi sensory integration of the human fingertip. Building on our previously developed sensor Fazal et al. (2024), the new version introduces significant advancements in sensor architecture, sensing performance, and system integration. The fingertip module combines optoelectronic tactile sensing, long-range Time-of-Flight (ToF) proximity detection, and a 9-axis Inertial Measurement Unit (IMU) within an ultra-compact form factor measuring 12 × 15 mm, with a 3 mm curved upper surface for anatomical conformity, resulting in total dimensions of 12 × 18 mm. The new sensor incorporates PWM-driven optoelectronic elements that enable dynamic control of light intensity, significantly enhancing sensitivity, reducing power consumption, and improving signal-to-noise ratio compared to the earlier version, where LEDs were driven in series. The upgraded 9-axis IMU extends the sensing capabilities by adding magnetometer-based orientation tracking, improving accuracy and robustness under dynamic motion. A new long-range, high-frequency ToF module replaces the previous short-range sensor, enabling real-time pre-touch distance estimation and high-speed depth perception. The fingertip module is designed specifically for seamless integration with the anthropomorphic hand designed by SEEDs robotic, offering real-time tactile, proximity, and inertial data fusion for adaptive grasping and contact-aware manipulation. Experimental evaluations demonstrate noise reduction, improved spatial resolution, and enhanced accuracy in object distance and contact estimation. The compact size, biomimetic geometry, and multi-sensor fusion capabilities make the proposed fingertip sensor a promising solution for next-generation human-robot interaction, delicate object handling, and soft robotic systems.

Achieving human-level dexterity in robotic manipulation requires not only precise actuation but also rich perceptual feedback at the point of contact. While vision provides global context, the fingertip intimately interacts with the object and must provide multi dimensional information about contact, motion, and geometry. Therefore, multi-modal fingertip sensing, the integration of tactile, proximity, and inertial modalities, has become a crucial frontier in robotics. Tactile sensing enables robots to detect contact forces, texture, slippage, and local geometry, while proximity (or pre-touch) sensing extends awareness to just before contact. In addition, the inertial measurement unit (IMU) provides information about orientation and slippage. By fusing complementary streams of data, these modalities allow a robot to behave more like a human hand, capable of adjusting in real time to object properties, grasp uncertainties, dynamic changes in the environment, and improve robustness to occlusion or poor lighting.

In recent years, robotics has advanced rapidly toward achieving human-like dexterity and adaptability, enabling robots to perform increasingly complex manipulation tasks across industrial, service, and collaborative environments. One of the most demanding challenges in this evolution is the ability to interact intelligently with objects, requiring not only precise actuation but also a deep understanding of the physical interaction between the robot's end effector and its environment. The success of such manipulation tasks critically depends on the quality and diversity of sensory feedback, allowing robots to adapt their behavior dynamically in the presence of uncertainties, variable contact conditions, or delicate materials. To meet these demands, multi-modal fingertip sensors have emerged as a pivotal technology, integrating complementary sensory modalities such as tactile, proximity, and inertial within a compact and bio-mimetic structure. The proposed fingertip sensor, inspired by the geometry and sensory richness of the human fingertip, exemplifies this approach by combining PWM-controlled optoelectronic tactile elements, a long-range Time-of-Flight (ToF) proximity sensor, and a $9-$axis Inertial Measurement Unit (IMU). This integration enables the robot to perceive contact pressure, detect nearby objects before contact, and estimate motion and orientation simultaneously, providing a unified perceptual framework for dexterous manipulation. Incorporating multiple sensing modalities within a compact form factor significantly enhances the functionality, precision, and adaptability of robotic systems while minimizing weight and wiring complexity, an essential advantage for anthropomorphic grippers and other size-constrained platforms. The fusion of tactile, proximity, and inertial data yields richer and more reliable perception, improving grasp stability, contact awareness, and environmental understanding. Moreover, by embedding these sensing functions in a single fingertip-sized module, the system achieves a cost-effective and easily integrable solution that can support a broad range of applications, from industrial and surgical robotics to human-robot interaction and soft manipulation.

The primary scientific and technological contributions of the extended research are outlined below:

- Design of a fingertip-shaped multi-modal sensor with optoelectronic, ToF, and 9-axis IMU sensing, tailored for the anthropomorphic robotic hand.
- Implementation of PWM-driven optoelectronic tactile sensing, allowing dynamic illumination control and reduced power consumption.
- Integration of a long-range, high-frequency ToF sensor enabling accurate pre-touch detection and extended depth sensing.
- Enhanced multi-sensor fusion capability through improved IMU performance and real-time data synchronization.

5.1 Sensor Technology

The proposed multi-modal fingertip sensor combines tactile, proximity, and inertial sensing within a compact fingertip module. The system is designed to perceive contact, pre-contact, and motion states simultaneously, providing comprehensive information about object interaction and finger dynamics. The sensor architecture is built around a multilayer PCB governed by the micro-controller, which manages data acquisition, synchronization, and communication from all sensing elements. A 3×3 array of tactile optoelectronic elements (taxels) is distributed beneath a deformable silicone layer that emulates the soft tissue of a human fingertip. A Time-of-Flight (ToF) module is mounted on the backside of the sensor, enabling pre-touch distance estimation without interfering with the tactile interface. A $9-$axis IMU provides real-time motion, orientation, and acceleration data. The integration of these sensing modalities allows the fingertip to perceive deformation during contact, estimate object proximity before contact, and measure dynamic motion, resulting in an intelligent sensing platform suitable for anthropomorphic grippers and dexterous robotic manipulation.

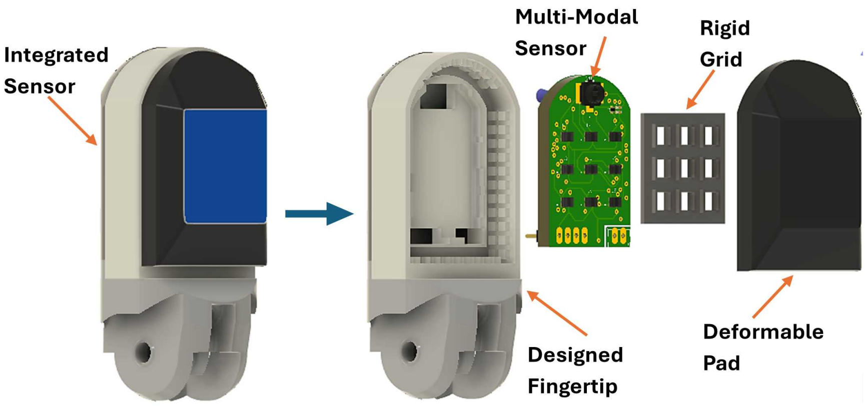


Figure 5.1 CAD of an integrated sensor (left) and exploded view with all components (right).

5.1.1 Sensing Modalities

The tactile sensing subsystem operates on the principle of photo-reflective transduction. An array of optoelectronic elements, each composed of an infrared Light-Emitting Diode (LED) and a phototransistor, is arranged beneath a deformable silicone pad that serves as the contact surface. When an external force is applied to the pad, local deformation alters the reflection path and intensity of the emitted light. The phototransistor detects these variations in the intensity of reflected light producing analog voltage signals proportional to the magnitude and distribution of contact pressure. For this sensor, the measured voltages signals are in $millivolts$.

To improve signal quality and energy efficiency, the infrared LEDs are driven by Pulse-Width Modulated (PWM) control signals, allowing dynamic adjustment of light intensity based on contact conditions. This approach enhances sensitivity, reduces power consumption, and minimizes optical noise compared to fixed-current illumination schemes. The resulting set of analog signals from the tactile array forms a spatial tactile map, enabling the estimation of contact forces, pressure distribution, and object geometry during manipulation.

The proximity sensing subsystem is based on the Time-of-Flight (ToF) principle, where the sensor emits short infrared light pulses and measures the time taken for the reflected light to return. This time delay is directly proportional to the distance between the fingertip and the target object. The technique allows non-contact distance measurement with $millimeter$ level precision, independent of surface color or texture. Integrated at the backside of the fingertip, the ToF sensor provides pre-touch awareness, enabling the robotic system to detect approaching objects, adjust its grasp trajectory, and avoid collisions before physical contact occurs.

The inertial sensing subsystem measures linear acceleration, angular velocity, and magnetic orientation along three orthogonal axes. These data streams are fused internally to determine the absolute orientation and dynamic motion of the fingertip in real time. The inertial information supports pose estimation, slippage detection, and contact dynamics monitoring during grasping and manipulation. By continuously tracking both static and dynamic states, the inertial subsystem enhances overall perceptual stability and ensures smooth transitions between approach, contact, and post-contact phases of interaction.The integrated IMU also has the capability to measure the temperature.

5.1.2 Electronics Section

The electronic design of the proposed multi-modal fingertip sensor has been optimized for compactness, low power consumption, and synchronized acquisition of tactile, proximity, and inertial data. The entire system is powered through a 3.3 V supply provided directly from the USB interface, eliminating the need for external voltage regulation and ensuring stable operation across all subsystems. The electronic architecture centers around an ESP32-PICO-V3 micro-controller, which manages signal acquisition, synchronization, and data communication with the host computer. The complete layout, including power distribution and signal flow, is illustrated in Fig. 5.3.

The electronic system of the proposed multi-modal fingertip sensor is entirely powered through a 3.3 V supply provided via USB, ensuring compactness and low power consumption while maintaining stable operation across all sensing subsystems. At its core, the sensor employs an ESP32-PICO-V3 microcontroller (Espressif Systems), featuring a dual-core $32-$bit processor (up to 240 MHz) with integrated Wi-Fi and Bluetooth connectivity, multiple $12-$bit ADC channels, and hardware support for PWM, I2C, SPI, and UART communication. The optoelectronic tactile array consists of nine photo-reflective elements (infrared LED and phototransistor pairs, NJL5912R; New Japan Radio Co.) arranged in a 3×3 configuration beneath a deformable silicone layer, reported in Fig. 5.2. In the earliest version of the tactile sensor De Maria et al. (2012), the infrared LEDs were powered directly through a constant voltage source. In the subsequent design Fazal et al. (2024), the LEDs were connected in series and driven by a constant current source supplying approximately 1 mA. This configuration exploited the high sensitivity of the optoelectronic component (NJL5912R) while reducing overall power consumption and improving signal uniformity across the array. In the proposed design, each LED is individually driven using Pulse-Width Modulated

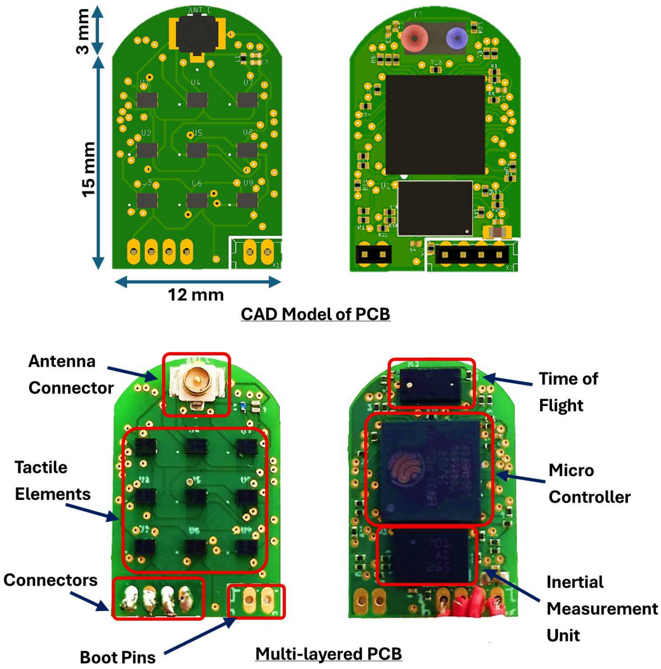


Figure 5.2 Multi-layered PCB with Front View on Left Side and Bottom View on Right Side

(PWM) signals generated by the ESP32 micro-controller, enabling dynamic control of illumination intensity. This approach enhances measurement sensitivity, provides adaptive light modulation under varying surface and contact conditions, and further decreases power consumption while maintaining a high signal-to-noise ratio.

The fingertip also integrates a 9−axis Inertial Measurement Unit (BNO055, manufactured by Bosch) that combines a three-axis accelerometer, gyroscope, and magnetometer with an embedded ARM Cortex-M0 processor running Bosch's sensor-fusion algorithm to provide drift-free orientation, acceleration, and angular velocity data. It is compatible with various communication protocols, including Serial, I2C, and SPI. however, the proposed sensor employs the I2C protocol.

Time-of-Flight (ToF) proximity sensor (VL53L4CD; STMicroelectronics), mounted on the backside of the sensor to perform precise pre-touch distance measurement by calculating the light-pulse round-trip time with millimeter-level accuracy. All digital sensors communicate with the ESP32 through the I2C bus, while the tactile subsystem uses the controller's ADC interface for real-time analog acquisition. The processed multi-modal data are transmitted to a host computer through UART serial communication or optionally via the onboard antenna connector for wireless operation. A ROS node running on the host PC handles data reception, timestamping, and topic publication for higher-level visualization and sensor fusion. The system operates at typical sampling frequencies of 500 Hz for tactile sensing, 100 Hz for inertial sensing, and 50 – 60 Hz for proximity sensing, offering a balanced trade-off between responsiveness and stability for real-time robotic manipulation tasks.

This compact module offers accessibility through a 4—way connector (Product code 53047-04, manufactured by Samtec) for communication with microcontroller. Further details about the ToF module connections and applications can be found by the interested readers in Cirillo et al. (2021b). The complete block diagram of all electronic components with connections is summarized in Fig. 5.3

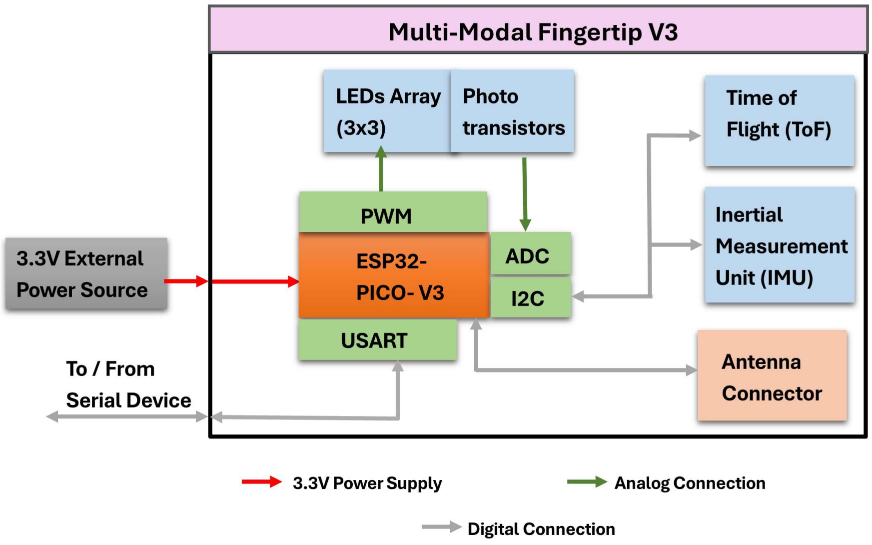


Figure 5.3 Block diagram of key electronic modules of fingertip

5.1.3 Mechanical Parts

The mechanical design of the proposed multi-modal fingertip sensor represents a complete redesign of the previous generation, with the objective of achieving a fully integrated, lightweight, and modular fingertip suitable for the SEEDs anthropomorphic robotic gripper. Unlike the earlier configuration, which relied on an aluminum mounting base, the present version is entirely 3D-printed, excluding the electronic PCB, thereby improving manufacturability and enabling rapid prototyping. The fingertip structure integrates the sensor directly into the last phalange of the robotic finger, allowing it to be easily attached or detached through the upper motion joint. This modularity supports efficient maintenance, replacement, and future sensor upgrades. The fingertip also includes a tendon routing cavity, specifically designed to ensure smooth motion transmission to the distal joint without interfering with the internal sensor layout.

A fundamental component of the fingertip assembly is the deformable silicone pad, which acts as the tactile interface between the environment and the internal optoelectronic sensing array. Depending on the application, the pad geometry can be modified to achieve different sensing characteristics. For instance, as detailed in Palli and Pirozzi (2019a), applications involving objects smaller than the pad, where the focus is on recognizing object features for manipulation (e.g., the shape of a grasped wire), a flat-surfaced pad may be utilized. Alternatively, for manipulation tasks involving torque and contact force data, a dome-shaped deformable pad can be used to reconstruct these parameters through a trained neural network applied to tactile map signals Costanzo et al. (2019a).However, in the current design, a flat surface pad has been adopted to facilitate uniform contact and accurate reconstruction of the geometry and characteristics of the surface of the object. The inner walls of the pad are black to ensure optical isolation between adjacent tactile cells, while the areas directly above each optoelectronic element are white, enhancing reflectivity and improving sensitivity, as shown in Fig. 5.4. The pad is fabricated using silicone elastomer, chosen for its low hysteresis, high elasticity, and excellent light diffusion properties compared to other deformable materials.

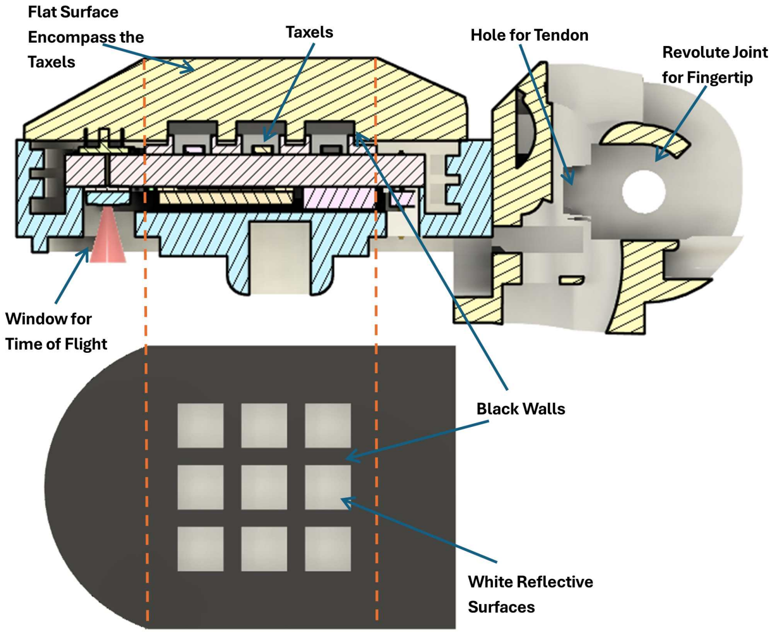


Figure 5.4 Cross section view for fingertip shell and deformable silicone pad characteristics.

Between the deformable silicone pad and the main PCB lies the rigid optical grid, whose primary function is to maintain the optoelectronic sensing elements within their monotonic operating range. This structural layer prevents non-linear signal behavior that occurs when the distance between the reflective surface and the photo-reflective element falls below approximately 300μm Costanzo et al. (2019b). In the present design, the grid geometry ensures a minimum spacing of 500μm between the reflective inner surface of the pad and the photo-reflectors, thereby guaranteeing stable and repeatable sensor responses. The upper side of the grid interfaces directly with the dark inner walls of the silicone pad, improving the mechanical coupling and structural integrity of the tactile interface. The grid is fabricated using high-resolution 3D printing in black ABS plastic, with a dimensional precision of approximately $\pm 100\mu$m. The black material was chosen to suppress lateral optical reflections and enhance contrast between adjacent sensing cells. The grid is bonded to the deformable pad on one side and to the PCB on the other (as shown in Fig. 5.1) using a thin layer of cyanoacrylate adhesive, ensuring precise alignment and robust assembly during repeated mechanical loading.

The main fingertip shell serves as both the mechanical enclosure and the support structure for the multi-modal PCB, housing the tactile array, IMU, and ToF modules in a compact volume. The ToF sensor is positioned on the backside of the fingertip, facing outward to capture pre-touch distance information without obstructing the tactile interface. The shell also provides mechanical isolation for the internal electronics and includes openings for the communication connector and tendon attachment. All parts are fabricated using high-resolution 3D printing technology, ensuring geometric accuracy, low weight, and reproducibility. (see Fig. 5.1).

5.2 Software Design

The software architecture of the proposed multi-modal fingertip sensor is organized into two functional layers:

1. The embedded firmware, running on the ESP32-PICO-V3 microcontroller, responsible for real-time data acquisition, synchronization, and transmission of tactile, inertial, and proximity data.
2. The host-side ROS (Robot Operating System) node, which handles data reception, decoding, time synchronization, and publishing of multi-modal information to higher-level robotic control and visualization modules.

This layered design ensures modularity, real-time operation, and straightforward integration with robotic systems based on the ROS ecosystem.

5.2.1 Microcontroller Firmware

The firmware executed on the ESP32-PICO-V3 manages data acquisition from the three sensing subsystems: the tactile array, inertial measurement unit (IMU), and Time-of-Flight (ToF) proximity sensor. The controller generates PWM signals to drive the infrared LEDs of the tactile array, acquires analog readings from the photo transistors through its $12-$bit ADC channels, and communicates with the IMU and ToF sensors via the I2C protocol. The collected data are processed, timestamped, and transmitted to the host computer using UART serial communication. The firmware is structured using a task-based architecture, enabling

Algorithm 3 Microcontroller Firmware

```
procedure MAIN
    Initialize system
    Configure PWM for tactile LEDs
    Configure ADC channels for tactile taxels (3×3 array)
    Configure I2C for ToF and IMU communication
    Configure UART serial interface
    while true do
        $rxData \leftarrow$ read $char$ from serial
        if $rxData ==$ a then
            for $k \leftarrow 1$ to 9 do
                Set PWM duty cycle for LED[$k$]
                $tactData \leftarrow$ read analog voltage (mV) from channel $k$
                Send $tactData$ over serial
            end for
        else if $rxData ==$ b then
            Send $SensorID$ and firmware version
        else if $rxData ==$ c then
            $tofData \leftarrow$ read value from ToF via I2C
            Send $tofData$ over serial
        else if $rxData ==$ d then
            $imuData \leftarrow$ read orientation, acceleration, and gyro data from IMU via I2C
            Send $imuData$ over serial
        end if
        Delay for next sampling interval
    end while
end procedure
```

independent sampling rates for each sensing modality. The tactile subsystem operates at 500 Hz, providing high temporal resolution for surface deformation mapping; the IMU runs at 100 Hz for accurate orientation and motion estimation; and the ToF sensor operates at 50–60 Hz, offering stable and accurate proximity measurements.

Pseudo-code reported in Algorithm 3 presents the developed firmware showing the operations carried out by the micro-controller. The firmware automatically synchronizes multi rate sensor data using the internal microsecond system clock, ensuring coherent temporal alignment between different modalities. Each data frame is encapsulated into a unified packet containing tactile voltage values (in mV), IMU readings (acceleration, angular velocity, quaternion orientation and temperature), and ToF range measurements (in mm), along with a CRC checksum for data integrity verification.

5.2.2 Elaboration Software (PC-Based)

The host-side interface was implemented as a ROS (Robot Operating System) node in `C++`, responsible for real-time acquisition, verification, and publication of the fingertip sensor data. The node establishes a serial connection with the ESP32 micro-controller via `/dev/ttyUSB0` at 1 Mbps, with low-latency mode enabled to minimize transmission delay. After initialization, a handshake command `'b'` is issued to request the sensor ID, confirming link integrity and firmware readiness. The node continuously exchanges command-based queries with the micro-controller: `'a'` for tactile, `'d'` for IMU data, and `'e'` for ToF. Each sensor response consists of a structured packet containing a 2-byte header, payload, and 1-byte `XOR` checksum. The node validates packet integrity, decodes sensor data, and publishes it to dedicated ROS topics. The tactile array (9 × 1 values, in mV) is published as `std_msgs::Int16MultiArray` on `/tactile_data`; the ToF range in (mm) as `std_msgs::Int16` on `/tof_data`; and IMU data (Euler angles, angular velocity, linear acceleration, magnetic field, gravity, and temperature) as `geometry_msgs::Vector3Stamped` and `std_msgs::Float32` messages under the `/imu namespace`.

Algorithm 4 PC-Side ROS Node

```
procedure MAIN
    Initialize ROS node and publishers
    Configure Serial Port ($/dev/ttyUSB0$, 1 Mbps, low_latency)
    Send $'b'$ to request SensorID
    Read and verify SensorID response
    while ROS is running do
        Send $'a'$ to request tactile data
        Read packet and verify XOR checksum
        if packet valid then
            Parse 9 tactile values (mV)
            Publish on $/tactile\_data$ topic
        end if
        Send $'d'$ to request IMU data
        Read packet and verify checksum
        if packet valid then
            Parse orientation, gyro, accel, mag, gravity, temperature
            Publish on $/imu/*$ topics
        end if
        Send $'e'$ to request ToF data
        Read packet and verify checksum
        if packet valid then
            Parse range (mm)
            Publish on $/tof\_data$ topic
        end if
        Delay according to sensor update rates
    end while
end procedure
```

5.3 Integration and Experimental Validation

This phase of research focused on the integration of the compact multi-modal fingertip sensor, developed with PWM-driven optoelectronic technology, into the SEEDs anthropomorphic robotic hand as presented in Fig 5.5. Designed to replicate the geometry and size of a human fingertip, the sensor was seamlessly embedded into the distal phalanges of the hand, achieving full mechanical and electrical integration without compromising dexterity or motion range. The modular design allows each fingertip to operate as an independent sensory unit, facilitating scalable deployment across multiple fingers.

Figure 5.5 Integration of compact fingertip on index finger of SEEDs anthropomorphic Hand

All sensing modules communicate with the onboard ESP32-PICO-V3 microcontroller via the I2C bus, while the tactile array outputs analog signals read through the controller's ADC channels. The synchronized data from all three sensing modalities are processed and transmitted to the host computer through UART or wireless communication, where a dedicated ROS node performs timestamping, decoding, and topic publishing for higher-level perception and control tasks. The system operates at typical sampling rates of 500 Hz for tactile sensing, 100 Hz for inertial data, and $50-60$ Hz for proximity measurements, ensuring responsive and reliable performance for real-time anthropomorphic manipulation and human–robot interaction.

Figure 5.6 shows the time series data from the multi channel tactile array integrated into the fingertip of the anthropomorphic hand. Each colored line represents the output voltage of an individual taxel as the finger repeatedly transitions between touch and untouched positions. During the initial interval $(0-18\text{s})$, all channels remain nearly constant, indicating no external contact and a stable baseline. Between 20s and 35s, distinct voltage rises appear across multiple channels, corresponding to contact events that occur as the finger come in contact and applies pressure on the target surface. The amplitude and timing differences between the channels reflect the spatial pressure distribution across the fingertip. Successive peaks confirm multiple contact cycles, while the rapid return of signals to baseline demonstrates good sensor recovery and repeatability. Overall, the results verify that the tactile array reliably detects the onset of contact, maintains a stable output during sustained load, and exhibits low noise and hysteresis. These responses confirm the suitability of the system for real-time grasp detection and force feedback within the anthropomorphic gripper.

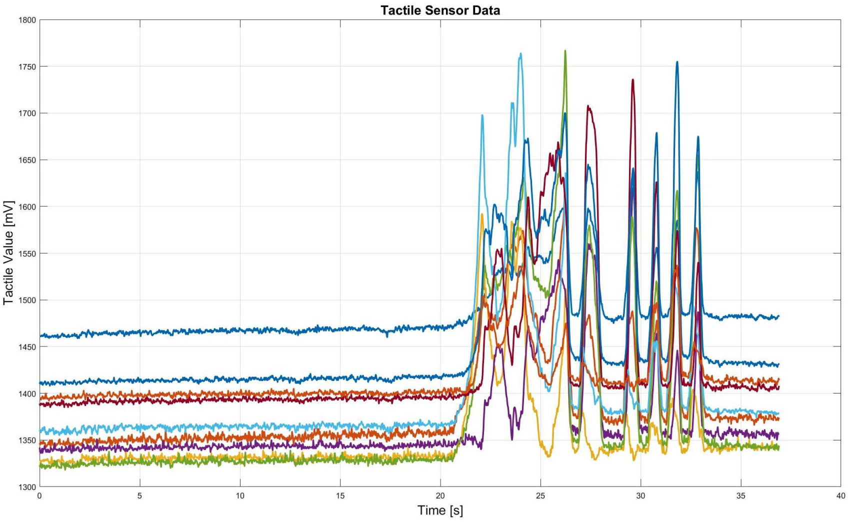


Figure 5.6 Tactile sensor outputs recorded during object grasping

Figures 5.7 and 5.8 illustrate the inertial measurements captured from the IMU mounted on the fingertip during repeated open and close of the anthropomorphic finger. Data include orientation, angular velocity, linear acceleration, accelerometer, gravity, and temperature readings acquired in real time. The orientation traces show clear step-like transitions that correspond to discrete finger movements. The periodic changes in the orientation axes (X,Y,Z) align with each actuation, indicating stable attitude tracking throughout the motion sequence. The angular velocity plot exhibits peaks that coincide with the start and end of each finger movement, confirming accurate detection of rotational dynamics and absence of drift in static intervals. Linear acceleration signals show transient spikes during motion and return to baseline when the finger remains stationary, validating correct gravity compensation and dynamic response.

In Figure 5.8, the accelerometer outputs capture both the gravitational component and motion-induced accelerations. During static phases, one axis remains near $9.81\,\mathrm{m/s^2}$, confirming proper calibration of the sensor. The gravity vector components rotate smoothly and maintain

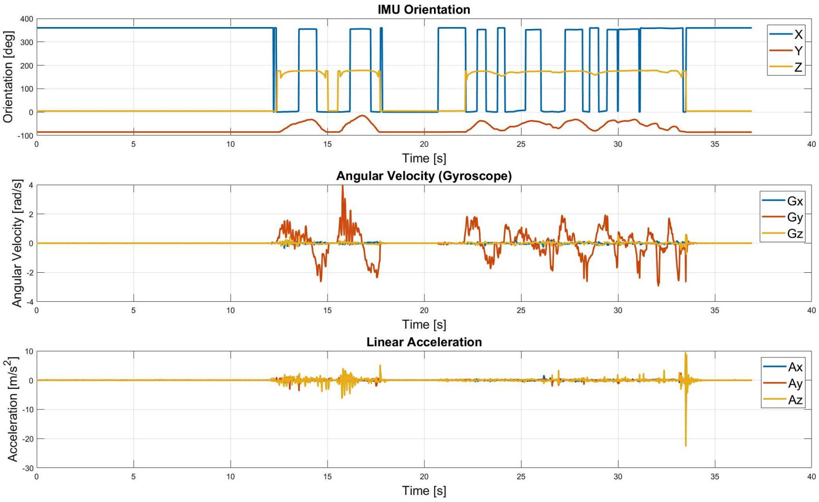


Figure 5.7 IMU orientation, angular velocity, and linear acceleration data recorded during finger actuation

a nearly constant magnitude, reflecting reliable orientation estimation. The board temperature curve indicates a slight rise from ambient to a stable operating value of approximately 30°C, demonstrating thermal stability over the experiment. Overall, the IMU plots confirm

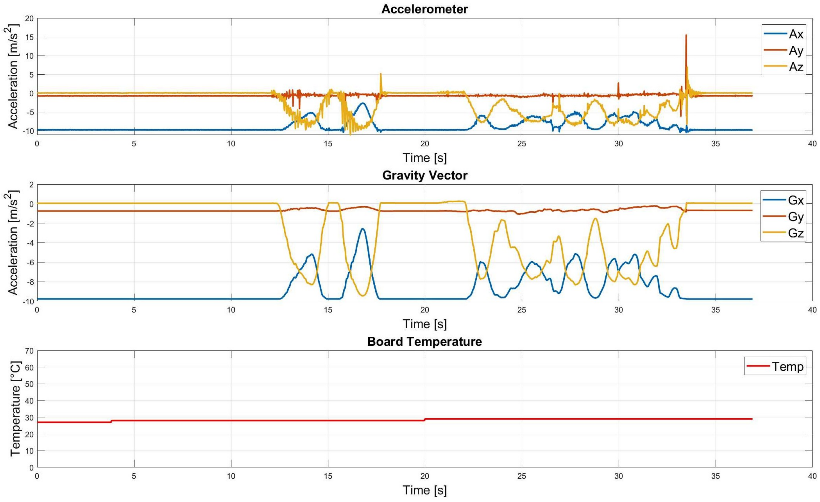


Figure 5.8 Accelerometer, gravity vector, and temperature data from the fingertip IMU during finger motion

synchronized and stable sensor performance during finger actuation. The consistent correlation between orientation, angular velocity, and acceleration signals verifies accurate motion tracking and reliable operation of the fingertip sensing module within the anthropomorphic hand.

Part III

Cross Domain Applications & Extensions

Chapter 6

Endoscopic Stereo Vision for Robotic 3D Detection of Thin Wire Features

Vision-based systems have become increasingly prevalent in robotic applications, enabling robots to perceive and interpret their surrounding for autonomous decision-making. However, achieving accurate 3D reconstruction remains a complex challenge, particularly when dealing with thin, flexible, and deformable linear objects. Existing commercial 3D cameras capable of modeling small-scale objects are often bulky, costly, and difficult to integrate within compact robotic platforms.

To address these limitations, this research presents a compact stereo vision system designed for seamless integration into a robotic end-effector. The system employs two low-cost, off-the-shelf endoscopic cameras to detect and characterize fine wire-like objects within the workspace, specifically estimating their spatial location and diameter. The proposed approach leverages advanced vision algorithms, typically used with macro-scale cameras, and adapts them to a miniature endoscopic configuration. Experimental results demonstrate that the system achieves a diameter estimation error below 10% and a positional estimation error below 3% for wires with diameters in the millimeter and less than millimeter range. This methodology is generalizable to various endoscopic camera configurations and can be extended to the detection of other slender or filamentary objects in robotic manipulation tasks.

The manipulation and perception of deformable linear objects (DLOs) present significant challenges in robotic applications. DLOs are characterized by one dominant dimension compared to their other two, encompassing objects such as wires, cables, tubes, and medical hoses. Due to their diverse physical properties and the frequent presence of extremely small diameters, sometimes less than one millimeter, accurate shape reconstruction and three-dimensional feature extraction remain particularly demanding when using conventional vision systems.

This phase of research addresses the problem of estimating both the spatial position and diameter of thin DLOs through a 3D reconstruction approach based on stereo vision. The proposed method uses compact endoscopic cameras to capture high-resolution stereoscopic imagery, enabling precise modeling of slender, flexible objects that are otherwise difficult to perceive with standard optical setups.

The principal contribution of this research is the design and development of a complete vision system, covering both hardware and software aspects based on a pair of low-cost endoscopic macro cameras. The system is specifically engineered to estimate the spatial location and diameter of thin wire-like objects, with diameters ranging from 0.5 mm to 4 mm. Through reconstruction of the three-dimensional (3D) scene, the generated point cloud is further processed and segmented into clusters to identify individual wires, their central axes, and orientations within the workspace.

The proposed methodology extends and adapts the established techniques described in Zhang (2002), Heikkila and Silvén (1997) for camera calibration, disparity map computation, and region of interest (ROI) selection, tailoring them to the optical characteristics and constraints of endoscopic imaging. The image data acquired from the stereo cameras is subjected to edge detection algorithms to accurately delineate wire boundaries. Subsequently, the diameter estimation is performed by correlating the detected edges with the depth information derived from the 3D reconstruction, allowing precise measurement of wire dimensions.

To validate the reconstructed 3D data and confirm the accuracy of the proposed vision-based estimation, a series of robotic experiments were conducted. In these experiments, wires were autonomously grasped from a warehouse environment using robotic fingers previously developed by some of the authors in Cirillo et al. (2021a). These fingers were equipped with embedded tactile sensors, which provided real-time feedback to assess grasp quality and ensure proper handling of detected wires. The integration of tactile sensing and stereo vision thus offers a comprehensive framework for reliable perception and manipulation of thin deformable linear objects.

6.1 Methodology

The objective of this research phase is to develop a robust methodology for accurately estimating the spatial position and diameter of thin wires stored within a warehouse environment. The considered wire diameters range from 0.5 mm to 4 mm. The proposed approach is based on a stereo vision framework that enables a precise 3D reconstruction of the workspace, facilitating the identification and selection of the correct wire for robotic grasping according to its estimated diameter.

To achieve this goal, a systematic calibration of the stereo camera pair is first performed to ensure accurate geometric alignment and depth perception. Once calibrated, the cameras are used to capture stereoscopic images of the scene, from which 3D reconstruction is carried out.

6.1.1 Camera Calibration

The calibration of the stereo camera system is performed to determine both the intrinsic and extrinsic parameters of each individual camera, as well as the geometric relationship between the two cameras. The intrinsic parameters include quantities such as the focal length, optical center, and skew coefficient, which define the internal characteristics of the camera lens and sensor. The extrinsic parameters, on the other hand, describe the rotation and translation of each camera with respect to the calibration target, thereby defining their spatial orientation within the experimental setup.

In addition, the inter-camera geometry representing the spatial relationship between the two cameras of the stereo pair is characterized by the relative rotation and translation vectors, along with the computation of the fundamental and essential matrices. As illustrated in Fig 2.5, the two cameras are positioned in a general relative configuration, and the stereo coordinate frame $\sum_c$ is fixed at the principal point of Camera 1. Each camera is modeled according to the pinhole camera model, establishing the mathematical relationship between a three-dimensional point $[x, y, z]^T$ in the scene and its corresponding two-dimensional image projection $[u, v]^T$, as expressed in Zhang (2002).

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \quad \text{with } \mathbf{A} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.1}$$

In the projection model, s represents an arbitrary scale factor, while $\mathbf{A}$ denotes the intrinsic camera matrix. The parameters (u_0, v_0) correspond to the principal point in pixel coordinates, γ accounts for the skew between the image axes, α and β define the focal lengths along the u and v axes respectively, expressed in pixels. The spatial relationship between the world coordinate system and the camera coordinate system is described by the rotation matrix $\mathbf{R}$

and the translation vector **t**.

The estimation of camera parameters requires a set of correspondences between known three-dimensional (3D) world points and their two-dimensional (2D) image projections. These correspondences are typically acquired by capturing multiple images of a calibration pattern from different viewpoints. Common calibration targets include custom pattern detectors, asymmetric circle grids, or checkerboards. The checkerboard pattern, comprising alternating black and white squares of uniform size, offers the advantage of high corner detection accuracy while avoiding 180° ambiguity. The control points are defined at the intersections of the squares and can be automatically detected in 2D images using a corner detection algorithm.

By placing the origin in the lower right corner of the top left square, the 3D coordinates of the control points can be computed based on the checkerboard geometry, including the number of squares along each row and column and the fixed square size. The calibration algorithm initially assumes zero lens distortion and computes a closed-form solution for both intrinsic and extrinsic parameters. This preliminary solution is then refined through non-linear least-squares optimization to estimate all parameters, including lens distortion coefficients. The general calibration procedure is illustrated in Fig 6.1, and its accuracy is quantified using the mean re-projecting error, defined as the average Euclidean distance, in pixels between the detected image points and their reprojected points.

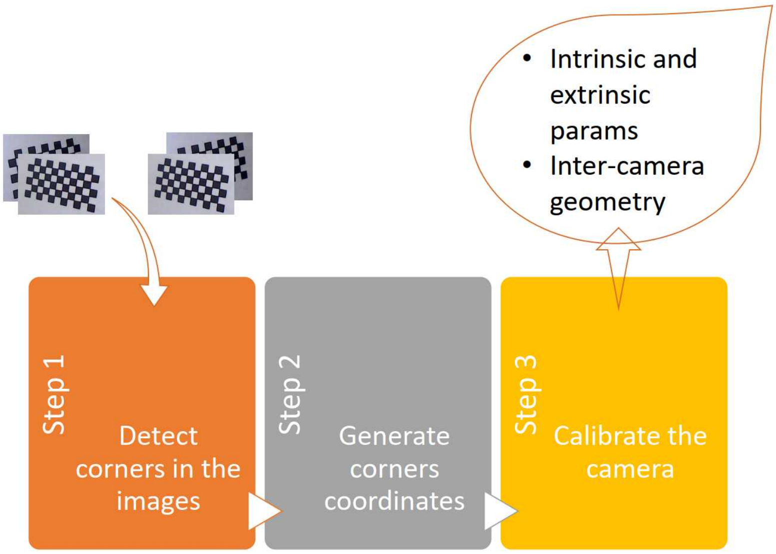


Figure 6.1 Steps in the calibration procedure

6.1.2 3D Reconstruction of Scene

The intrinsic and extrinsic camera parameters, together with the inter-camera geometry, are fundamental for accurate three-dimensional (3D) scene reconstruction. To achieve this, two essential intermediate steps are performed: stereo image rectification and disparity map computation.

The rectification process projects both stereo images onto a common image plane such that corresponding points share identical row coordinates. This transformation provides the visual effect of parallel camera alignment and serves a dual purpose: (i) to undistort the images by compensating for lens distortion, and (ii) to geometrically rectify the stereo pair. The resulting undistorted and rectified image pair constitutes the input for the disparity computation stage.

In the disparity estimation phase, the displacement between corresponding pixels in the two images is calculated using the Semi-Global Matching (SGM) algorithm. This method requires the definition of a disparity range, characterized by the minimum and maximum disparity values that span the horizontal shift between matching pixels. Additionally, a uniqueness threshold is employed to ensure reliable disparity estimation; a disparity value for a pixel is considered invalid if the minimum Hamming distance within the range is lower than the product of this threshold and the Hamming distance corresponding to the selected disparity.

The SGM approach computes the matching cost matrix from the Hamming distance between corresponding pixels, obtained through the Census Transform of the two rectified images. Each pixel is then assigned a disparity value that minimizes this matching cost. The computed disparity map serves as the sole input for the 3D reconstruction algorithm, which derives the spatial coordinates of the scene points with respect to the stereo reference frame $\sum_c$.

Once the 3D point cloud is generated, it is further processed to determine the position and orientation of the wires within the workspace. Specifically, the position of each wire $p_{ci}$and its orientation θ_i in the image plane, for $i = 1, 2, \ldots, n$ where n is the number of wires detected, are computed as follows. A Region of Interest (ROI) is first extracted from the point cloud using a Kd-tree–based search algorithm, a binary space-partitioning structure that divides data points recursively using the median-split criterion. The selected ROI is then segmented into clusters according to two conditions: a minimum Euclidean distance between clusters and a minimum number of points within each group.

The centroid of each i^{th} cluster is calculated as the mean of its constituent points, representing the 3D position p_{ci} of the wire relative to the reference frame $\sum_c$. The wire orientation θ_i in the image plane is estimated by fitting a first-order polynomial through two points corresponding

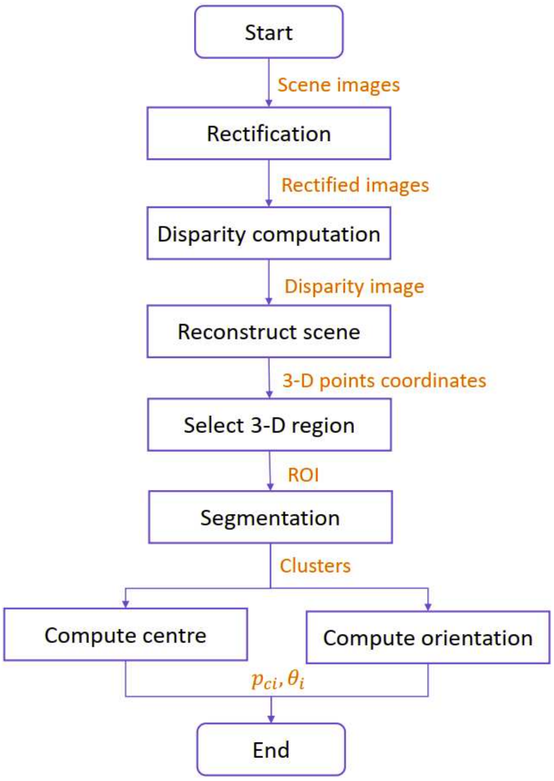


Figure 6.2 Flowchart of 3D scene reconstruction algorithm

to the mean positions of the first and second halves of the cluster along the $y-$axis. A schematic overview of the entire 3D scene reconstruction pipeline, including rectification, disparity computation, and point cloud analysis, is illustrated in Fig 6.2.

6.1.3 Diameter Detection

The reconstruction of the three-dimensional (3D) scene is of paramount importance for determining the depth information of the objects within the workspace. This depth data provides the estimated distance of each wire from the camera, which can be utilized not only for the wire grasping operation but also for accurately computing the wire diameter. To minimize the influence of noise inherent in the 3D point cloud, the wire diameter is calculated directly from the 2D image while incorporating the corresponding $z-$coordinate information from the reconstructed 3D scene.

The wire is first localized in the 2D image by referencing the point p_{ci}. From this position, the two lateral edges of the wire are identified. The coordinates of p_{ci} are then used to determine the wire center (u_{c_i}, v_{c_i}) in the 2D image. For simplification, several assumptions are made: the image skew is neglected ($\gamma = 0$), the reference frame is aligned with the camera coordinate system $\sum_c$, the rotation and translation parameters are set as $\mathbf{R} = \mathbf{I}$, and $\mathbf{t}= [0\,0\,0]^T$, and the image is rectified such that $\alpha = \beta = f$. Considering a unitary scale factor ($s = 1$), Equation 6.1 can therefore be rewritten as:

$$u = \frac{f}{z}x + u_0 \tag{6.2}$$

$$v = \frac{f}{z}y + v_0 \tag{6.3}$$

Since the wire in the image may not be perfectly aligned with the vertical axis, the image is rotated by an angle θ_i for each detected wire to achieve proper alignment. This rotation yields the transformed pixel coordinates (u_r, v_r). The original wire center coordinates (u_{c_i}, v_{c_i}) are then mapped onto the rotated image by applying the corresponding two-dimensional rotational transformation in the image plane.

$$\begin{bmatrix} u_r \\ v_r \end{bmatrix} = \mathbf{R}_{2D} \left(\begin{bmatrix} u \\ v \end{bmatrix} - \mathbf{c} \right) + \mathbf{c} \tag{6.4}$$

$$\mathbf{R}_{2D} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{6.5}$$

Here, c denotes a two-element vector representing the coordinates of the image center. The edges within the image are detected using the Sobel gradient operator, which produces a binary matrix of the same dimensions as the input image—assigning a value of 1 to pixels corresponding to detected edges and 0 otherwise. The resulting edge image is then analyzed along the $v-$axis, specifically by examining the row that contains the pixel (u_{c_i}, v_{c_i}), in order to identify the edge points closest to the center of the wire. Once the edge coordinates u_{e1} and u_{e2} are determined, Equation 6.2 is inverted to compute the actual wire diameter in millimeters.

$$d = \frac{z}{f}(u_{e2} - u_{e1}) \tag{6.6}$$

The complete procedure for wire diameter computation is summarized in Algorithm 1, which presents the corresponding pseudo-code implementation.

Algorithm 5 Diameter computation

```
**Input:** $I$ (2D image), $p_{c_i}$
**for** $i = 0;\ i <$ number of wires; $i{+}{+}$ **do**
    Compute wire centre in $I$                                ▷ from Eqs.6.2-6.3
    $I_r \leftarrow \text{imrotate}(I, \theta_i)$
    $(u_{c_i}, v_{c_i}) \leftarrow$ Remap image centre        ▷ from Eq. 6.4
    $I_{edges} \leftarrow \text{edge}(I_r, \text{'Sobel'}, \text{threshold})$
    $I_{row} \leftarrow I_{edges}(v_{c_i}, :)$
    $edges \leftarrow \text{find}(I_{row} == 1)$
    $u_{e1} \leftarrow \max(edges(\text{find}(edges < u_{c_i})))$
    $u_{e2} \leftarrow \min(edges(\text{find}(edges > u_{c_i})))$
    Compute $d_i$                                             ▷ from Eq. 6.6
**end for**
**Output:** $d$
```

6.2 Experimental Validation

To validate the proposed methodology, a comprehensive set of experiments was conducted using a carefully designed experimental setup. The system comprises a UR5e robotic arm (Universal Robots) equipped with a Robotiq Hand-E parallel gripper. The gripper is fitted with a pair of custom-designed fingers developed by the authors, each integrating embedded tactile sensors for grasp validation. Further details regarding the tactile sensor design and functionality can be found in Cirillo et al. (2021a). The developed vision system employs two

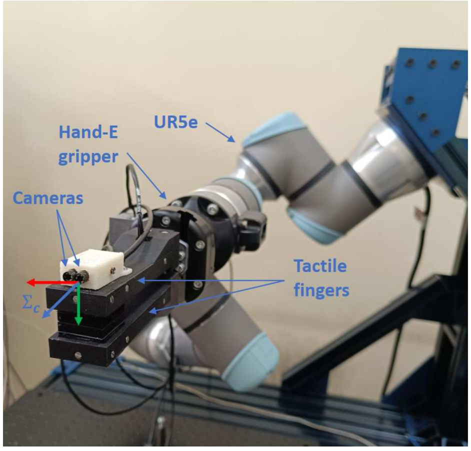


Figure 6.3 Hardware setup of stereo cameras

commercial off-the-shelf endoscopic cameras with a resolution of 640×480 pixels, securely mounted within a custom-designed housing attached to the rear side of one of the tactile fingers (see Fig 6.3). The primary objective of the experiment is the automatic identification and picking of wires stored in a warehouse, based on their estimated diameters. While the global position of the warehouse with respect to the robot base is known, the exact distance of each wire from the cameras is not predetermined, elaborated in Fig 6.4. Therefore, the stereo vision system estimates both the diameter and depth ($z-$coordinate) of each wire relative to the camera coordinate frame $\sum_c$.

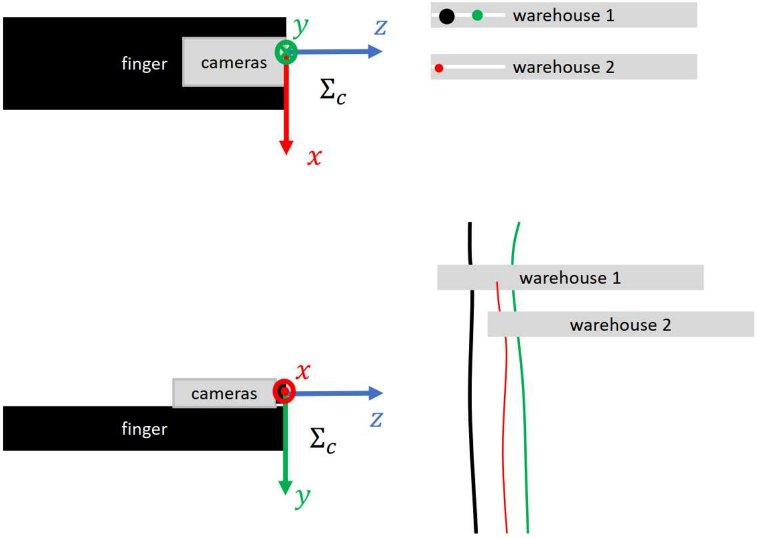


Figure 6.4 Wires arranged in the warehouse, top view (top) and lateral view (bottom).

The diameter estimation was validated using four wires placed in the scene, while the grasping trials were performed using two of these wires. The camera calibration procedure, described in Section **"Camera Calibration"**, was executed in MATLAB using 13 images per camera, captured from various viewpoints by moving the robotic arm. The calibration employed a checkerboard pattern with six rows and eleven columns, each square measuring 2 mm. These parameters were determined empirically through trial and error to achieve an optimal balance between accuracy and field coverage. The resulting mean reprojection error was 0.27 pixels, confirming high calibration precision.

After calibration, the diameter estimation algorithm was validated by capturing stereo image pairs containing four wires of nominal diameters 2.15 mm, 0.50 mm, 1.20 mm, and 3.90 mm (from left to right in Fig 6.5). The wires were positioned at different known distances from the cameras: 150.0 mm, 95.0 mm, 138.0 mm, and 132.0 mm, respectively. Ground-truth

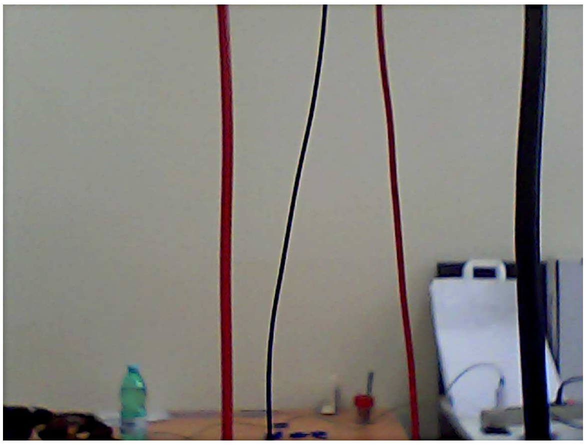


Figure 6.5 Wires of different diameters, from the left: 2.15 mm, 0.50 mm, 1.20 mm, and 3.90 mm, at different distances from cameras.

diameters were measured using a vernier caliper with a resolution of 0.05 mm. Image acquisition was performed within a focal range of 90 – 140 mm, corresponding to the optimal focus distance of the selected cameras.

Figure 6.6 Disparity map of the four wires frame.

The image processing pipeline, implemented in MATLAB following the flowchart in Section **"3D Reconstruction of Scene"**, included image rectification, disparity map computation, and 3D scene reconstruction. For the disparity map refer to Fig 6.6, the parameters were set to a disparity range of 30–94 pixels and a uniqueness threshold of 40. To reduce background noise, a Region of Interest (ROI) was selected along the $z-$axis between 30 mm and 180 mm. Clustering of the 3D point cloud was performed using a Euclidean distance threshold of 0.8 and a minimum cluster size of $3,000$ points. For each cluster, the center point p_{ci} and the orientation angle θ_i were computed. Fig 6.7 shows the four identified clusters, each represented in a distinct color, with their corresponding centers highlighted.

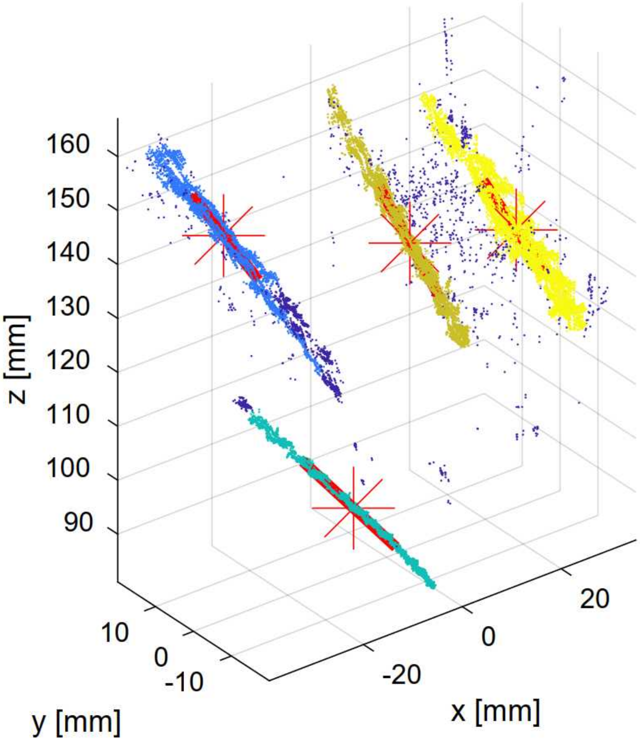


Figure 6.7 Highlighted segmentation with identified wires centers.

The reconstructed wire distances from the cameras were 148.7 mm, 95.7 mm, 139.7 mm, and 135.3 mm, yielding percentage errors of 0.9%, 0.7%, 1.4%, and 2.5%, respectively (shown in table 6.1), when compared to the nominal distances.

Table 6.1 Comparison between nominal and reconstructed distances

Distance		
Nominal distance	**Reconstructed distance**	**Distance error**
150.0 mm	148.7 mm	0.9 %
95.0 mm	95.7 mm	0.7 %
138.0 mm	139.7 mm	1.4 %
132.0 mm	135.3 mm	2.5 %

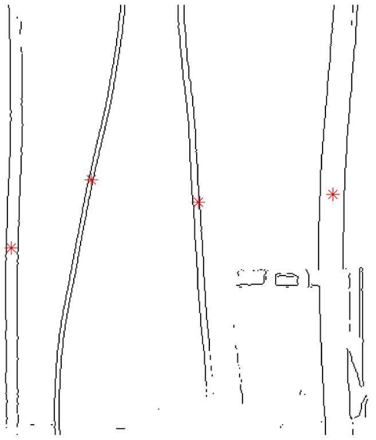


Figure 6.8 Highlighted edges and centers of wires.

Following the procedure outlined in Algorithm 5, the cluster centers were projected onto the 2D image using Equations 6.2– 6.3, with parameters $f = 896.96$,$u_0 = 309.21$ and $v_0 = 220.43$ pixels. Each image was then rotated by its corresponding θ_i, and the point p_{ci} was remapped in the rotated image, considering the image center coordinates $[258.5, 220.5]$ and the transformation in Equation 6.4. Edge detection was performed using a threshold of 0.09 (see Fig 6.8), allowing the extraction of edge coordinates u_{e1} and u_{e2}, from which the diameter d_i was computed.
The estimated diameters obtained using Equation 6.6 were 1.99 mm, 0.53 mm, 1.09 mm, and 3.63 mm, corresponding to percentage errors of 7.4%, 6.0%, 9.2%, and 6.9%, respectively as presented in table 6.2.

Table 6.2 Comparison between nominal and reconstructed diameters

	Diameter	
Nominal diameter	**Reconstructed diameter**	**Diameter error**
2.15 mm	1.99 mm	7.4 %
0.50 mm	0.53 mm	6.0 %
1.20 mm	1.09 mm	9.2 %
3.90 mm	3.63 mm	6.9 %

For the wire picking validation, the system was integrated into a Robot Operating System (ROS) network, where MATLAB handled image processing as a callable service. Two wires, with nominal diameters of 1.20 mm and 2.15 mm, were placed in the warehouse. The robot was positioned such that both cameras viewed both wires simultaneously, and the image processing service was invoked. The wires were placed at known distances of 112.0 mm and 100.0 mm, respectively, to verify accuracy. The algorithm estimated their depths as 110.1 mm and 102.9 mm, corresponding to percentage errors of 1.7% and 2.9%, respectively. The detected diameters were 1.10 mm and 1.96 mm, with mean diameter errors of 8.3% and 8.8%, respectively. Fig 6.9 illustrates the segmented 3D point cloud with distinct color clusters.

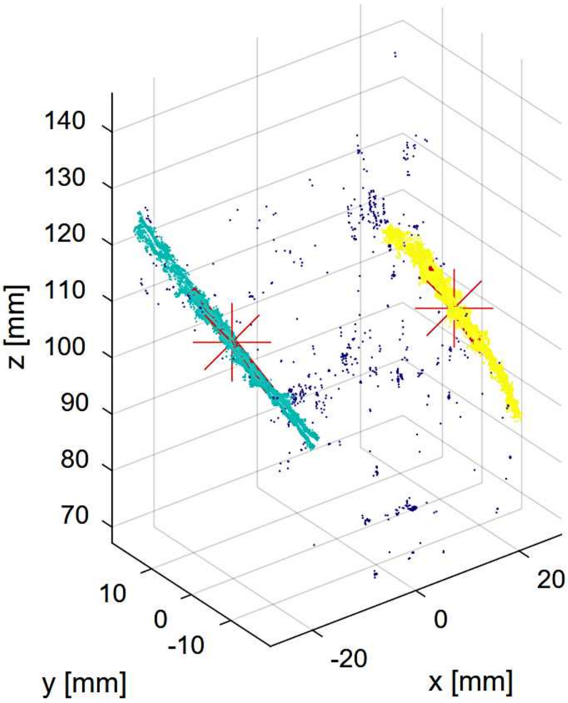


Figure 6.9 3D Point cloud segmentation for the wire picking task.

The user can select the target wire based on its estimated diameter. If a wire with the specified diameter is present, the robot aligns its fingers with the wire's x and y coordinates and uses the estimated $z-$coordinate for precise picking. Finally, the tactile sensors validate successful grasping. The entire process is summarized in the flowchart shown in Fig 6.10.

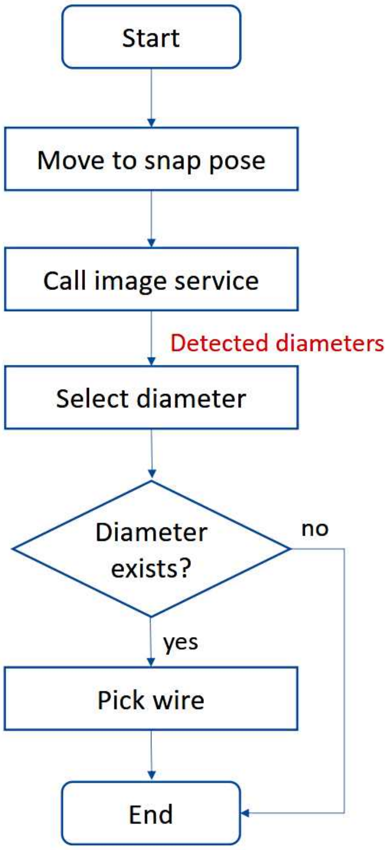


Figure 6.10 Flowchart for the wire picking task.

Chapter 7

Multi-Modal Sensor to Control 3 Axis Attitude of a Satellite System

The developed system presents the application of the designed multi-modal sensor for the attitude control of three-axis satellites. With satellite missions becoming increasingly complex, the need for highly precise and reliable attitude control systems has grown substantially. Conventional sensing technologies often encounter limitations in accuracy, sensitivity, and adaptability to dynamic space environments. To address these challenges, we introduce a novel approach that leverages the multi-modal sensing capabilities of the designed sensor to fuse data from multiple sources, including gyroscopes and accelerometers integrated within the inertial measurement unit (IMU). This integration enhances both the robustness and precision of attitude estimation and control.

The sensor architecture employs advanced sensor fusion techniques to optimize the performance of attitude control algorithms, enabling real-time corrections to satellite orientation. In particular, the IMU component demonstrates high precision in measuring angular velocity and acceleration, thereby significantly improving the satellite's ability to maintain stable orientation even under adverse operational conditions such as external disturbances or rapid maneuvers.

Beyond its demonstrated effectiveness in robotic grippers, this research highlights the adaptability of the multi-modal sensor for aerospace applications. By extending its utility to satellite attitude control, the study contributes to the development of intelligent and versatile sensing technologies. This cross-domain applicability underscores the potential of the multi-modal sensor as a foundation for future advancements in both robotic manipulation and space systems engineering.

The increasing complexity of satellite missions in modern space exploration necessitates the adoption of innovative technologies to achieve accurate and reliable attitude control. Today, satellites are deployed for a wide range of applications, including military, scientific, and commercial purposes. This growing demand has motivated numerous organizations and private companies to pursue the launch of their own satellites. To fully capitalize on the advantages of satellite-based systems, it is imperative that satellites offer flexibility in mission scheduling while minimizing the costs associated with on-orbit deployment. In this context, small satellites have emerged as a highly attractive alternative to larger platforms.

However, the compact nature of small satellites introduces unique challenges. The limited onboard volume constrains the integration of components, necessitating highly compact equipment designs. Furthermore, the reduced power budget of these satellites requires all subsystems to operate under strict energy constraints, demanding efficiency in both hardware and software utilization Jayaram (2009).

Among the critical subsystems, the attitude control system plays a pivotal role in mini-satellite operations, particularly for missions that demand precise orientation and steering capabilities. Satellite attitude refers to the angular orientation of the spacecraft with respect to a defined orbital reference frame. Maintaining the desired attitude throughout the satellite's operational lifetime is crucial to ensure mission success, whether for Earth observation, communication, or scientific measurements. This continuous control is achieved through the integration of dedicated hardware components, such as actuators and sensors, in conjunction with software-based controllers that process sensor data, compute corrective actions, and command the actuators to sustain stable orientation.

7.1 Hardware Design of Satellite

The hardware design was the first critical step in this research, as it provided the physical foundation for testing and evaluating the proposed control system. The hardware consisted of two primary subsystems: the testing platform and the actuator system(reaction wheels).

7.1.1 Testing Platform

To approximate the free-space dynamics of a satellite on Earth, the experimental setup required frictionless rotation in three degrees of freedom (roll, pitch, and yaw). For this purpose, a spherical air bearing was selected (Fig. 7.1). This bearing allows smooth, low-resistance rotation by supporting the platform on a thin film of compressed air.

The globe-shaped air bearing was designed to provide a travel angle of ±45° around the roll and pitch axes. The bearing was capable of sustaining a load capacity of 160 kg at an input pressure of 80 psi, which ensured that the combined weight of the testing platform and actuators could be supported without significant frictional effects. To achieve the required precision, the spherical surface and its corresponding base were manufactured using a high-precision CNC machining process, guaranteeing a near-perfect fit between the contacting surfaces. The final assembly provided stable and smooth motion, closely resembling the near-frictionless rotational behavior of satellites in orbit.

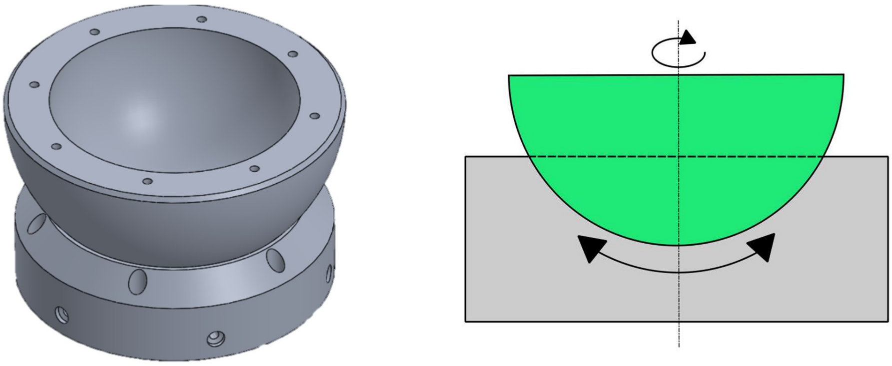


Figure 7.1 Spherical air bearing (left) and globular air bearing: degrees of freedom (right)

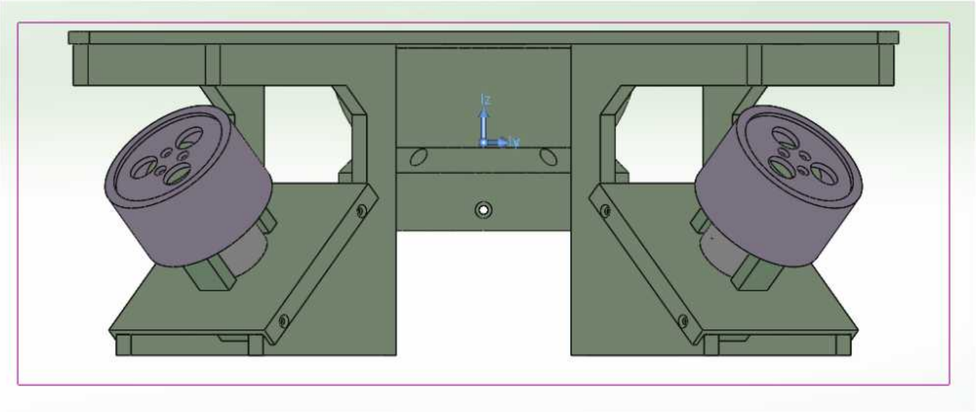


Figure 7.2 Approximate center of gravity point

The platform structure, which was mounted on the air bearing, was specifically designed to maintain balance and symmetry, presented in Fig 7.3. The critical requirement was to align the center of gravity (CoG) of the platform with the geometric center of rotation of the bearing (Fig. 7.2). To achieve this, the top plate was designed with a symmetric pattern of holes and equal spacing of mounting points. This ensured uniform mass distribution and minimized gravitational disturbance torques.

Special care was taken in the design of the actuator mounting hangers, which were manufactured with identical dimensions and material properties to avoid introducing asymmetry. After machining all components, balancing tests were conducted. The platform demonstrated proper equilibrium about the roll and pitch axes, confirming the accuracy of the CoG alignment.

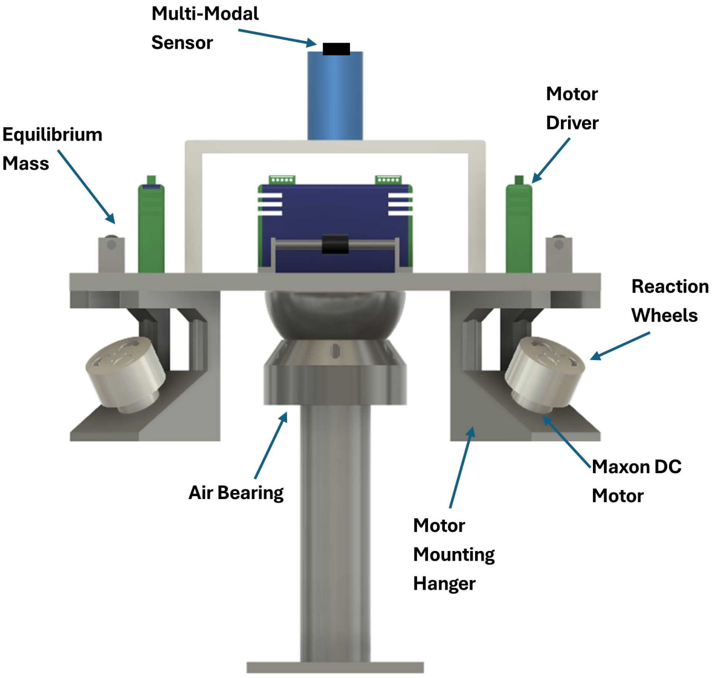


Figure 7.3 CAD model of complete assembly of satellite system

7.1.2 Actuator System

The chosen actuators for the platform were reaction wheels, which are widely used in satellite attitude control for their ability to generate precise control torques without expending propellant. Four wheels were installed on the platform in a pyramidal configuration, oriented at 45° relative to each axis (Fig. 7.4). This arrangement enabled full three-axis attitude control with built-in redundancy: in the event of a single wheel failure, the remaining three wheels could continue to provide control torque in all axes.

Each reaction wheel was designed to produce a maximum torque of approximately 20

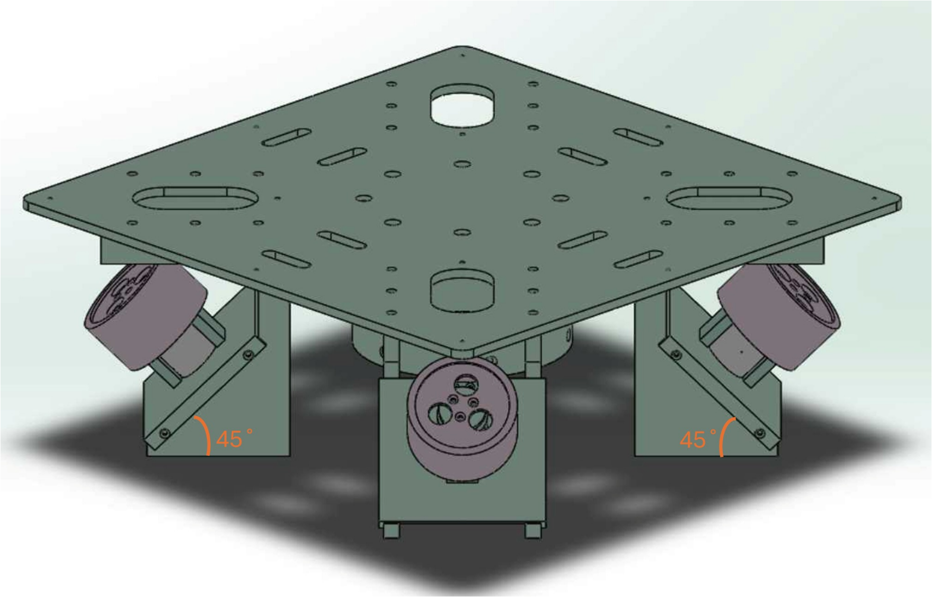


Figure 7.4 Reaction wheel configuration (pyramidal layout)

mNm, sufficient to overcome expected disturbance torques during terrestrial experiments. The wheels were driven by Maxon 242478 DC motors, rated at 15 W, with a maximum speed of 5860 rpm. When coupled to the custom-designed wheel disks, the actuators provided a wide operational torque range with relatively long acceleration and deceleration periods, allowing both fine control and larger maneuvers.

The wheel body was manufactured from lightweight, rigid material to maximize efficiency, while the shaft was mounted on precision ball bearings to minimize internal friction and vibration. This ensured smooth operation and consistent torque output. The final assembly of the testing platform with installed actuators is presented in Fig. 7.5.

7.2 Control Architecture

Following the development of the hardware system, the next step involved designing of the control architecture for the satellite platform. The controller is responsible for maintaining the desired attitude by regulating the speed of the reaction wheels according to the measured angular rates and orientation errors. The control system was designed to achieve three main objectives:

- Maintain precise attitude tracking with various reference commands.
- Compensate for external disturbance torques, such as gravitational effects.

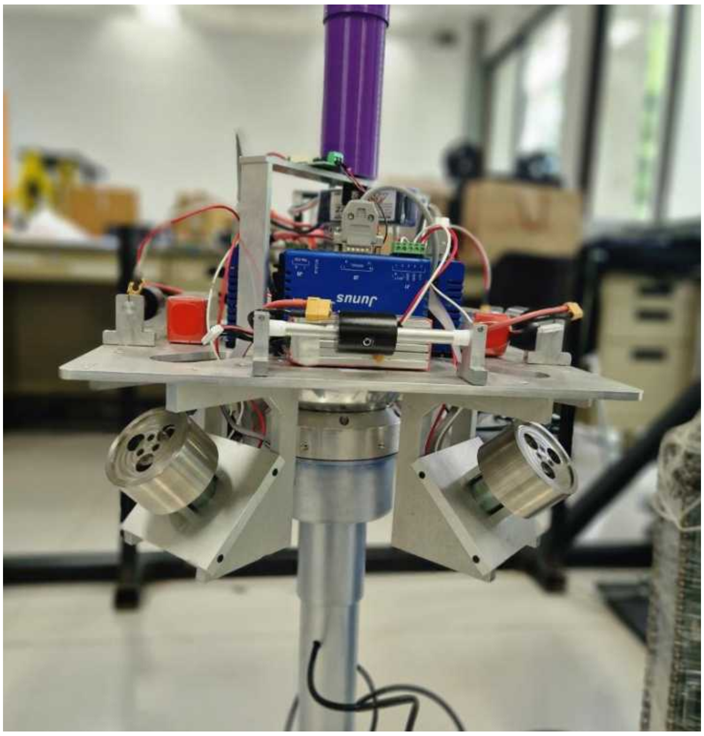


Figure 7.5 Final assembly of testing platform and satellite system

- Prevent actuator saturation during prolonged operation.

To accomplish these goals, a Proportional–Derivative (PD) control law with quaternion feedback was implemented as the primary attitude control strategy, supplemented by a saturation limitation handler to manage wheel speed limits and ensure continuous actuator availability.

7.2.1 Reaction Wheels Configuration

The platform was equipped with four reaction wheels, oriented in a pyramidal configuration at 45 with respect to each body axis. This configuration enables full 3-axis attitude control while providing fault tolerance, even if one wheel fails, the remaining three can still generate the required control torques through vector redundancy. The mapping between the reaction wheel torques and the body-frame torques is represented by the wheel configuration matrix

A_w, which defines how individual wheel torques contribute to the total control torque vector:

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = [A_w] \begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \\ \tau_{c4} \end{bmatrix} \tag{7.1}$$

Since A_w is a non-square (over-actuated) matrix, a torque allocation algorithm was implemented to distribute control commands efficiently across all wheels. A minimum-norm optimization criterion was used to compute actuator torques while minimizing the total energy consumption:

$$\min_{\tau_{c1},\tau_{c2},\tau_{c3},\tau_{c4}} \left\| \begin{bmatrix} \tau_{c1} & \tau_{c2} & \tau_{c3} & \tau_{c4} \end{bmatrix} \right\|^2 \quad \text{subject to: } [A_w] \begin{bmatrix} \tau_{c1} \\ \tau_{c2} \\ \tau_{c3} \\ \tau_{c4} \end{bmatrix} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{7.2}$$

This ensures a balanced torque distribution among the reaction wheels, improving efficiency and reducing mechanical wear.

7.2.2 Control Algorithm

The control law adopted for the satellite platform is based on a Proportional–Derivative (PD) controller formulated in quaternion space. The quaternion representation was chosen over Euler angles to avoid singularities (gimbal lock) and to provide smooth, continuous attitude representation suitable for real-time computation on the embedded controller.

The quaternion error q_e is defined as the difference between the commanded quaternion q_{cmd} and the current measured quaternion q_c. The angular velocity error ω_e is obtained as the difference between the desired and measured angular rates.

The control torque for each axis is computed using the following equations:

$$\hat{\tau}_{wx} = -2K_{px}q_{e1}q_{e4} - K_{dx}\omega_x \tag{7.3}$$

$$\hat{\tau}_{wy} = -2K_{py}q_{e2}q_{e4} - K_{dy}\omega_y \tag{7.4}$$

$$\hat{\tau}_{wz} = -2K_{pz}q_{e3}q_{e4} - K_{dz}\omega_z \tag{7.5}$$

where:

- $q_{e1}, q_{e2}, q_{e3}, q_{e4}$ are the components of the quaternion error q_e;
- $\omega_x, \omega_y, \omega_z$ are the angular velocities measured by the IMU;
- $K_{p(x,y,z)}$ and $K_{d(x,y,z)}$ are the proportional and derivative control gains, respectively.

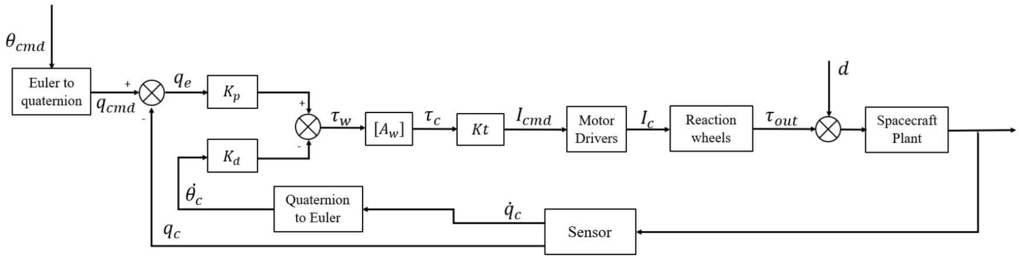


Figure 7.6 Block diagram of PD controller for satellite

The proportional term contributes to error correction based on the current orientation difference, while the derivative term provides damping proportional to angular rate, reducing oscillations and improving stability.A schematic representation of the PD control loop is shown in Fig. 7.6.The controller continuously compares the reference attitude with the current attitude, computes the error, and generates the appropriate control torque commands for the reaction wheels.

7.3 Sensor Fusion for Attitude Estimation

Accurate attitude estimation is essential for stable satellite control. In this system, a 6-axis IMU, consisting of a 3-axis accelerometer and a 3-axis gyroscope, is used to measure the satellite's motion. Since each sensor has distinct limitations, accelerometers are sensitive to noise and dynamic acceleration, while gyroscopes suffer from drift, a sensor fusion algorithm is implemented to combine their complementary characteristics.

7.3.1 Accelerometer-Based Estimation

The accelerometer measures gravitational acceleration, allowing the estimation of roll and pitch as:

$$\phi_{acc} = \arctan\left(\frac{a_y}{a_z}\right) \tag{7.6}$$

$$\theta_{acc} = \arctan\left(\frac{-a_x}{\sqrt{a_y^2 + a_z^2}}\right) \tag{7.7}$$

where a_x, a_y, and a_z are the accelerometer outputs along the x, y, and z axes, respectively. These estimates are accurate under static or quasi-static conditions.

7.3.2 Gyroscope-Based Estimation

The gyroscope provides angular velocity measurements $(\omega_x, \omega_y, \omega_z)$, which are integrated over time to estimate the change in attitude:
To obtain a reliable estimate, a Complementary Filter was implemented, combining high-frequency gyroscope data with low-frequency accelerometer data:

$$\phi_{gyro}(t) = \phi_{gyro}(t - \Delta t) + \omega_x \Delta t \tag{7.8}$$

$$\theta_{gyro}(t) = \theta_{gyro}(t - \Delta t) + \omega_y \Delta t \tag{7.9}$$

$$\psi_{gyro}(t) = \psi_{gyro}(t - \Delta t) + \omega_z \Delta t \tag{7.10}$$

To combine both sensing modalities, a Complementary Filter is implemented, blending high-frequency gyroscope data with low-frequency accelerometer data. The fusion equations for roll and pitch are given by:

$$\phi(t) = \alpha\,[\phi(t - \Delta t) + \omega_x \Delta t] + (1 - \alpha)\,\phi_{acc} \tag{7.11}$$

$$\theta(t) = \alpha\,[\theta(t - \Delta t) + \omega_y \Delta t] + (1 - \alpha)\,\theta_{acc} \tag{7.12}$$

where $0 < \alpha < 1$ is the blending coefficient. In this study, $\alpha = 0.98$ was used to achieve a balance between responsiveness and stability.

7.3.3 Quaternion Representation

The fused roll, pitch, and yaw angles are converted into quaternion form for attitude representation:

$$
\begin{aligned}
q_w &= \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)+\sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)\\
q_x &= \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)-\cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)\\
q_y &= \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)+\sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)\\
q_z &= \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)-\sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)
\end{aligned}
\tag{7.13}
$$

This quaternion output ensures a continuous, singularity-free representation of attitude, suitable for use in the control loop.

7.3.4 Summary

The implemented Complementary Filter effectively combines accelerometer and gyroscope data to provide stable and drift-free estimates of roll, pitch, and yaw. The quaternion representation of the fused attitude enables precise and real-time feedback for the PD controller in the satellite attitude control system.

7.4 Experimental Results

This section presents and discusses the experimental results obtained from the robotic satellite platform equipped with four reaction wheels in a pyramidal configuration.

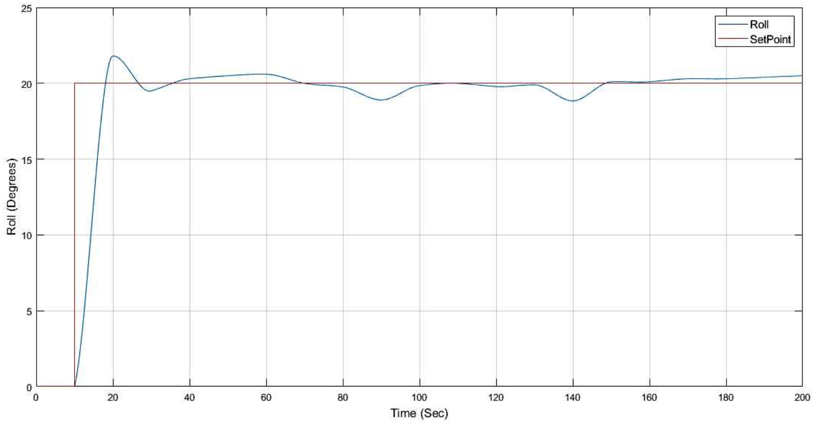


Figure 7.7 Roll angle response: commanded vs. actual attitude

The experiments were conducted to validate the effectiveness of the developed attitude control system using the proposed Proportional–Derivative (PD) controller and the multi-modal sensor for real-time feedback. The experimental setup aimed to evaluate three primary aspects: the accuracy of attitude stabilization, the influence of gravitational disturbances, and the behavior of the saturation limitation handler during extended operation.

Figures 7.7, 7.8, and7.9 show the attitude responses for roll, pitch, and yaw axes, respectively.

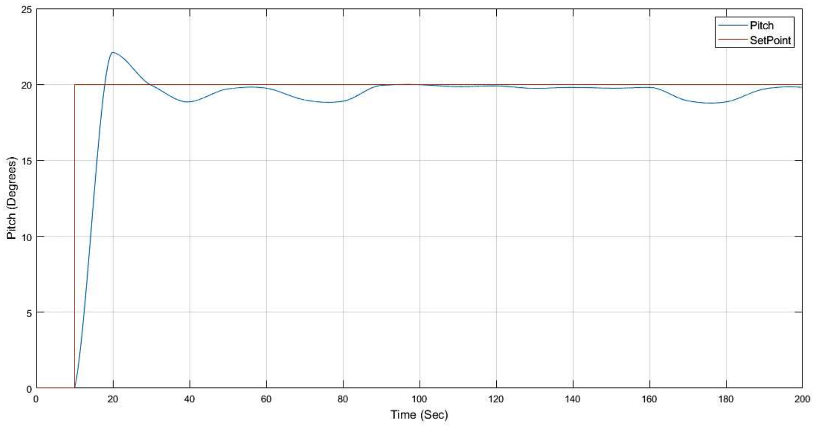


Figure 7.8 Pitch angle response: commanded vs. actual attitude

Each plot compares the commanded reference with the measured attitude obtained from the sensor fusion system. At the start of each test, a step input was applied to initiate rotation. The system responded rapidly and exhibited a short transient oscillation before converging to the desired orientation. The results demonstrate that the PD controller effectively minimized the angular error, achieving stable convergence within a few seconds.

Figures 7.7 and 7.8 illustrate the roll and pitch attitude responses, respectively. Both axes exhibit a slight overshoot following the commanded step input, which is expected in an under-damped second-order system tuned for fast response. The pitch axis shows a comparatively larger overshoot and longer settling time due to the influence of gravitational torque acting about the vertical plane, while the roll axis stabilizes more quickly with minimal residual oscillation.

The small oscillations observed near steady-state in the pitch response correspond to the operation of the saturation limitation handler, which temporarily suspends reaction wheel torque to prevent over-speeding. Despite these brief fluctuations, overall system performance remains stable, demonstrating the PD controller's ability to achieve accurate attitude tracking in both axes. Figure 7.9 indicates smooth and well-damped responses for the yaw axes.

The yaw axis achieved the commanded attitude with minimal overshoot and negligible

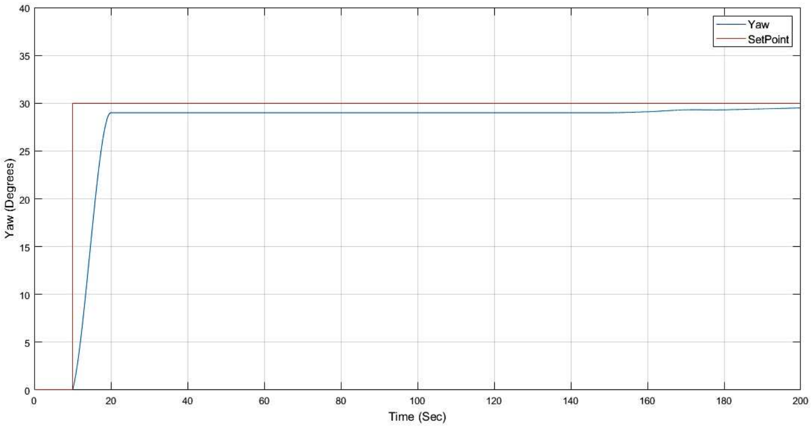


Figure 7.9 Yaw angle response: commanded vs. actual attitude

steady-state error, confirming precise gain tuning and effective dynamic compensation.The yaw response, derived primarily from gyroscope integration, demonstrated slightly slower settling but remained within acceptable accuracy limits for attitude control.

The results confirm the robustness of the PD controller and the reliability of the sensor fusion system for small-satellite attitude regulation.

Chapter 8

Conclusion & Future Directions

8.1 Conclusion

This thesis has presented the design, development, and experimental validation of a series of multi-modal sensing systems and their integration into various robotic platforms to enhance perception, dexterity, and adaptability in manipulation tasks. The research addressed the growing need for human-inspired sensing and control in robotic systems by combining tactile, proximity, inertial, and visual feedback into compact and modular hardware architectures suitable for real-world applications in industrial, agri-food, and space robotics.

The initial phase of this research focused on the development of a foundational multi-modal sensing architecture, which integrates optoelectronic tactile elements, Time-of-Flight (ToF) proximity sensing, and an Inertial Measurement Unit (IMU) within a unified platform. This architecture allows simultaneous measurement of acceleration, angular velocity, and temperature, supporting real-time perception and control. The compact form factor and modular design enable seamless integration into multi-fingered grippers or robotic hands. Individual sensing capabilities were experimentally characterized in terms of sampling frequency, noise, and precision, confirming their robustness for robotic manipulation and future applications in control and data fusion.

The subsequent research stage introduced an advanced multi-modal sensor integrated with the Robotiq 3-Finger Adaptive Gripper, substantially improving its capability to manipulate delicate and deformable objects. The integration of tactile and inertial feedback with a Proportional-Integral (PI) control strategy ensured precise finger positioning, stable force regulation, and adaptive performance under varying interaction conditions. Experimental validation demonstrated the system's ability to maintain stable, flexible, and safe grasping, even in collaborative scenarios. Moreover, a centroid-based HRI control approach was

developed, enabling intuitive interaction between humans and robots. By detecting force direction through tactile feedback, the gripper autonomously executes context-aware release or retention actions, ensuring safety and responsiveness during shared manipulation.
Parallel to this, the research presented the design and validation of a reconfigurable four-finger robotic gripper, developed for agri-food applications requiring careful handling of fragile produce. The gripper was modeled and tested in a virtual environment before fabrication and integration with custom multi-modal fingertip sensors. Mounted on a UR5e robotic arm, it demonstrated stable grasping, smooth control response, and strong adaptability across various grasp configurations, confirming its potential for automation in delicate food-handling tasks.
Building upon these developments, Chapter 5 introduced the third-generation compact fingertip sensor, marking a major advancement in sensor miniaturization and integration. This version was designed to replicate the size and geometry of a human fingertip and was equipped with PWM-driven optoelectronic tactile elements, an integrated 9-axis IMU, and a ToF proximity sensor. The sensor was successfully integrated into the SEEDs anthropomorphic robotic hand, achieving full mechanical and electronic compatibility. This design provided synchronized acquisition of tactile, proximity, and inertial data, enabling fine-grained perception of contact conditions and object geometry. Experimental tests confirmed the sensor's stability, low noise, and responsiveness, establishing its suitability for human-like dexterous manipulation and human–robot interaction applications.
A comparative analysis with representative tactile sensing systems from the literature, summarized in Table 8.1, further emphasizes the significance of this contribution. While many existing solutions focus primarily on high-resolution tactile perception using a single sensing modality, the proposed fingertip sensor distinguishes itself through true multi-modal integration at the fingertip level, combining tactile, proximity, and inertial sensing within a compact anthropomorphic form factor. Unlike planar tactile skins or externally mounted sensors, the proposed design is fully embedded within the distal finger joint, enabling natural interaction, anticipatory grasping, and enhanced situational awareness during manipulation. This integrated sensing capability provides complementary information that is not simultaneously available in most commercial solutions, supporting more informed grasp planning, adaptive control, and safer interaction.

In Chapter 6, the research extended into the domain of vision-based perception, presenting the design and validation of an endoscopic stereo vision system for the 3D reconstruction and geometric estimation of thin deformable linear objects (DLOs) such as wires and cables. The system, based on low-cost miniature cameras, enabled accurate estimation of both spatial position and diameter with less than 10% diameter error and 3% distance error. These results confirmed the method's potential for high-precision perception in manipulation tasks involving small or flexible objects, offering a cost-effective and compact alternative to conventional depth cameras.

Finally, Chapter 7 explored the extended use of the IMU subsystem for attitude estimation and control of satellite-like robotic systems. The IMU-based feedback loop was integrated into a reaction wheel setup, demonstrating precise attitude stabilization and control under varying rotational conditions. This experiment showcased the versatility of the developed sensing framework beyond terrestrial robotics, highlighting its applicability in space robotics and autonomous platform stabilization.

Collectively, the outcomes of this research advance the state of the art in multi-modal sensing for robotic manipulation. Through the integration of tactile, proximity, inertial, and visual modalities within compact, modular architectures, this thesis demonstrates how multi-sensory perception can enhance robotic intelligence, adaptability, and safety. The PI-based control framework and centroid-driven HRI strategy provide concrete evidence of how tactile and inertial feedback can be leveraged for responsive, human-aware robotic behavior.

The developed systems successfully bridge the gap between biological and robotic perception, offering scalable solutions that can be adapted for various robotic platforms, from industrial and agri-food systems to space robotics and assistive technologies. The compact fingertip sensor, in particular, represents a significant milestone, proving that human-like sensing capability can be achieved without compromising mechanical performance or integration flexibility.

Table 8.1 Comparison of the proposed multi-modal fingertip sensor with representative tactile sensing systems.

Specification	**Proposed Fingertip**	**XELA uSkin**	**PapillArray v2.0**	**BioTac SP**	**GelSight Mini**
Sensing modalities	Tactile + ToF + IMU	Tactile	Tactile	Tactile + vibration + temperature	Vision-based tactile
Tactile technology	Optoelectronic (IR LED–phototransistor)	Magnetic (Hall-based)	Optical displacement pillars	Fluid-filled electrodes	Camera + elastomer
Tactile resolution	3 × 3 (9 taxels)	4 × 4 (16 taxels)	3 × 3 (9 pillars)	24 electrodes	Image-based
Proximity sensing	ToF (pre-touch)	–	–	–	–
Inertial sensing	9-axis IMU	–	–	–	–
Physical dimensions (mm)	12 × 15 × 18	24 × 28 × 5.5	Vendor specification	Not reported	Vendor specification
Maximum sampling rate	500 Hz (tactile)	100 Hz	Up to 1000 Hz	73 Hz (pressure)	25 FPS
Communication interface	UART (USB), Wi-Fi	Vendor interface	USB	SPI / host board	USB
ROS compatibility	Yes	Optional / vendor-dependent	Yes	Varies	Yes
Form factor	Anthropomorphic fingertip	Flat flexible skin	Planar tactile array	Biomimetic fingertip	External vision sensor

8.2 Future Research Directions

Building upon the achievements of this thesis, several promising research directions are proposed:

1. **Sensor Fusion for Enhanced Spatial Awareness:** Future work will focus on combining tactile and IMU data for real-time pose estimation and contact orientation tracking. This fusion will enhance the robot's ability to adapt its grasp strategy dynamically during interaction with moving or irregular objects.

2. **Advanced Slippage Detection and Control:** The integration of data-driven algorithms for early slippage detection will be explored, leveraging tactile signal dynamics and machine learning models. Predictive control methods will be developed to automatically adjust grip force and maintain stable contact under varying load conditions.

3. **Monocular and Visual-Tactile Integration:** Building upon the stereo vision system, a monocular eye-on-hand setup will be investigated to reduce hardware complexity and enable compact deployment. Combining tactile and visual sensing will allow more robust perception in cluttered or unstructured environments.

4. **Full Integration with Anthropomorphic Hands:** Extending the compact multimodal fingertip design to all fingers of the SEEDs robotic hand will enable multi-finger coordination and advanced in-hand manipulation. This will support complex behaviors such as dynamic regrasping and fine object reorientation.

5. **Model-Based and Learning-Driven Control:** Future research will explore hybrid control architectures that combine physics-based models with machine learning algorithms to improve the autonomy, adaptability, and decision-making capabilities of robotic grippers in collaborative and assistive contexts.

6. **Cross-Domain Applications:** The sensing and control frameworks developed in this thesis will be further investigated for applications in space robotics, assistive prosthetics, and smart agricultural automation, where adaptability, precision, and safety are equally critical.

References

Abyarjoo, F., Barreto, A., Cofino, J., and Ortega, F. R. (2014). Implementing a sensor fusion algorithm for 3d orientation detection with inertial/magnetic sensors. In *Innovations and advances in computing, informatics, systems sciences, networking and engineering*, pages 305–310. Springer.

Caccavale, R., Finzi, A., Laudante, G., Natale, C., Pirozzi, S., and Villani, L. (2023). Manipulation of boltlike fasteners through fingertip tactile perception in robotic assembly. *IEEE/ASME Transactions on Mechatronics*, 29(2):820–831.

Caporali, A., Galassi, K., and Palli, G. (2023). Deformable linear objects 3d shape estimation and tracking from multiple 2d views. *IEEE Robotics and Automation Letters*, 8(6):3852–3859.

Cirillo, A., Costanzo, M., Laudante, G., and Pirozzi, S. (2021a). Tactile sensors for parallel grippers: Design and characterization. *Sensors*, 21(5):1915.

Cirillo, A., Laudante, G., and Pirozzi, S. (2021b). Proximity sensor for thin wire recognition and manipulation. *Machines*, 9(9):188.

Cirillo, A., Laudante, G., and Pirozzi, S. (2021c). Tactile sensor data interpretation for estimation of wire features. *Electronics*, 10(12):1458.

Cirillo, A., Laudante, G., and Pirozzi, S. (2022). Wire grasping by using proximity and tactile sensors. In *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS)*, pages 01–06. IEEE.

Cirillo, P., Laudante, G., and Pirozzi, S. (2021d). Vision-based robotic solution for wire insertion with an assigned label orientation. *IEEE Access*, 9:102278–102289.

Cop, K. P., Peters, A., Žagar, B. L., Hettegger, D., and Knoll, A. C. (2021). New metrics for industrial depth sensors evaluation for precise robotic applications. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5350–5356. IEEE.

Costanzo, M., De Maria, G., and Natale, C. (2019a). Two-fingered in-hand object handling based on force/tactile feedback. *IEEE Transactions on Robotics*, 36(1):157–173.

Costanzo, M., De Maria, G., Natale, C., and Pirozzi, S. (2019b). Design and calibration of a force/tactile sensor for dexterous manipulation. *Sensors*, 19(4):966.

Dahiya, R. S., Metta, G., Valle, M., and Sandini, G. (2010). Tactile sensing—from humans to humanoids. *IEEE Transactions on Robotics*, 26(1):1–20.

De Gregorio, D., Poggi, M., Ramirez, P. Z., Palli, G., Mattoccia, S., and Di Stefano, L. (2021). Beyond the baseline: 3d reconstruction of tiny objects with single camera stereo robot. *IEEE Access*, 9:119755–119765.

De Maria, G., Natale, C., and Pirozzi, S. (2012). Force/tactile sensor for robotic applications. *Sensors and Actuators A: Physical*, 175:60–72.

Eddine, B. J., Yijiang, D., and Zhong, W. (2015). Anti-disturbance pd controller design for flexible spacecraft attitude stabilization. In *2015 34th Chinese Control Conference (CCC)*, pages 5687–5690. IEEE.

Fazal, T. A., Laudante, G., Mirto, M., Pennacchio, O., and Pirozzi, S. (2024). Multi-modal sensor for fingertips of anthropomorphic grippers. In *2024 20th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, pages 1–7. IEEE.

GelSight (2022). Gelsight gs mini datasheet. Version dated December 20, 2022. Accessed: October 1, 2023.

Harjoko, A., Hujja, R. M., and Awaludin, L. (2017). Low-cost 3d surface reconstruction using stereo camera for small object. In *2017 International Conference on Signals and Systems (ICSigSys)*, pages 285–289. IEEE.

Heikkila, J. and Silvén, O. (1997). A four-step camera calibration procedure with implicit image correction. In *Proceedings of IEEE computer society conference on computer vision and pattern recognition*, pages 1106–1112. IEEE.

Hirschmuller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutual information. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, pages 807–814. IEEE.

Ismail, Z. and Varatharajoo, R. (2010). A study of reaction wheel configurations for a 3-axis satellite attitude control. *Advances in Space Research*, 45(6):750–759.

Jayaram, S. (2009). Design and analysis of nano momentum wheel for picosatellite attitude control system. *Aircraft Engineering and Aerospace Technology*, 81(5):424–431.

Kaboli, M., Feng, D., Cheng, K., and Cheng, G. (2020). Active tactile transfer learning for object discrimination in an unstructured environment using multimodal robotic skin. *The International Journal of Robotics Research*, 39(4):402–430.

Khamis, H., Xia, B., and Redmond, S. J. (2021). Real-time friction estimation for grip force control. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1608–1614. IEEE.

Kim, P. (2011). *Kalman filter for beginners: with MATLAB examples*. CreateSpace.

Kim, Y., Yuk, H., Zhao, R., Chester, S. A., and Zhao, X. (2018). Printing ferromagnetic domains for untethered fast-transforming soft materials. *Nature*, 558(7709):274–279.

Klimaszewski, J., Janczak, D., and Piorun, P. (2019). Tactile robotic skin with pressure direction detection. *Sensors*, 19(21):4697.

Liu, C., Zhuang, Y., Nasrollahi, A., Lu, L., Haider, M. F., and Chang, F.-K. (2020). Static tactile sensing for a robotic electronic skin via an electromechanical impedance-based approach. *Sensors*, 20(10):2830.

Loeb, G. E., Fishel, J., Wettels, N., Santos, V. J., and Peck, R. (2012). Measurement of sliding friction-induced vibrations for biomimetic tactile sensing. US Patent 8,181,540.

Makihata, M., Muroyama, M., Tanaka, S., Nakayama, T., Nonomura, Y., and Esashi, M. (2018). Design and fabrication technology of low profile tactile sensor with digital interface for whole body robot skin. *Sensors*, 18(7):2374.

Monguzzi, A., Cella, C., Zanchettin, A. M., and Rocco, P. (2023). Vision-based state and pose estimation for robotic bin picking of cables. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3114–3120. IEEE.

Muja, M. and Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340):2.

Neto, J. A. B., Lima, J. L., Pereira, A. I., and Costa, P. (2021). Low-cost 3d lidar-based scanning system for small objects. In *2021 22nd IEEE International Conference on Industrial Technology (ICIT)*, volume 1, pages 907–912. IEEE.

Palli, G. and Pirozzi, S. (2019a). A tactile-based wire manipulation system for manufacturing applications. *Robotics*, 8(2).

Palli, G. and Pirozzi, S. (2019b). A tactile-based wire manipulation system for manufacturing applications. *Robotics*, 8(2):46.

Robotiq (2020). 3-finger adaptive robot gripper. Accessed: 2024-06.

Sheng, H., Wei, S., and Yu, X. (2020). Image segmentation and object measurement based on stereo vision. In *2020 Chinese Automation Congress (CAC)*, pages 3637–3641. IEEE.

Shirazi, A. and Mirshams, M. (2014). Pyramidal reaction wheel arrangement optimization of satellite attitude control subsystem for minimizing power consumption. *International Journal of Aeronautical and Space Sciences*, 15(2):190–198.

Suen, M.-S. and Chen, R. (2018). Capacitive tactile sensor with concentric-shape electrodes for three-axial force measurement. In *Proceedings*, volume 2, page 708. MDPI.

Suzuki, T., Nakashima, A., Yasuda, K., and Natori, N. (1990). Attitude and orbit control subsystem for ers-1 and its subsystem test. In *Automatic Control in Aerospace 1989*, pages 81–86. Elsevier.

Wang, Y., Chen, J., and Mei, D. (2019). Flexible tactile sensor array for slippage and grooved surface recognition in sliding movement. *Micromachines*, 10(9):579.

Wettels, N., Santos, V. J., Johansson, R. S., and Loeb, G. E. (2008). Biomimetic tactile sensor array. *Advanced robotics*, 22(8):829–849.

Xela Robotics (2023). Xela robotics. Official website. Accessed: October 1, 2023.

Yuan, W., Dong, S., and Adelson, E. H. (2017). Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762.

Zhang, Z. (2002). A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334.

Zhaowei, S., Yunhai, G., Guodong, X., and Ping, H. (2004). The combined control algorithm for large-angle maneuver of hitsat-1 small satellite. *Acta Astronautica*, 54(7):463–469.

Zhou, Z., Zuo, R., Ying, B., Zhu, J., Wang, Y., Wang, X., and Liu, X. (2024). A sensory soft robotic gripper capable of learning-based object recognition and force-controlled grasping. *IEEE Transactions on Automation Science and Engineering*, 21(1):844–854.

Appendix A

Source Code of Multi-Modal Sensor Suites

A.1 Micro-controller Firmware

```
/**
  Generated Main Source File

  Company:
    Microchip Technology Inc.

  File Name:
    main.c

  Summary:
    This is the main file generated using PIC10 / PIC12 / PIC16 / PIC18 MCUs

  Description:
    This header file provides implementations for driver APIs for all
    modules selected in the GUI.
    Generation Information :
        Product Revision  :  PIC10 / PIC12 / PIC16 / PIC18 MCUs - 1.81.8
        Device            :  PIC16F19155
        Driver Version    :  2.00
*/
```

```
#include "mcc_generated_files/mcc.h"
#include "mcc_generated_files/examples/i2c1_master_example.h"


#define SAD                 0x6B
#define CTRL1               0x10
#define CTRL2               0x11
#define FUNC_CFG_ACCESS     0x01
#define PAGE_RW             0x17
#define PAGE_SEL            0x02
#define PAGE_ADDRESS        0x08
#define PAGE_VALUE          0x09
int counter;


uint8_t a;
uint16_t value;
uint8_t byteH;
uint8_t byteL;
uint8_t data[24];
int8_t data_imu[14];
int8_t pedometer;
uint8_t b;
uint8_t c;
uint8_t FingerName[5]="S010\0";

uint8_t sourceData[14] =
{0x20,0x21,0x22,0x23,0x24,0x25,0x26,0x27,0x28,0x29,0x2A,0x2B,0x2C,0x2D};

/*
                         Main application
 */
void main(void)
{
```

```
// initialize the device
 SYSTEM_Initialize();
 //I2C1_Initialize();
    I2C1_Write1ByteRegister(SAD,CTRL2,0b00001100);
    I2C1_Write1ByteRegister(SAD,CTRL1,0b00001100);



// When using interrupts, you need to set the Global and
Peripheral Interrupt Enable bits
// Use the following macros to:

// Enable the Global Interrupts
//INTERRUPT_GlobalInterruptEnable();



while (1)
{
    // Add your application code
            if (EUSART1_is_rx_ready())
    {
       a=EUSART1_Read();

       if (a=='a'){

       //value=ADCC_GetSingleConversion(RA0);
       value=ADCC_GetSingleConversion(channel_ANA0);    //U1
       ADCON0bits.ADGO = 0;
       ADPCH = 0x3b;
       data[0]=ADRESL;
       data[1]=ADRESH;

       b=0;
       EUSART1_Write(data[0]);
       while(b==0){if (EUSART1_is_tx_done()){b=1;}}
       b=0;
```

```
    EUSART1_Write(data[1]);
    while(b==0){if (EUSART1_is_tx_done()){b=1;}}

  // data[0]=ADRESH;data[1]=ADRESL;
  // ADCC_DischargeSampleCapacitor();
    //          __delay_us(100);
    value=ADCC_GetSingleConversion(channel_ANA1);    //U2
    ADCON0bits.ADGO = 0;
    ADPCH = 0x3b;
    data[2]=ADRESL;
    data[3]=ADRESH;

    b=0;
    EUSART1_Write(data[2]);
    while(b==0){if (EUSART1_is_tx_done()){b=1;}}
    b=0;
    EUSART1_Write(data[3]);
    while(b==0){if (EUSART1_is_tx_done()){b=1;}}


    value=ADCC_GetSingleConversion(channel_ANA2);    //U3
    ADCON0bits.ADGO = 0;
    ADPCH = 0x3b;
    data[4]=ADRESL;
    data[5]=ADRESH;

    b=0;
    EUSART1_Write(data[4]);
    while(b==0){if (EUSART1_is_tx_done()){b=1;}}
    b=0;
    EUSART1_Write(data[5]);
    while(b==0){if (EUSART1_is_tx_done()){b=1;}}


    value=ADCC_GetSingleConversion(channel_ANA3);    //U4
```

```
ADCON0bits.ADGO = 0;
ADPCH = 0x3b;
data[6]=ADRESL;
data[7]=ADRESH;

b=0;
EUSART1_Write(data[6]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}
b=0;
EUSART1_Write(data[7]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}


value=ADCC_GetSingleConversion(channel_ANA4);    //U5
ADCON0bits.ADGO = 0;
ADPCH = 0x3b;
data[8]=ADRESL;
data[9]=ADRESH;

b=0;
EUSART1_Write(data[8]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}
b=0;
EUSART1_Write(data[9]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}


value=ADCC_GetSingleConversion(channel_ANA6);    //U6
ADCON0bits.ADGO = 0;
ADPCH = 0x3b;
data[10]=ADRESL;
data[11]=ADRESH;

b=0;
EUSART1_Write(data[10]);
```

```
while(b==0){if (EUSART1_is_tx_done()){b=1;}}
b=0;
EUSART1_Write(data[11]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}

value=ADCC_GetSingleConversion(channel_ANB0);    //U8
ADCON0bits.ADGO = 0;
ADPCH = 0x3b;
data[14]=ADRESL;
data[15]=ADRESH;

b=0;
EUSART1_Write(data[14]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}
b=0;
EUSART1_Write(data[15]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}


value=ADCC_GetSingleConversion(channel_ANA7);    //U7
ADCON0bits.ADGO = 0;
ADPCH = 0x3b;
data[12]=ADRESL;
data[13]=ADRESH;

b=0;
EUSART1_Write(data[12]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}
b=0;
EUSART1_Write(data[13]);
while(b==0){if (EUSART1_is_tx_done()){b=1;}}



}
```

```
           if (a=='b'){for(uint8_t p = 0; p < 5; p++)
         EUSART1_Write(FingerName[p]);}

         if(a=='d')
         {
             for (counter=0; counter < 14; counter ++)
             {
                 data_imu[counter]
                 = I2C1_Read1ByteRegister(SAD, sourceData[counter]);
                 c=0;
                 EUSART1_Write(data_imu[counter]);
                 while(c==0){if (EUSART1_is_tx_done()){c=1;}}
             }

                }
            }
    }
}
/**
 End of File
*/
```

A.2 PC Side ROS Node

```
#include "read_sensor.h"

#include <ros/ros.h>
#include <serial/serial.h>

#include <algorithm>
#include <cstdint>
#include <string>
#include <vector>

/* ------------- Minimal globals ---------------- */
```

```
static Sensor_t        g_sensor{};
static serial::Serial* g_serial = nullptr;

static uint16_t g_offset_samples = 0U;

/* ------------- Small utilities ------------- */
static std::string sanitize(std::string s)
{
  std::replace(s.begin(), s.end(), '/', '_');
  return s;
}

static bool hasTag(const std::string& name, char tag)
{
  return name.find(tag) != std::string::npos;
}

/* Probe a serial port: send 'b', expect a 4-byte sensor name */
static bool probePort(const std::string& port, Sensor_t& out_sensor)
{
  try
  {
    serial::Serial ser(port, 1000000U, serial::Timeout::simpleTimeout(1000U));
    if (ser.write("b") == 0U) return false;

    const std::string name = ser.read(4U);
    if (name.size() != 4U) return false;

    out_sensor.SensorName = name;
    out_sensor.SerialPort = port;
    out_sensor.SensorID   = 0U;
    return true;
  }
  catch (...)
  {
```

```
    return false;
  }
}

/* Find first sensor on /dev/ttyUSB* or /dev/ttyACM* */
static bool autoDetectSensor(Sensor_t& out_sensor)
{
  const auto ports = serial::list_ports();
  for (const auto& p : ports)
  {
    if (p.port.find("ttyUSB") ==
    std::string::npos && p.port.find("ttyACM") ==
    std::string::npos)
      continue;

    if (probePort(p.port, out_sensor))
      return true;
  }
  return false;
}

/* Configure sensor metadata (channels/imu count) */
static void configureSensor(Sensor_t& s)
{
  s.NumIMU = 0U;
  s.pedometer = 0U;

  if (hasTag(s.SensorName, 'P'))
  {
    // Proximity: device tells number of elements via 'c'
    (void)g_serial->write("c");
    g_serial->read(&s.NumSensingElements, 1U);
  }
  else if (hasTag(s.SensorName, 'F'))
  {
```

```
    s.NumSensingElements = 25U;
  }
  else if (hasTag(s.SensorName, ’S’))
  {
    s.NumSensingElements = 12U;
    s.NumIMU = 7U;
    s.pedometer = 1U;
  }

  if (s.NumSensingElements == 0U && s.NumIMU == 0U)
  {
    ROS_ERROR("Sensor has no valid sensing elements.");
    std::exit(EXIT_FAILURE);
  }
}

int main(int argc, char** argv)
{
  ros::init(argc, argv, "read_sensor_node");
  ros::NodeHandle nh("~");

  /* Optional parameter: serial_port (e.g., /dev/ttyUSB0) */
  std::string port_param;
  nh.param<std::string>("serial_port", port_param, "");

  /* Detect sensor */
  bool ok = false;
  if (!port_param.empty())
  {
    ok = probePort(port_param, g_sensor);
  }
  else
  {
    ok = autoDetectSensor(g_sensor);
  }
```

```
if (!ok)
{
  ROS_ERROR("No compatible sensor found (ttyUSB*/ttyACM*).");
  return 1;
}

/* Open serial */
g_serial = new serial::Serial(g_sensor.SerialPort, 1000000U,
serial::Timeout::simpleTimeout(1000U));
configureSensor(g_sensor);

ROS_INFO("Using sensor %s on %s", g_sensor.SensorName.c_str(),
g_sensor.SerialPort.c_str());

/* Publishers */
ros::Publisher pub_prox;
ros::Publisher pub_tactile_raw;
ros::Publisher pub_tactile;
ros::Publisher pub_imu;

if (hasTag(g_sensor.SensorName, 'P'))
{
  pub_prox =
  nh.advertise<read_sensor::proximity_sensor_data_all>("proximity/raw", 1);
}
else
{
  pub_tactile_raw =
  nh.advertise<read_sensor::tactile_sensor_data>("tactile/raw", 1);
  pub_tactile     =
  nh.advertise<read_sensor::tactile_sensor_data>("tactile", 1);
  pub_imu         = nh.advertise<read_sensor::data_imu>("imu", 1);

  g_sensor.Offset.assign(g_sensor.NumSensingElements, 0.0f);
```

```
  g_offset_samples = 0U;
}

/* Loop rate */
const uint16_t rate_hz = hasTag(g_sensor.SensorName, 'P') ? 50U : 500U;
ros::Rate rate(rate_hz);

std::vector<uint8_t> raw;
std::vector<uint8_t> imu_bytes;

read_sensor::proximity_sensor_data_all msg_prox;
read_sensor::tactile_sensor_data msg_tactile_raw;
read_sensor::tactile_sensor_data msg_tactile;
read_sensor::data_imu msg_imu;

while (ros::ok())
{
  /* ---------------- Proximity ---------------- */
  if (hasTag(g_sensor.SensorName, 'P'))
  {
    (void)g_serial->write("a");

    msg_prox.distance_measure.clear();
    msg_prox.return_rate.clear();
    msg_prox.reference_rate.clear();
    msg_prox.return_conv_time.clear();
    msg_prox.reference_conv_time.clear();
    msg_prox.return_count.clear();
    msg_prox.reference_count.clear();
    msg_prox.return_ambient_count.clear();
    msg_prox.reference_ambient_count.clear();

    const uint8_t bytes_per_channel = 18U;
    g_serial->read(raw, g_sensor.NumSensingElements * bytes_per_channel);
```

```
    for (uint8_t k = 0U; k < g_sensor.NumSensingElements; ++k)
    {
      const size_t b = static_cast<size_t>(k) * bytes_per_channel;

      msg_prox.distance_measure.push_back(static_cast<float>(raw.at(b)));
      msg_prox.return_rate.push_back(static_cast<float>
              (raw.at(b + 1U) + 256U * raw.at(b + 2U)) / 128.0f);
      msg_prox.reference_rate.push_back(static_cast<float>
              (raw.at(b + 3U) + 256U * raw.at(b + 4U)) / 128.0f);
      msg_prox.return_conv_time.push_back(static_cast<float>
              (raw.at(b + 5U) + 256U * raw.at(b + 6U)));
      msg_prox.reference_conv_time.push_back(static_cast<float>
              (raw.at(b + 7U)));
      msg_prox.return_count.push_back(static_cast<float>
              (raw.at(b + 8U) + 256U * raw.at(b + 9U) +
              65336U * raw.at(b + 10U)));
      msg_prox.reference_count.push_back(static_cast<float>
      (raw.at(b + 11U) + 256U * raw.at(b + 12U) + 65336U *
      raw.at(b + 13U)));
      msg_prox.return_ambient_count.push_back(static_cast<float>
      (raw.at(b + 14U) + 256U * raw.at(b + 15U)));
      msg_prox.reference_ambient_count.push_back(static_cast<float>
      (raw.at(b + 16U) + 256U * raw.at(b + 17U)));
    }

    msg_prox.header.stamp = ros::Time::now();
    pub_prox.publish(msg_prox);
  }

  /* ---------------- Tactile + IMU ---------------- */
  else
  {
    // Tactile read
    (void)g_serial->write("a");
    msg_tactile_raw.tactile_sensor_data.clear();
```

```
msg_tactile.tactile_sensor_data.clear();

g_serial->read(raw, g_sensor.NumSensingElements * 2U);

for (uint8_t k = 0U; k < g_sensor.NumSensingElements; ++k)
{
  const uint16_t adc = static_cast<uint16_t>(raw.at(2U * k) +
  (raw.at(2U * k + 1U) << 8));
  const float v = static_cast<float>(adc) * 3.3f / 4096.0f;

  msg_tactile_raw.tactile_sensor_data.push_back(v);

  if (g_offset_samples < TACTILE_OFFSET_NUM_SAMPLES)
    g_sensor.Offset.at(k) += v;
  else if (g_offset_samples == TACTILE_OFFSET_NUM_SAMPLES)
    g_sensor.Offset.at(k) /=
    static_cast<float>(TACTILE_OFFSET_NUM_SAMPLES);
  else
    msg_tactile.tactile_sensor_data.push_back(v-g_sensor.Offset.at(k));
}

if (g_offset_samples <= TACTILE_OFFSET_NUM_SAMPLES) ++g_offset_samples;

msg_tactile.header.stamp = ros::Time::now();
pub_tactile_raw.publish(msg_tactile_raw);
pub_tactile.publish(msg_tactile);

// IMU read (only if present)
if (g_sensor.NumIMU > 0U)
{
  (void)g_serial->write("d");
  msg_imu.tactile_data_imu.clear();

  g_serial->read(imu_bytes, g_sensor.NumIMU * 2U);
```

```
        // Same conversion as your original code (first 7 channels)
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(0) +
        imu_bytes.at(1)  * 256U) / 256.0f) + 25.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(2) +
        imu_bytes.at(3)  * 256U)) * 4.375f / 1000.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(4) +
        imu_bytes.at(5)  * 256U)) * 4.375f / 1000.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(6) +
        imu_bytes.at(7)  * 256U)) * 4.375f / 1000.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(8) +
        imu_bytes.at(9)  * 256U)) * 0.061f / 1000.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(10) +
        imu_bytes.at(11) * 256U)) * 0.061f / 1000.0f);
        msg_imu.tactile_data_imu.push_back(((int16_t)(imu_bytes.at(12) +
        imu_bytes.at(13) * 256U)) * 0.061f / 1000.0f);

        msg_imu.header.stamp = ros::Time::now();
        pub_imu.publish(msg_imu);
      }
    }

    ros::spinOnce();
    rate.sleep();
  }

  delete g_serial;
  g_serial = nullptr;
  return 0;
}
```

Appendix B

Source Code of Multi-Modal Fingertip

B.1 Micro-controller Firmware

```
#include <Arduino.h>
#include <Wire.h>
#include <vl53l4cd_class.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BNO055.h>
#include <utility/imumaths.h>

// LEDC settings
#define LEDC_TIMER_7_BIT 7
#define LEDC_BASE_FREQ 600000

// LED pins and corresponding channels
#define LED_PIN_1 20
#define LED_PIN_2 33
#define LED_PIN_3 10
#define LED_PIN_4 7
#define LED_PIN_5 8
#define LED_PIN_6 5
#define LED_PIN_7 19
#define LED_PIN_8 9
#define LED_PIN_9 15
```

```
#define LEDC_CHANNEL_1 0
#define LEDC_CHANNEL_2 1
#define LEDC_CHANNEL_3 2
#define LEDC_CHANNEL_4 3
#define LEDC_CHANNEL_5 4
#define LEDC_CHANNEL_6 5
#define LEDC_CHANNEL_7 6
#define LEDC_CHANNEL_8 7
#define LEDC_CHANNEL_9 9

// ADC Pins
#define ADC1 4
#define ADC2 2
#define ADC3 13
#define ADC4 12
#define ADC5 32
#define ADC6 14
#define ADC7 27
#define ADC8 26
#define ADC9 25

// ToF Sensor
#define DEV_I2C Wire
VL53L4CD sensor_vl53l4cd_sat(&DEV_I2C, -1);

// IMU Sensor
Adafruit_BNO055 bno = Adafruit_BNO055(55, 0x28, &Wire);

// Protocol Constants
const uint8_t SYNC_HEADER[2] = {0xAA, 0x55};
const uint8_t FOOTER = 0x0D;
const size_t NUM_TACTILE_SENSORS = 9;
const size_t NUM_TOF_SENSORS = 1;
const size_t TACTILE_PACKET_SIZE = 2 + (NUM_TACTILE_SENSORS * 2) + 1;
const size_t TOF_PACKET_SIZE = 2 + (NUM_TOF_SENSORS * 2) + 1;
```

```
const size_t IMU_PACKET_SIZE = 2 + (19 * sizeof(float)) + 1;
                                    // 19 float values + checksum

// Sensor Identification
const char SENSOR_ID[] = "S001";  // 4 bytes including null terminator
const char ID_CMD = ’b’;           // Identification command
const char TACTILE_DATA_CMD = ’a’; // Tactile data request command
const char TOF_DATA_CMD = ’c’;     // ToF data request command
const char IMU_DATA_CMD = ’d’;     // IMU data request command

// Individual duty cycles for each LED
int dutyCycle1 = 59;
int dutyCycle2 = 51.5;
int dutyCycle3 = 55.5;
int dutyCycle4 = 51.5;
int dutyCycle5 = 52;
int dutyCycle6 = 59;
int dutyCycle7 = 49;
int dutyCycle9 = 54;

// EMA filter settings
#define ALPHA 0.1      // Smoothing factor
                (0.1 = heavy smoothing, 0.3 = light smoothing)

// EMA filter variables for each ADC
float ema_adc1 = 0;
float ema_adc2 = 0;
float ema_adc3 = 0;
float ema_adc4 = 0;
float ema_adc5 = 0;
float ema_adc6 = 0;
float ema_adc7 = 0;
float ema_adc8 = 0;
float ema_adc9 = 0;
```

```
uint8_t tactileTxBuffer[TACTILE_PACKET_SIZE];
uint8_t tofTxBuffer[TOF_PACKET_SIZE];
uint8_t imuTxBuffer[IMU_PACKET_SIZE];

// Function to apply EMA filter
float applyEMA(float new_val, float prev_ema, float alpha) {
  return alpha * new_val + (1 - alpha) * prev_ema;
}

void setup() {
  Serial.begin(1000000);

  // Initialize I2C for ToF sensor
  DEV_I2C.begin();

  // Configure ToF sensor
  sensor_vl53l4cd_sat.VL53L4CD_SetRangeTiming(200, 0);
  sensor_vl53l4cd_sat.VL53L4CD_StartRanging();

  // Initialize IMU sensor
  if (!bno.begin()) {
    Serial.println("ERR:BNO055 not detected");
    while (1);
  }
  delay(1000); // Allow sensor to stabilize

  // Initialize EMA with first ADC reading
  ema_adc1 = analogReadMilliVolts(ADC1);
  ema_adc2 = analogReadMilliVolts(ADC2);
  ema_adc3 = analogReadMilliVolts(ADC3);
  ema_adc4 = analogReadMilliVolts(ADC4);
  ema_adc5 = analogReadMilliVolts(ADC5);
  ema_adc6 = analogReadMilliVolts(ADC6);
  ema_adc7 = analogReadMilliVolts(ADC7);
  ema_adc8 = analogReadMilliVolts(ADC8);
```

```
  ema_adc9 = analogReadMilliVolts(ADC9);

  // Configure LEDC channels for each pin
  ledcAttachChannel(LED_PIN_1, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_1);
  ledcAttachChannel(LED_PIN_2, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_2);
  ledcAttachChannel(LED_PIN_3, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_3);
  ledcAttachChannel(LED_PIN_4, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_4);
  ledcAttachChannel(LED_PIN_5, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_5);
  ledcAttachChannel(LED_PIN_6, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_6);
  ledcAttachChannel(LED_PIN_7, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_7);
  ledcAttachChannel(LED_PIN_8, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_3);
  ledcAttachChannel(LED_PIN_9, LEDC_BASE_FREQ,
            LEDC_TIMER_7_BIT, LEDC_CHANNEL_8);
}

void sendIdentification() {
  Serial.write(SENSOR_ID, sizeof(SENSOR_ID)-1);
                          // Send without null terminator
}

void buildTactileDataPacket() {
  tactileTxBuffer[0] = SYNC_HEADER[0];
  tactileTxBuffer[1] = SYNC_HEADER[1];

  int16_t* dataPtr = (int16_t*)&tactileTxBuffer[2];

  // Using EMA values for the packet
```

```
  dataPtr[0] = (int16_t)ema_adc1;
  dataPtr[1] = (int16_t)ema_adc2;
  dataPtr[2] = (int16_t)ema_adc3;
  dataPtr[3] = (int16_t)ema_adc4;
  dataPtr[4] = (int16_t)ema_adc5;
  dataPtr[5] = (int16_t)ema_adc6;
  dataPtr[6] = (int16_t)ema_adc7;
  dataPtr[7] = (int16_t)ema_adc8;
  dataPtr[8] = (int16_t)ema_adc9;

  uint8_t checksum = 0;
  for(size_t i = 2; i < TACTILE_PACKET_SIZE - 1; i++) {
    checksum ^= tactileTxBuffer[i];
  }
  tactileTxBuffer[TACTILE_PACKET_SIZE - 1] = checksum;
}

void buildTofDataPacket() {
  uint8_t NewDataReady = 0;
  VL53L4CD_Result_t results;
  uint8_t status;

  tofTxBuffer[0] = SYNC_HEADER[0];
  tofTxBuffer[1] = SYNC_HEADER[1];

  do{
    status =
    sensor_vl53l4cd_sat.VL53L4CD_CheckForDataReady(&NewDataReady);
  } while (!NewDataReady);

  int16_t* dataPtr = (int16_t*)&tofTxBuffer[2];

  if ((!status) && (NewDataReady != 0)) {
    // (Mandatory) Clear HW interrupt to restart measurements
    sensor_vl53l4cd_sat.VL53L4CD_ClearInterrupt();
```

```
    sensor_vl53l4cd_sat.VL53L4CD_GetResult(&results);
    dataPtr[0] = (int16_t)results.distance_mm;
  }

  uint8_t checksum = 0;
  for(size_t i = 2; i < TOF_PACKET_SIZE - 1; i++) {
    checksum ^= tofTxBuffer[i];
  }
  tofTxBuffer[TOF_PACKET_SIZE - 1] = checksum;
}

void buildImuDataPacket() {
  // Read all sensor data
  sensors_event_t orientationData, angVelocityData, linearAccelData,
                  magnetometerData, accelerometerData, gravityData;

  bno.getEvent(&orientationData,
              Adafruit_BNO055::VECTOR_EULER);
  bno.getEvent(&angVelocityData,
              Adafruit_BNO055::VECTOR_GYROSCOPE);
  bno.getEvent(&linearAccelData,
              Adafruit_BNO055::VECTOR_LINEARACCEL);
  bno.getEvent(&magnetometerData,
              Adafruit_BNO055::VECTOR_MAGNETOMETER);
  bno.getEvent(&accelerometerData,
              Adafruit_BNO055::VECTOR_ACCELEROMETER);
  bno.getEvent(&gravityData,Adafruit_BNO055::VECTOR_GRAVITY);

  // Get temperature and calibration
  int8_t boardTemp = bno.getTemp();
  uint8_t system, gyro, accel, mag = 0;
  bno.getCalibration(&system, &gyro, &accel, &mag);

  imuTxBuffer[0] = SYNC_HEADER[0];
  imuTxBuffer[1] = SYNC_HEADER[1];
```

```
float* dataPtr = (float*)&imuTxBuffer[2];

// Orientation
dataPtr[0] = orientationData.orientation.x;
dataPtr[1] = orientationData.orientation.y;
dataPtr[2] = orientationData.orientation.z;

// Gyro
dataPtr[3] = angVelocityData.gyro.x;
dataPtr[4] = angVelocityData.gyro.y;
dataPtr[5] = angVelocityData.gyro.z;

// Linear acceleration
dataPtr[6] = linearAccelData.acceleration.x;
dataPtr[7] = linearAccelData.acceleration.y;
dataPtr[8] = linearAccelData.acceleration.z;

// Magnetometer
dataPtr[9] = magnetometerData.magnetic.x;
dataPtr[10] = magnetometerData.magnetic.y;
dataPtr[11] = magnetometerData.magnetic.z;

// Accelerometer
dataPtr[12] = accelerometerData.acceleration.x;
dataPtr[13] = accelerometerData.acceleration.y;
dataPtr[14] = accelerometerData.acceleration.z;

// Gravity
dataPtr[15] = gravityData.acceleration.x;
dataPtr[16] = gravityData.acceleration.y;
dataPtr[17] = gravityData.acceleration.z;

// Temperature (converted to float)
dataPtr[18] = (float)boardTemp;
```

```
  uint8_t checksum = 0;
  for(size_t i = 2; i < IMU_PACKET_SIZE - 1; i++) {
    checksum ^= imuTxBuffer[i];
  }
  imuTxBuffer[IMU_PACKET_SIZE - 1] = checksum;
}

void loop() {
  // Write different duty cycles to each channel
  ledcWriteChannel(LEDC_CHANNEL_1, dutyCycle1);
  ledcWriteChannel(LEDC_CHANNEL_2, dutyCycle2);
  ledcWriteChannel(LEDC_CHANNEL_3, dutyCycle3);
  ledcWriteChannel(LEDC_CHANNEL_4, dutyCycle4);
  ledcWriteChannel(LEDC_CHANNEL_5, dutyCycle5);
  ledcWriteChannel(LEDC_CHANNEL_6, dutyCycle6);
  ledcWriteChannel(LEDC_CHANNEL_7, dutyCycle7);
  ledcWriteChannel(LEDC_CHANNEL_3, dutyCycle3);
  ledcWriteChannel(LEDC_CHANNEL_8, dutyCycle9);

  // Update EMA values
  ema_adc1 = applyEMA(analogReadMilliVolts(ADC1), ema_adc1, ALPHA);
  ema_adc2 = applyEMA(analogReadMilliVolts(ADC2), ema_adc2, ALPHA);
  ema_adc3 = applyEMA(analogReadMilliVolts(ADC3), ema_adc3, ALPHA);
  ema_adc4 = applyEMA(analogReadMilliVolts(ADC4), ema_adc4, ALPHA);
  ema_adc5 = applyEMA(analogReadMilliVolts(ADC5), ema_adc5, ALPHA);
  ema_adc6 = applyEMA(analogReadMilliVolts(ADC6), ema_adc6, ALPHA);
  ema_adc7 = applyEMA(analogReadMilliVolts(ADC7), ema_adc7, ALPHA);
  ema_adc8 = applyEMA(analogReadMilliVolts(ADC8), ema_adc8, ALPHA);
  ema_adc9 = applyEMA(analogReadMilliVolts(ADC9), ema_adc9, ALPHA);

  // Handle serial commands
  if(Serial.available() > 0) {
    char cmd = Serial.read();
```

```
    if(cmd == ID_CMD) {
      sendIdentification();
    }
    else if(cmd == TACTILE_DATA_CMD) {
      buildTactileDataPacket();
      Serial.write(tactileTxBuffer, TACTILE_PACKET_SIZE);
      Serial.flush();
    }
    else if(cmd == TOF_DATA_CMD) {
      buildTofDataPacket();
      Serial.write(tofTxBuffer, TOF_PACKET_SIZE);
      Serial.flush();
    }
    else if(cmd == IMU_DATA_CMD) {
      buildImuDataPacket();
      Serial.write(imuTxBuffer, IMU_PACKET_SIZE);
      Serial.flush();
    }
  }

  delay(2);
}
```

B.2 PC Side ROS Node

```
    #include <ros/ros.h>
#include <serial/serial.h>
#include <std_msgs/Int16MultiArray.h>
#include <std_msgs/Int16.h>
#include <geometry_msgs/Vector3Stamped.h>
#include <std_msgs/Float32.h>

class MultiSensorNode {
private:
    ros::NodeHandle nh_;
```

```
    serial::Serial ser_;
    ros::Publisher tactile_pub_;
    ros::Publisher tof_pub_;

    // IMU Data Publishers
    ros::Publisher orientation_pub_;
    ros::Publisher gyro_pub_;
    ros::Publisher linear_accel_pub_;
    ros::Publisher mag_pub_;
    ros::Publisher accel_pub_;
    ros::Publisher gravity_pub_;
    ros::Publisher temp_pub_;

    std::string sensor_id_;
    int num_tactile_sensors_;
    bool initialized_;

public:
    MultiSensorNode() : num_tactile_sensors_(9), initialized_(false)
    {   tactile_pub_ =
        nh_.advertise<std_msgs::Int16MultiArray>("tactile_data", 1);
        tof_pub_ = nh_.advertise<std_msgs::Int16>("tof_data", 1);

        // Initialize IMU data publishers
        orientation_pub_ =
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/orientation",1);
        gyro_pub_ =
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/gyro",1);
        linear_accel_pub_ =
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/linear_accel",1);
        mag_pub_ =
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/mag",1);
        accel_pub_ =
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/accel",1);
        gravity_pub_ =
```

```
        nh_.advertise<geometry_msgs::Vector3Stamped>("imu/gravity",1);
        temp_pub_ =
        nh_.advertise<std_msgs::Float32>("imu/temp",1);

        printf("\033[34m\n\n=== Multi-Sensor Controller ===\033[0m\n");
        printf("\033[34m Detected %d tactile sensors
        configured\033[0m\n", num_tactile_sensors_);
        printf("\033[34m Enter '0' to begin initialization...\033[0m\n");
    }

    bool initializeSerial() {
        try {
            ser_.setPort("/dev/ttyUSB0");
            ser_.setBaudrate(1000000);
            serial::Timeout to = serial::Timeout::simpleTimeout(1000);
            ser_.setTimeout(to);
            ser_.open();

            std::string cmd = "setserial ";
            cmd += ser_.getPort();
            cmd += " low_latency";
            system(cmd.c_str());

            if(ser_.isOpen()) {
            std::cout << "\033[32m" << "Serial port ...
            initialized successfully" << "\033[0m" << std::endl;
            return true;
            }
        } catch (const std::exception& e) {
            ROS_ERROR("Serial port error: %s", e.what());
        }
        return false;
    }

    bool getSensorID() {
```

```
        if(!ser_.isOpen()) return false;

        ser_.write("b");
        std::string response = ser_.read(4); // Expect "S001"

        if(response.empty()) {
            ROS_WARN("No response to ID request");
            return false;
        }

        sensor_id_ = response;
        std::cout << "\033[34m" << "Sensor ID: "...
        << sensor_id_ << "\033[0m" << std::endl;

        return (response[0] == ’S’);
    }

    void requestTactileData() {
        uint8_t buffer[21]; // Header + 9x2 bytes + checksum

        ser_.write("a");
        size_t bytes_read = ser_.read(buffer, sizeof(buffer));

        if(bytes_read == sizeof(buffer)) {
            uint8_t checksum = 0;
            for(size_t i = 2; i < 20; i++) checksum ^= buffer[i];

            if(checksum == buffer[20]) {
                std_msgs::Int16MultiArray msg;
                msg.data.resize(num_tactile_sensors_);
                memcpy(&msg.data[0], &buffer[2], 18);
                tactile_pub_.publish(msg);
            } else {
                ROS_WARN_THROTTLE(1, "Tactile data checksum failed");
            }
```

```
    } else if(bytes_read > 0) {
      ROS_WARN("Incomplete tactile packet: %zu bytes", bytes_read);
    }
}

void requestTofData() {
    uint8_t buffer[5]; // Header + 1x2 bytes + checksum

    ser_.write("c");
    size_t bytes_read = ser_.read(buffer, sizeof(buffer));

    if(bytes_read == sizeof(buffer)) {
        uint8_t checksum = 0;
        for(size_t i = 2; i < 4; i++) checksum ^= buffer[i];

        if(checksum == buffer[4]) {
            std_msgs::Int16 msg;
            memcpy(&msg.data, &buffer[2], 2);
            tof_pub_.publish(msg);
        } else {
            ROS_WARN_THROTTLE(1, "ToF data checksum failed");
        }
    } else if(bytes_read > 0) {
        ROS_WARN("Incomplete ToF packet: %zu bytes", bytes_read);
    }
}

void requestImuData() {
    const size_t imu_packet_size = 2 + (19 * sizeof(float)) + 1;
    uint8_t buffer[imu_packet_size];

    ser_.write("d");
    size_t bytes_read = ser_.read(buffer, imu_packet_size);

    if(bytes_read == imu_packet_size) {
```

```
uint8_t checksum = 0;
for(size_t i = 2; i < imu_packet_size - 1; i++)
checksum ^= buffer[i];

if(checksum == buffer[imu_packet_size - 1]) {
    float* float_data = reinterpret_cast<float*>(&buffer[2]);
    ros::Time current_time = ros::Time::now();

    // Publish Orientation (Euler angles)
    geometry_msgs::Vector3Stamped orientation_msg;
    orientation_msg.header.stamp = current_time;
    orientation_msg.header.frame_id = "imu_link";
    orientation_msg.vector.x = float_data[0];
    orientation_msg.vector.y = float_data[1];
    orientation_msg.vector.z = float_data[2];
    orientation_pub_.publish(orientation_msg);

    // Publish Gyro
    geometry_msgs::Vector3Stamped gyro_msg;
    gyro_msg.header = orientation_msg.header;
    gyro_msg.vector.x = float_data[3];
    gyro_msg.vector.y = float_data[4];
    gyro_msg.vector.z = float_data[5];
    gyro_pub_.publish(gyro_msg);

    // Publish Linear Acceleration
    geometry_msgs::Vector3Stamped linear_accel_msg;
    linear_accel_msg.header = orientation_msg.header;
    linear_accel_msg.vector.x = float_data[6];
    linear_accel_msg.vector.y = float_data[7];
    linear_accel_msg.vector.z = float_data[8];
    linear_accel_pub_.publish(linear_accel_msg);

    // Publish Magnetometer
    geometry_msgs::Vector3Stamped mag_msg;
```

```
            mag_msg.header = orientation_msg.header;
            mag_msg.vector.x = float_data[9];
            mag_msg.vector.y = float_data[10];
            mag_msg.vector.z = float_data[11];
            mag_pub_.publish(mag_msg);

            // Publish Accelerometer
            geometry_msgs::Vector3Stamped accel_msg;
            accel_msg.header = orientation_msg.header;
            accel_msg.vector.x = float_data[12];
            accel_msg.vector.y = float_data[13];
            accel_msg.vector.z = float_data[14];
            accel_pub_.publish(accel_msg);

            // Publish Gravity
            geometry_msgs::Vector3Stamped gravity_msg;
            gravity_msg.header = orientation_msg.header;
            gravity_msg.vector.x = float_data[15];
            gravity_msg.vector.y = float_data[16];
            gravity_msg.vector.z = float_data[17];
            gravity_pub_.publish(gravity_msg);

            // Publish Temperature
            std_msgs::Float32 temp_msg;
            temp_msg.data = float_data[18];
            temp_pub_.publish(temp_msg);

        } else {
            ROS_WARN_THROTTLE(1, "IMU data checksum failed");
        }
    } else if(bytes_read > 0) {
        ROS_WARN("Incomplete IMU packet: %zu bytes", bytes_read);
    }
}
```

```
    void requestDataLoop() {
        //ros::Rate rate(100); // 100Hz

        while(ros::ok()) {
            //requestTactileData();
            //requestTofData();
            requestImuData();
            //rate.sleep();
        }
    }

    void mainLoop() {
        std::string user_input;

        while(ros::ok() && !initialized_) {
            std::cout << "> ";
            std::getline(std::cin, user_input);

            if(user_input == "0") {
                if(initializeSerial()) {
                    if(getSensorID()) {
                        initialized_ = true;
                        std::cout << "\033[32m" << ...
                        "Starting data acquisition"<<"\033[0m"<< std::endl;
                        requestDataLoop();
                    } else {
                        ROS_ERROR("Invalid sensor ID");
                    }
                }
            } else {
                ROS_WARN("Please enter ’0’ to begin");
            }
        }
    }
};
```

```
int main(int argc, char** argv) {
    ros::init(argc, argv, "multi_sensor_node");
    MultiSensorNode node;
    node.mainLoop();
    return 0;
}
```