Hybrid Rigid–Soft Industrial Gripper: Actuation, Design Enhancement, Multi-Modal Sensorization, and Real-Time Coordinated Control for Automotive Assembly

Università di Genova

Muhammad Usman Khalid

Department of Mechanical, Energy, Management, and Transport Engineering (DIME)

University of Genova

Supervisor:
Prof. Prof. Matteo Zoppi

In partial fulfillment of the requirements for the degree of

Doctor of Philosophy

November, 2025

Abstract

The growing complexity of industrial automation and the shift toward human–robot collaborative manufacturing demand robotic grippers that combine mechanical precision, adaptive compliance, and intelligent sensing. This doctoral thesis addresses these needs through comprehensive research on actuation mechanisms, kinematic analysis, design enhancement, advanced sensorization, and industrial implementation of a versatile universal gripper system capable of handling components from delicate items to complex rigid parts. In parallel, it contributes to industrial robot control within the EU Horizon SESTOSENSO project.

The work begins with a detailed kinematic analysis of a novel three-finger gripper architecture. Each rigid mechanical finger integrates a Chebyshev–parallelogram linkage mechanism with a thermoplastic polyurethane (TPU) contact interface. The mechanism produces near-linear trajectories with a deviation of ± 0.033 mm and a mechanical advantage of 6.06:1. Independent actuation is achieved using JVL stepper motors with embedded programmable logic controllers (PLCs), communicating via Modbus remote terminal unit (RTU). The control architecture supports torque-based and velocity-based stall detection, both operating in real time with configurable thresholds. These strategies enable reliable grasping without dedicated force sensors by leveraging internal motor feedback parameters.

Gripper enhancement is guided by a quantitative deflection coefficient to assess finger wrapping and by quasi-static force–displacement testing. The original four-bar parallelogram was redesigned into a six-bar linkage with compliant pads. This eliminates link interference limitations while preserving the essential kinematics, resulting in adaptive grasping capability. Pull-out tests demonstrated improved force profiles for complex automotive parts and reliable manipulation of objects from 100 g to 7.5 kg.

Vision-based sensorization was achieved through an embedded Raspberry Pi Camera V3. The integration of vision-based sensing within

the additively manufactured soft finger structure establishes the feasibility of achieving multiple sensing modalities with a single compact embedded system while retaining the characteristic properties of the fingers. The proposed system successfully estimates normal interaction forces, measures internal deformation (Z-displacement), classifies the position of the applied force, and detects slip events with the complete sensing pipeline processed on an embedded platform while avoiding complex signal disambiguation challenges and occlusion issues. Complementing this, a fully flexible resistive sensor was fabricated via fused deposition modeling (FDM) printing and embedded in the finger for contact and bending detection. A novel light-angle sensor array was also developed using a custom four-layer rigid-flex printed circuit board (PCB), where prototype sensors successfully demonstrate distributed tactile sensing capabilities.

The universal gripper and sensing systems were validated on a COMAU six-degrees-of-freedom (6-DOF) industrial robot in diverse grasping trials, confirming adaptability, robustness, and sensing reliability. Separately, within the SESTOSENSO project, real-time control strategies were developed for coordinating a KUKA KR150 robot with a UR10 cobot via robot sensor interface (RSI) and robot operating system (ROS) in an automotive roof assembly task. This work addressed control architecture, real-time trajectory correction, and safe human–robot collaboration in confined, visually occluded environments.

This thesis advances the state of the art in hybrid gripper systems by integrating rigid precision, soft adaptability, and intelligent sensing with industrially validated control strategies. The outcomes directly support Industry 4.0/5.0 objectives, enabling flexible, high-performance automation adaptable to diverse manufacturing requirements.

Contents

List of Figures

Nomenclature

Acronyms / Abbreviations

ADC	Analog-to-Digital Converter
AI	Artificial Intelligence
API	Application Programming Interface
ASTM	American Society for Testing and Materials
CAD	Computer-Aided Design
CNN	Convolutional Neural Network
COBOT	Collaborative Robot
CRF	Centro Ricerche Fiat
CSV	Comma-Separated Values
DOF	Degrees of Freedom
ETH	Ethernet
EU	European Union
FDM	Fused Deposition Modeling
FEA	Finite Element Analysis
FPC	Flexible Printed Circuit
FPS	Frames Per Second
FR4	Flame Retardant 4 (PCB substrate)
GUI	Graphical User Interface
HRI	Human-Robot Interaction
Hz	Hertz
IEEE	Institute of Electrical and Electronics Engineers
IFAC	International Federation of Automatic Control
IMU	Inertial Measurement Unit

IR	Infrared
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
KRC	KUKA Robot Controller
KRL	KUKA Robot Language
LED	Light-Emitting Diode
MA	Mechanical Advantage
PCB	Printed Circuit Board
PID	Proportional-Integral-Derivative
PLA	Polylactic Acid
PLC	Programmable Logic Controller
PWM	Pulse Width Modulation
ROI	Region of Interest
ROS	Robot Operating System
RS-485	Recommended Standard 485
RSI	Robot Sensor Interface
RTDE	Real-Time Data Exchange
SDK	Software Development Kit
SMD	Surface-Mount Device
SPI	Serial Peripheral Interface
STL	Stereolithography
TCP	Tool Center Point
TPU	Thermoplastic Polyurethane
UART	Universal Asynchronous Receiver-Transmitter
UDP	User Datagram Protocol
UR	Universal Robots
USB	Universal Serial Bus
UTM	Universal Testing Machine
XML	Extensible Markup Language
YCB	Yale-CMU-Berkeley (object dataset)

Chapter 1

Introduction

1.1 Thesis Overview

Industrial automation and robotics have witnessed unprecedented growth in recent decades, driven by the demands of Industry 4.0 and the need for flexible, adaptive manufacturing systems and now leading to Industry 5.0 with the adoption of human–robot collaborative manufacturing Guerra-Zubiaga *et al.* (2022); Park *et al.* (2020). The increase in the complexity of industrial automation and the evolution toward Industry 5.0 have created a demand for robotic grippers that combine mechanical precision, adaptive compliance, and intelligent sensing. In modern production environments, manipulators must have the capability to handle objects that vary in size, shape, and material properties while maintaining the precision, robustness, and reliability required for high-throughput manufacturing Demirsoy *et al.* (2024a); Hernandez *et al.* (2023).

This doctoral thesis addresses these challenges through comprehensive research on advanced robotic manipulation systems for industrial applications. The work encompasses two parallel research streams: the study of a novel universal industrial gripper with multi-modal sensorization and the implementation of advanced robot control strategies for human-robot collaborative manufacturing within the EU Horizon SESTOSENSO project European Commission (2022). The gripper research spans actuation mechanism, kinematic analysis, mechanical enhancement, and multi-modal sensorization, resulting in a versatile grasping solution that combines the precision of rigid mechanics with the adaptability of soft robotics for handling diverse industrial components ranging from delicate items to complex rigid objects. Separately, the thesis contributes to the SESTOSENSO project through the design and implementation of advanced control strategies for the real-time coordination of an industrial robot and a collaborative robot in an industrial assembly application.

The core gripper design of the versatile three-finger universal gripper bridges

the gap between traditional rigid grippers and emerging soft robotic systems. The gripper architecture features modular fingers; each mechanical finger integrates a Chebyshev–parallelogram linkage with a thermoplastic polyurethane (TPU) self -adaptable finger as the contact interface. Thus, it benefits from the precision and ruggedness of its mechanical structure, along with the flexibility of the compliant material for safe object interface. The compound linkage mechanism transforms the linear input at the Chebyshev linkage into angular motion driving the parallelogram linkage interfaced with the self-adaptable finger, maintaining near-perfect linear motion ($\pm$0.033 mm deviation) with a 6.06:1 mechanical advantage. Independent finger actuation is provided by JVL stepper motors with embedded programmable logic controllers (ePLCs) communicating via modbus remote terminal unit (RTU). The developed control architecture supports both torque-based and velocity-based stall detection strategies, with configurable thresholds enabling grasping without dedicated sensors by leveraging internal motor feedback parameters.

Beyond the baseline gripper design, systematic enhancement methods were explored to achieve improved grasping. Improvements in the soft finger design and the linkage mechanism were investigated. The research uses a deflection coefficient quantification approach for the evaluation of soft finger wrapping behavior, and quasi-static force–displacement testing allowed validation through assessment of the finger deflection under load. To address the rigid link interference issue in the original four-bar parallelogram linkage and improve adaptive grasping, the four-bar mechanism was transformed into a six-bar linkage with an integrated compliant pad. These enhancements significantly improved the gripper's ability to conform to complex geometries and enabled higher grasping forces.

The fundamental innovation of this work lies in the development of three distinct multi-modal sensorization approaches for the self-adaptable fingers. The primary sensing approach introduces a novel vision-based sensing solution for self-adaptable soft fingers using an embedded camera, demonstrating proprioceptive and exteroceptive capabilities. The system achieves normal force estimation, Z-displacement estimation, slip detection, and force application position classification by integrating the Pi Camera V3 sensor into the structure of an additively manufactured self-adaptable soft finger. The sensing solution combines two algorithms: an optical flow-based method for reliably detecting slip events and a convolutional neural network (CNN) for force, displacement estimation, and position classification. The slip detection algorithm uses a dual-threshold approach that combines the median absolute deviation (MAD) and standard deviation. The complete processing pipeline was implemented on an embedded platform and achieved reliable performance with slip detection operating at 28.5 Hz and a CNN update rate of 8 Hz.

The second sensing modality utilizes resistive sensors fabricated using multi-

material fused deposition modeling (FDM) 3D printing techniques. This approach enables the creation of fully integrated, 3D printable sensorized fingers with embedded conductive TPU tracks, providing reliable contact detection and bending feedback. The third sensorization approach presents a novel light-angle sensor array architecture implemented through a custom four-layer rigid-flex PCB design developed in collaboration with the University College Dublin (UCD). This approach measures the deformation of the silicone pillars using optical principles. Individual sensor prototypes successfully demonstrate distributed tactile sensing capabilities, thus offering comprehensive tactile feedback across the finger contact surface, which is essential for precision grasping and handling tasks.

Customized rigs integrated with a universal testing machine (UTM) were used for self-adaptable finger design evaluation, sensor characterization, dataset acquisition, slip detection parameter optimization and pull-out testing with the universal gripper. Moreover, validation experiments were performed with a six-degrees-of-freedom (6-DOF) COMAU robot to mimic industrial manipulation scenarios.

In parallel, this thesis contributes to advanced robot control strategies through the SESTOSENSO project addressing the challenge of coordinating multiple robots in collaborative assembly operations, specifically focusing on automotive roof installation. The demonstration was initially planned for the Stellantis manufacturing facility, but later the University of Genoa (UniGe) was declared as the demonstration leader; thus, the complete robot cell for the demonstration was set up at UniGe. Major contributions to the SESTOSENSO project involved developing real-time control strategies for coordinating a KUKA KR150 R3100-2 industrial robot with a Universal Robot-UR10e collaborative robot via the robot operating system (ROS) and KUKA robot real-time trajectory correction using the robot sensor interface (RSI), enabling safe human–robot interaction in confined spaces and visually occluded assembly environments.

Collectively, this research advances the state of the art in hybrid gripper systems, soft adaptability, intelligent sensing, and validated industrial control strategies. The outcomes directly support the Industry 4.0/5.0 objectives, enabling flexible automation systems that can adapt to diverse manufacturing requirements without extensive reconfiguration.

1.2 Background and Motivation

Robotic grippers are considered the fundamental component in various industrial setups when grasping and object manipulation have to be performed Birglen & Schlicht (2018); Li *et al.* (2021). Manufacturing industries such as the automotive sector, packaging, pharmaceutical, electronics, agriculture, food, warehousing,

and logistics industries use grippers for manipulation and handling of strong parts and delicate items, pick and place of products, loading and unloading trucks, inventory management and to perform various precision gripping tasks Hernandez *et al.* (2023); Wang *et al.* (2021). The advancement in industrial gripping solutions is directly influenced by the progress in the field of robotics, as the kinematic design and actuation mechanism of the gripper determine its ability w.r.t a certain application Wolf & Schunk (2018).

1.2.1 Evolution of Industrial Manipulation Systems

The field of industrial automation has evolved significantly since the beginning of mechanized production, as depicted in Figure 1.1. The third industrial revolution brought about the development of programmable logic controllers (PLCs) and industrial robots, which facilitated adaptable automation and set the groundwork for modern manufacturing processes. Currently, we are at the cusp of the transition from the fourth to the fifth industrial revolution. Industry 4.0, shaped by cyber-physical systems, the Internet of Things (IoT), and data-centric production, requires automation systems to exhibit exceptional flexibility and adaptability. The shift from Industry 4.0 to Industry 5.0 has altered manufacturing prerequisites, highlighting the importance of human-robot collaboration and flexible automation Guerra-Zubiaga *et al.* (2022); Pasupuleti (2025); Xu *et al.* (2021).

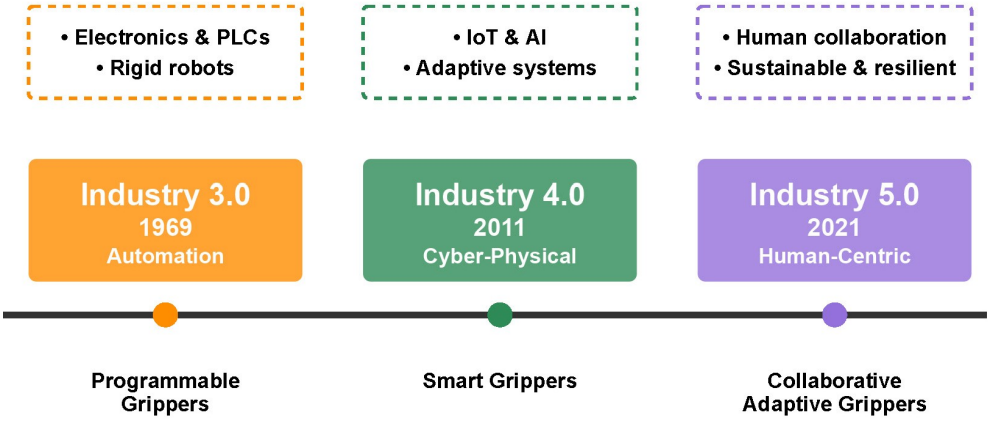


Figure 1.1: Evolution of industrial paradigms from Industry 3.0 to Industry 5.0, showing the progression of robotic gripper technologies and their key characteristics

With the move from mass production to mass customization, production batch sizes have decreased, and numerous product variations may be manufactured each day in some facilities. This high degree of variability makes traditional customized

tooling impractical and necessitates the need for universal manipulation solutions. The automotive industry highlights these challenges, where a single assembly line might accommodate multiple car models, each comprising thousands of parts that differ in size, material, and fragility.

Progressing from Industry 4.0 to 5.0 imposes distinct demands on robotic grasping systems, which include:

- **Adaptability:** Systems need to manage the growing diversity of products without major reconfiguration
- **Safety:** Built-in safety protocols for seamless human-robot interaction while maintaining efficiency
- **Intelligence:** Advanced sensory input for independent decision-making and quality oversight
- **Sustainability:** Optimized resource use and adherence to circular economy principles
- **Resilience:** Steady performance despite changes in supply chain and shifting requirements

1.2.2 Emergence of Soft Robotics in Industrial Applications

Soft robotics has emerged as a research area that is concerned with making robots compliant, thus opening up new possibilities and applications with safe interactions with humans and general fragile objects and tools Abeywickrama *et al.* (2023); Lee *et al.* (2017); Schmitt *et al.* (2018). In industrial environments, there is a growing interest in soft robots for jobs that require gentle handling, the ability to adapt to irregularly shaped objects, and secure human-robot collaboration. As soft grippers are typically fabricated from silicone elastomers or thermoplastic materials, they can conform to object geometries, providing secure grasps without precise positioning. Soft grippers have found use in sensitive operations such as picking fruits, handling food, and assembling electronics, where conventional grippers could cause damage AboZaid *et al.* (2024); El-Atab *et al.* (2020); Wang *et al.* (2024). As a result, an increasing number of commercial solutions have been introduced. Figure 1.3 presents some commercially available grippers: piSOFT-GRIP, SG grippers, and Gentle-Duo, which reflect the latest advancements in commercial soft grasping technology.

Figure 1.2: Examples of commercial soft robotic grippers and application of soft grippers in handling delicate objects

1.2.3 Limitations of Current Gripper Technologies

Rigid Systems:

- Application-specific designs requiring multiple end-effectors
- Inability to handle fragile objects without extensive force control
- Safety concerns in human-robot collaboration scenarios
- Limited adaptation to object geometry variations

Soft Systems:

- Payload limitations typically below 1 kg
- Positioning accuracy degradation under load

- Absence of integrated sensing for closed-loop control
- Material durability concerns in industrial environments
- Slower actuation speeds compared to rigid systems

1.2.4 The Hybrid Gripper Paradigm

The recognition of the complementary advantages and drawbacks of rigid and soft grippers has motivated the emergence of hybrid architectures designed to combine the strengths of both paradigms. In such systems, rigid mechanical structures deliver precise motion control and high force output, while soft contact interfaces provide adaptability to varying object geometries and ensure safe, compliant interactions Liu *et al.* (2023a).

State-of-the-art hybrid implementations include designs where rigid actuators drive soft fingertips, mechanisms with tunable stiffness for variable compliance, and modular configurations capable of switching between rigid and soft operating modes D'Avella *et al.* (2023); Govindan & Krishna (2023); Jamil *et al.* (2022). While these developments mark important progress toward versatile robotic manipulation, many existing solutions still suffer from only partial integration of rigid and soft elements, leading to trade-offs that limit the full exploitation of their combined benefits.

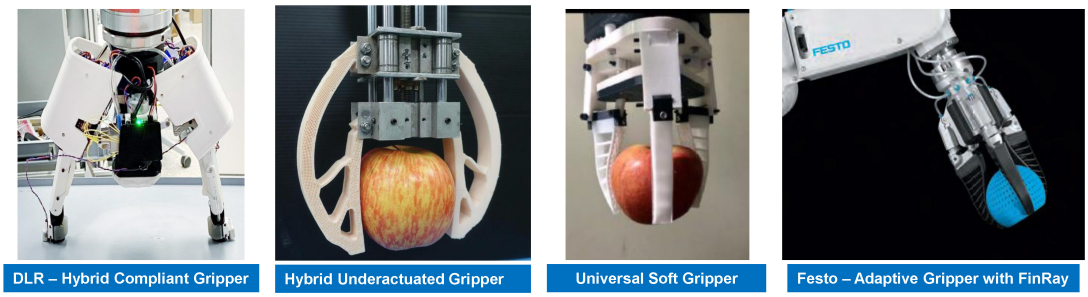


Figure 1.3: Examples of state-of-the-art hybrid grippers in research and industry: DLR Hybrid Compliant Gripper, Hybrid Underactuated Gripper, Universal Soft Gripper with Optimized FinRay, and Festo Adaptive Gripper with FinRay.

1.2.5 Sensing Challenges in Compliant Systems

A key obstacle to the broader adoption of soft and hybrid robotic systems lies in the absence of robust, comprehensive sensory feedback. Traditional rigid robots rely extensively on position encoders, force sensors, and vision systems to achieve precise manipulation. By contrast, embedding equivalent sensing capabilities

within soft robotic structures is particularly challenging because of their large deformations, nonlinear material properties, and complex geometries Hegde *et al.* (2023).

Traditional sensing methods are often unsuitable in this context. For example, strain gauges and standard force sensors can compromise the intrinsic compliance of soft materials, while external cameras are prone to occlusion and struggle to capture the distributed, continuous contact dynamics of soft robotic interactions. The central challenge, therefore, is to design sensing approaches that maintain material compliance while still delivering accurate, real-time feedback for intelligent control Zhou *et al.* (2024).

Existing approaches to soft robot sensorization include embedded conductive materials, optical sensing, magnetic field sensing, and pneumatic pressure monitoring. However, these methods face tightly coupled trade-offs between sensing fidelity and mechanical softness. Difficulties include avoiding structural stiffening, mitigating nonlinear and noisy signals, ensuring long-term durability, achieving calibration, and enabling energy-efficient systems across varied environments Ambaye *et al.* (2024); Collins *et al.* (2022); Low *et al.* (2021); Luo *et al.* (2024). Until these challenges are addressed, the lack of practical, reliable sensorization solutions continues to limit soft and hybrid grippers from attaining the level of intelligence required for industrial deployment.

1.2.6 Motivation and Approach

This doctoral research is motivated by the need for robotic manipulation systems that can combine precision, adaptability, and intelligent feedback in industrial environments increasingly shaped by Industry 4.0 and 5.0 paradigms. While the development of a universal gripper concept had already begun within our laboratory, the original prototype lacked comprehensive kinematic analysis, advanced actuation control, and embedded sensing capabilities. Addressing these gaps provided both a scientific challenge and a practical opportunity to extend the gripper toward industrially viable performance.

The starting point was a detailed study of the gripper's mechanical performance, undertaken through systematic kinematic analysis and the development of actuation and motor control strategies to enable independent finger operation. From there, the research progressed to finger redesign, focusing on enhancing adaptability and mechanical behavior moving from a four-bar to a six-bar linkage mechanism.

Recognizing that adaptability without sensing remains insufficient for intelligent manipulation, the thesis introduces multiple sensorization strategies, most notably a vision-based sensing solution that simultaneously achieved proprioceptive (Z-displacement and position estimation) and exteroceptive (force estimation,

slip detection) sensing using a single embedded camera without compromising the self-adaptable finger compliance. The systems additional sensing concepts, including light-angle arrays and 3D-printed resistive sensors, were also explored to extend feedback capabilities.

The final dimension of the research was integration into industrial contexts. Through collaboration with partners in the EU Horizon SESTOSENSO project, control architectures were developed and validated for coordinating robots (KUKA KR-150 and UR-10) in real-time, human–robot collaborative assembly tasks. This ensured that the work was continuously aligned with genuine industrial requirements.

1.3 Research Objectives and Contributions

This research systematically advances industrial gripper technology through kinematic analysis, actuation and control, intelligent sensorization, and industrial validation. The objectives and resulting contributions are organized across three interconnected themes.

1.3.1 Kinematic Analysis, Actuation, and Control Systems

Objective: Establish a comprehensive understanding of gripper mechanics and develop control strategies for independent finger operation, validated through integration with a COMAU 6DOF robot.

Contributions:

- Complete kinematic characterization of the Chebyshev–parallelogram linkage, quantifying a 6.06:1 mechanical advantage and ± 0.033 mm motion linearity.
- Development of modular actuation control using JVL stepper motors with embedded PLCs and Modbus RTU communication.
- Implementation of sensorless grasping via torque-based and velocity-based stall detection, enabling real-time grasp confirmation without explicit force sensors.
- Kinematic transformation analysis from four-bar to six-bar linkage, demonstrating 95.3% force transmission efficiency and improved adaptability through compliant pads.

- Validated performance enhancement, including a 273% increase in retention force (287 N), 150% payload increase (up to 7.5 kg), and 23% expansion in graspable diameter (270 mm).
- Experimental validation of soft finger and gripper through compression and pull-out tests using a UTM.

1.3.2 Multi-Modal Sensorization

Objective: Enable intelligent perception in compliant fingers without compromising their mechanical compliance—the critical innovation required for industrial deployment.

Contributions:

- **Camera-embedded sensing architecture (Primary contribution):** A novel approach to embedding a miniature camera within a compliant finger, achieving multi-modal sensing while preserving mechanical compliance, was developed and demonstrated. This system enables the following:
  - Normal force estimation with RMSE <0.54 N.
  - Z-displacement tracking with RMSE <0.46 mm.
  - Real-time slip detection at 28.5 Hz.
  - Position classification of applied forces with 95% accuracy.

  The complete sensing pipeline was implemented on an embedded Raspberry Pi platform for real-time inference using TensorFlow Lite (TFLite).
- **Slip detection innovation:** A dual-threshold slip algorithm that combines median absolute deviation (MAD) and standard deviation metrics with temporal filtering was designed and validated to achieve robust performance in real-time operation.
- **Complementary sensing modalities:**
  - 3D-printable resistive sensors for contact and bending detection embedded within compliant fingers.
  - Light-angle sensor arrays, extending a single prototype sensor into a distributed rigid-flex PCB implementation, enabling tactile feedback across the finger surface.

1.3.3 Industrial Integration and Control Development

Objective: (i) Integration of the universal gripper on a COMAU robot for manipulation tasks and (ii) development of real-time robot control architectures within the EU Horizon SESTOSENSO project.
Contributions:

- **Gripper validation:** Successful integration and testing of the universal gripper with a COMAU 6DOF robot, demonstrating the manipulation of industrially relevant objects including automotive components.
- **Industrial robot control (SESTOSENSO - HORIZON EUROPE):**
  - Development of real-time control strategies for a KUKA KR150 robot using the RSI.
  - Coordination of KUKA and UR10 robots for cooperative automotive roof assembly.
  - Implementation of safety protocols for human–robot collaboration.

1.4 Thesis Organization

The thesis is structured as follows:

- **Chapter 1** introduces the motivation, problem statement, research objectives and contributions, and the overall structure of the thesis.
- **Chapter 2** provides a literature review of rigid, soft, and hybrid grippers, sensorization strategies, and industrial robot control frameworks.
- **Chapter 3** presents the kinematic analysis and actuation control strategies for the universal gripper.
- **Chapter 4** details the enhancement of self-adaptive fingers, including the transition from four-bar to six-bar linkages with compliant elements.
- **Chapter 5** introduces the camera-embedded sensing approach for multimodal feedback in compliant fingers.
- **Chapter 6** explores additional sensorization techniques, including resistive 3D-printed sensors and light-angle tactile arrays.
- **Chapter 7** documents the industrial implementation of real-time robot control strategies within the EU Horizon SESTOSENSO project.
- **Chapter 8** concludes the thesis by summarizing the contributions, discussing limitations, and outlining directions for future research.

1.5 Research Publications

The research conducted during this doctoral study has been disseminated through the following peer-reviewed publications:

Primary Contributions

1. *Design and Development of a Novel Universal Gripper Having Rigid Mechanics With Self-Adaptable Fingers for Industrial Applications*, IEEE Access, 2025. DOI: 10.1109/access.2025.3587328

2. *A Camera-Embedded Self-Adaptable Finger With Multi-Modal Sensing Capabilities for Robotic Manipulation*, IEEE Access, 2025. DOI: 10.1109/access.2025.3561582

3. *Universal Gripper for Industrial Manipulation with Enhanced Rigid Mechanics and Self-Adaptable Fingers*, Advanced Robotics Research (WILEY), 2025. Accepted.

4. *Design and Fabrication of Resistive Sensor Using Additive Manufacturing Techniques for Robotic Application*, International Journal of Advanced Manufacturing Technology, 2025. DOI: 10.1007/s00170-025-15977-y

Supporting Contributions

5. *Investigating Multi-Material Additive Manufacturing for Disassembly and Reparability of Adhesive Joints by Precision Heating*, Adhesives, 2025. DOI: 10.3390/adhesives1010004

6. *Thermal Curing of Adhesive Joints Enabled by Precision Heating Multi-Material Additive Manufacturing*, Journal of Manufacturing and Materials Processing, 2025. DOI: 10.3390/jmmp9050151

Submitted Work

7. *HRI Experiments with Sensorized High Payload Robots*, submitted to IFAC 2026.

Beyond publications, elements of this research were presented in project meetings, and demonstrations within the EU Horizon SESTOSENSO consortium.

Chapter 2

Literature Review

2.1 Introduction

The design and deployment of robotic grippers for industrial manipulation has evolved through multiple technological paradigms, from rigid, task-specific mechanisms to compliant and hybrid architectures with intelligent sensing. This chapter reviews the key developments that inform the contributions of this thesis and establishes the theoretical foundation for the contributions presented in subsequent chapters.

2.2 Robotic Grippers

Robotic grippers serve as the main interface between robots and their surroundings. Their evolution reflects broader industrial demand shaped over years of automation. Rigid grippers, generally powered by electric motors, pneumatic cylinders, or hydraulic systems, prevail in industrial tasks because of their substantial load capacities, precise positioning (accuracy < 0.1 mm), and robustness Wang *et al.* (2021).

Grippers can be categorized by the number of fingers, the type of actuation, the operating mechanism, and the prehension approach. Grippers may feature 2, 3, or 4 fingers, or they may be anthropomorphic with more than four fingers. Robotic grippers are actuated through hydraulic, pneumatic, magnetic, or electric means, using mechanisms like screws, worm gears, rack and pinion, cams and followers, or pulleys. The grasping method includes impactive, astrictive, or ingressive techniques Monkman *et al.* (2007); Raval & Patel (2016). As depicted in Figure 2.1, commercial solutions from companies such as Festo, Schunk, and SMC primarily employ pneumatic or electrical actuation, with mechanical designs tailored to specific part shapes Kumar *et al.* (2025). This focus on special-

ization makes rigid grippers generally unsuitable for handling parts with diverse sizes, shapes, or surface complexities without undergoing considerable redesign or adjustment.

Grippers are classified based on the number of fingers, type of actuation, mechanism, and method of prehension. The gripper can be 2-finger, 3-finger, 4-finger, or anthropomorphic i.e., having more than four fingers. Robotic grippers can be actuated hydraulically, pneumatically, magnetically, or electrically and have a screw, worm gear, rack and pinion, cam and follower, or pulley-based mechanism. The grasping method can be impactive, astrictive, or ingressive Monkman *et al.* (2007); Raval & Patel (2016). Commercial grasping solutions from manufacturers like Festo, Schunk, and SMC shown in Figure 2.1 utilize pneumatic or electric actuation, with mechanical designs optimized for specific part geometries Kumar *et al.* (2025). This specialization means rigid grippers often lack the flexibility to handle parts with varying shapes, sizes, or surface complexities without significant redesign or changeover which may consume hours or even days, significantly reducing production flexibility. Moreover, rigid grippers are unsuitable for grasping fragile objects due to limited adaptability and flexibility and may cause damage or deformation Al Abeach *et al.* (2017); Hernandez *et al.* (2023); Kattan Urrutia & Carrasco Bardales (2024).

Figure 2.1: Examples of commercial rigid-grippers by Festo and Schunk

2.2.1 Soft Robotic Grippers

Soft grippers have developed from advancements in materials science and bio-inspired design. By exploiting compliance in elastomers and smart polymers, these systems adapt passively to object geometry and are lighter and easier to manufacture Shintake *et al.* (2018b); Wang & Chortos (2022). The field of soft robotics focuses on creating compliant robots, facilitating new opportunities and

applications for safe interactions with humans and fragile items or instruments. Unlike traditional rigid robots with defined degrees of freedom, soft robots possess a continuous, flexible structure, resulting in high DOFs. Therefore, conventional actuation methods are unsuitable for soft robots. The predominant actuation techniques in soft robotics include fluidic systems, electro-active polymers (EAP), and variable tendons Jumet *et al.* (2022); Lee *et al.* (2017). As a result, numerous industries are interested in compliant grippers for handling delicate and easily damaged parts or products Otti *et al.* (2025). Currently, industrial soft grippers integrating under-actuation and compliance operate without any control strategies; however, their pose and state can change passively due to unpredictable external forces, complicating their position control Shintake *et al.* (2018b); Wang & Chortos (2022). Soft grippers typically support payloads of less than 1 kg, have reduced positional precision, slower actuation, and durability issues in demanding industrial environments Hernandez *et al.* (2023); Muller *et al.* (2020); Yap *et al.* (2016).

2.2.2 Hybrid Grippers

Hybrid architectures aim to merge the strength and accuracy of rigid mechanics with the flexibility of soft interfaces. Instances of such designs involve rigid linkages with compliant fingertips, mechanisms with adjustable stiffness, and modular end-effectors capable of switching between different modes. These systems have been effective in managing objects of various sizes and enhancing safety in collaborative environments. Nonetheless, numerous designs depend on pneumatic systems or add complexity to control, which hampers widespread industrial use Li *et al.* (2024); Wang *et al.* (2023); Yoon *et al.* (2025).

One significant advantage of hybrid grippers is their enhanced versatility, which enables them to grasp and manipulate a wider range of objects with varying shapes, sizes, weights, and degrees of fragility than grippers that are entirely rigid or soft Khin *et al.* (2024); Park *et al.* (2019); Trinh *et al.* (2024). For example, a hybrid gripper that employs rigid fingers to execute traditional gripping tasks while using inflatable soft telescopic fingers to achieve gentle interaction with irregularly shaped objects that can stably grasp objects up to approximately 450 g Gerez *et al.* (2020). Another example shows how a hybrid gripper equipped with passive pneumatic soft joints around rigid links can improve handling efficiency and reduce the gripping distance when grasping deformable and thin items like paper, outperforming rigid grippers in similar scenarios Tran *et al.* (2024). A key advantage of hybrid grippers is their higher payload capacity. The incorporation of rigid supports into a soft pneumatic network (PneuNet) can significantly enhance the fingertip force and the actuation speed, enabling it to handle heavier objects up to 1.3 kg while retaining the compliance required to

grasp irregular shapes Park *et al.* (2019). A novel gripper using hybrid actuation for improved grasping of multi-shape objects using combined PneuNet and layer jamming actuation is presented in Tennakoon *et al.* (2024). Moreover, Sun *et al.* (2023) attains higher load capacity over 2 kg and versatile grasping by integrating a vacuum-assisted bio-inspired adhesive surface with a bidirectionally bendable pneumatic actuator. The controlled motion of rigid links and the flexible properties of soft fingers in hybrid grippers provide potential for dexterous grasping tasks. Hybrid grippers can perform complex in-hand manipulation tasks that are difficult for solely rigid or soft grippers. For example, a hybrid gripper inspired by the human hand that incorporates pneumatic bellows as soft actuators within a rigid structure has demonstrated lateral compliance and the ability to perform anthropomorphic dexterous manipulations Zhu *et al.* (2023).

Zhang et al. Zhang *et al.* (2020) provided a comprehensive review of hybrid grippers in the field of agricultural robotics. These grippers combine various actuation techniques and materials—such as the integration of pneumatic and suction systems or the use of both rigid and flexible components—to enhance the handling of delicate objects. In their RG series, OnRobot incorporated compliant pads on the grippers' fingertips, enabling them to handle objects with diverse shapes OnRobot A/S (2025). Designs featuring a mix of rigid and soft materials have emerged to offer both compliance and skillful manipulation Chang *et al.* (2019); Romanov *et al.* (2022); Zhu *et al.* (2023). To replicate the dexterity of human fingers, hybrid grippers equipped with optimized variable stiffness mechanisms have been developed Dinakaran *et al.* (2023); Li *et al.* (2018). Jamming-based robotic grippers, designed to manipulate various small items like bottle caps, screwdrivers, and coils, have also been introduced Wang *et al.* (2021). These grippers are capable of adapting to unfamiliar objects of different orientations, surfaces, and shapes by conforming to the object and employing suction for gripping Brown *et al.* (2010). Nevertheless, most adaptive gripping research and development has primarily targeted soft objects, focusing on solutions such as pick-and-place for the food and agriculture sectors or managing small logistics and industrial tools Demirsoy *et al.* (2024b).

2.3 Fin Ray Principle-Inspired Grippers

In the realm of bioinspired soft grasping techniques, the Fin Ray principle stands out for its aptitude in facilitating adaptive hybrid grasping in industrial settings. This is primarily due to its intrinsic ability to adapt passively, having compliance that varies with load, and ease of fabrication Crooks *et al.* (2016). The principle draws from the biomechanical function of a fish's tailfin; under pressure, the fin curves inward and its tip moves in opposition to the load Shan & Birglen (2020)

as illustrated in Figure 2.2.

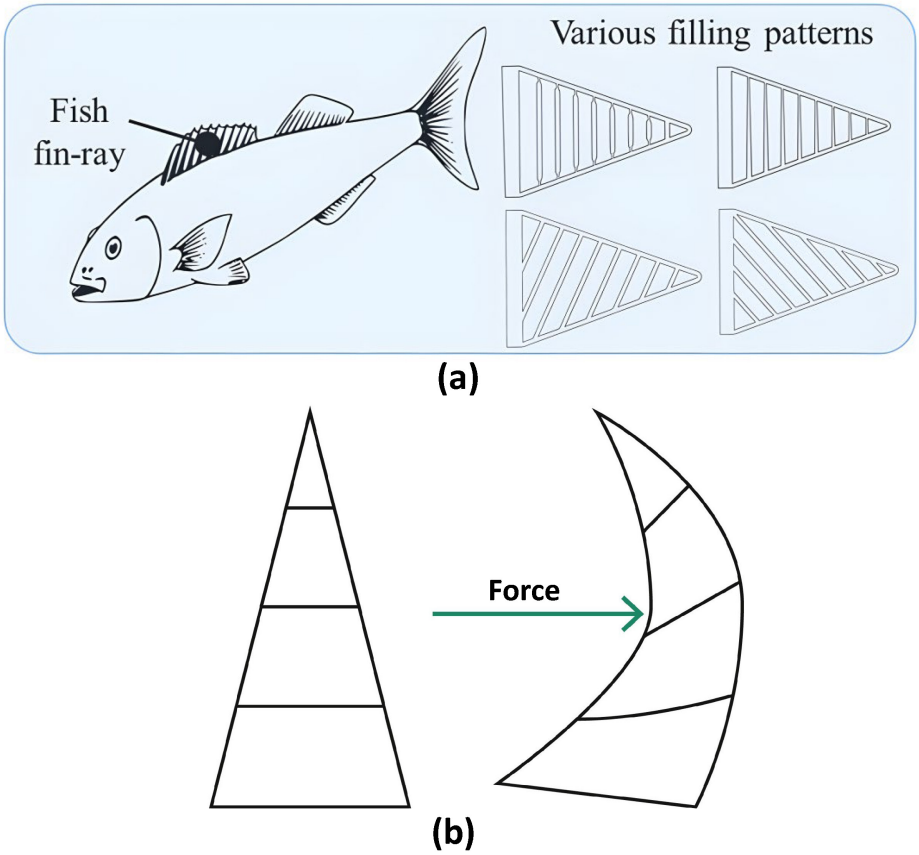


Figure 2.2: (a) Biological inspiration: the fin-ray structure of a fish, illustrating various internal filling patterns and (b) The Fin Ray® effect: when a force is applied to the structure, both the base and tip bend toward the load, demonstrating adaptive deformation similar to a fish fin.

Researchers have explored various ways to enhance Fin Ray-inspired fingers by adjusting the quantity, orientation, thickness, and shape of the internal cross-beams to meet particular grasping objectives Yao *et al.* (2023a, 2024a). FESTO Group pioneered the creation of adaptive fingers modeled after the Fin Ray design, featuring a triangular construction with crossing beams inside to redistribute forces and produce deformation Bannasch & LK (2012); Festo (2025). This efficient and modifiable system has spurred further research into hybrid grippers Muller *et al.* (2020). Hyeon Shin et al. Shin *et al.* (2021) devised a gripper with refined finger architecture based on the Fin Ray concept, significantly enhancing the gripper's weight capacity (up to 1.2 kg) and allowing it to switch between parallel and centric grasping styles. Additionally, the gripper is electric-powered

and can manipulate diverse objects, showcasing its flexibility in manufacturing contexts. Crooks et al. Crooks *et al.* (2017) introduced a bio-inspired gripper that employs the gripping mechanism observed in the Manduca sexta and integrates the adaptable form of Fin Ray fingers. Utilizing 3D printing technology, they created a single-piece design comprising both soft and rigid elements, able to handle various everyday items up to 530 g. Birglen Birglen (2015) illustrated enhanced versatility and safety with adaptive robotic fingers that can be paired with conventional industrial grippers. By conducting kinetostatic analysis and experimental validation, this work emphasized how integrating such adaptive solutions into existing industrial systems could mitigate hesitancy in their adoption. Ali et al. Ali *et al.* (2019) developed a three-fingered hybrid Fin Ray gripper actuated by a single stepper motor for efficient pick-and-place tasks, specifically evaluating the effectiveness of NinjaFlex 3D printing filaments for fabricating compliant fingers.

Current studies on Fin Ray structures aim to enhance their mechanical efficiency, incorporate sophisticated sensing technologies, and broaden their use across industries such as manufacturing and agriculture. Nonetheless, there are still there are several challenges to overcome: increasing grip strength, avoiding out-of-plane deformations, implementing effective sensor technologies that maintain finger flexibility, and refining control precision for complex object shapes. The passive adaptability, structural simplicity, and scalability of the Fin Ray principle make it an excellent choice for industrial hybrid grippers, thus motivating its selection for this study.

2.4 Sensorization in Soft Robotic Grippers

The transition from rigid to soft robotics marks a notable advancement in robotic grasping and manipulation, where precision and adaptability are essential Shintake *et al.* (2018a). Soft grippers, which harness the advantages of flexibility, have garnered significant interest Yao *et al.* (2023b). These devices automatically conform to the shape of the objects they handle, offering an effective mix of gentle yet secure grasping. The inherent adaptability of soft grippers facilitates the manipulation of objects with varied shapes, sizes, and material properties, minimizing harm to delicate items while ensuring firm control over heavier and more solid objects.

The capability to detect interaction forces and slip, similar to human manipulation, is critical Welle *et al.* (2023). Sensory integration in these compliant systems poses unique challenges due to the substantial deformations, nonlinear material behavior, and necessity of preserving flexibility. Hence, soft robots leverage their flexible materials to interact with objects effectively in an open loop

system absent of sensory feedback Truby *et al.* (2019).

A key obstacle lies in developing integrated proprioceptive and exteroceptive sensors for soft grippers, essential for recognizing the gripper's configuration and its interactions with objects. Realizing such sensory integration in flexible robotic systems demands creative solutions that preserve the gripper's inherent compliance Truby *et al.* (2018); Yang *et al.* (2024).

2.4.1 Conventional Sensing Methods

Rigid robots typically employ force–torque sensors, encoders, and external cameras. However, traditional rigid sensors cannot be used for soft robots due to the flexible nature of the materials, which can affect the robot's compliance, thereby limiting their performance Donato *et al.* (2024); Yang *et al.* (2024, 2022). In addition, soft robots exhibit highly nonlinear properties with infinite degrees of freedom and complex deformations that are imperceptible with traditional sensor placement approaches Kim *et al.* (2021).

2.4.2 Embedded Soft Sensors

- **Resistive and capacitive sensors:** In soft robotics sensing technologies, advancements have been made to develop stretchable sensors capable of sensing the complex deformations and interactions of flexible robotic structures Pagoli *et al.* (2022). These sensing modalities are important for soft robots to interact with their environment and perform complex operations intelligently. Resistive sensors measure changes in electrical resistance upon deformation of the conductive material, allowing the sensing of various parameters, such as strain, force, and curvature Li *et al.* (2017). Resistive sensors can be integrated into the soft robot body or the actuation mechanism to detect bending or pressure conditions. In Zhao *et al.* (2023) a flexible resistive sensor network is developed with multiple independent strain sensors whose correlated values are used to capture complex deformation patterns in soft structures. Capacitive sensors provide tactile feedback by measuring the capacitive changes that occur as soft material deforms. Capacitive sensors have high sensitivity and repeatability and enable force, strain, shear, and bending measurements Li *et al.* (2017). They are also more accurate and durable compared to resistive sensorsPagoli *et al.* (2022), The development of a highly stretchable capacitive strain sensor for soft systems is given in White *et al.* (2017).

- **Magnetic sensors:** Magnetic sensors with magnetic components can measure changes in magnetic fields due to the orientation and position of mag-

nets embedded in soft robots and provide information about robot movement and deformation. However, the use of these sensors in soft robots is limited due to the rigid magnetic and sensing elements Hao *et al.* (2022).

- **Optical methods:** Another interesting class of sensors that has seen increasing usage in soft robotics is optoelectronic-based sensors. Optoelectronic sensors show high sensitivity and fast response rate, accommodate noncontact sensing, have low power consumption, exhibit lower hysteresis, are immune to electromagnetic interference, and are resistant to chemical corrosion. In addition, by utilizing flexible and stretchable optical fibers inside soft robots, sensors can be easily integrated within soft robots. These sensors typically consist of an emitter, receiver, and waveguide. These sensors can measure displacement fields and strain within a robot's soft structure by detecting changes in the optical signal Hao *et al.* (2022); Pagoli *et al.* (2022).

  Light-angle optical sensors infer 3D force and displacement by analyzing the direction and intensity of light reflected within a deformable structure, such as a silicone pillar. These systems use embedded reflectors and quadrant photodiodes to detect subtle deformations, enabling compact, high-resolution tactile sensing without cameras. These sensors offer robust performance for soft robotic manipulation, including slip detection and vibration sensing Leslie *et al.* (2023b).

Integrating these compliant sensors into soft grippers, manipulators, actuators, and other robotic technologies such as wearable devices and medical robots has been a key area of research. By integrating soft sensors, these technologies enable better environmental interaction, understanding of their state, and improved performance Kim *et al.* (2021); Sirithunge *et al.* (2024). However, significant limitations and challenges must be overcome, such as their performance in different environmental conditions, hysteresis, limited spatial resolution, complex calibration, inability to differentiate between different types of forces, and reduced accuracy in delivering real-time feedback, especially during significant deformations Bhirangi *et al.* (2021); Loo *et al.* (2022); Yoshimoto *et al.* (2024). In addition, the mechanical properties of the soft robot can be influenced by embedding soft sensors in its flexible structure Lalegani Dezaki *et al.* (2023); Qu *et al.* (2024).

2.4.3 Vision-Based Sensorization

Embedding cameras inside compliant fingers represents an emerging approach, these sensors use a camera to provide high-resolution image data by capturing

visual information about the contact surface. Unlike traditional soft sensors, vision-based systems use image processing and machine learning techniques to detect detailed geometric changes, object slippage, and forces without mechanical contact Pang *et al.* (2021); Welle *et al.* (2023); Zhang *et al.* (2023). Various vision-based sensing systems have been developed for improved grasping by providing high-spatial-resolution measurements of geometry, force, and slip. These advanced sensors utilize a deformable contact medium, illumination unit, and integrated camera to detect the deformation of the contact medium Taylor *et al.* (2022); Zhang *et al.* (2023). The most prominent of these vision-based sensors are GelSight Dong *et al.* (2017); Yuan *et al.* (2017), TacTip Lepora (2021), FingerVision Yamaguchi & Atkeson (2016) and DIGIT Lambeta *et al.* (2020). However, most of these camera-based sensing solutions are typically bulkier than traditional sensors such as capacitive and resistive soft sensors, making their seamless integration into soft robot structures more difficult. Therefore, there are limited implementations of soft robot tactile sensing using vision sensors. She *et al.* (2020) presents an exoskeleton-covered soft finger with embedded cameras capable of tactile perception and proprioception by using GelSight technology. GelSight Fin Ray integrates the GelSight sensor into a customized Soft-inspired finger that passively adapts to any object being grasped, performs tactile reconstruction, estimates object orientation, and measures shear and torsional forces Liu *et al.* (2023b). However, GelSight adds a silicon pad over the already deformable finger; therefore, the sensing system must distinguish between the deformation of the sensing silicon pad and a change in the state of Soft's compliant material Liu & Adelson (2022). A neuromorphic event camera-based sensorization of a Soft-inspired finger has been presented in Faris *et al.* (2023). The event camera was placed outside the finger to observe the side profile of the finger. The motion data of internal markers is used with a CNN to predict the two-dimensional deformation of three markers placed on the finger and slippage events, providing a unified solution for proprioception and exteroception.

Table 2.1 presents the state-of-the-art in vision sensing technologies explored in research. It outlines the types of cameras employed and the associated methodologies, highlighting their primary applications and implementation approach in recent studies.

Table 2.1: Comparison of vision sensor technologies and their key characteristics

Sensor	Camera Model	Approach	Measurable Parameters	Sensing Methodology	Platform
GelSight	High-res camera	Deformable elastomer (gel-coated surface) as medium of contact	Surface texture, shape, material hardness, contact area, pressure distribution, force magnitude and direction	Open CV marker detection, Optical flow, Gel deformation analysis	PC
TacTip (Family)	ELP camera module	Biomimetic, flexible membrane with embedded markers	Contact location, force direction, torque, object shape	Open CV, Forces (contact, shear) inferred from the degree of pin displacement	PC/Raspberry Pi
DIGIT (built on GelSight technology)	Omnivision OVM7692 60fps	Soft Transparent elastomeric gel surface embedded with markers and array of LEDs for illumination	Force Magnitude and Direction (normal + shear), contact location, surface texture	Marker displacement using computer vision, and sensor calibration for mapping displacements to normal and shear force components	PC
DIGITAC (with tactip surface and DIGIT sensor base)	Similar to DIGIT	Enhanced DIGIT	Forces and displacements, tested using Edge following	Marker displacement detection, image processing	PC
Optical Tactile Sensor (by ETH Zurich)	ELP USBFHD06H	Deformation of a soft material with markers at different depths within the gel	Normal Force distribution, displacements	OpenCV Centroid detection, Sparse & Dense Optical Flow, Marker displacement & DNN for normal force.	PC
Tactile Reactive Grasper	Raspberry Pi Camera	Roller mechanism equipped with GelSight sensors	Force distribution, 3D Reconstruction, marker detection.	GelSight technology with roller sensing area.	Raspberry Pi as interface b/w PC & Viko
VIKO	Raspberry Pi Camera	Tracking of Dense random pixel markers	Contact area, Shear force and marker displacement	Dense inverse search (DIS) optical flow algorithm	Raspberry Pi as interface b/w PC & Viko

2.5 Human–Robot Collaboration and Control Architectures in Industrial Robotics

Over the last six decades, industrial robots have reshaped manufacturing by lowering expenses, boosting efficiency, and removing people from the most strenuous and dangerous jobs. The earliest robots, like *Unimate*; the inaugural industrial robot engineered by Unimation Inc. in the 1950s Nof (1999) were used primarily for heavy lifting and routine operations; they operated in isolated areas with minimal sensing capabilities and simple programming. This segregation shielded human workers from hazardous environments while allowing the robots to perform repetitive tasks swiftly. Moving into the 1980s, significant progress in electronics, mechanics, and computer sciences greatly enhanced robot functionalities: robots gained increased positional precision, sophisticated drive systems, and could undertake precision tasks. The incorporation of sensors, such as force/torque and vision systems, enabled robots to sense their environment, check process status, and execute more advanced, semi-autonomous operations Gasparetto & Scalera (2019). Despite these advancements, robots often remained confined within safety cages due to the excessive risk they posed in shared human environments. Since the 2000s, innovations like machine learning and advanced sensing have propelled a shift towards more adaptable manufacturing. Contemporary industrial robots gather extensive sensor data and are increasingly employing learning and adaptation to optimize task execution Lange *et al.* (2021).

This evolution has allowed robots to break free from isolated cells and engage in collaborative efforts with human operators: cobots and hybrid workspaces enhance ergonomics, increase versatility, and facilitate more intricate assembly tasks Matthias & Smarra (2022). Simultaneously, robotic applications are spreading beyond traditional factory floors: autonomous robots are now found in warehouses for logistics, in healthcare settings as surgical assistants, as well as social and service robots active in public venues, and intelligent domestic robots within private residences Siciliano & Khatib (2016).

Despite these advances, several challenges remain unresolved. Coordinating industrial robots and cobots in shared workcells requires deterministic communication, robust synchronization, and safe motion adaptation under uncertainty. These challenges form the motivation behind large-scale initiatives such as the Horizon Europe **SESTOSENSO** project, which aims to develop perception and control technologies that allow robots to operate collaboratively in realistic industrial scenarios with humans in the loop.

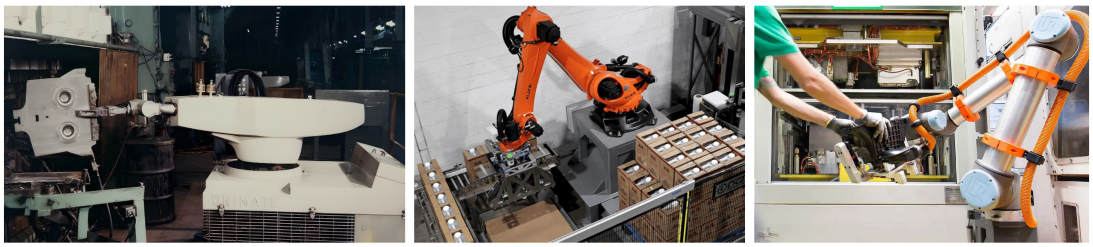


Figure 2.3: Evolution of industrial robots: from early heavy-duty manipulators confined within safety cells to modern collaborative robots. (a) Early industrial robot: Unimate; (b) caged production cell with KUKA industrial robot; (c) modern collaborative robot (cobot) working alongside a human operator.

2.5.1 Sensing and Control Trends for Collaborative Assembly

Operating safely and effectively in dynamic, partially sensed environments requires both advanced sensors and control strategies that exploit sensor feedback in real time. Two interrelated developments are particularly relevant:

- **Sensor-rich robotic platforms.** Modern systems integrate vision, force/-torque sensing, and embedded proprioceptive measurements to perceive both the workpiece and human activity. These sensors enable perception of uncertainties (part tolerances, human interventions, occlusions) that would otherwise break strictly position-based programs.
- **Real-time and adaptive control.** Controllers that can modify trajectories and actions online using tight feedback loops and deterministic communication links are essential for maintaining performance and safety when the environment or task changes. Real-time correction interfaces (e.g., vendor-specific APIs such as KUKA RSI KUKA GmbH (2025), ABB EGM ABB Robotics (2024)) and field-level communication protocols (EtherCAT, PROFINET) are part of this trend.

Together, sensing and real-time control make it possible to orchestrate complex collaborative assembly tasks in which industrial robots provide precision and payload capacity while cobots and humans handle flexible positioning and final assembly steps.

2.6 Research Gaps and Opportunities

Despite the rapid evolution of robotic grippers and collaborative robotic systems, several critical gaps remain between laboratory concepts and robust industrial deployment. These challenges are the focus of the investigations in this thesis.

Mechanical Design Gaps

Hybrid grippers that integrate rigid linkages with compliant finger structures have demonstrated potential, yet most existing designs are limited by ad hoc development. There is a lack of systematic kinematic analysis methods to evaluate parameters such as motion linearity, force transmission efficiency, and mechanical advantage across their entire operating range. Moreover, enhancement strategies are typically pursued in an empirical manner without analytical validation, pointing to a need for more robust design optimization.

Sensorization Challenges

Intelligent manipulation requires comprehensive feedback, but compliant finger structures complicate sensing integration. Traditional rigid sensors compromise compliance, while embedded technologies such as resistive, capacitive, and optical sensors suffer from hysteresis, drift, or mechanical interference. Vision-based sensing systems offer great promise, but many current solutions rely on external cameras prone to occlusion or require added elastomer layers that alter mechanical response. The opportunity exists to establish embedded, compact, and multimodal sensing systems that that retain the adaptability of compliant fingers while delivering accurate force, deformation, and slip data.

Control Architecture Limitations

Industrial adoption requires controllers that not only actuate grippers but also interface seamlessly with manufacturing infrastructure. Present research prototypes depend on lab-specific control hardware, lacking compatibility with standard industrial communication protocols such as Modbus RTU, EtherCAT, or PROFINET. In addition, robust sensorless control strategies utilizing motor-internal feedback for object detection as a sensorless control strategy remains largely unexplored and could serve as a lightweight alternative to dedicated force sensors.

System Integration and Validation Gaps

Many hybrid gripper prototypes remain remain untested under real-world industrial conditions. There is limited evidence of their ability to manipulate diverse objects from fragile items to heavy automotive components, while consistently delivering repeatable and reliable performance. Additionally, while collaborative assembly is gaining traction, coordinated control of heterogeneous robots (industrial arms and cobots) in safety-critical, human-in-the-loop scenarios is still a largely unexplored research area, particularly under constraints of visual occlusion, confined geometries, and industrial cycle-time requirements.

Research Opportunities

Addressing these gaps presents several research opportunities:

- Development of systematic kinematic frameworks for hybrid gripper analysis and enhancement.
- **Integration of compact, embedded vision-based sensing to enable multi-modal feedback without compromising compliance.**
- Exploration of additional sensing modalities (resistive, light-angle) to improve perception capabilities.
- Design of modular control architectures compatible with industrial communication standards and supporting sensorless grasp.
- Real-world validation of hybrid grippers on industrial robots such as COMAU platforms to assess adaptability and robustness.
- Development of coordinated control strategies for heterogeneous robots (KUKA and UR10) in collaborative automotive assembly, including real-time safety controls and trajectory correction for KUKA industrial robots using RSI.

These opportunities define the focus of this thesis: advancing hybrid gripper performance through in-depth analysis, novel sensorization strategies, and pragmatic industrial control implementations, while extending contributions to the broader domain of human-robot collaborative assembly.

Chapter 3

Kinematic Analysis and Control of the Universal Gripper

3.1 Introduction

Robotic grippers serve as the crucial link between manipulators and their respective operating environments. This chapter delivers a comprehensive kinematic analysis along with the development of a control system for a universal gripper featuring rigid mechanics and self-adaptable fingers. Although the mechanical design originated from previous work in our laboratory, the detailed kinematic characterization, actuation control strategies, and systematic experimental validation discussed here are vital contributions toward industrial deployment. This chapter establishes both the mechanical and control framework while also highlighting constraints that led to later improvements.

The primary objectives encompass analyzing a compound Chebyshev parallelogram linkage mechanism that transforms linear actuation from an integrated stepper motor into a curvilinear translation gripper motion with minimal parasitic forces; developing a control architecture supporting multiple finger and independent actuation, and an option to rapidly interchange the self-adaptable fingers as per application requirements; characterizing the mechanical properties of TPU 98A for soft finger design; experimentally evaluating gripper performance across various object geometries; and demonstrating practical implementation with industrial robotic systems. In comparison with pneumatic actuation, this solution eliminates the compressor infrastructure, achieving a high mechanical advantage without sealing degradation or contamination sensitivity in pneumatic systems. Compared with cable-driven systems, the compound rigid linkage provides a more predictable force transmission without the cable stretch, friction losses, and maintenance issues associated with cable-driven designs. The mechanical advantage

remains constant throughout the range of motion, unlike in tendon-driven systems. By integrating precision kinematics with self-adaptable fingers, we achieve positioning accuracy that is not possible with pure soft designs while maintaining adaptive grasping capability.

3.2 Design Requirements and Specifications

The original gripper was conceived as a universal end-effector for industrial applications requiring both rigid grasping and compliance. The requirements guiding its evaluation included:

- **Payload capacity:** Capable of handling objects up to 5 kg.
- **Adaptability:** Accommodate a wide range of geometries, from cylindrical parts to irregular components.
- **Industrial integration:** Compatibility with communication protocols (Modbus RTU) for seamless integration with industrial robots.
- **Compliance:** Provide adaptive grasping without complex control, through bio-inspired Fin Ray fingers.
- **Compactness:** Overall weight less than 3 kg for robot arm compatibility.

3.3 Gripper Design and Architecture

The gripper has been developed to handle components representative of a large variety of shapes, densities, and dimensions encompassing various parts used in the industrial assembly, intralogistics, and powertrain automation. The target of the gripper design is a secure and stable grasp of these complex parts by self-encompassing around the part being manipulated. Consequently, a compliant fingertip-based approach has been utilized with specific finger kinematics developed to ensure reachability towards a component while having a curved trajectory. Stepper motors having lead screws mounted with nuts were used for the actuation of each finger of the gripper enabling precise and repeatable linear motion.

3.3.1 Gripper Design and Material

Since the objective was the development of a gripper for industrial assembly, powertrain automation, intralogistics, and other similar industrial sectors with effective grasping in various applications, it was established that the design should

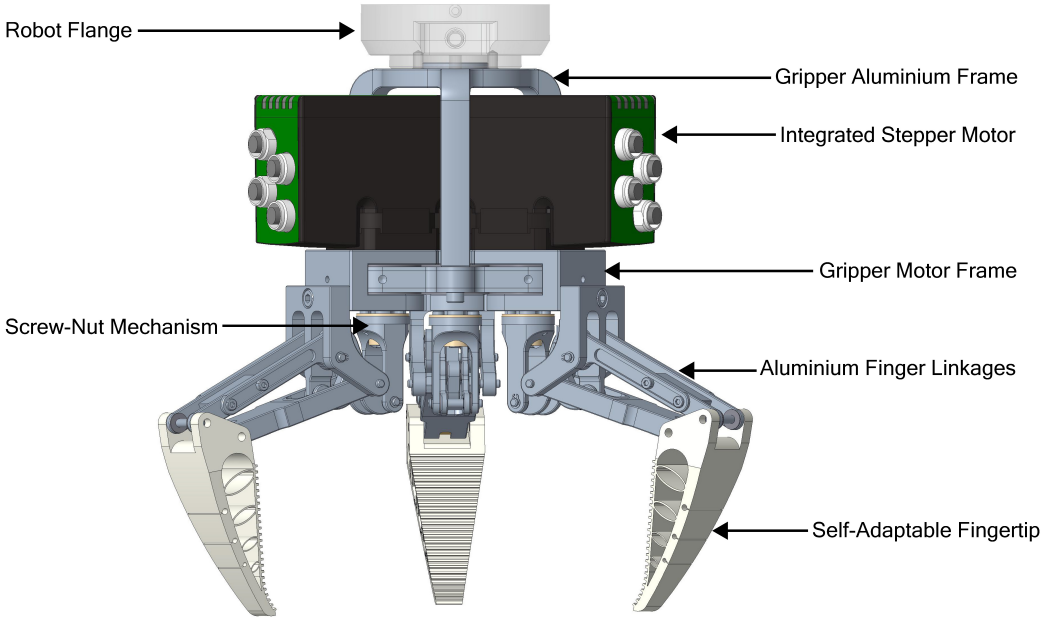


Figure 3.1: The gripper designed for industrial assembly, powertrain automation, intralogistics, and other similar industrial applications

possess a robust mechanical structure with the passive adaptive characteristics of a soft robotic gripper, making the gripper capable of handling fragile, rigid, and complexly shaped objects, thus adding versatility to the implemented design. The current state-of-the-art gripping methodologies using rigid mechanical structures as well as soft robotic gripping approaches were investigated, as reported in the previous sections. Multiple prototypes of the individual components and subsystems were synthesized starting from the state of the art and analyzed before addressing all the design input driven by the applications. A three-finger design was chosen to achieve greater universality. The three fingers are positioned 120 degrees apart from one another as a standard configuration. However, each finger can easily be set at different positions manually, enhancing the grasping range capabilities of the gripper.

The finalized grasping solution, as shown in Fig. 3.1, comprises three identical fingers individually actuated using smart integrated stepper motors, allowing for independent control of each finger. The robot flange is the mounting interface for connecting to the robot arm.

Each finger consists of a motor for actuation, a lead screw transmission with a screw-nut mechanism for interfacing the motor with the aluminum finger linkages, and a self-adaptable fingertip mounted on the coupler of the parallelogram four-bar linkage. The gripper frame and the linkages were designed according to

the industrial partner's (Stellantis-CRF) input for the machining process. The selected material for these components is a 7075 aluminum alloy due to its mechanical properties and corrosion resistance. The gripper aluminum frame is the main structure holding the three-finger structure, having a compact yet sturdy build, ensuring stability. The soft fingers were developed expressly to be additively manufactured (AM-ed) to use the shape complexity without added cost, which is one of the most relevant advantages of the AM process. The Fused Deposition Modeling (FDM) process was used to realize the soft fingers as the most cost-effective with regard to the design specifications. The material used for the soft components, Flexfill TPU 98A Fillamentum (2019) by Filamentum, is a thermoplastic rubber. The 3D printer used was the Prusa i3 MK3S by PRUSA Prusa Research (2023).

3.3.1.1 Self-Adaptable Finger

Although soft, passive Fin Ray fingers have gained much attention from the industrial sector and researchers, a standardized framework for the optimal design of such fingers for particular applications is still an open topic Deng & Li (2021). Thus, considering the application requirements, a general design methodology is followed to attain a suitable finger structure. First, gripping task requirements and parameters such as load capacity, object shapes and sizes, and adaptability level are defined. Then, a suitable material that exhibits the desired characteristics of compliance and robustness is selected. Subsequently, a conceptual structure is designed, and tests are carried out for validation either by simulation analysis or prototype testing. Finally, the performance of the finger is evaluated, and the design is refined iteratively until the requirements are met Deng & Li (2021); Elgeneidy *et al.* (2020); Lee *et al.* (2019). However, as the reason for opting for soft gripping is to have the ability to handle various objects, even after following this methodology, passive fingers may not perform as required in some real-world scenarios.

Our design of the self-adaptable fingers is inspired by pioneering research in the field of adaptive grasping and soft robotics. The Fin Ray principle was used as the design paradigm because of its inherent adaptability to object contours. The Fin Ray structure is characterized by a triangular structure with internal beams that enable passive adaptation to contact surfaces while preserving structural stability. The Fin Ray effect provides an ideal foundation for industrial applications that require both conformability and accuracy; various finger structural concepts based on this principle have been presented in Bannasch & LK (2012). The study by Jiaqiang Yao *et al.* (2024b) addresses a gap in understanding how the structural design of the internal beams of soft robotic Fin Ray fingers affects grasping performance in terms of enhanced adaptability and improved energy efficiency.

This inspired us to investigate the use of various crossbeam structures. Shin et al.'s insightful work on optimized Fin Ray fingers Shin *et al.* (2021) demonstrated enhancement in the finger structure by altering the number of internal beams and utilizing various slope configurations of these beams. Additional inspiration was drawn from research on underactuated adaptive robotic fingers with a focus on enhanced versatility in grasping various objects Birglen (2015); Shan & Birglen (2020).

While the existing research established foundational principles, our designs required custom finger structures that were compatible with gripper mechanics and application requirements. The self-adaptable finger development process involved iterative prototyping and testing of multiple designs to explore variations in the overall finger geometry, internal structure, and contact surface texture. Given the absence of standardized Fin Ray design methodologies, we adopted an experimental approach ,and compression tests were performed on each prototype to evaluate the basic force-displacement characteristics. Our optimization focused on achieving variable compliance along the finger length using geometry alone rather than complex multi-material approaches and ensuring single-step production using standard FDM printing without support structures or assembly, unlike many existing designs requiring multi-part construction. Through this iterative process, three designs with promising performance characteristics were selected for comprehensive evaluation.

Fig. 3.2 shows the three finger designs, each with an identical length of 104 mm and contact surface length of 76 mm. All designs have the same object contact surface width profile, tapering from 35 mm at the mounting base to 12 mm at the tip end. Finger 1 and 2 have similar profile heights, measuring 28 mm, while Finger 3 has an increased height of 46 mm at the mounting base. All three finger designs exhibit a decreasing height toward the tip and incorporate the standardized mounting holes for modular attachment to the gripper's rigid mechanism. Finger 1, as shown in Fig. 3.2 image (a), has a triangular contour with a network of elastic bridges within the contour to attain the desired stiffness and deformation, allowing the finger to adapt to object contours while maintaining structural integrity during force application. The tapered profile gradually reduces in width toward the tip, creating a stiffness gradient that enables firm grasping at the base while enabling delicate gripping at the tip. The interconnected bridge network distributes loads throughout the structure, unlike discrete beam designs that create stress concentrations. Finger 2, shown in Fig. 3.2 image (b), has a similar overall shape to Finger 1 but has horizontal crossbeams distributed along the finger length. The horizontal beams function as flexural hinges during interaction with objects. Finger 3, shown in Fig. 3.2 image (c), has a larger geometry with greater flexibility in regions aligned with elastic bridges, thereby enabling complex adaptive behavior. All designs share a common base

size and mounting interface to ensure interchangeability with the modular gripper platform and test equipment.

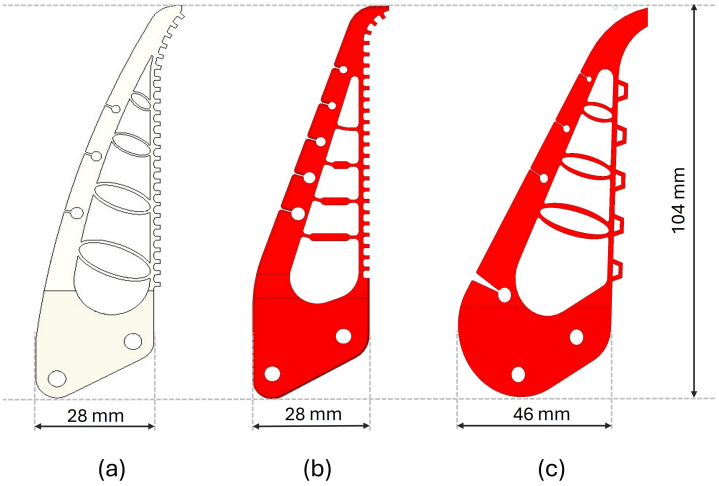


Figure 3.2: Adaptable finger designs: (a) Finger 1, (b) Finger 2, (c) Finger 3

To assess the performance characteristics of each finger design, compression tests were performed at three positions along each finger: the tip, one-third (1/3) length, and two-thirds (2/3) length. The tests were repeated five times for each finger at each compression position with a constant displacement rate of 10 mm/min, with a 5 N preload to ensure proper finger contact and a 50 N maximum force limit to prevent finger damage and retain structural integrity. This testing approach enabled a thorough characterization of force-displacement behavior throughout the finger structure, providing insights into how each design would perform when grasping objects. Compression tests were performed using the UTM, The setup consists of a base plate attached to the UTM, which acts as the platform for the testing setup. A vertical stand attached to two linear guides on its sides is mounted on the interface plate, and the finger to be tested is mounted on a finger interface block that is attached to a slider on the linear guide. A counterweight is attached to the slider on the linear guide opposite to the finger, and a pulley system is used for uniform distribution of the weight, which counters the load effect due to the finger interface block. The finger interface securely holds the finger during the application of forces regulated by the UTM. This setup is shown in Fig. 3.3. The stop block is positioned at the desired test point under the finger and acts as the object pressing into the finger surface at that particular test point, the thick black arrow indicates the UTM probe applying downward force.

The compression tests at the tip position Fig. 3.4a revealed varying mechanical

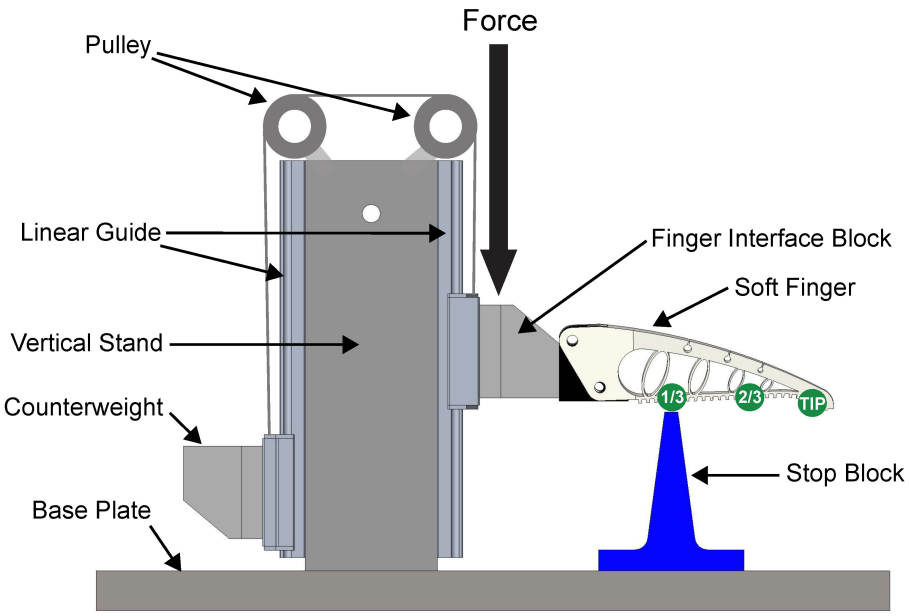
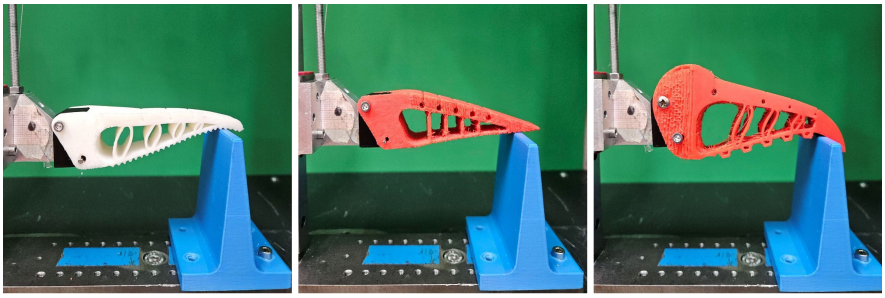


Figure 3.3: Experimental setup for finger compression tests identifying the three test positions

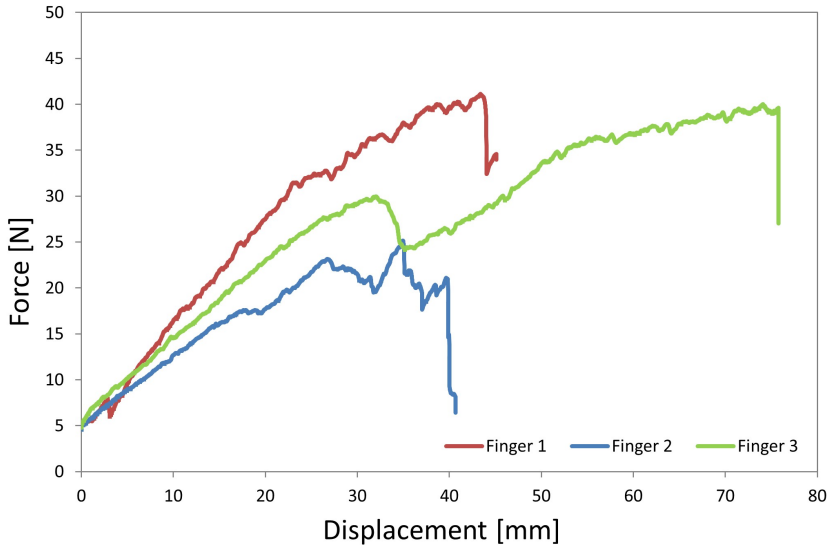
responses of the three soft finger designs Fig. 3.4b. Finger 1 reached a maximum of 41 N with a displacement of approximately 43 mm before total slippage occurred, demonstrating high force capacity. Finger 2 demonstrated the lowest force capacity with a more gradual increase in force and reached a maximum force of 25 N at a displacement of 35 mm before slippage. Finger 3 exhibited a comparatively more linear response initially, with a plateau at around 30 N followed by a gradual increase in force to 40 N at nearly 75 mm displacement before slippage. These responses indicate that Finger 1 provides superior force capacity at the tip for precision grasping, while Finger 3 offers extended displacement capability, which may be advantageous for conforming to complex geometries.

Across the five times the compression tests were repeated all designs showed consistent behavior. All finger designs showed acceptable repeatability with coefficient of variation (CoV). Peak force repeatability at tip position (n=5 trials each) for Finger 1: 41±4.2 N (CoV=10.3%), Finger 2: 25±3.2 N (CoV=12.8%), Finger 3: 40±5.7 N (CoV=14.2%).

The compression tests at the two-third position Fig. 3.5a exhibited a relatively more balanced load capacity and compliance Fig. 3.5b. The three fingers showed similar initial responses up to a displacement of approximately 9 mm and a force of 13 N. Beyond this point, the response of the finger designs significantly differed. Finger 1 exhibited the most rapid force increase, reaching 50 N at a displacement



(a)

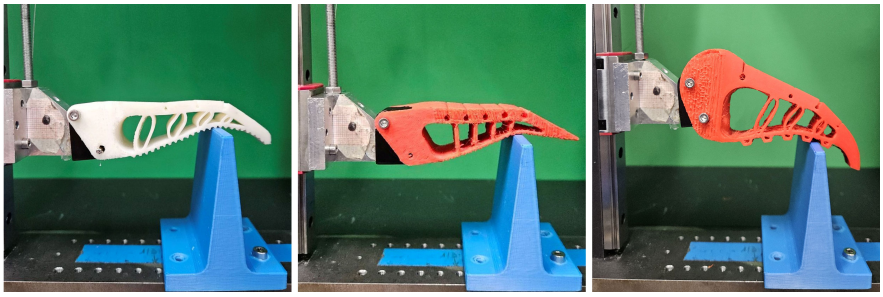


(b)

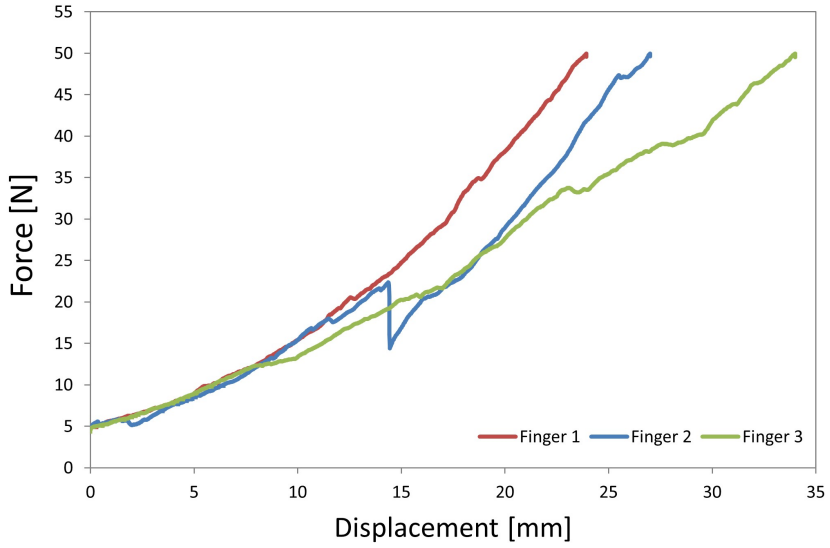
Figure 3.4: Compression test - TIP position (a) Experimental setup and (b) Compression test representative curve

of 24 mm. Finger 2 shows a drop in force after reaching 22 N, indicating compression of the internal structure as the Fin Ray effect occurs. Subsequently, a steep increase in the force is observed, reaching 50 N at 27 mm of displacement. The response curve of Finger 3 shows greater compliance with a more gradual force build-up, reaching 50 N at a displacement of 34 mm. At the two-thirds position, all fingers consistently reached the 50 N force across the trials, demonstrating good test repeatability. These results show that the two-third position allows for an optimal combination of force capacity and displacement range, making it the preferred contact region for grasping an object.

The compression tests at the one-third position Fig. 3.6a revealed higher stiffness characteristics for all fingers Fig. 3.6b. At the one-third position near the



(a)

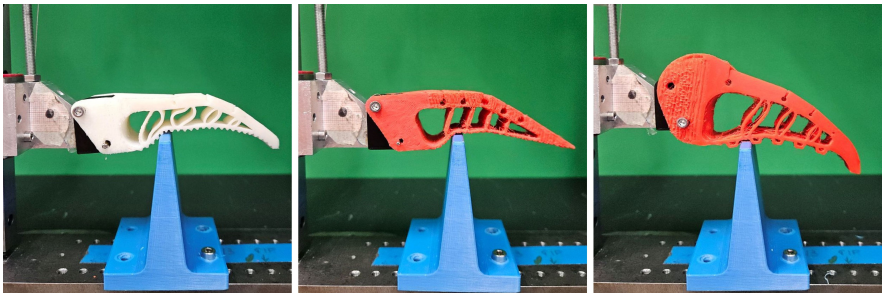


(b)

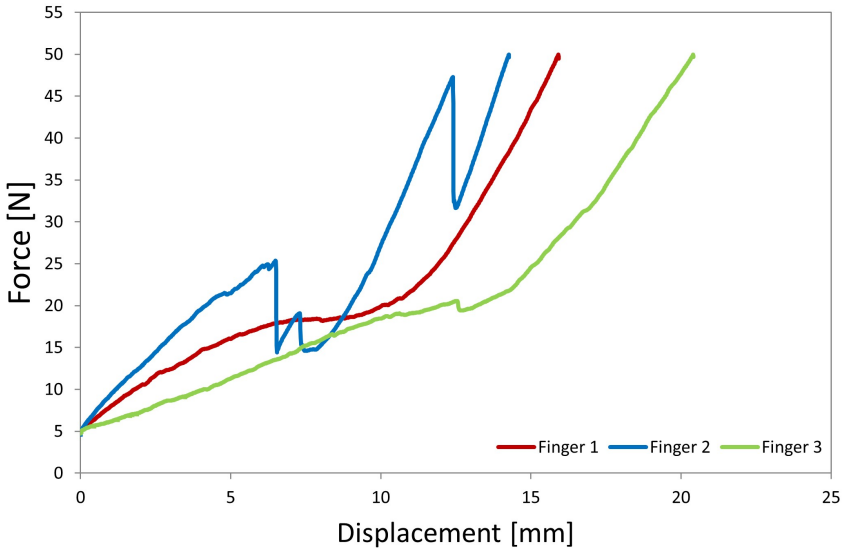
Figure 3.5: Compression test - Two Third position (a) Experimental setup and (b) Compression test representative curve

base, all fingers reached the 50 N limit at lower displacement values due to the higher structural stiffness in this region, with consistent behavior across all trials.

Finger 1 exhibits greater compliance with lower force and shows a slight decline in curve around 8 mm of displacement, followed by a rapid increase in force reaching 50 N near 16 mm of displacement. Finger 2 demonstrates a complex deformation response and exhibited the steepest force increase initially; however, as the displacement reaches 6.5 mm, there is a sudden drop in force from 25 N to 14 N, followed by another small peak of force indicating the compression of the internal beams. Afterward, a steep increase in the force was observed until 47 N, followed by another drop in the force as the finger tried to conform further, and the internal structure was compressed before reaching 50 N. Finger 3



(a)



(b)

Figure 3.6: Compression test - One Third position (a) Experimental setup and (b) Compression test representative curve

demonstrated the most gradual force progression up to 20 N, followed by a more rapid increase in force, reaching 50 N at a displacement of 20.5 mm. The force profiles at this location underscore the significance of examining the force distribution along the finger length for stable grasping. The swift increase in force in all designs at this position indicates higher structural rigidity closer to the finger base.

The compression testing at multiple positions (tip, one-third, and two-thirds) revealed nuanced performance differences among the three finger designs, providing a basis for particular design selection in specific applications. All designs showed consistent behavior across trials. Finger 1 consistently demonstrated superior force capability across all test positions, with the most consistent and

predictable force-displacement characteristics. This consistency was manifested as smoother force curves with fewer plateaus or sudden transitions, indicating more reliable grasping behavior. While Finger 3 demonstrated the advantage of extended displacement ranges, which is potentially beneficial for grasping irregular or larger objects, these benefits came at the cost of less predictable force progression and reduced fingertip force capability. Finger 2 exhibited the most significant variability in performance across all test positions, indicating less consistent behavior. The performance differences among the three designs can be attributed to their structural mechanisms. Finger 1's elastic bridge architecture distributes forces throughout the structure as circular voids provide graduated flexibility without compromising structural continuity, preventing stress concentration and enabling higher load capacity. Finger 2's horizontal crossbeams create discrete hinge points that concentrate stress, reducing the overall strength. Finger 3's larger voids and increased height provide greater compliance but reduce structural rigidity, resulting in lower force capacity despite the extended displacement range. The optimal beam angle and thickness relationships warrant future investigation through parametric studies.

For industrial powertrain applications that prioritize secure retention and consistent performance, the balance of force capability, conformity, and cross-position consistency established Finger 1 as the most suitable design. Thus, for this study, we selected Finger 1 for subsequent integration into the gripper and further testing. However, the modular architecture of the gripper system allows application-specific finger selection, This flexibility demonstrates the gripper's versatility for various industrial grasping scenarios.

3.3.1.2 Gripper Mechanics

The modular design of the gripper incorporates interchangeable soft fingers, allowing rapid reconfiguration with alternative finger morphologies and materials suitable for specific applications. The three-finger modules, comprising a rigid mechanism and soft fingers, can be easily separated from the gripper frame, and the integrated stepper motors actuating them can be easily removed from the motor frame. This extends the functional versatility of the gripper across various object shapes and tasks, from delicate object manipulation requiring compliant materials to applications requiring robust composite fingers.

To reconfigure the orientation of the gripper fingers, the finger modules can be rotated around the lead screw axis on the gripper frame in fixed angular positions adjusted using locking pins. Fig. 3.7 shows the locking pin mechanism of the finger module orientation configuration. This provision to reconfigure the gripper enhances versatility, adaptability, and efficiency across various industrial sectors and applications.

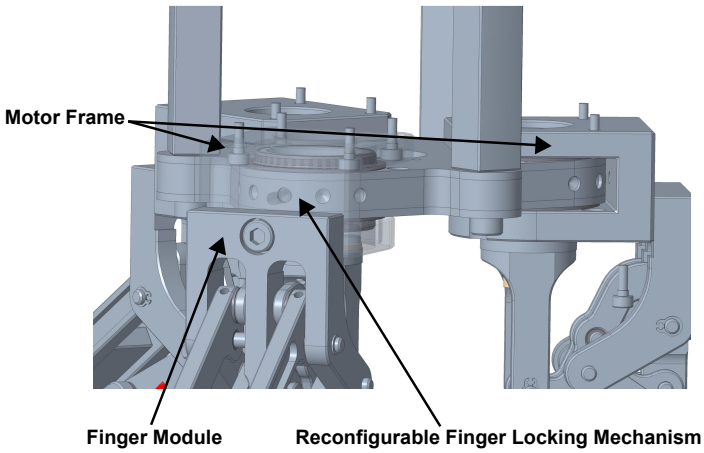


Figure 3.7: Reconfigurable finger locking mechanism with fixed angular positions

Fig. 3.8 shows some possible reconfigurations of the fingers to attain different grasp modes.

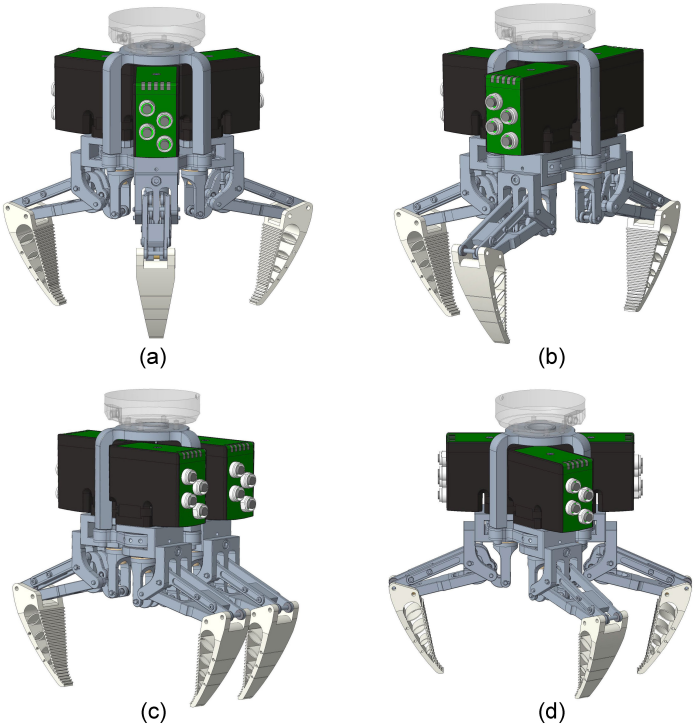


Figure 3.8: Gripper reconfigurable grasp modes (a) Equiangular configuration (120° spacing), (b) Two-finger parallel grasp, (c) Three-finger envelope grasp, and (d) Custom angular configuration

The default orientation for the gripper is the three-finger equiangular configuration with 120° spacing. The other modes are two-finger pinch or parallel grasp (two fingers opposing one another), three- finger parallel envelope grasp (two fingers on one side opposing the third), or customized angular finger arrangement for specific applications. Moreover, each finger can be controlled individually, allowing dexterous movements.

The gripper has a maximum grasping diameter of 219 mm when fully opened in a standard finger arrangement, as depicted in Figure 3.9. Much effort was put into reducing the overall weight of the gripper as components were specifically machined in steel and aluminum with a compact design for the finger, gripper frame, and link assemblies. The weight of the developed gripper is less than 2.8 kg.

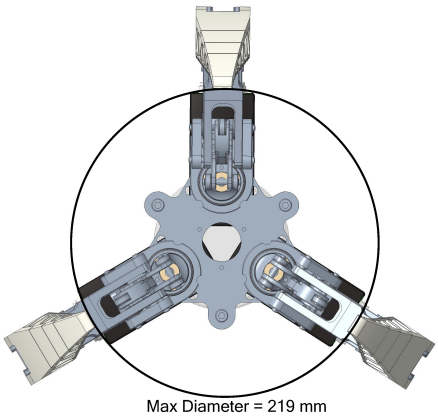


Figure 3.9: Gripper bottom view showing maximum grasping diameter

The length of the gripper when fully extended is 340 mm, and the width from the center of the flange to the motor front of a finger in the standard gripper configuration is 108 mm. The contact interface length of the self-adaptable finger is 76 mm. These dimensions are shown in Figure 3.10.

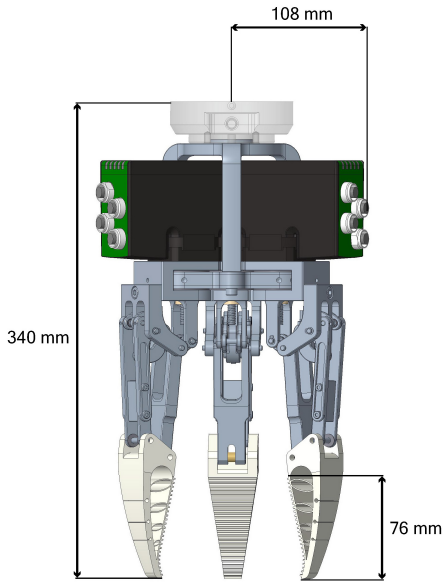


Figure 3.10: Gripper dimensions

Figure 3.11 illustrates the grasping motion of the universal gripper when approaching an object. The gripper has a curvilinear translation motion that involves both linear and rotational movements. Such motion causes the gripper to move in a curved trajectory toward the object, allowing it to grasp complex shapes. This would not be possible with a straight path motion toward an object with irregular geometry. As the gripper reaches the object in a curved path, the self-adaptable finger conforms to the shape of the object, allowing for an overall stable and firm grasp.

3.3.2 Gripper Architecture and Analysis

The kinematic chain of each finger of the gripper comprises of two subchains, a Chebyshev lambda linkage (also called Hoekens linkage) and a parallelogram four-bar linkage, as shown in Figure 3.12. This architecture was designed to transform the linear motion of a lead screw actuator into a precise gripping motion. The Chebyshev lambda linkage transforms the linear motion of the lead screw (point P on Figure 3.13) to a rotary motion of its output link L_2, which is also a member of the parallelogram four-bar linkage. The parallelogram mechanism maintains a constant orientation of the finger as it approaches the component for grasping.

Straight-line mechanisms convert angular or rotary motion to straight-line motion or vice versa. Watt, Chebyschev, Hoeken, Evans, Kempe, and Peaucellier are the most common straight-line mechanisms. An accurate straight line can only be attained using six or more links with only revolute joints. However, four-

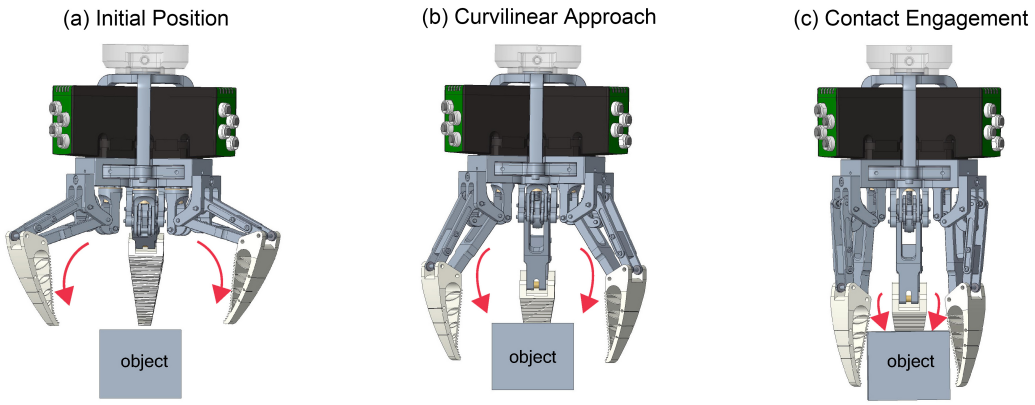


Figure 3.11: Gripper motion sequence during object grasping: (a) Initial Position - starting position with maximum finger opening, (b) Curvilinear Approach - fingers maintain orientation during curved trajectory (arrows indicate motion paths), (c) Contact Engagement - object contact with finger adaptation. Note: Illustration shows motion principle

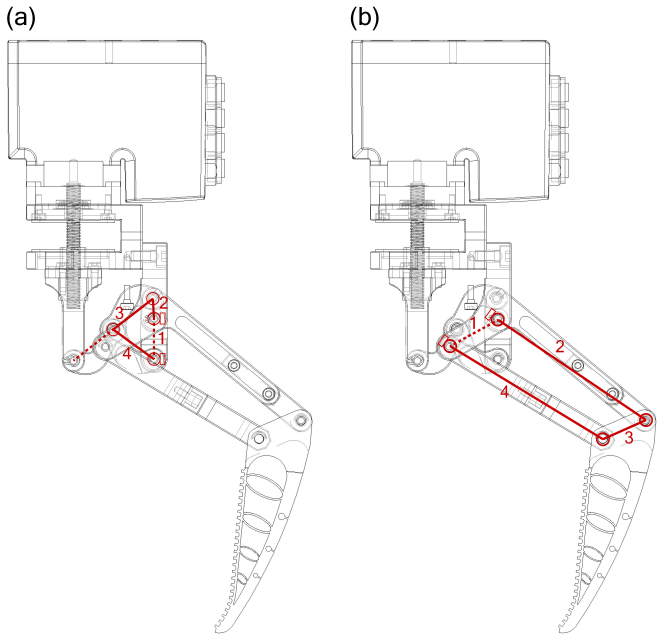


Figure 3.12: Schematic of gripper finger showing linkages: (a) Chebyshev lambda linkage, (b) Parallelogram fourbar linkage

bar approximate straight-line linkages are preferred because of their simplicity of design. In the automation industry, straight-line motion is required in numerous machines. The Chebyshev lambda linkage (also known as Hoekens linkage, a

cognate mechanism of the Chebyshev linkage) is a four-bar mechanism that can generate a partial straight line trajectory with less than one percent tolerance. The Chebyshev lambda linkage can be driven by a motor as it is a crank-rocker Lu *et al.* (2014); Norton (2014). This makes it an optimal linkage for the gripper design, for the conversion of linear input to angular motion. The gripper weight, maximum diameter of grasp, range of motion, and the link length ratios as per required motion were the factors considered in determining the dimensions of the subchains. Table 3.1 shows the dimensional parameters of the linkages.

Table 3.1: Dimensional parameters of the mechanism in millimeter (mm).

Linkage Type	**Linkage Lengths**			
Chebyshev lambda	L_1	L_2	L_3	L_4
	18	9	22.5	22.5
Parallelogram fourbar	L_5	L_6	L_7	L_8
	25	80	25	80

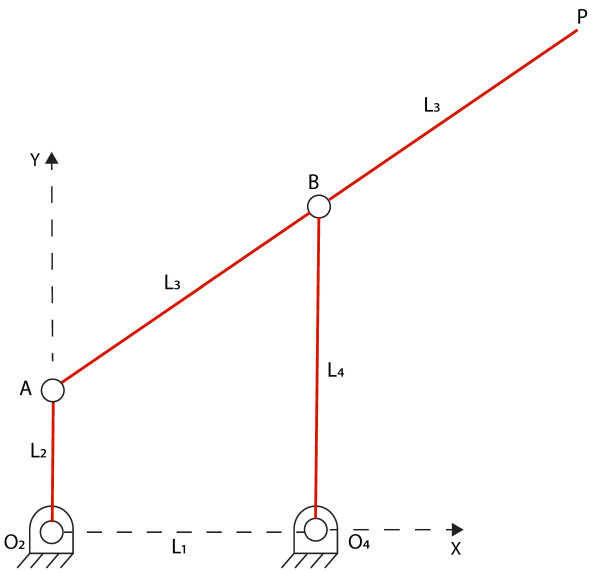


Figure 3.13: The Chebyshev lambda linkage

A reference schematic of the Chebyshev lambda mechanism with joint angles in the reference configuration is shown in Figure 3.13. The link length configuration of the linkage is L_1:L_2:L_3:L_4 = 2:1:2.5:2.5. Due to the links length configuration of the Chbyshev lambda linkage there always exists a right angle

ΔAO_4P at point O_4 as shown in Figure 3.14. This geometrical property is used to solve the kinematics of the Chebyshev lambda linkage Ju *et al.* (2024); Lu *et al.* (2014).

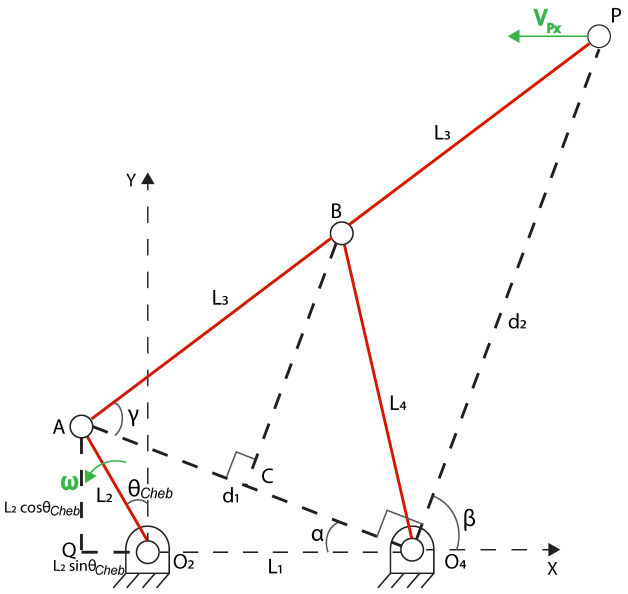


Figure 3.14: Geometrical analysis of the Chebyshev lambda linkage

The link lengths are denoted as, $L_1 = L_{O_2O_4}$, $L_2 = L_{O_2A}$, $L_3 = L_{AB}$ and $L_4 = L_{BO_4}$ with $L_{O_2O_4}$, L_{O_2A}, L_{AB} and L_{BO_4} being the lengths between specified points. For a given input angle θ_{Cheb}, the position of point A is determined by:

$$A_x = L_2 cos(\theta_{Cheb}) \tag{3.1}$$

$$A_y = L_2 sin(\theta_{Cheb}) \tag{3.2}$$

The distance d_1 between points A and O_4 can be calculated using ΔAQO_4:

$$\begin{aligned} d_1 &= \sqrt{(L_1 + L_2 \sin(\theta_{Cheb}))^2 + (L_2 \cos(\theta_{Cheb}))^2} \\ d_1 &= \sqrt{(2L_2 + L_2 \sin(\theta_{Cheb}))^2 + (L_2 \cos(\theta_{Cheb}))^2} \\ &= L_2\sqrt{5 + 4\sin(\theta_{Cheb})} \end{aligned} \tag{3.3}$$

where $L_1 = 2L_2$ in accordance with the link length ratio and θ_{Cheb} as the angle

that defines the angular movement of the crank with the revolute joint at point O_2.

A perpendicular bisecting the vertex angle and base of the isosceles triangle ΔAO_4B results in two congruent triangles. Thus, the distance d_1 can also be calculated considering ΔACB:

$$d_1 = 2L_3 \cos\gamma \tag{3.4}$$

Based on the link length configuration $L_3 = \frac{5}{2}L_2$, giving:

$$d_1 = 5L_2 \cos\gamma \tag{3.5}$$

By equating Equation (4.2) and Equation (3.5) the value of $\cos\gamma$ can be evaluated:

$$cos\gamma = \frac{1}{5}\sqrt{5 + 4\sin(\theta_{Cheb})} \tag{3.6}$$

Equation (3.6) can be used to evaluate the value of $\sin\gamma$, given as:

$$\begin{aligned} sin\gamma &= \sqrt{1 + \frac{1}{25}(-5 - 4\sin(\theta_{Cheb}))} \\ &= \frac{2}{5}\sqrt{5 - \sin(\theta_{Cheb})} \end{aligned} \tag{3.7}$$

Considering the triangle ΔAO_4P the distance d_2 between P and O_4 is given by:

$$\begin{aligned} d_2 &= 2L_3 \sin\gamma \\ &= 2L_2\sqrt{5 - \sin(\theta_{Cheb})} \end{aligned} \tag{3.8}$$

Using ΔAQO_4 $sin\alpha$ can be determined, where α complements β that is the angle between PO_4 and x-axis:

$$\begin{aligned} sin\alpha &= \frac{L_2 cos(\theta_{Cheb})}{d_1} \\ &= \frac{cos(\theta_{Cheb})}{\sqrt{5 + 4\sin(\theta_{Cheb})}} \end{aligned} \tag{3.9}$$

Equation (3.9) can be used to find the value of $\cos\alpha$ as:

$$cos\alpha = \sqrt{\frac{(2+\sin(\theta_{Cheb}))^2}{5+4\sin(\theta_{Cheb})}} \tag{3.10}$$

From the above relations, the position P of the coupler corresponding to the crank angle θ_{Cheb} can be evaluated:

$$\begin{aligned} P_x &= L_1 + d_2\cos\beta = L_1 + d_2\sin\alpha \\ &= 2L_2\left(1 + \frac{\cos(\theta_{Cheb})\sqrt{5-\sin(\theta_{Cheb})}}{\sqrt{5+4\sin(\theta_{Cheb})}}\right) \end{aligned} \tag{3.11}$$

$$\begin{aligned} P_y &= d_2\sin\beta = d_2\cos\alpha \\ &= 2L_2\frac{\sqrt{5-\sin(\theta_{Cheb})}(2+\sin(\theta_{Cheb}))}{\sqrt{5+4\sin(\theta_{Cheb})}} \end{aligned} \tag{3.12}$$

The velocity of components of point P can be obtained as:

$$V_{Px} = \frac{dP_x}{d\theta_{Cheb}}\omega \tag{3.13}$$

$$V_{Py} = \frac{dP_y}{d\theta_{Cheb}}\omega \tag{3.14}$$

The term ω represents the angular velocity of the link L_2, expressed in rad/s. The velocity of point A, the Chebyshev linkage output, can be calculated by differentiating its component equations with respect to time as follows:

$$V_{Ax} = -L_2\omega sin(\theta_{Cheb}) \tag{3.15}$$

$$V_{Ay} = L_2\omega cos(\theta_{Cheb}) \tag{3.16}$$

The four-bar linkage is a closed-chain linkage having four binary links with one DoF. Figure 3.15 shows a parallelogram linkage which is considered a special-case Grashof linkage, having opposite sides equal in length. The parallelogram linkage is advantageous as it precisely duplicates the rotary motion of the input link at the output links. The coupler of the parallelogram four-bar linkage has a curvilinear translation motion, maintaining the same angle while traveling in a curved path. Hence, the coupler retains its orientation throughout the motion Norton (2014). This property of the parallelogram four-bar linkage allows the

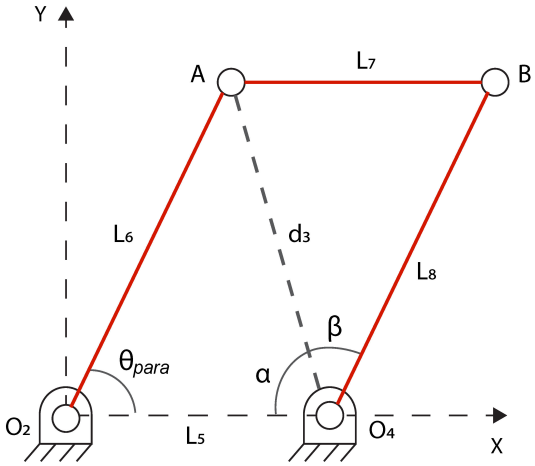


Figure 3.15: Geometrical analysis of four-bar parallelogram linkage

gripper fingertips to move in a curved path without altering its orientation when approaching an object.

In the parallelogram four-bar mechanism, the positions of A and B should be evaluated over time. As θ_{para} is the angle between the crank and fixed frame on the x-axis. The x and y coordinates of A are given by:

$$A_x = L_6 cos(\theta_{para}) \tag{3.17}$$

$$A_y = L_6 sin(\theta_{para}) \tag{3.18}$$

Here, θ_{para} is derived from the position of point A in the Chebyshev linkage. The angle of the input link to the parallelogram is given by:

$$\theta_{para} = \arctan(\frac{A_y}{A_x}) \tag{3.19}$$

Here, A_x and A_y are obtained using Chebyshev linkage kinematics to calculate θ_{para}.

The link lengths are defined as, $L_5 = L_{O_2O_4}$, $L_6 = L_{O_2A}$, $L_7 = L_{AB}$ and $L_8 = L_{BO_4}$. The distance d_3 between points A and O_4 can be found using law of cosines:

$$d_3 = \sqrt{L_5^2 + L_6^2 - 2L_5L_6\cos(\theta_{para})} \tag{3.20}$$

Next, we can calculate the angle α between L_5 and d_3 using the law of sines:

$$sin\alpha = \frac{L_6 \sin(\theta_{para})}{d_3} \tag{3.21}$$

The relation for the distance between points A and B is:

$$L_7 = \sqrt{d_3^2 + L_8^2 - 2d_3 L_8 \cos\beta} \tag{3.22}$$

Equation (3.22) can be solved further to get the relation for β as:

$$cos\beta = \frac{d_3^2 + L_8^2 - L_7^2}{2d_3 L_8} \tag{3.23}$$

Hence, the x and y coordinates of point B are given by:

$$B_x = L_5 - L_8 \cos(\alpha + \beta) \tag{3.24}$$

$$B_y = L_8 \sin(\alpha + \beta) \tag{3.25}$$

In this configuration, because $L_5 = L_7 = 25$ mm and $L_6 = L_8$, the motion of point B is a direct translation of point A. Therefore, the coordinates of point B can be found alternatively using a direct relation between points A and B, as the offset between them remains constant in orientation while only the magnitude changes:

$$B_x = A_x + (L_5 - L_6 \cos(\theta_{para})) \tag{3.26}$$

$$B_y = A_y + L_6 \sin(\theta_{para}) \tag{3.27}$$

The angular velocity of the input link of the four-bar parallelogram linkage is derived from the motion of point A as follows:

$$\omega_{para} = \frac{d\theta_{para}}{d\theta_{Cheb}}\omega \tag{3.28}$$

The velocity components of B can be found by differentiating :

$$V_{Bx} = V_{Ax} + \omega_{para} L_6 \sin(\theta_{para}) \tag{3.29}$$

$$V_{By} = V_{Ay} + \omega_{para} L_6 \cos(\theta_{para}) \tag{3.30}$$

The Chebyshev lambda linkage and the parallelogram linkages are coupled such that the link L_2 of the Chebyshev lambda linkage acts as the input to the parallelogram linkage, causing an angular rotation of the link L_6 of the parallelogram four-bar linkage. To understand the gripper dynamics, we performed a velocity analysis using a standard input angular velocity of $\omega = 1$ rad/s. This unit angular velocity reference allows easy scaling for different operating speeds

and simplifies the interpretation of mechanical advantage by directly comparing velocity magnitudes.

The velocity analysis results in Figure 3.16 show that the velocity magnitude (9.00 mm/s) of point A (Chebyshev output) is constant throughout the motion and is equal to $L_2\omega$, this reflects the uniform circular motion. Point P (lead screw connection) maintains a nearly constant velocity magnitude (12.02 mm/s±0.20%) throughout the operating range ($\theta_{\text{Cheb}} = 60°$ to 120°). This is achieved with the optimized 1:2.5 link ratio configuration in the Chebyshev lambda linkage.

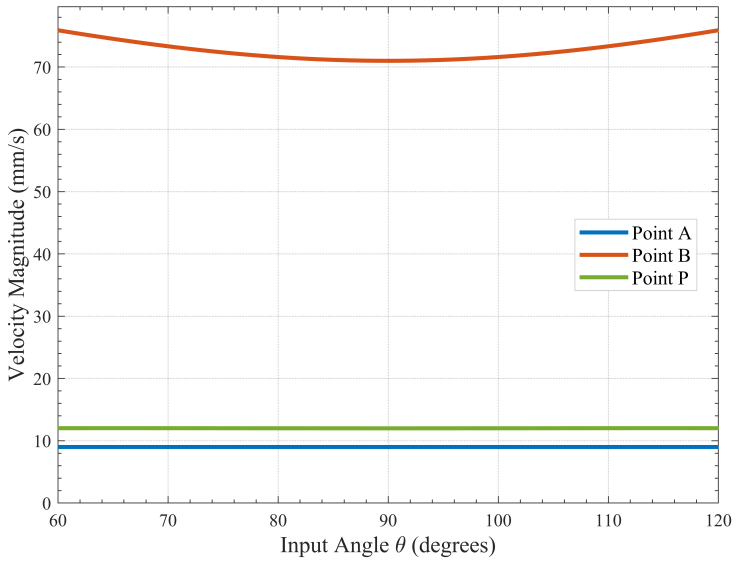


Figure 3.16: Velocity Magnitude vs Input Angle

Unlike points A and P, the velocity magnitude of point B (parallelogram linkage output) exhibits a nonlinear profile across the input angle range. The velocity of point B is highest at the extremes of the motion range ($\theta_{\text{Cheb}} = 60°$ and 120°) and reaches a minimum near the middle of the range (approximately at ($\theta_{\text{Cheb}} = 90°$). This velocity variation (70-77 mm/s) arises from the geometric coupling of the linkages, where the angular velocity of the input link of the parallelogram (ω_{para}) approaches zero for an instant in the mid-range. The variable velocity profile of point B provides various operational advantages; higher velocities at motion extremes enable rapid approach and closure, improving cycle times; reduced mid-range velocity provides finer control during critical object contact and grasp motion; and smooth transitions between these states result in gradual acceleration profiles that minimize vibration and shock loads during grasping operation.

The mechanical advantage of the system is calculated as the velocity ratio between points B and P, which demonstrates the force multiplication character-

istics of the mechanism. This ratio averaged 6.06:1, with a maximum of 6.31:1, indicating that while point B moves approximately six times faster than the lead screw input at point P.

For precise gripper control through the lead screw motion, linear motion throughout the operating range is critical because it eliminates lateral forces that would otherwise cause premature wear, reduce efficiency, and potentially create binding in actuator-linkage interface. The linear motion analysis of the system is shown in Figure 3.17. Point P exhibits near-perfect linear motion (y = -0.0000x + 36.0201), showing minimal deviations (maximum 0.0333 mm, average 0.0147 mm) across the complete operating range. Point P travels 12 mm (from x = 24 mm to x = 12 mm) while maintaining a constant height of 36.02 mm, representing the linear displacement required from the lead screw actuator.

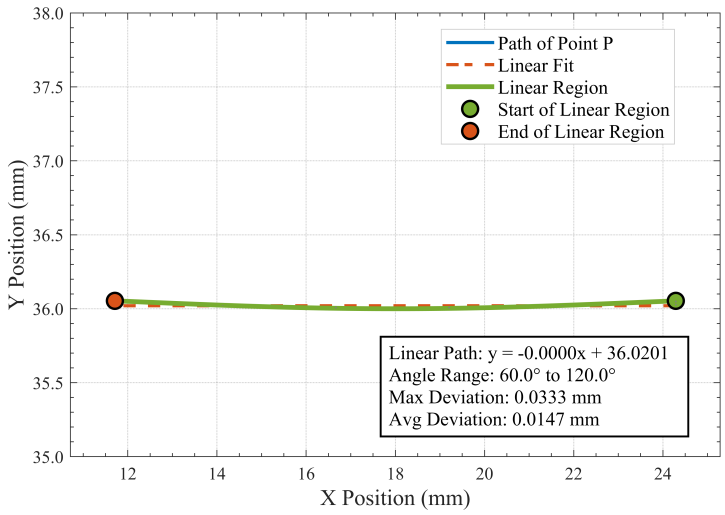


Figure 3.17: Linear Motion Analysis of Point P

The Figure 3.17 shows two overlapping elements: the blue line representing point P's actual trajectory and the green line highlighting regions with deviation <0.05mm from perfect linearity. These appear identical because 100% of the path qualifies as a linear region, as all the deviations are below the threshold.

The kinematic analysis demonstrated that the proposed gripper architecture with the combined Chebyshev and parallelogram linkages achieves exceptional motion characteristics for precision gripping applications, with near-perfect linear motion (maximum deviation ±0.033 mm), consistent velocity control, and significant mechanical advantage (up to 6.31). This motion precision meets industrial standards for automated assembly operations, which generally require ±0.1 mm for general object handling and ±0.05 mm for precision tasks ISO (1998); SCHUNK GmbH & Co. KG (2025).

3.4 Actuation and Control

This section presents the development and evolution of the actuation system and control architecture for a three-fingered robotic gripper. The system employs JVL integrated stepper motors (MIS171 Series) with a two-phase development approach: an initial LabVIEW implementation that established core control functionality and validated the hardware architecture, followed by an enhanced Python implementation that extended capabilities with advanced features including synchronized multi-motor operation and multi-modal feedback mechanisms. This progressive development approach enabled rapid prototyping while ultimately achieving a production-ready control system.

3.5 Hardware Architecture

3.5.1 Motor Specifications

The gripper comprises three independent underactuated fingers, each actuated by JVL A/S Integrated Stepper Motors (MIS171 Series) with lead screw mechanisms as shown in Figure 3.18. Key specifications include:

- **Integrated Design**: Motor, encoder, controller, and communication board in a single unit
- **Closed-loop Operation**: Continuous rotor position monitoring with stall-free operation
- **Performance**: Running torque of 0.18 Nm, resolution of 409,600 counts/revolution
- **Power**: Motor power 12–72 VDC (max 4.0 A), control power 7–28 VDC (<200 mA)

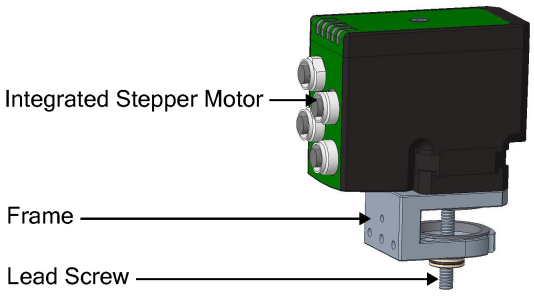


Figure 3.18: Integrated Stepper Motor

The motors support both MacTalk (proprietary JVL protocol) and Modbus RTU (industrial standard) over RS-485 serial interface. The system utilizes Modbus RTU at 19,200 baud with 8 data bits, even parity, and 1 stop bit for operational control.

3.6 Control Architecture Development

3.6.1 Phase 1: LabVIEW-Based Control Interface

The initial control system was developed using National Instruments LabVIEW, providing a graphical programming environment well-suited for rapid prototyping and hardware validation, the architecture is shown in Figure 3.19. This phase successfully established the fundamental control architecture and validated the motor specifications and communication protocols.

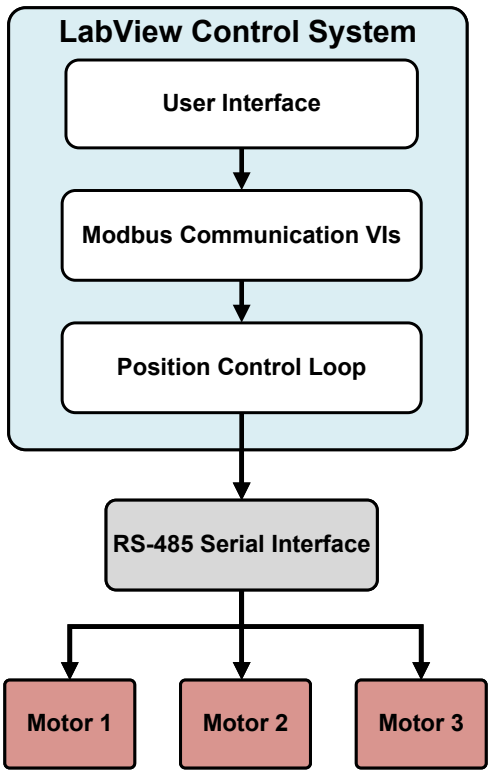


Figure 3.19: LabVIEW control system architecture with sequential motor control

Algorithm 1 presents the LabVIEW control structure, which established basic position control capabilities.

Algorithm 1 LabVIEW Control Architecture

procedure SystemInitialization($\mathcal{M}$)
  for all $m_i \in \mathcal{M} = \{M_1, M_2, M_3\}$ **do**
    addr(m_i) $\leftarrow i$
    if WaitForMacTalk(10 s) **then**
      ContinueWithMacTalk(m_i)
    else
      SetModbusRegister(m_i, 125) ▷ Switch to Modbus RTU
      ConfigureModbus(19200, 8, E, 1)
    end if
  end for
end procedure
procedure PositionControl(m_i, p_{target})
  $p_{\text{current}} \leftarrow$ ReadPosition(m_i)
  while $|p_{\text{target}} - p_{\text{current}}| > 0$ **do**
    $\Delta p \leftarrow p_{\text{target}} - p_{\text{current}}$
    SendCommand($m_i, \Delta p$)
    $p_{\text{current}} \leftarrow$ ReadPosition(m_i)
  end while
end procedure
procedure GripperControl($\mathcal{M}, \theta$) ▷ $\theta \in [0, 1]$
  $P_{\text{sync}} \leftarrow$ CalculatePositions(θ)
  for all $m_i \in \mathcal{M}$ **do** ▷ Sequential execution
    PositionControl($m_i, P_{\text{sync}}[i]$)
  end for
end procedure

The LabVIEW implementation successfully achieved its primary objectives:

- **Hardware Validation**: Confirmed motor specifications and communication protocol functionality
- **Control Paradigm**: Established position-based control architecture for three independent fingers
- **Communication Reliability**: Achieved communication latency <50 ms at 19,200 baud with consistent operation
- **Position Control**: Demonstrated command response time <100 ms with inter-finger synchronization error <2 mm
- **Operational Range**: Validated full range of motion with consistent 2-second closing time

- **Protocol Implementation**: Successfully configured motors from MacTalk to Modbus RTU operational mode

These achievements provided confidence in the hardware selection and established the foundational control approach, enabling progression to more advanced features.

3.6.2 Phase 2: Enhanced Python Implementation

Building upon the validated LabVIEW foundation, a Python-based control system was developed to extend functionality and enable more sophisticated control strategies. This transition facilitated implementation of advanced features including truly synchronized motor control, comprehensive feedback monitoring, and intelligent grip detection. Figure 3.20 shows the enhanced architecture.

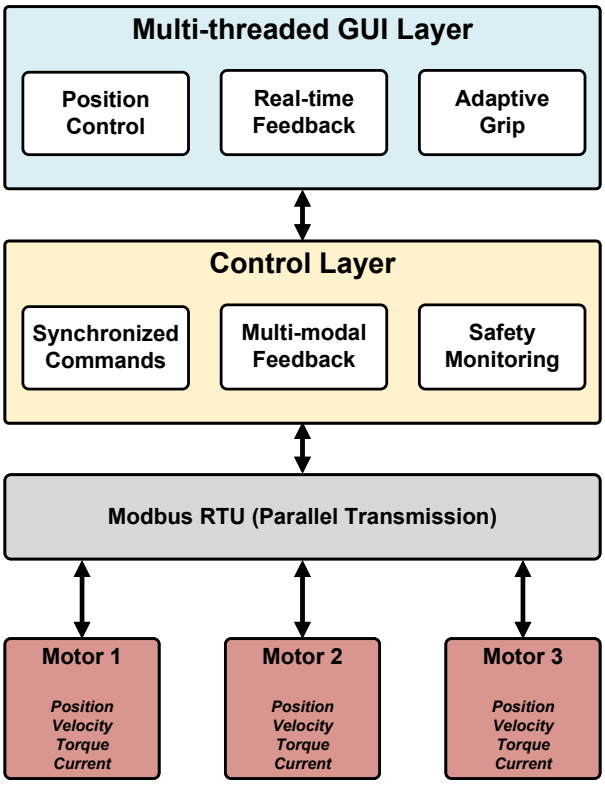


Figure 3.20: Python control system with synchronized commands and multi-modal feedback

3.7 Advanced Features in Python Implementation

3.7.1 Synchronized Command Transmission

The most significant enhancement over the LabVIEW system is the implementation of truly synchronized motor control, eliminating the sequential command delay observed in the initial system. Algorithm 2 details this approach.

Algorithm 2 Synchronized Multi-Motor Control

procedure SendSynchronizedCommands(motor_positions)
  Phase 1: Pre-build command frames ▷ No I/O operations
  command_frames ← []
  for all $(m_{\text{id}}, p_{\text{target}}) \in$ motor_positions **do**
    frame ← BuildModbusFrame($m_{\text{id}}, p_{\text{target}}$)
    command_frames.append(frame)
  end for
  Phase 2: Clear communication buffers
  SerialPort.ClearInputBuffer()
  SerialPort.ClearOutputBuffer()
  Phase 3: Rapid transmission
  $t_{\text{start}} \leftarrow$ GetCurrentTime()
  for all frame ∈ command_frames **do**
    SerialPort.Write(frame) ▷ $< 1\mu$s between writes
  end for
  SerialPort.Flush() ▷ Force immediate transmission
  Phase 4: Non-blocking response collection
  responses ← CollectResponsesAsync(timeout = 10ms)
  $t_{\text{transmission}} \leftarrow$ GetCurrentTime() $- t_{\text{start}}$
  return (responses, $t_{\text{transmission}}$)
end procedure

Performance Achievement: Synchronization delay reduced from 100–300 ms (LabVIEW sequential implementation) to <1 ms (Python parallel transmission).

3.7.2 Multi-Modal Feedback Integration

Extending beyond the position-only monitoring in the LabVIEW system, the Python implementation monitors four parameters simultaneously through dedicated register reads: position (register 10), velocity (register 11), torque (register

12), and current (register 13). The `read_all_feedback()` method provides real-time monitoring of all parameters, enabling intelligent grip detection through two complementary strategies.

3.7.2.1 Velocity-Based Stall Detection

Figure 3.21 illustrates the velocity-based detection logic, which identifies object contact when motor velocity drops below 500 encoder units/second for 5 consecutive readings despite active position commands. The system also monitors position change (tolerance: 5000 units) to confirm genuine stalling versus commanded stops.

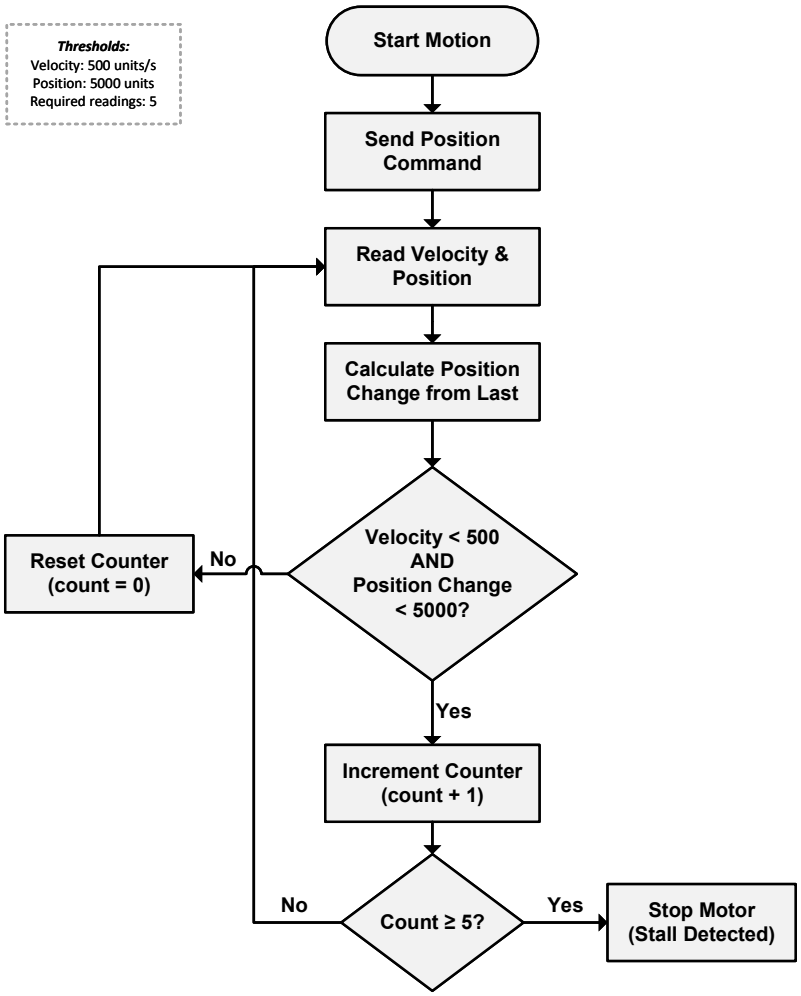


Figure 3.21: Velocity-based stall detection flowchart

The system monitors motor velocity and position in encoder units, the native

output from the motor's integrated encoder (409,600 counts/revolution). A velocity below 500 encoder units/second combined with position change less than 5000 units for five consecutive readings (250 ms) indicates a stalled condition, signaling object contact.

3.7.2.2 Torque-Based Grip Detection

Torque monitoring provides direct force feedback, enabling adaptive gripping based on object resistance. Motor-specific thresholds (M_1: 15,000, M_2: 25,000, M_3: 20,000 encoder units) compensate for mechanical variations. Algorithm 3 presents this strategy.

Algorithm 3 Torque-Based Grip Detection

procedure TorqueGripDetection(m_{id})
  $T_{\text{thresh}} \leftarrow \{M_1 : 15000, M_2 : 25000, M_3 : 20000\}$
  $\tau_{\text{baseline}} \leftarrow$ ReadTorque(m_{id})
  while MotorMoving(m_{id}) **do**
    $\tau_{\text{current}} \leftarrow$ ReadTorque(m_{id})
    $\Delta\tau \leftarrow |\tau_{\text{current}} - \tau_{\text{baseline}}|$
    if $\Delta\tau > T_{\text{thresh}}[m_{\text{id}}]$ **then**
      $F_{\text{grip}} \leftarrow$ EstimateForce($\Delta\tau$)
      ExecuteControlledStop(m_{id})
      return (GRIP_DETECTED, F_{grip})
    end if
    UpdateBaseline($\tau_{\text{baseline}}, \tau_{\text{current}}$)
  end while
end procedure

3.7.3 Hybrid Control Strategy

The system combines velocity and torque detection for robust operation across diverse object types. Figure 3.22 shows the integrated control flow.

3.7.4 Safety and Monitoring Features

The Python implementation includes several safety mechanisms absent in the LabVIEW system:

1. **Emergency Stop**: Dual-action stop sequence that first sets velocity to zero, then locks position to prevent unintended movement. Response time <100 ms.

2. **Position Limits**: Software-enforced bounds (minimum: -2.1×10^6, maximum: 7×10^6 encoder counts) prevent mechanical damage from over-travel.

3. **Multi-threaded Monitoring**: Background thread updates position displays every 500 ms without blocking command execution, providing real-time system awareness.

4. **Communication Validation**: CRC16 verification on all Modbus transactions ensures data integrity, with automatic retry on checksum failures.

These safety features complement the intelligent grip detection, creating a robust control framework suitable for production environments.

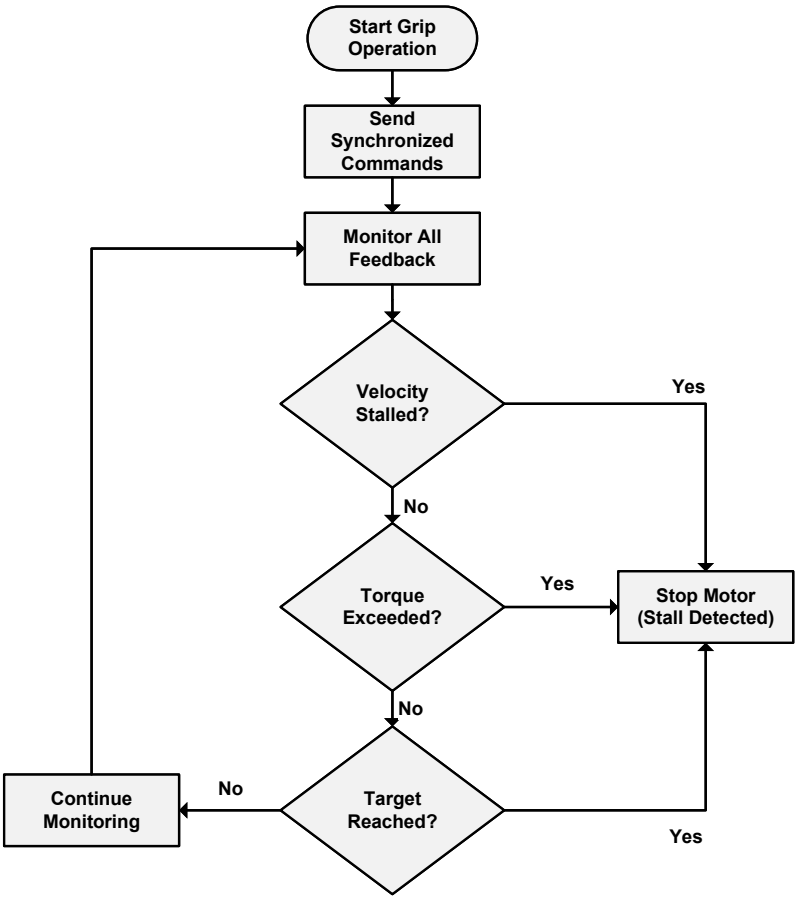


Figure 3.22: Hybrid control strategy integrating multiple feedback modes

The two-phase development approach proved effective for this robotic gripper system. The LabVIEW implementation successfully established and validated the fundamental control architecture, confirming hardware specifications

and demonstrating reliable position-based control. This foundation enabled confident progression to more advanced features.

The Python implementation built upon this validated foundation to introduce significant enhancements:

1. **Coordinated Motion**: Parallel command transmission architecture eliminated sequential delays, enabling truly synchronized three-finger operation

2. **Comprehensive Sensing**: Utilizing all available motor feedback parameters (position, velocity, torque, current) enabled development of intelligent control strategies

3. **Adaptive Control**: Hybrid velocity-torque grip detection accommodates diverse object properties, preventing damage to delicate items while ensuring secure grasps

4. **Robust Safety**: Multi-layered emergency stop, position limits, and continuous monitoring ensure safe operation in practical applications

The motor-specific torque thresholds (M_1: 15,000, M_2: 25,000, M_3: 20,000) demonstrate the system's capability to compensate for hardware non-idealities, a critical requirement for practical robotic systems. Rather than treating all fingers identically, the system adapts to actual mechanical variations.

This section presented the progressive development of the gripper actuation system through two implementation phases. The initial LabVIEW system successfully established position-based control with reliable communication (<50 ms latency), validated hardware specifications, and demonstrated consistent gripper operation with 2-second closing times and inter-finger synchronization within 2 mm. Building upon this validated foundation, the Python implementation introduced substantial enhancements: improvement in motor synchronization (<1 ms), comprehensive 4-parameter feedback monitoring at 20 Hz, and intelligent grip detection. These advancements, combined with robust safety features, provide a production-ready control system suitable for precise robotic manipulation tasks. The two-phase approach demonstrates the value of progressive development: rapid prototyping to validate concepts, followed by targeted enhancements to achieve advanced functionality.

3.8 Experiments and Results

To evaluate the complete gripper assembly's object retention and holding capabilities, tensile (pull-out) and grasping tests were conducted. Each tensile test was performed five times to ensure statistical validity. A range of objects was tested

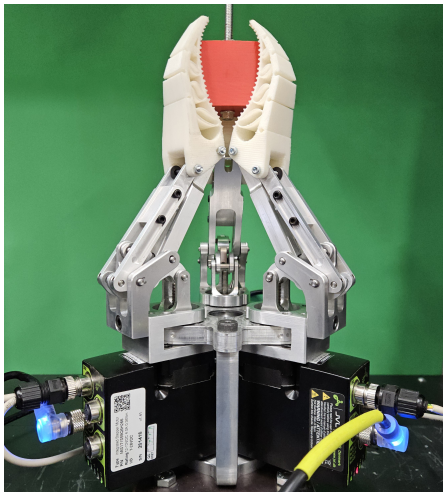
for the suitability of the gripper, with a primary focus on automotive industrial parts, while grasping tests were performed with the gripper mounted on A 6DOF Comau industrial robot.

3.8.1 Tensile (pull-out) tests

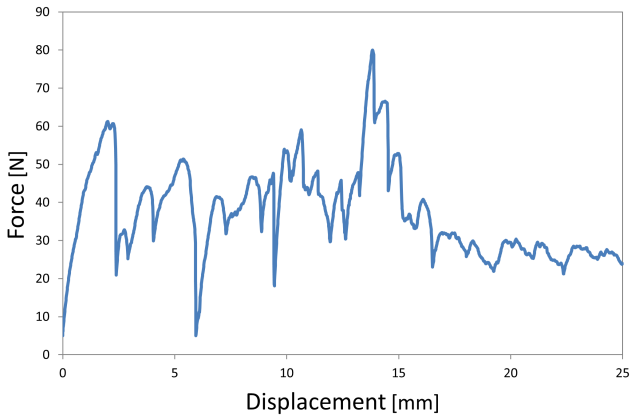
Pull-out experiments were carried out with the gripper assembly mounted on the UTM to measure the force and displacement values. While the gripper's modular design enables multiple configurations, the default three-finger equiangular configuration (120° spacing) was selected for retention testing because it represents the optimal arrangement for maximum grasping stability and force distribution. Moreover, using the default configuration, the tests provide the baseline inherent gripper retention capability. In practical applications, alternative configurations can be selected in accordance with object geometry. The tests were performed at a controlled displacement rate of 10 mm/min with a 5 N preload, allowing sufficient data resolution to capture the characteristic force peaks and plateaus during gripping and slippage. The tests were performed five times and demonstrated good repeatability with a CoV ranging from 9-13% for peak retention forces.

First, a 3D printed cube of 40 mm was used to test the three-finger grasp, as shown in Fig. 3.23a. The corresponding result revealed a complex pattern with multiple peaks varying between 31 N to 79 N of force, followed by smaller peaks varying between 21 N to 29 N as the object is near total slippage after 17 mm displacement, as shown in Fig. 3.23b. The initial force peak of 61 N at 2 mm of displacement represents the static friction threshold of the grasp, while subsequent peaks indicate partial slippage and grip adjustment events as the test implements a continuous pull-out of the object. The region where smaller peaks are observed indicates the dynamic friction phase, where the object is undergoing continuous, slow slippage through the fingers, approaching total loss of contact. Maximum retention force observed across trials with the 3D printed cube was 79±7.8 N (CoV=9.9%) and initial peak variation of 61±6 N (CoV=9.8%).

The equiangular configuration proved effective for the tested cubic shape, with three contact points simultaneously engaging different faces of the cube. The sustained dynamic friction force observed in our experiments fluctuated between 21 N and 29 N, indicating a range of grip stability during controlled slippage, translating to a practical load capacity varying approximately between 2.1 and 2.9 kg (accounting for gravitational acceleration).



(a)



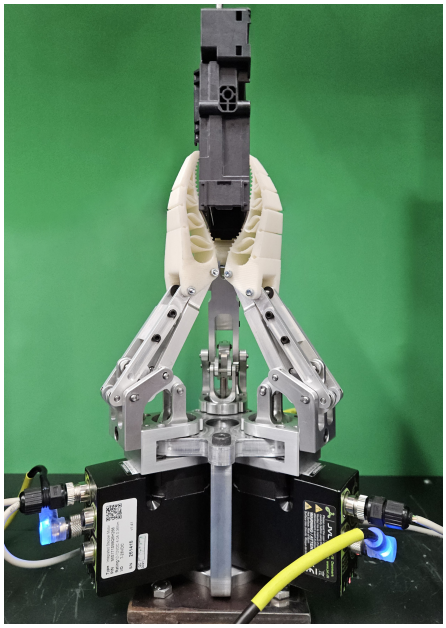
(b)

Figure 3.23: Tensile test - Cube (a) Test setup and (b) Tensile test representative curve

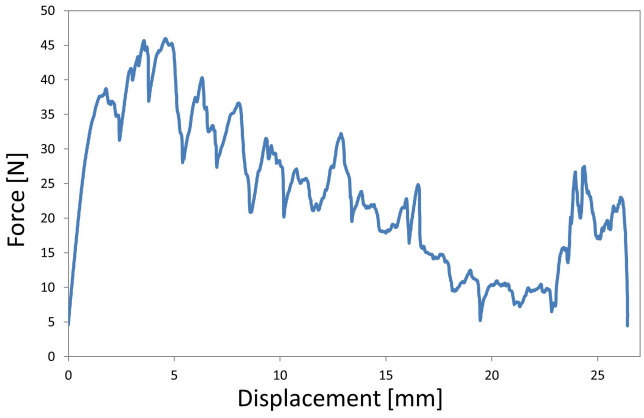
The tensile test of a flat industrial part shown in Fig. 3.24a produced a rapid force build-up, reaching a force of 45 N at approximately 3.5 mm displacement with an initial finger self-adjustment at 1.7 mm of displacement with 39 N of force, as shown in Fig. 3.24b. The initial increase in force indicates a secure grasp of the object, followed by a gradual reduction in force due to continuous slippage of the object. Force fluctuations were observed, demonstrating the dynamic adjustment of the self-adaptable fingers to conform to the object as the object continues to slip and starts to stabilize around 8-10 N after 20 mm displacement. Subsequently, a final force recovery is observed at 23-26 mm, indicating engagement with a different geometric feature of the object as it continued to slip through the grasp.

Peak force repeatability observed during the five trials was 45±5 N (CoV=11.1%).

The relatively lower sustained force during the slippage phase compared to the cube demonstrates the challenge of maintaining friction on flat-surfaced objects. Nevertheless, the gripper demonstrated an effective initial hold before the start of the controlled slippage, confirming its capability to grasp objects with limited geometric features that allow for a secure grasp.



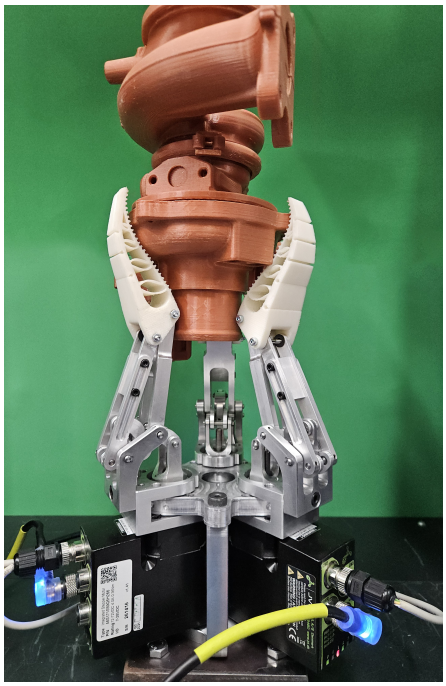
(a)



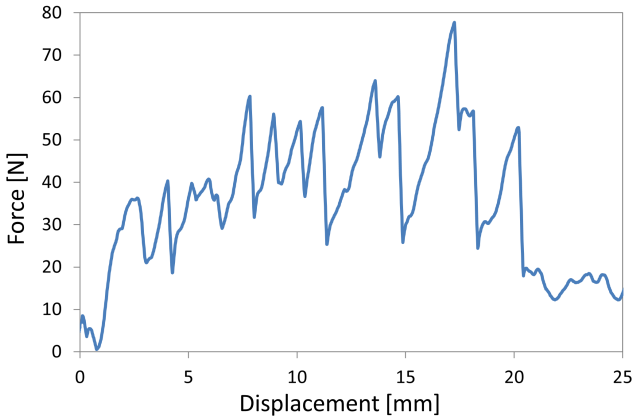
(b)

Figure 3.24: Tensile test - Flat industrial part (a) Test setup and (b) Tensile test representative curve

Furthermore, a pull-out test was carried out using a prototype of a complex industrial part. The experimental setup is shown in Fig. 3.25a. The complex shape demonstrated an intricate force profile with multiple significant retention force peaks across the displacement range, as shown in Fig. 3.25b.



(a)



(b)

Figure 3.25: Tensile test - Complex industrial part (a) Experimental setup and (b) Representative curve

An initial force of 7.8 N, followed by a variation in the force as the displacement increases, indicates the adjustment of the fingers to the shape of the object. Subsequently, a significant increase in force of approximately 36 N at a displacement of around 2.5 mm is observed, followed by a series of increasingly higher force peaks: approximately 60 N at 7.8 mm, multiple intermediate force peaks in the 40-55 N range between 8-12 mm, a significant peak of 63 N at 14.5 mm, culminating in the maximum retention force of approximately 77 N observed around 17 mm displacement. However, as the axial pull-up force by the UTM exceeds force peaks with a slight decrease in the force of 56 N and 53 N occurs before the force transitions to a steady-state of 12-20 N beyond 20 mm displacement. The response pattern demonstrates a dynamic grip-slip-regrip mechanism as the object is pulled through the grasp. The individual peaks represent the redistribution of grip forces across the self-adaptable fingers or engagement with a different geometric feature of the object. The significant force peak at 77 N highlights the grippers' ability to adapt to complex geometries and exploit the geometric features of complex industrial components. During repeated tests, a maximum retention force of 77±10 N (CoV=13.0%) was observed, with multiple intermediate peaks of 40-60 N range showing similar repeatability.

The tensile testing results demonstrate the gripper's object retention capabilities across varied geometries. Maximum retention forces of 45 N (flat surfaces) to 77 N (complex geometry) were achieved. The sustained forces during slippage (15- 25N) translate to practical load capacities of 1.5-2.5kg, with peak retention supporting loads up to 7.7 kg. These results confirm the suitability of the proposed gripper for diverse industrial components.

3.8.2 Grasping assessment with gripper integrated with a 6DOF industrial robot

In order to validate the gripper's performance, we performed grasping experiments by integrating the gripper with a 6DOF industrial robot arm (Comau NS-16) having a payload capacity of 16 kg, as shown in Fig. 3.26.

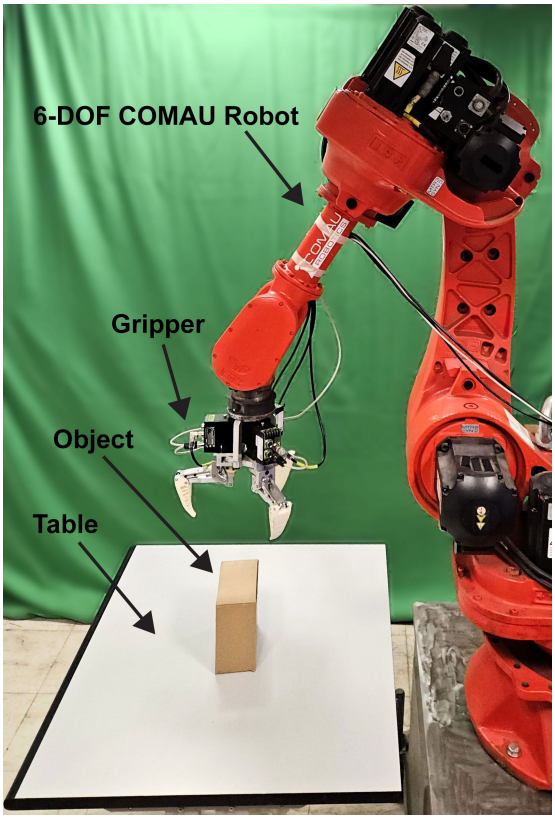


Figure 3.26: Experimental setup with gripper integrated with a 6DOF Comau Industrial Robot

Test objects representing varying weights, geometries, and surface properties were selected to test the gripper's performance. Table 3.2 presents the tested objects, their weights, and dimensions. Each object was positioned in a predefined location, and the robot was programmed to approach the object, securely grasp, lift, hold the object for 20 seconds in the elevated position, and finally return the object to its location on the table. To reliably assess grasping performance, the test was repeated five times for each object. Moreover, to validate the gripper's versatility, different finger configurations were used for testing. The default equiangular configuration was applied to the car fan blower, orange, and clutch pressure plate. The two-finger parallel grasp was employed for the car power control unit, bearing, and power supply box. Lastly, the three-finger envelope grasp was used for the car tail light, pneumatic cylinder, and suspension upper control arm, as shown in Fig. 4.34.

Table 3.2: Properties of grasped objects with gripper mounted on Comau robot

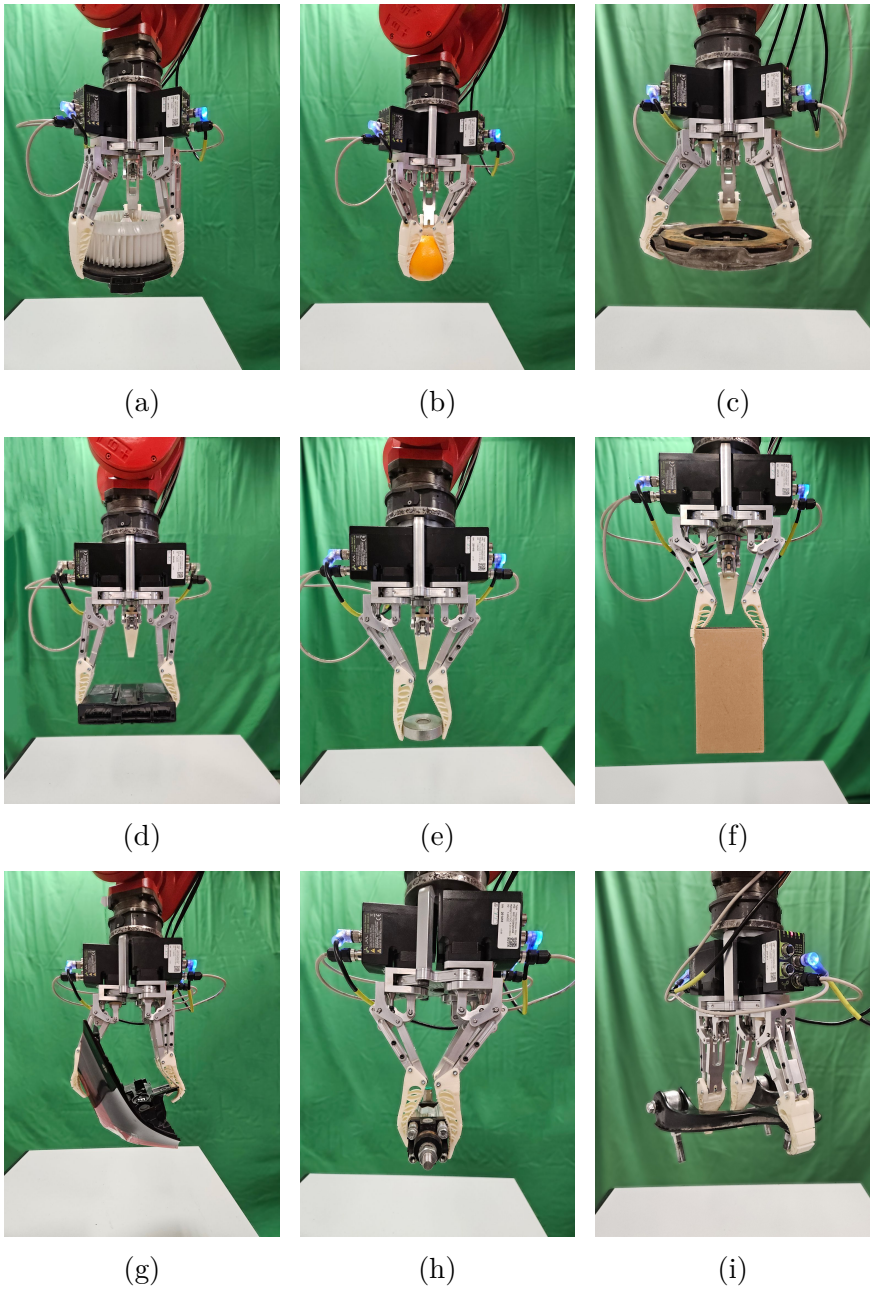
Object	Weight (g)	Dimensions (mm)
Car Fan Blower	1633	180 (D), 140 (H)
Orange	200	76 (D)
Clutch Pressure Plate	3315	216 (D), 40 (H)
Car Power Control Unit	260	160×138×35 (L×W×H)
Bearing	400	60 (D)
Power Supply Box	1400	163×162×85 (L×W×H)
Car Tail Light	1550	Complex
Pneumatic Cylinder	600	179×45×45 (L×W×H)
Suspension Upper Arm	2400	Complex

The grasp tests utilizing the default three-finger configuration with 120° spacing demonstrated good versatility across the tested objects. The car fan blower (1533 g) was grasped reliably, with the fingers effectively conforming to its circular base geometry. The orange (200 g) was handled with consistent success because the self-adaptable fingers conformed to the shape of the orange without causing any damage or deformation, demonstrating the gripper's ability to handle fragile objects. Grasping tests of the clutch pressure plate (3315 g) demonstrated the gripper's capability in the upper range of its capacity. Although the initial grasp of the pressure plate was secure, as visible in Fig. 5.9c, partial slippage was observed during the 20 second holding phase in two of the five test cycles. This indicates 3-3.5 kg as the practical upper limit for grasping with the grippers' default equiangular configuration, aligning well with the retention force measurements from the tensile tests.

Testing using the two-finger parallel configuration was more effective for grasping objects with parallel surfaces. Both the car power control unit (260 g) and the bearing (400 g) were grasped successfully in all the trials, with the adaptable fingers conforming to the surface edges and maintaining a secure grasp. The power supply box (1400 g) was also successfully grasped, and a stable grasp was maintained in all trials, demonstrating the gripper's ability to hold moderately heavy prismatic objects.

The three-finger envelope configuration is particularly effective for complex objects. The car tail light (1550 g) was successfully grasped in the trials, with the fingers conforming to its complex geometric features. The pneumatic cylinder (600 g) was also reliably manipulated in the envelope configuration with a stable hold. The suspension upper control arm (2400 g) presented moderate difficulty because of its weight and complex shape. The gripper was able to securely hold the suspension control arm during the holding phase in four of the five tests, with

minor slippage in one trial.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3.27: Grasping assessment with gripper grasping various objects: (a) Car fan blower, (b) Orange, (c) Clutch pressure plate, (d) Car Power control unit, (e) Bearing, (f) Power supply box, (g) Car tail light,(h) Pneumatic cylinder and (i) Suspension upper arm

The experimental results demonstrate that the practical weight limit for secure and reliable grasping was closely aligned with the sustained force values observed in the tensile tests. Objects weighing up to approximately 2.5 kg were grasped reliably, while objects approaching or exceeding 3 kg showed minor occasional slippage. Although force–torque sensing was not available, quantitative performance metrics were collected for repeated tests. The default equiangular configuration achieved 100% grasp success for objects weighing up to 2.5 kg but demonstrated reduced performance with the clutch plate weighing 3.3 kg, experiencing partial slippage in 2 of 5 trials. In contrast, the two-finger parallel configuration demonstrated 100% success across all tested objects. The three-finger envelope configuration achieved around 95% overall success, with partial slippage occurring in 1 of 5 trials when handling the suspension upper arm. Operational consistency was observed through consistent gripper closing times of 2.0 ± 0.2 seconds across all gripper configurations. All successful grasps maintained stability throughout the full 20 second hold period. Notably, no visible damage was observed to delicate objects (orange) in any of the tests.

Objects with complex geometry and sharp curvy edges presented challenges due to limited contact area, indicating the possible benefits of application-specific soft finger designs. Grasp performance was significantly influenced by the optimal gripper configuration, as each configuration demonstrated specific advantages for different object classes. The equiangular configuration was optimal for grasping objects that require a balanced force distribution. The two-finger parallel configuration was better for grasping prismatic objects, and the three-finger envelope grasp demonstrated better grasping performance for irregular objects.

The test results validate the practical use for industrial powertrain applications involving components weighing up to 2.5 kg with high reliability and up to 3.3 kg with moderate reliability. The modular and reconfigurable design of the gripper proved particularly valuable during testing, enabling rapid reconfiguration to efficiently grasp the tested objects. For industrial deployment based on our results, we recommend a safety factor of 2.0, limiting operational loads to 1.25 kg for continuous operation and a maximum recommended payload of 2.5 kg for low-speed operations. These factors account for the observed transition from peak retention force to sustained force during controlled slippage, ensuring reliable operation in industrial environments. Typical automotive components in our test range fall within safe operational limits when using the appropriate gripper configuration. The 3315 g clutch plate represents the upper limit requiring an envelope configuration for optimal safety. The primary failure mode was gradual slippage and no drastic failure modes were observed during testing.

3.9 Chapter Summary

This chapter detailed the kinematic analysis, actuation, and control of the universal gripper. An analytical model of the Chebyshev and four-bar linkage was developed, revealing nearly linear fingertip movements and considerable force enhancement. Independent actuation and sensorless control capabilities were enabled through the use of integrated stepper motors. Tests conducted with a Universal Testing Machine and a COMAU 6DOF robot demonstrated the gripper's versatility and industrial applicability. These results provide a foundational benchmark for performance, which will serve as a reference for evaluating future design improvements in Chapter 4.

Chapter 4

Gripper Enhancement

4.1 Introduction

The original universal gripper design introduced in Chapter 3 demonstrated the feasibility of combining rigid linkages with compliant fingers to achieve adaptable industrial manipulation. While effective in handling a range of objects, limitations were observed in grasp stability, adaptability to varying geometries, and load capacity, along with rigid link interference due to unwanted interaction of complex-shaped objects with the inner part of the linkage. Addressing these challenges required systematic analysis of the soft finger and exploration of enhanced linkage configurations.

This chapter presents enhancements to our universal gripper. The redesigned self-adaptable fingers, validated through finite element analysis and experimental testing, achieved an improvement in the wrapping capability while increasing the maximum retention force. The transformation from a four-bar to six-bar linkage mechanism with an integrated compliant pad eliminated rigid link interference issues, enabling articulation angles up to δ_1 = -22° and δ_2 = -15° while maintaining 95.3% force transmission efficiency. These enhancements expanded the operational capacity from 3 kg to 7.5 kg maximum payload and increased the grasping diameter by 23% to 270 mm. Comprehensive validation through pull-out tests and robotic grasping experiments with objects ranging from 247 g to 7500 g demonstrated the performance of the enhanced industrial gripper with 100% success rate. The enhanced gripper combines soft robotic adaptability with industrial reliability through passive compliance, enabling flexible manufacturing.

4.2 Design Enhancement Methodology

4.2.1 Performance Gap Analysis

Comprehensive analysis of the initial gripper's performance in industrial trials identified specific enhancement requirements:

1. **Force limitations:** While the original design achieved 77 N peak pull-out force, automotive assembly applications required forces exceeding 150 N for secure manipulation of heavy components during dynamic operations.

2. **Geometric Constraints:** The four-bar parallelogram linkage experienced mechanical interference, limiting the gripper's ability to fully envelop large objects. This interaction not only compromised the gripper's structural integrity but also caused damage to the objects during the handling process.

3. **Finger Adaptation:** Initial finger designs demonstrated adequate compliance but suboptimal wrapping behavior for irregular geometries, resulting in reduced contact area and force distribution.

4. **Material Durability:** Extended testing revealed wear patterns in TPU fingers, particularly at high-stress concentration points, necessitating design modifications for improved longevity.

4.2.2 Design Enhancement Strategy

The enhancement strategy employed parallel development paths:

1. **Quantitative Finger Evaluation:** Development of the deflection coefficient metric to systematically evaluate and optimize finger wrapping behavior.

2. **Develop enhanced architecture:** Introduce a six-bar linkage with compliant pad integration to improve adaptability.

3. **Validation Protocol:** Establishment of standardized testing procedures for comparative evaluation of self-adaptable fingers and the universal gripper.

4.3 AM-ed Material characterization

The Flexfill TPU 98A thermoplastic rubber material was characterized according to the ASTM D412 normative, which defines the procedures to be used to

evaluate the tensile properties of vulcanized thermoset rubber and thermoplastic elastomers. The specimens shown in Figure 4.1 were AM-ed for the characterization experiment.

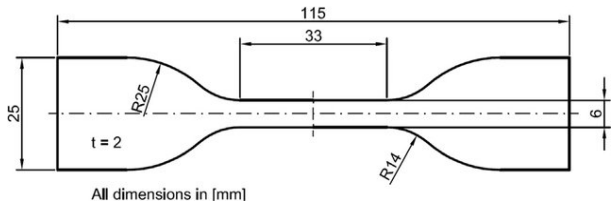


Figure 4.1: Tensile test specimen with dimensions

Five tensile test repetitions were carried out, a representative result curve is shown in Figure 4.2.

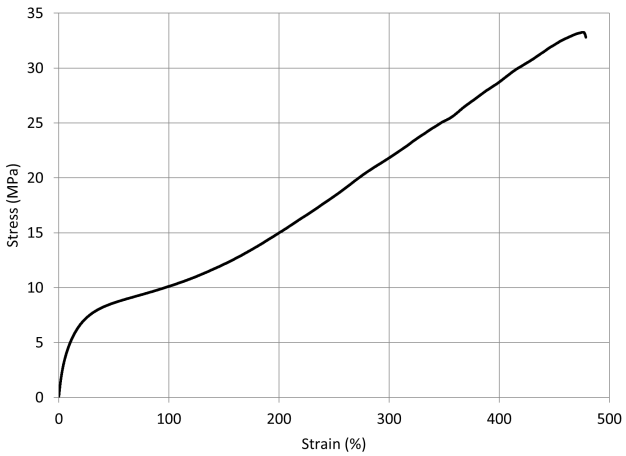


Figure 4.2: Tensile test result, representative curve

The averaged tensile value were used as input for the ANSYS curve fitting tool to estimate the hyperelastic material law constants. To describe the material behavior a 5 parameters Mooney-Rivlin model Bergström (2015) was utilized. The five-parameter Mooney–Rivlin model can be expressed in its general form as shown in Equation (4.1).

$$W = C_{10}(I_1 - 3) + C_{01}(I_2 - 3) + C_{11}(I_1 - 3)(I_2 - 3) + C_{20}(I_1 - 3)^2 + C_{02}(I_2 - 3)^2 + (1/d)(J - 1)^2 \tag{4.1}$$

In this formulation, W denotes the strain energy function, while $I_1$and I_2 represent the first and second invariants of the deviatoric strain tensor, respectively. The coefficients C_{ij} are material parameters that describe the material's response to deviatoric deformation, and d is a dimensionless parameter associated with the material's incompressibility. The constant values are reported in Table 4.1.

Table 4.1: Material constants for Mooney–Rivlin five-parameter model

Parameter (units)	C_{10} **(MPa)**	C_{01} **(MPa)**	C_{11} **(MPa)**	C_{20} **(MPa)**	C_{02} **(MPa)**	d **(-)**
Values	-26.25	38.14	-0.87	0.13	9.19	0

The obtained material card was validated performing in ANSYS the virtual testing of the tensile test using 3-D 10-Node tetrahedral structural solid element and mesh size of 0.8 mm. The force-displacement result is shown in Figure 4.3.

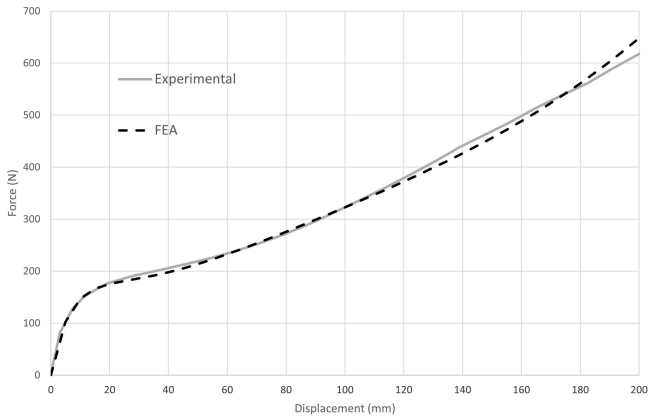


Figure 4.3: Material card validation, comparison between FEA and experimental results

4.4 Self-Adaptable Finger Design

The design of the self-adaptable finger presented in Chapter 3, had a triangular structure with elastic bridges inside the contour to attain the wrapping of the object, as shown in Figure 4.4. These fingers were able to self-adapt substantially to the shape of the object and perform a secure grasp; however, it was observed that when the force in the pull-out tests exceeded 70 N, object slippage occurred.

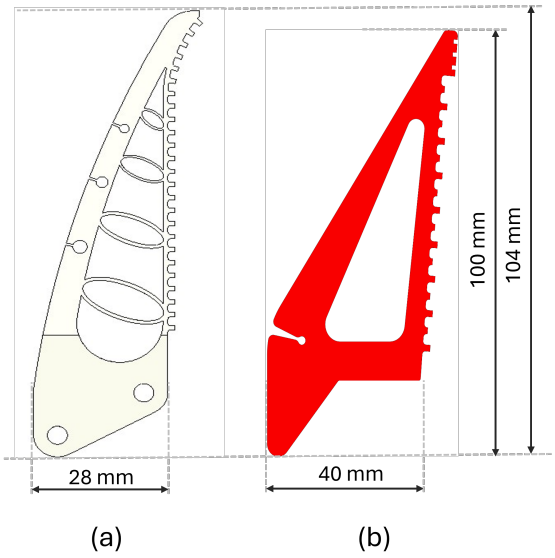


Figure 4.4: (a) Finger 1, Self-adaptable finger With internal bridges, (b) Finger 2, Redesigned self-adaptable finger With no internal structure

Because the universal industrial gripper is intended to handle parts with varied rigidity and dimensions, high gripping force is essential along with self-adaptability. The redesign of the soft finger was investigated with the aim of achieving a higher gripping force while maintaining the characteristic benefits of self-adaptability. This would allow the gripper to securely grasp heavier rigid and soft workpieces, even at a high acceleration rate. The structural optimization method Suder *et al.* (2021) was used as the starting point for redesigning the self-adaptable finger. In this approach, a mathematical method to evaluate the ability of a passive adaptable soft finger to wrap around an object being grasped was introduced. Various finger designs with different components and internal structures were evaluated using FEA and experimental tests, and it was concluded that the finger with no internal structure had superior adaptability. Moreover, it was observed that if the thickness of the contact side and the opposite side of the finger were increased, a higher force was required to achieve a certain deflection.

Initially, a compliant finger with a triangular contour with variable thickness and no internal elastic bridges was designed, and the design was refined iteratively. The designed iterations were first evaluated on the basis of the approach defined in Suder *et al.* (2021) and then by comparison of the force–displacement curves, with the intent of achieving better results than the previous design shown in Figure 4.4(a). The experimental tests were performed using a specifically designed rig allowing accurate evaluation and controlled motion. The achieved self-adaptable

soft finger design is shown in Figure 4.4(b). A different pattern of friction teeth was also incorporated into the finger design along with a step at the fingertip to enhance stability and prevent the slippage of objects.

Testing for finger evaluation was performed using the experimental setup shown in Figure 3.3.

4.4.1 Finger Wrapping Assessment

The methodology defined in Suder *et al.* (2021) is based on the deflection coefficient, which is calculated while the finger is being pressed by an increasing load. To perform this test, the finger must be fixed at the mounting end while the positions of certain points changing under load are noted. The two positions to be observed are the contact point where the object is pressed into the finger and the fingertip. The finger is pressed by the object in steps of 1 millimeter, and at each step, the perpendicular distance of the points from their positions in the unloaded state is measured. Figure 4.5 shows the distances measured, where Dc is the deflection at the edge of the inward face of the finger where it comes into contact with the object and Dt is the deflection measured at the fingertip.

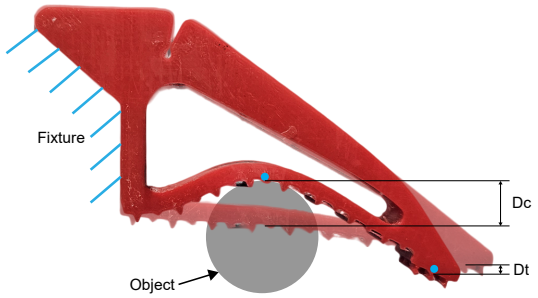


Figure 4.5: Distance measurement with finger under load

Using the distances measured, the finger deflection coefficient can be calculated. The deflection coefficient δ is defined as:

$$\delta = \frac{\sum_{i=1}^{n} \frac{Dt_i}{Dc_i}}{n} \tag{4.2}$$

where Dt and Dc are the distances measured in the ith compression step. In a comparison of different designs of soft FinRay fingers, the finger with the smallest value of δ is considered to have the best wrapping characteristic.

The fingers were preliminary virtual tested in ANSYS replicating the setup described in Suder *et al.* (2021) using the values obtained for finger 1 as reference.

The fingers developments aimed to minimize the δ coefficients. This iterative optimization led to the shape of finger 2. Virtual testing full-field displacement results are reported in Figure 4.6. To evaluate the coefficient δ displacement results were evaluated each millimeter and the values Dt and Dc were assessed using probes at the positions displayed in Figure 4.5. The values of Dt and Dc are shown in Table 4.2.

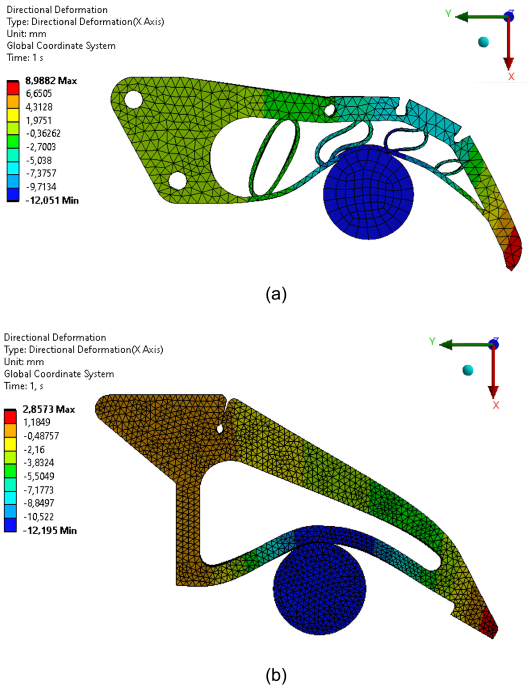


Figure 4.6: FEA results, x-axis displacement plot for both fingers: a) Finger 1; b) Finger 2, redesigned self-adaptable finger

To carry out the same characterization experimentally, assessing Dt and Dc to calculate the coefficient δ of both compliant fingers, the setup shown in Figure 3.3 was utilized with the UTM applying compression force. The test was performed with a total compression of 12 mm, making n equal to 12. To precisely measure the fingertip and object contact points perpendicular to the contact surface, the Tracker video analysis tool (OSP) was used along with some measuring tools needed to set the scale of the video. The test setup for the new finger design is shown in Figure 4.7. R is used as a reference to measure the distance of the contact point C as C remains stationary during the test. Even though the direction of R is opposite to that of C, they will have the same travel distance. F shows the direction of the fingertip after the consecutive steps. The values of Dt and Dc attained with this experiment give the deflection coefficient of the

respective fingers and are shown in Table 4.2. The value of δ for Finger 2 is much less than that for Finger 1; hence, it has a better wrapping property.

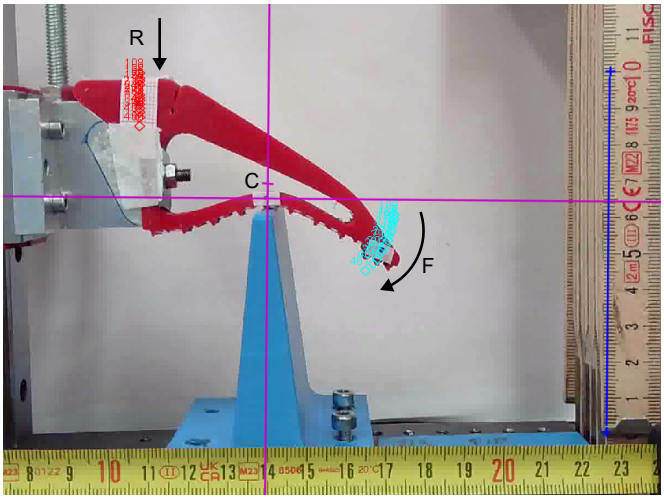


Figure 4.7: Finger deflection coefficient video analysis setup

Table 4.2: Deflection coefficient for Finger 1 and Finger 2

Finger	**Structure**	**Compression Depth (mm)**	**Deflection Coefficient (FEA)**	**Deflection Coefficient (Experimental)**
Finger 1 (white) Fig. 4.4(a)	Internal bridges	12	0.45	0.39
Finger 2 (red) Fig. 4.4(b)	No internal structure	12	0.12	0.1

4.4.2 Quasi-Static Force-Displacement Test

The grasping solution aimed required good conformity along with a higher load-carrying capacity so that the gripper could handle various fragile and rigid objects. To evaluate and quantify whether the redesigned finger (finger 2) geometry would perform better at grasping than the previous iteration of the self-adaptable finger (finger 1), the force and displacement behavior of both fingers were analyzed.

The force–displacement curve allows the assessment of finger deflection under load and force generation. These quasi-static tests were performed using a UTM by slow application of force up to 45 N with a preload value of 5N, ensuring proper contact, finger stability, and elimination of slack. A 45 N limit was set as the finger will subjected to lesser force when mounted on the gripper. The testXpert software was used to control the UTM and continuously collect force and displacement data during the test. To assess the grasping performance of both fingers at different positions, the stop block was placed at three different points with the fingers having inward face down. This setup allows the application of force at the one-third, two-thirds, and tip of the fingers. The arrangement is shown in Figure 4.8.

Figure 4.8: Stop block positions for Quasi-Static Force-Displacement Testing

An instant of the quasi-static test for both finger 1 and finger 2 with the object at the two-third position of the fingers is shown in Figure 4.9. The figure corroborates that the fingers are deforming at the desired point when an external force is applied. The force-displacement curves of finger 1 and finger 2 are shown in Figure 4.10. At the two-third position, the curve of finger 1 illustrates that the deformation is complex and can vary based on changing loading conditions. Initially, the force increases gradually with displacement. However, as the displacement reaches 16 mm, a decline in force combined with sudden drops and peaks is observed, indicating the initial compression of the internal structure of finger 1 as the FinRay effect occurs. Furthermore, a less abrupt response with increasing force is observed until the displacement reaches 30 mm, and then drops and peaks are observed when the internal elastic beams of the finger are fully compressed. This sudden series of drops and peaks occurs because of the slippage of the object as the finger adjusts to the object during deformation. While a higher displacement indicates more adaptability, finger 1 demonstrates an over-compliant behavior, which signifies a lower grasping force and loss of stability during grasping. The force-displacement curve of finger 2 with no cross beams at the two-third position exhibits low initial stiffness, implying that it can

deform with a small application of force from the object. This characteristic is important to ensure adaptability without exerting too much force, which might damage the object being grasped. As the force on the finger increases, relatively smooth progressive stiffening behavior is observed, which is necessary for slippage prevention and grasp stability while conforming to the object shape. Finger 1 reaches a force of 45 N with approximately 15 mm of displacement which signifies fingers gradual deformation with increasing force.

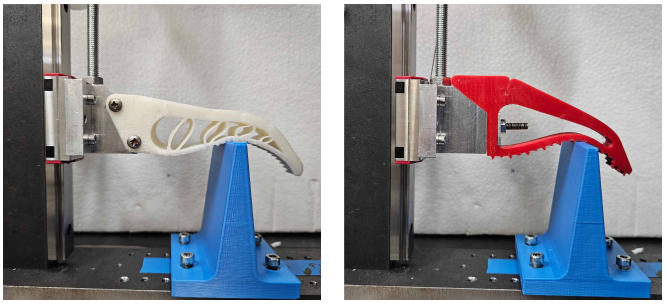


Figure 4.9: Force-displacement test at two-thirds (2/3) position of finger 1 (left) and finger 2 (right)

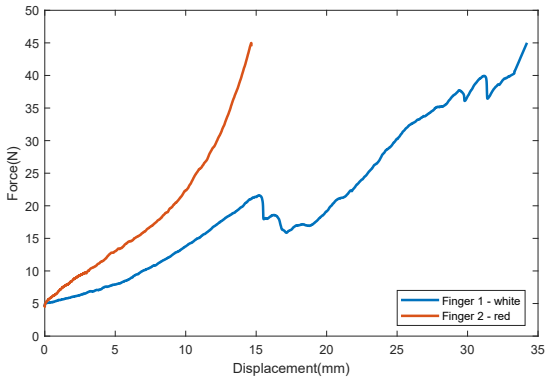


Figure 4.10: Force-displacement curve at finger 2/3 position

An image of the test performed with the stop block placed at the one-third position of the fingers is shown in Figure 4.11 and the corresponding force-displacement curve is presented in Figure 4.12. This position of the fingers is closer to the fixture and should demonstrate a stiffer response compared to the force-displacement behavior at the two-third position. The force-displacement curve shows that finger 1 exhibits more compliance with lower application of force but shows a slight declination in the curve around 8 mm of displacement as the finger starts to conform then around 16 mm of displacement sudden drops

and peaks are observed in the curve this abrupt behavior occurs due to slippage of fingers contact teeth contacting the stop block and also due to compression of the internal finger bridges. These sudden changes in the curve may result in a potential loss of grasp. In contrast, the force-displacement curve of finger 2 shows uniform compliance throughout with no sudden changes in stiffness or localized peaks.

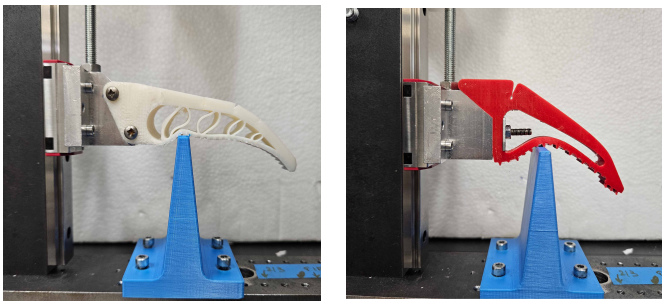


Figure 4.11: Force-displacement test at one-third (1/3) position of finger 1 (left) and finger 2 (right)

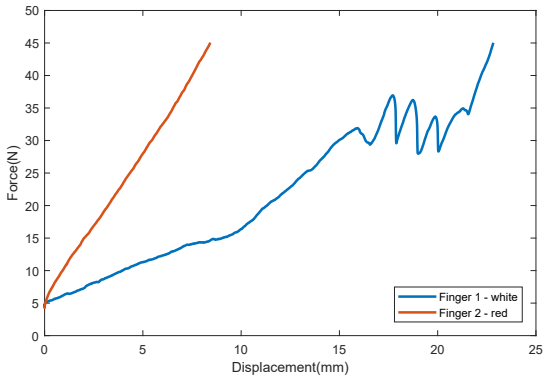


Figure 4.12: Force-displacement curve at finger 1/3 position

Moreover, the fingers were tested by placing a stop block at their tips. A captured image of the test is shown in Figure 4.13 and the corresponding force-displacement graph is shown in Figure 4.14. The force-displacement curve of finger 1 shows that after reaching a force of 21 N at the fingertip, a drop in force is observed; this dip is due to finger deformation. Following the drop in force, finger 1 behaves more rigidly after the initial deformation, and a gradual increase in force is observed until slippage occurs at approximately 37 N of force followed by a series of slips identified by the peaks in the curve. Contrarily for finger 2, the force and displacement increase gradually without major disruptions as finger 2

retains its structural integrity until the subjected force reaches 50 N. The Finger 1 force-displacement curve indicates a higher degree of flexibility, and after the initial dip in force, it offers a higher rigidity until object slippage occurs. However, finger 1 is unable to reach a higher peak force as deformation occurs at 21 N, whereas finger 2 shows more resistance to deformation and maintains stability when subjected to higher loads until slippage occurs at 50 N. Thus, finger 2 is a more viable option for tasks that require a strong and stable grip from the fingertip with consistent behavior.

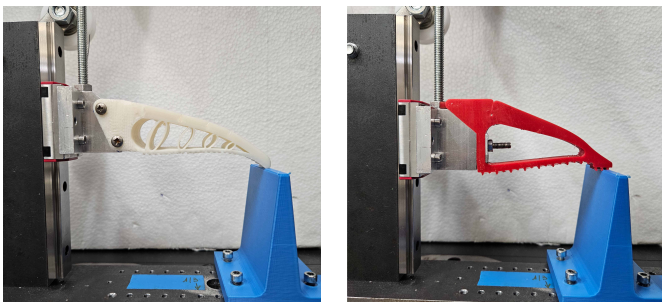


Figure 4.13: Force-displacement test at the fingertip of finger 1 (left) and finger 2 (right)

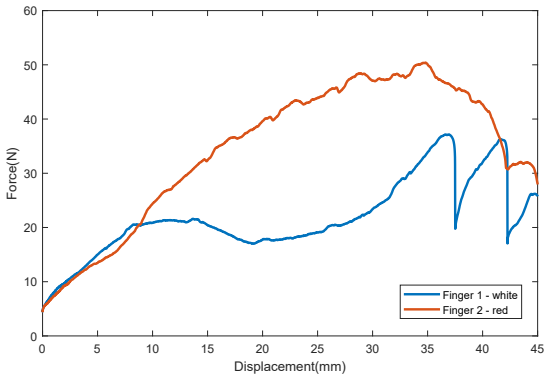


Figure 4.14: Force-displacement curve at fingertip position

The force-displacement tests allowed us to gain valuable insights into the mechanical responses of both fingers. The quantified data attained from the UTM was crucial for improving the design of the self-adaptable finger for the universal gripper. From the results presented above it can be concluded that the redesigned finger provides a more stable and adaptable behavior while having a higher load-handling capacity. Hence, the consistent behavior of finger 2 makes it more versatile as it may be used to handle a broader range of fragile to rigid objects thus making it more suitable for our required industrial application.

4.5 Gripper Linkage Design and Enhancement

The gripper comprises three identical fingers, each of which incorporates two kinematically coupled linkage mechanisms. The Chebyshev lambda linkage translates the screw–nut mechanism's linear motion to a rotary motion, with its output connected to the four-bar linkage, which maintains the position of the low side of the parallelogram where the finger interface exists. This allows the finger to maintain its orientation throughout its motion range.

The Chebyshev lambda linkage is a four-bar mechanism that is usually employed to produce an accurate straight-line motion from rotation input, with proper linkage proportions. However, in this case, is used to convert the linear input motion from the lead screw to the rotational output. The schematic of the Chebyshev lambda linkage with link lengths L_1,L_2,L_3,L_4 = 18, 19, 22.5, 22.5 mm and length ratios of L_1:L_2:L_3:L_4 = 2:1:2.5:2.5.is shown in Figure 4.15.

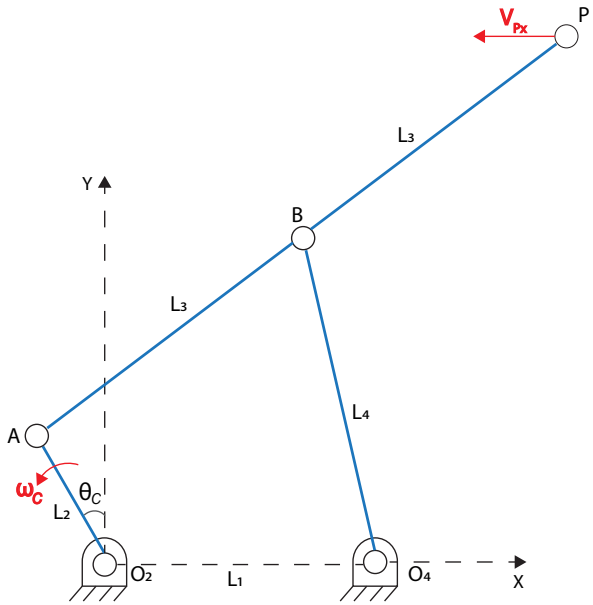


Figure 4.15: Chebyshev lambda linkage geometry

The length ratios of the Chebyshev lambda linkage are critical for the optimal straight-line motion of point P, which is linearly driven by the actuator. The position of output point A for a given crank angle θ_C is:

$$A_x = L_2 cos(\theta_C) \tag{4.3}$$

$$A_y = L_2 sin(\theta_C) \tag{4.4}$$

For the operating range θ_C from 60° to 120°, point P maintains a nearly perfect linear motion. The relationship between the coupler position P and angle θ_C is given by:

$$P_x = 2L_2\left(1 + \frac{\cos(\theta_C)\sqrt{5 - \sin(\theta_C)}}{\sqrt{5 + 4\sin(\theta_C)}}\right) \tag{4.5}$$

$$P_y = 2L_2\left(\frac{\sqrt{5 - \sin(\theta_C)}(2 + \sin(\theta_C))}{\sqrt{5 + 4\sin(\theta_C)}}\right) \tag{4.6}$$

The motion of point P, driven by the lead screw, exhibits a travel range of 12–24 mm, while point P maintains a near-constant position at y = 36.02 mm with a maximum deviation of only 0.0333 mm from the straight-line trajectory. These results correspond to a linearity of 99.997%, confirming that the Chebyshev linkage provides a near-perfect conversion of linear actuator input to consistent horizontal displacement. This motion is essential to ensure consistent force transmission from the actuator along the Chebyshev lambda linkage and subsequently to the coupled linkage.

The second stage of the kinematic mechanism of the gripper employs a parallelogram four-bar linkage that receives rotational input from point A of the Chebyshev linkage and drives the soft finger of the gripper. The link dimensions of the four-bar mechanism are L_5,L_6,L_7,L_8 = 25, 80, 25, 80 mm and the schematic is shown in the Figure 4.16. The parallel configuration with L_5=L_7 and L_6=L_8 and with parallelogram geometry, the output link angle ψ remains equal to the input angle θ_P, ensuring that the output link maintains the same orientation as the input link throughout the motion. The input angle to the four-bar linkage, θ_P, is derived from the output point A of the Chebyshev linkage using the following relation:

$$\theta_P = \arctan(\frac{A_y}{A_x}) \tag{4.7}$$

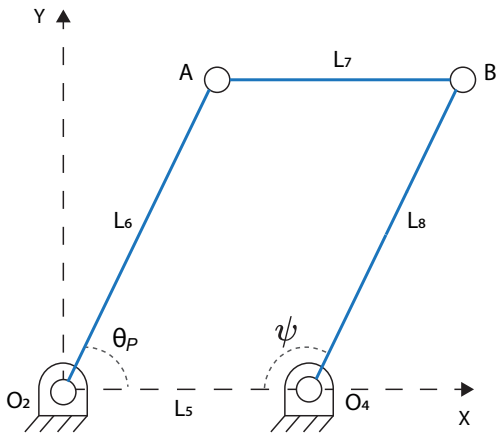


Figure 4.16: Four-bar linkage geometry

The output position B of the parallelogram mechanism is given by:

$$\begin{aligned} B_x &= O_{4x} + L_8 \cos(\psi) \\ B_y &= O_{4y} + L_8 \sin(\psi) \end{aligned} \tag{4.8}$$

The velocity characteristics demonstrate the mechanical advantage of the system. With a linear input velocity of V_P at the lead screw, the angular velocity of the Chebyshev crank is calculated as follows:

$$\omega_C = \frac{V_{Px}}{dP_x/d\theta_C} \tag{4.9}$$

The velocity at point A is given by $V_A = L_2\omega_C$ giving a value of 9.0 mm/s for unit reference velocity. Ultimately, going through the parallelogram linkage yields the velocity at point B, V_B of 71.0 mm/s, and velocity ratio of $V_B/V_P = 5.91$, representing the system's mechanical advantage. The integrated Chebyshev-parallelogram results in a gripper mechanism that achieves highly predictable and precise kinematic behavior with robust motion control and force transmission performance.

Despite excellent overall kinematic performance, the four-bar parallelogram mechanism suffers from a critical limitation: rigid link interference that constrains secure grasping strategies. With the soft finger object interface side of approximately 75 mm, the gripper can use finger grasp to contact objects of various heights, but the rigid link L_8 constrains the ability of the gripper to execute enveloping or multi-point grasps, especially for objects with complex or tall profiles.

In particular, tall objects with a height greater than 50 mm can only be contacted with the soft fingers, thus limiting the gripper to finger grasps. While such grasp configurations are generally sufficient for lifting, they are prone to object tilting or slippage and are generally less secure. As attempting secure enveloping grasps often result in a collision between the object being grasped and the rigid link L_8 of the four-bar parallelogram mechanism. Because of this limitation, complex-shaped objects also pose difficulties while grasping because contact with the soft finger is local due to the object's irregular profile and the inability to utilize the full finger length prevents distributed force application, reducing overall grasp security. Such inability results in precarious grasps because such objects require multi-point contact for stable grasps.

The link interference problem cannot be solved using dimensional optimization because the parallelogram constraint requires specific link length ratios. The rigid L_8 link was converted into a segmented, articulated chain to resolve this issue, introducing additional DOF that allow the mechanism to adapt to object geometry while preserving kinematic performance in the unloaded state.

The enhanced kinematic chain of the finger with two subchains: the Chebyshev lambda linkage and a six-bar linkage with a parallelogram shape are shown in Figure 4.17, with the link numbers identified.

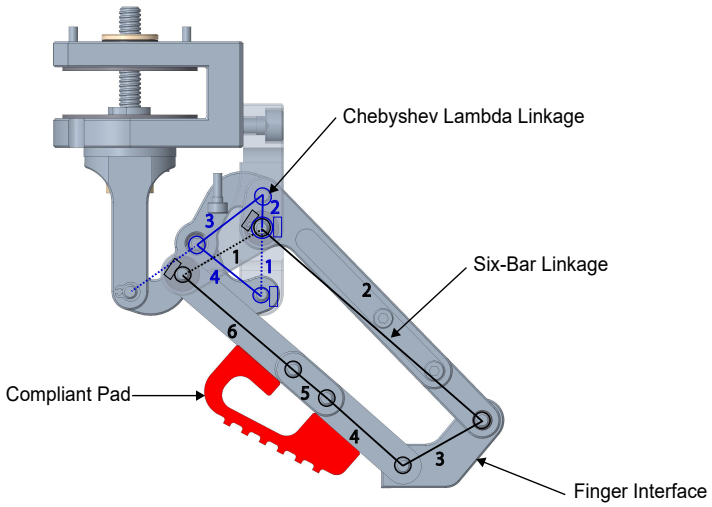


Figure 4.17: Enhanced linkage mechanism showing the two kinematic subchain with integrated compliant TPU pad and finger mount interface

The enhancement divides the link L_8 into three segments: segment c (40.0 mm, 50% of L_8) is the base segment connected to O_4, segment b (12.0 mm, 15% of L_8) is the coupling segment, and segment a (28.4 mm, 35% of L_8) is the terminal segment on the side where the self-adaptable finger is mounted as shown in the

six-bar mechanism schematic in Figure 4.18. This segmentation introduces two additional revolute joints that connect these segments, thus converting the four-bar parallelogram into a six-bar linkage with controlled articulation capability enabling passive adaptation to external contact.

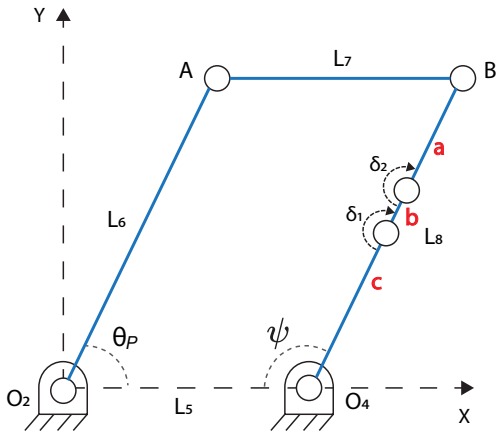


Figure 4.18: Six-bar linkage geometry

Figure 4.19 shows another view of the modified linkage. This modification allows for a more intricate motion, which may also lead to irregular motion patterns. To counter any irregularity and benefit from enhanced adaptability at the inner side of the linkage, in the physical implementation a compliant TPU pad (Flexfill TPU 98A) with one end mounted on link-4 of the six-bar and the other mounted on link-6 of the six-bar linkage was installed. This placement of the compliant pad constraints the movement of the three-bar link comprising of link-4 (segment a of L_8), link-5 (segment b of L_8), and link-6 (segment c of L_8) of the six-bar linkage. Hence, there are two benefits of using the compliant pad. The first is to minimize the movement of the revolute joints of the three-bar link, allowing the six-bar mechanism to function like a four-bar mechanism during actuation. The second benefit is to aid in a more stable grasp, reduce wear and tear, and the absorption of forces caused by the grasped object pushing the inner side of the linkage. While the compliant pad is present in all experimental tests, its deformation is not included in the kinematic analysis presented, which treats the mechanism as a rigid-body system.

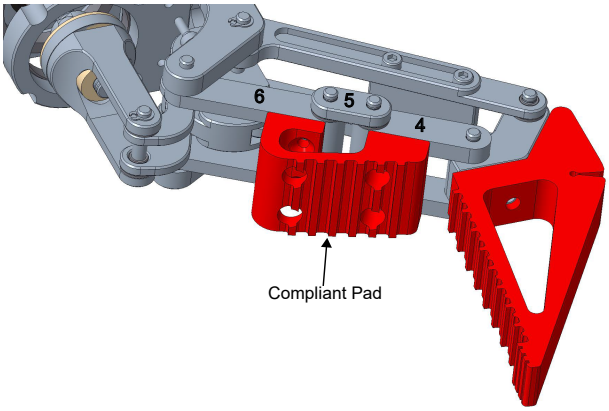


Figure 4.19: Bottom view of the six-bar linkage mechanism illustrating the compliant pad placement between links 4 and 6 for controlled articulation and interference prevention

The segmentation of the link L_8 of the parallelogram mechanism introduces three angular parameters ψ, the angle of segment c with respect to the horizontal at O_4; δ_1, the relative angle between segments c and b; and δ_2, the relative angle between segments b and a. In the unloaded state, the behavior of the parallelogram is preserved as $\delta_1 = \delta_2 = 0$ and $\psi = \theta_P$. A forward kinematic chain is constructed starting from point O_4 to determine the position of point B of the six-bar. The joint between segments c and b is positioned at:

$$\begin{aligned} x_{cb} &= O_{4x} + c.\cos(\psi) \\ y_{cb} &= O_{4y} + c.\sin(\psi) \end{aligned} \tag{4.10}$$

From this point, between segments b and a can be located as:

$$\begin{aligned} x_{ba} &= x_{cb} + b.\cos(\psi + \delta_1) \\ y_{ba} &= y_{cb} + b.\sin(\psi + \delta_1) \end{aligned} \tag{4.11}$$

Finally, the position point B is found using:

$$\begin{aligned} B_x &= x_{ba} + a.\cos(\psi + \delta_1 + \delta_2) \\ B_y &= y_{ba} + a.\sin(\psi + \delta_1 + \delta_2) \end{aligned} \tag{4.12}$$

These equations provide the output configuration in terms of joint angles and segment lengths; however, the loop closure constraint must be satisfied. The position of point B calculated by solving the linkage chain must be equal to that of point B calculated through the segmented path.

$$O_2 + L_6(\cos\theta_P, \sin\theta_P) + L_7(1, 0) = O_4 + \text{path to B} \tag{4.13}$$

The path to B is the method used to obtain point B using 4.12. This loop closure constraint gives two equations (in x and y) for three unknowns ψ, δ_1 and δ_2. The system behavior depends on the loading condition in the unloaded state $\delta_1 = \delta_2 = 0$ and $\psi = \theta_P$. In the loaded state, δ_1 and δ_2 are nonzero and are determined through contact-driven interaction and ψ adjusts to maintain the closure constraint.

A key feature of the six-bar mechanism is its passive self-adaptive behavior during object interaction. When the inner segments of the linkage contact an object during grasping, the articulation angles δ_1 and δ_2 are passively induced. Consequently, the angle ψ deviates from the input angle θ_P, causing the entire finger to reorient. This contact-driven articulation enables the gripper to achieve enveloping grasps through mechanical compliance. In the unloaded state, the enhanced mechanism preserves the original four-bar kinematic performance while preserving path linearity, the velocity ratio range, and repeatability summarized in Table 4.3. This validation demonstrates that the segmentation enhancement adds adaptive capability without compromising the original design's proven kinematic characteristics.

Table 4.3: Comparison of kinematic parameters between four-bar and six-bar mechanisms in unloaded configuration

Parameter	**Four-bar**	**Six-bar (Unloaded)**	**Deviation**
Path linearity at P	99.997%	99.997%	$< 0.001\%$
Velocity ratio (V_B/V_P)	5.91-6.31:1	5.91-6.31:1	0%
Mechanical advantage	Mean: 6.06:1	Mean: 6.06:1	0%
Angular relation	ψ=θ_P	ψ=θ_P	0°
Repeatability	±0.05 mm	±0.05 mm	-

Figure 4.20 shows that the velocity profile of the original four-bar mechanism is completely preserved in the unloaded state of the six-bar design. Point B maintains a characteristic velocity variation from 71 to 76 mm/s across the operating range, whereas Point P exhibits a nearly constant velocity of approximately 12 mm/s. The mechanical advantage varied from 5.91 to 6.31:1, with a mean of 6.06:1. Therefore, the additional revolute joints do not degrade the transmission properties of the original mechanism.

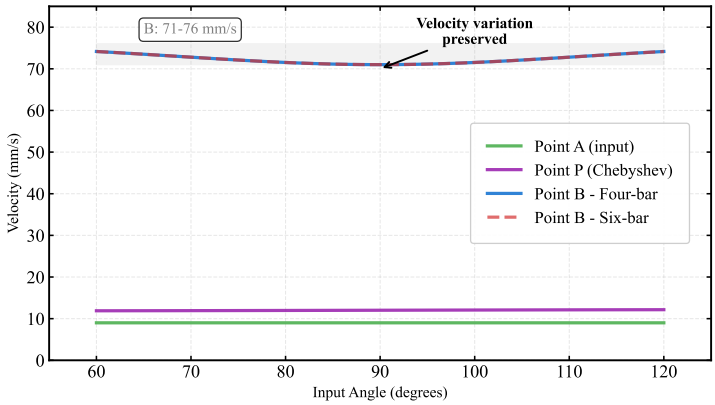


Figure 4.20: Velocity profiles showing complete preservation of the varying velocities throughout the operating range

A right-handed 2D coordinate system was used, with counterclockwise (CCW) rotations considered as positive. Object contact induces clockwise rotation of the segmented links during object interaction while grasping, resulting in negative articulation angles (δ_1, $\delta_2 < 0$). The link angle ψ compensates by increasing from its nominal value θ_P ($\Delta\psi > 0$) to maintain loop closure.

Figure 4.21 shows the progressive articulation of the six-bar mechanism from unloaded (rigid) to maximum articulation. The segmented link demonstrates controlled inward bending with δ_1 = -22° and δ_2 = -15° at maximum, while maintaining kinematic loop closure through base link adjustment $\Delta\psi$ = 16.3°. The positive $\Delta\psi$ compensates for the inward shortening of the segmented side link and enables the mechanism to wrap around the object.

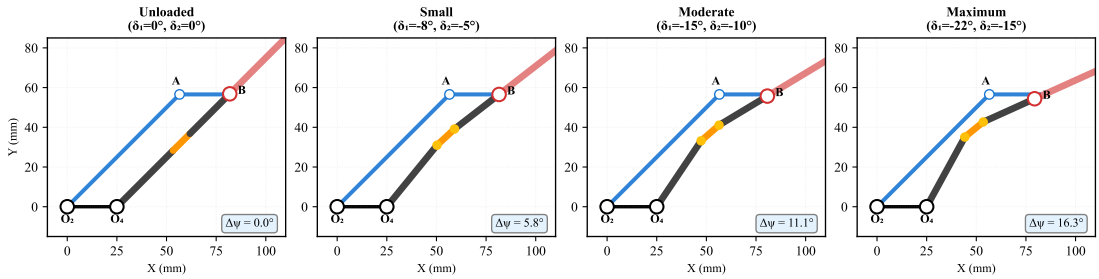


Figure 4.21: Progressive articulation of the six-bar mechanism. The compensatory ψ deviation maintains the kinematic loop closure while enabling adaptive grasping

Table 4.4 summarizes the kinematic configurations and the measured parameters across the operating range. As the articulation increases from the baseline

Table 4.4: Kinematic configurations and performance metrics of the six-bar mechanism across the operating range

Configuration	$\boldsymbol{\delta_1}$ (°)	$\boldsymbol{\delta_2}$ (°)	$\boldsymbol{\Delta\psi}$ (°)	**Total (°)**	**MA Range**	**Eff (%)**
Baseline	0	0	0.0	0.0	5.91–6.31	100.0
Small	−5	−3	3.6	−4.4	5.85–6.23	99.6
Moderate	−12	−8	8.8	−11.2	5.75–6.11	97.9
Large	−18	−12	13.3	−16.7	5.67–6.01	96.4
Maximum	−22	−15	16.3	−20.7	5.61–5.94	95.3

δ_1, δ_2: Articulation angles (negative = inward bending); $\Delta\psi$: Base link deviation from θ_P

Total: $\delta_1 + \delta_2 + \Delta\psi$; MA: Mechanical advantage; Eff: Force transmission efficiency

to the maximum configuration, the mechanical advantage decreases slightly from 5.91–6.31:1 to 5.61–5.94:1. The force transmission efficiency remains above 95%, even at maximum articulation the mechanism maintains 95.3% efficiency. The analysis confirms that the six-bar enhancement successfully addresses the link interference limitation while preserving the original mechanism's key kinematic properties. Negative articulation behavior with positive compensation represents a mechanically intelligent solution that exploits the additional DOF to achieve adaptive envelope grasping.

4.6 Gripper Design - Enhanced

The gripper design with the redesigned finger and linkage mechanism with a compliant pad is presented in Figure 4.22. These enhancements were made while keeping track of the added weight to the gripper with the target of not exceeding 3 kg.

The modifications to the finger design with its straighter profile, compared to the inward-tilted geometry of the original, expanded the maximum grasping diameter in the default orientation from 219 mm in the original design to 270 mm, significantly broadening the range of industrial components that can be handled.

The self-adaptable finger interface allows fingers to be easily mounted and interchanged, allowing them to be substituted or swapped with a finger having a different geometry or stiffness profile. Thus, making the gripper a more versatile platform for diverse industrial applications.

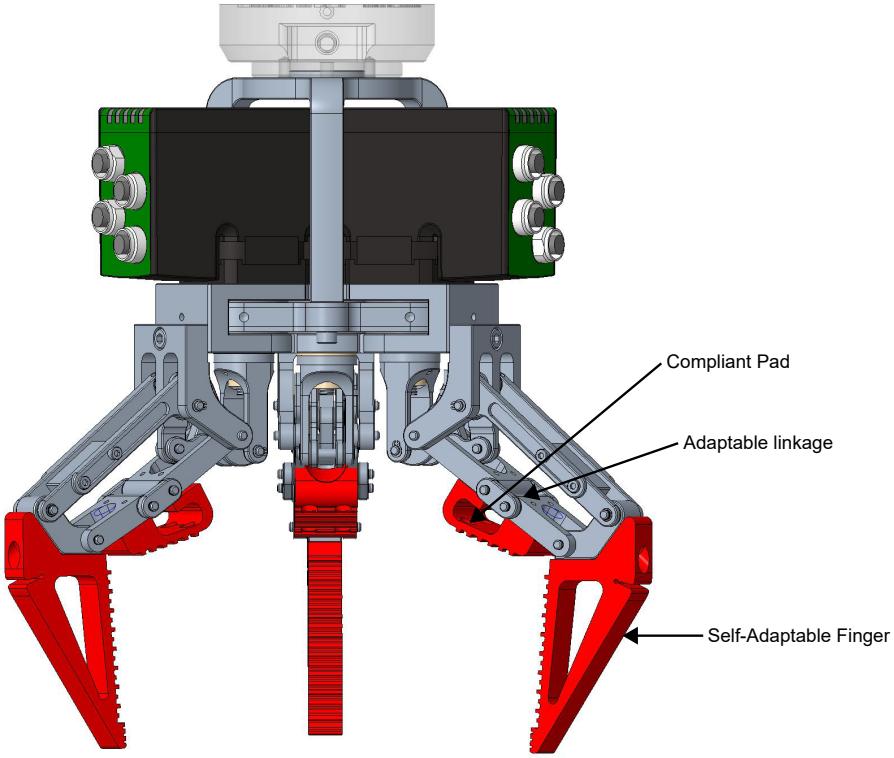


Figure 4.22: Enhanced universal gripper featuring redesigned self-adaptable fingers, six-bar linkage mechanisms with compliant pads, and retained modular architecture for industrial applications

4.7 Experimental Results and Discussion

To thoroughly assess the performance of the universal gripper, pull-out, and grasp tests were conducted. The objective of the experiments was to evaluate the holding capabilities of the universal gripper while grasping a range of objects and materials. The suitability of the gripper was evaluated for various industrial applications with a primary focus on automotive industrial parts.

4.7.1 Pull-Out Testing with Universal Testing Machine

The Universal Gripper was mounted to the UTM to perform tensile tests and collect force-displacement data for evaluation. First, a 3D printed stub shape object of 40x25x35mm was tested with the gripper configured to grip parallelly using two fingers Figure 4.23. The force-displacement curve for the same is shown in Figure 4.24.

The force-displacement profile exhibits variability due to intermittent slip-

pages and readjustments brought about by the finger's adaptability combined with the frictional and stepped features providing a dynamic element to the interaction. Three friction teeth slippages are observed under 8 mm with a maximum load of 80 N afterward the object reaches the step at the tip of the fingers. From this point, a smooth response with increasing load is observed until the force reaches 145 N. Subsequently, the friction teeth within the tip of the fingers start to slip and after 16 mm of displacement, the object starts to slip from the grip.

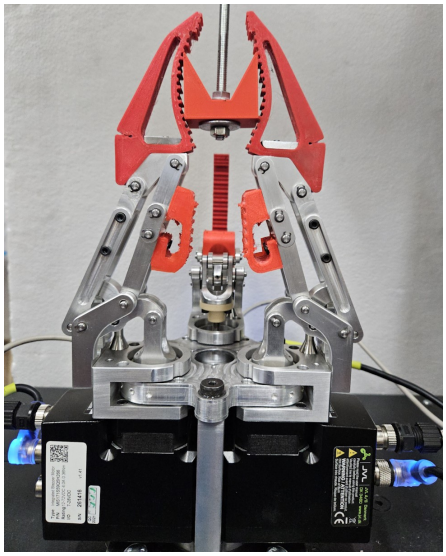


Figure 4.23: Stub shape pull-out test with two fingers parallel grasp

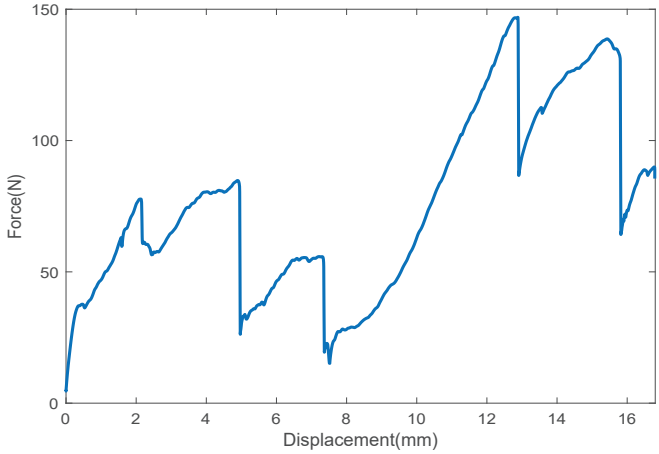


Figure 4.24: Representative Force-Displacement curve of two-fingers parallel grasp

Further, a 3D printed cube of 40 mm was used to test the three-finger grasp as shown in Figure 4.25. The corresponding result revealed a multifaceted mechanical behavior as shown in Figure 4.26. Slippages of the friction teeth are observed following force undershoot, but the force increases after the self-adaptable finger readapts to a maximum force of approximately 176 N. Variations in force occur when the gripper's teeth lose traction due to slippage across the cube's surface. After 15 mm displacement, the object starts to slip and consequently, a decremented force of about 20 N is observed.

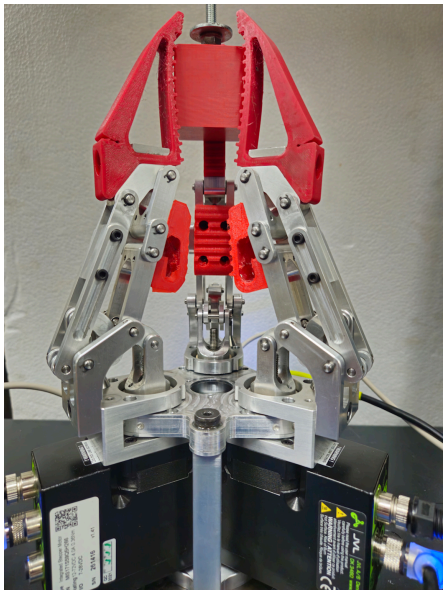


Figure 4.25: Pull-out test using 3D printed cube with three-finger grasp

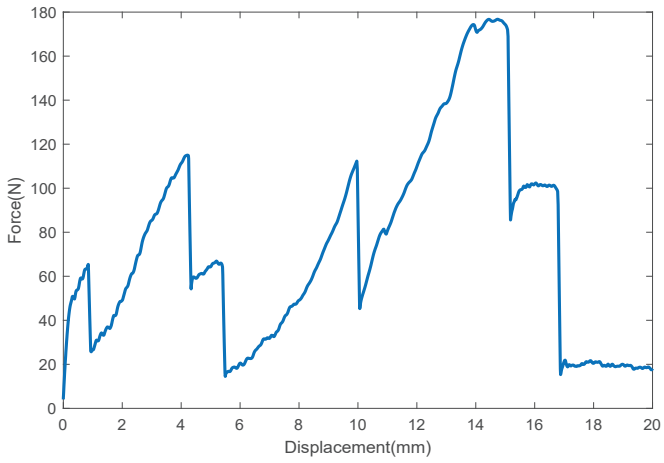


Figure 4.26: Representative Force-Displacement curve of three-finger cube grasp

The pull-out test on a 3D printed complex automotive object was performed as shown in Figure 4.27 and the resulting force-displacement graph is visible in Figure 4.28. At the initial contact, there is a significant increase in force, signifying secure engagement of the gripper with the object. Subsequently, the force starts to increase gradually until a notable drop is seen at 3.8 mm of displacement. At this stage, the fingers adapt to the geometry of the object, and accordingly, a considerable increase in force is observed, reaching a force of 156 N at a displacement of 9 mm, followed by force fluctuations due to fingers readjusting to the complex shape of the object. After a displacement of 11.7 mm, a substantial increase in force was observed, reaching a peak of 200 N at 15.8 mm of displacement. Although intermittent force fluctuations were observed, the upward trend in force signifies a firm grasp of the complex-shaped object.

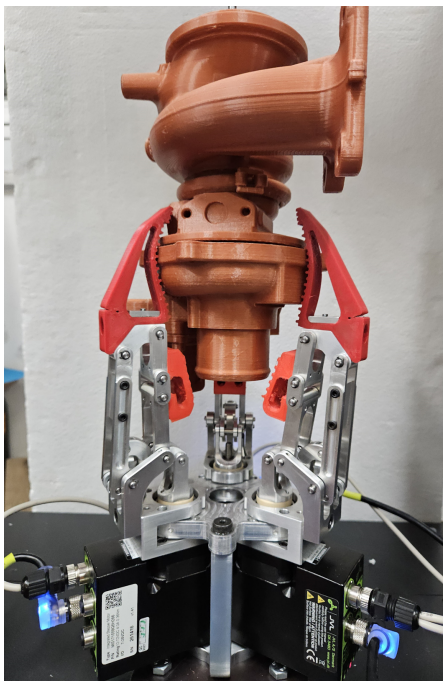


Figure 4.27: Pull-out test using 3D printed complex automotive part with three-finger grasp

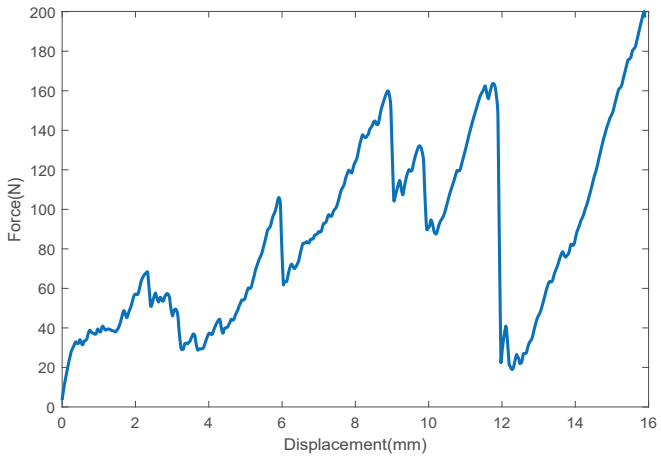


Figure 4.28: Representative Force-Displacement curve of complex automotive part pull-out test

Further, a tensile test with a car window switch panel was performed as shown in Figure 4.29 with the force-displacement response presented in Figure 4.30. Initially, a force of 38.8 N at 1.37 mm of displacement, implying a secure engagement, is seen. Afterward, a substantial increase in force to 100 N at 6.4 mm displacement, indicating a secure hold, is observed. This peak is followed by a drop to 13.9 N at 9.8 mm of displacement as the gripper readjusts. Later, a force of 130 N is reached at 14.5 mm of displacement, followed by force variations as the gripper adapts to the complex shape of the panel. A maximum momentary force peak of 173 N is observed at 33.5 mm of displacement as the object is held by the fingertip step. Subsequently, a decreasing force trend is observed as the object eventually slips out. Although the panel had intricate features and was very slippery, the compliant elbow pad along with the six-bar mechanism allowed for a more secure and encompassing hold. While the fluctuations represent the dynamic adaptability of the gripper in response to the changing features of the switch panel, the observed peaks at different stages demonstrate the gripper's resilience.

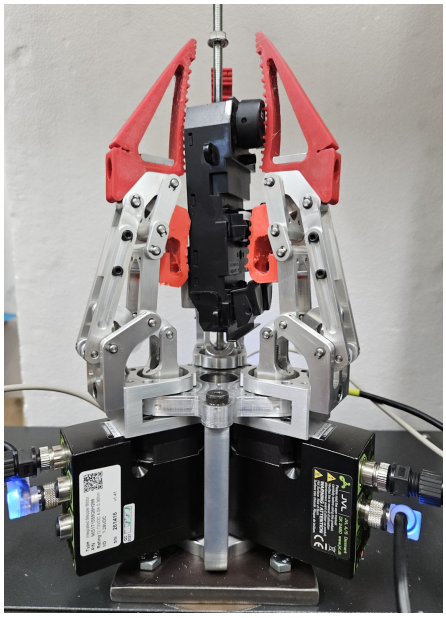


Figure 4.29: Pull-out test of car window switch panel with three-finger grasp

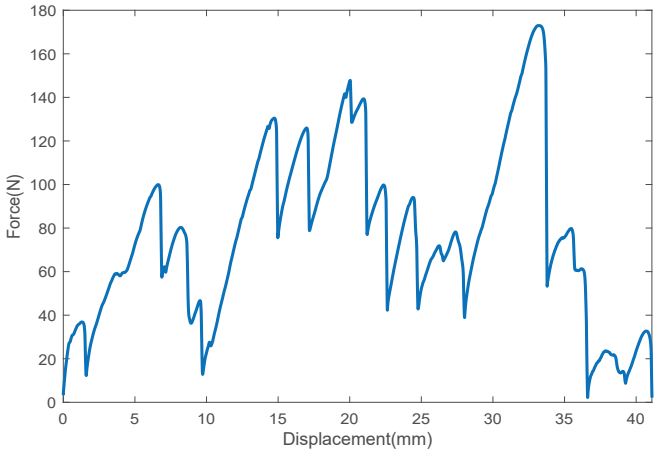


Figure 4.30: Force-displacement curve of pull-out test of car window switch panel

Finally, a tensile test of a clutch plate with a diameter of 195 mm was performed. An increasing force trend was observed from the start, indicating secure engagement of the gripper with the plate. The consistent force trend indicates that the fingers conformed to the shape of the plate edges, thereby ensuring a stable grasp. A maximum peak force of 287 N at 9.6 mm of displacement. After this point, the object starts to slip, and subsequent forces of 109 N at 9.77 mm and 144.3 N at 10.7 mm of displacement represent gripper finger adjustment.

Furthermore, at 13 mm of displacement, an increased force with a peak force of 174 N is observed. However, at 13.5 mm of displacement, breakout occurs.

Extensive pull-out tests on various components have provided valuable insights regarding gripper performance. The force–displacement graphs illustrate the gripper's adaptability and dynamic reaction when grasping materials of different levels of complexity, whereas the force peaks underscore the grasping capacity.

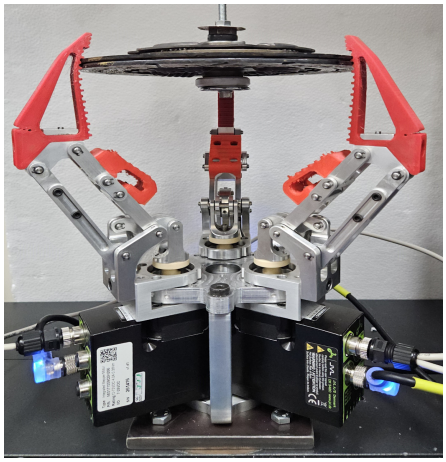


Figure 4.31: Pull-out test of car clutch plate with three-finger grasp

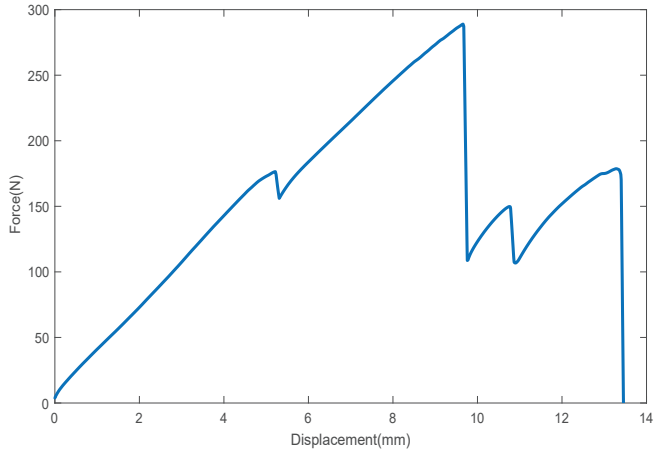


Figure 4.32: Force-displacement curve of pull-out test of car clutch plate

4.7.2 Grasping Assessment with a Universal Gripper Mounted on a Comau Robot

To further evaluate the gripper performance in a realistic object handling scenario, comprehensive grasping experiments were performed with the gripper mounted on a six-axis Comau NS-16 industrial robot. The NS-16 is widely deployed in automotive manufacturing for assembly, material handling, and quality inspection tasks, making it an ideal platform for evaluating the gripper. The experimental setup used for the gripper grasping assessment is shown in Figure 3.26.

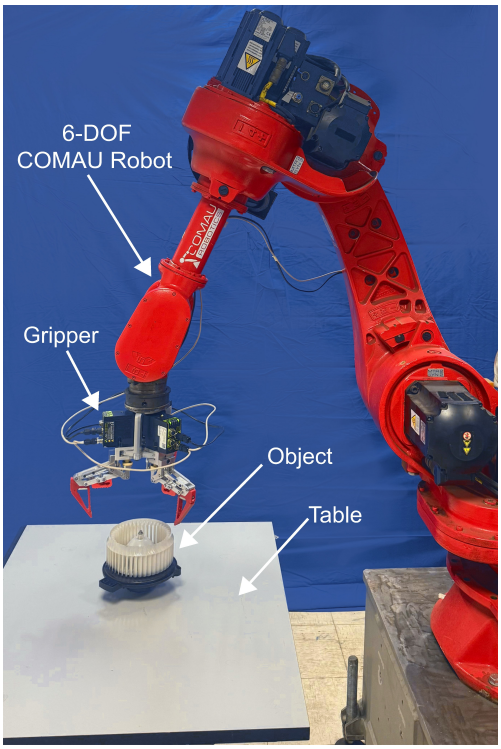


Figure 4.33: Experimental setup with gripper integrated with a 6DOF Comau Industrial Robot

The sequence of the test was such that the robot approached the object placed in a defined position on a table, the gripper securely grasped the object, the robot lifted the object, and the object was held static in air for 20 s. Finally, the object was placed at its original location and the object was released by the gripper. To obtain reliable conclusions, the experiment was iterated five times for each component.

A diverse set of automotive parts and some everyday objects were selected to test the versatility of the gripper across varying geometries, weights, stiffness, and

Table 4.5: Grasp validation with various objects with the enhanced gripper mounted on Comau 6DOF robot

Object	Weight (g)	Dimensions (mm)	Grasp type
Melon	2125	147 (D), 163 (H)	Enveloping
Car Fan Blower	1633	180 (D), 140 (H)	Enveloping
Helical Gear	1065	57 (D), 164 (H)	Enveloping
Power Supply Box	1400	163×162×85	Enveloping
Car Tail Light	1550	complex geometry	Enveloping
Connecting Rod	7500	complex geometry	Enveloping
Car Power Module	260	160×138×35	Fingertip
Cup	247	87 (D), 100 (H)	Fingertip
Clutch Plate	3315	216 (D), 40 (H)	Fingertip
Intake Manifold	2230	complex geometry	Fingertip
Piston	6500	147 (D), 156 (H)	Fingertip
Disk Brake	5000	265 (D)	Fingertip

surface properties. Table 4.5 lists the 12 objects used for grasp validation ranging from 247 g to 7500 g along with their dimensions and the achieved grasp type. These objects can be categorized as cylindrical objects (melon, car fan blower, helical gear, cup, clutch pressure plate, piston, and disk brake), prismatic/box-shaped components (power supply box, and car power module), and complex geometry parts (connecting rod, intake manifold, and car tail light).

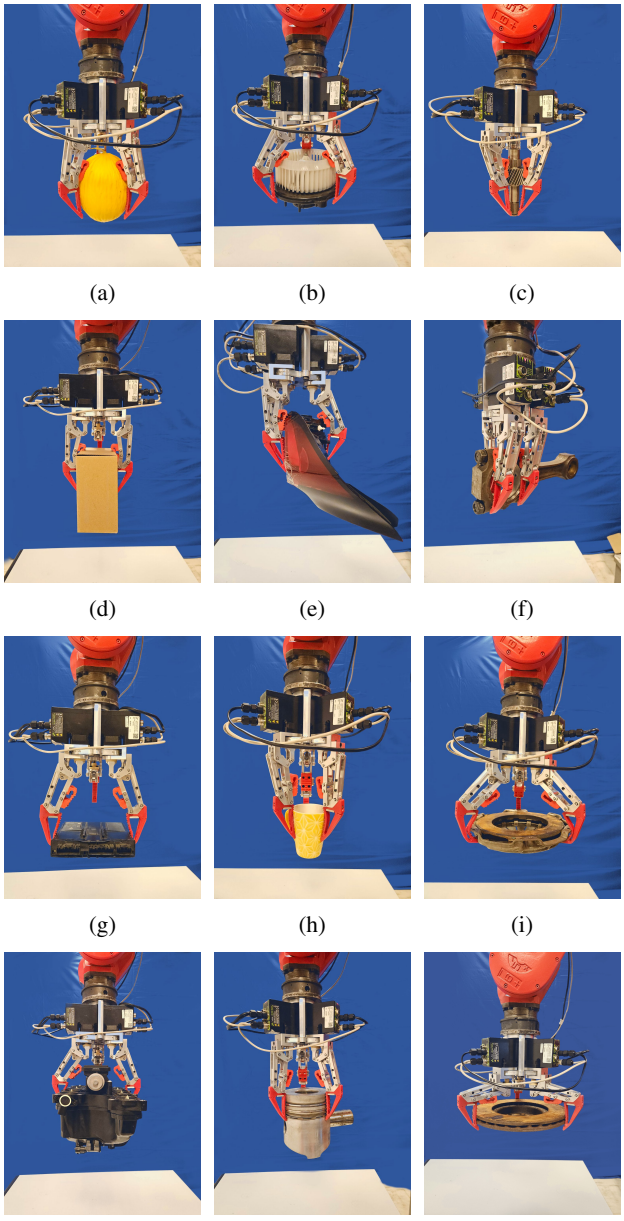


Figure 4.34: Grasping assessment with gripper grasping various objects: (a) Melon, (b) Car fan blower, (c) Helical gear, (d) Power supply box, (e) Car tail light , (f) Connecting rod, (g) Car power module ,(h) Cup (i) Clutch plate, (j) Intake manifold, (k) Piston and (l) Disk brake

Figure 4.34 shows the gripper handling a diverse set of objects. All grasp tests were successful without failure. The enhanced gripper exhibited two grasping

strategies suited to the object's characteristics. Enveloping grasps were successfully executed for the melon, car fan blower, helical gear, power supply box, car tail light, and connecting rod. The enhanced six-bar mechanism enabled enveloping grasps for larger cylindrical objects, such as a car fan blower with a weight of 1633 g and a diameter of 180 mm and a helical gear weighing 1065 g. The compliant pad effectively distributed the contact forces while the articulated linkage segments conformed to the object contours, preventing the rigid link interference issue observed in the original four-bar linkage design. This was also particularly evident when grasping the heaviest object tested, i.e., the 7500 g connecting rod with its complex irregular geometry, the segmented linkage with angles δ_1 and δ_2 adapting to wrap around the component's varying cross-sections. For objects requiring precision grasps, such as the car power module and cup, the redesigned self-adaptable fingers with friction patterns provided stable fingertip grasps. The fingertip design proved particularly effective for the disk brake weighing 5000 g and having a diameter of 265 mm, demonstrating the gripper's expanded grasping range from the original maximum diameter of 219 mm to 270 mm.

The melon and power supply box demonstrate the gentle yet secure grasping of delicate items, while the connecting rod, intake manifold, and piston showcase the gripper's capability with rigid, heavy, and irregularly shaped automotive components. The reconfigurability of the gripper was advantageous for optimizing grasp strategies. For example, for the complex geometry of the connecting rod, the three-finger parallel envelope mode shown in Figure 4.34(f) provided superior stability compared to the default 120-degree configuration. Similarly, the piston required the standard configuration for optimal force distribution around its circular profile, as shown in Figure 4.34(k).

Performance metrics across all tested objects revealed consistent grasp success with no failures during manipulation tasks. The enhanced deflection coefficient of the fingers translated to superior object wrapping, while the increased normal forces from the redesigned finger structure ensured reliable grasps even during dynamic robot movements. The dual function of the compliant pad, which constrains unwanted articulation during normal operation while enabling adaptive compliance during object contact, allowed the maintenance of grasp stability across the entire object spectrum.

The experimental validation demonstrated significant performance improvements over the original gripper design. The enhanced gripper achieved reliable manipulation of objects up to 7.5 kg, a 150% increase from the previous 3 kg limit with a 100% success rate across all grasping trials. Pull-out tests revealed peak retention forces reaching 287 N, compared to the previous maximum of 77 N, representing a substantial improvement in load capacity. The redesigned self-adaptable fingers with a deflection coefficient of 0.1 provided superior object conformity, while the rigid link interference that previously limited grasp

strategies was eliminated by the six-bar linkage mechanism with compliant pad. With a safety factor of 2.0, the enhanced gripper can reliably handle continuous operations with payloads up to 3.75 kg and maximum loads of 7.5 kg for controlled low-speed handling operations for industrial deployment. This expanded operational envelope encompasses the full range of typical automotive powertrain components, from lightweight sensors (247 g) to heavy connecting rods (7500 g), validating the gripper's universality for its intended industrial applications.

4.8 Chapter Summary

This chapter detailed the enhancement of the universal gripper. The redesigned self-adaptable fingers achieved a 74% improvement in wrapping capability (deflection coefficient 0.1 versus 0.39) while increasing load capacity to 287 N, representing a 273% improvement. The transformation from four-bar to six-bar linkage with integrated compliant pad eliminated rigid link interference, enabling articulation angles up to δ_1 = -22° and δ_2 = -15° while maintaining 95.3 force transmission efficiency. These enhancements expanded the operational envelope from 3 to 7.5 kg maximum payload and increased the grasping diameter by 23% to 270 mm. Comprehensive validation across 60 trials with objects ranging from 247 g to 7500 g evidenced 100% reliability. These enhancements lay the foundation for integrating advanced sensorization, presented in the subsequent chapter.

Chapter 5

Camera-Embedded Multi-Modal Sensorization of Self-Adaptable Fingers

5.1 Introduction

Industrial manipulation requires both adaptability to diverse object geometries and precise feedback for robust control. While soft and compliant fingers provide the adaptability necessary for handling irregular or fragile items, their integration into industrial systems has been constrained by a lack of reliable sensorization. Conventional rigid sensors compromise compliance, and external cameras suffer from occlusion and alignment issues in cluttered environments.

This chapter presents the design, development, and validation of a **camera-embedded self-Adaptable finger with multi-modal sensing capabilities for robotic manipulation**. The approach achieves simultaneous proprioceptive and exteroceptive sensing in a compact architecture while preserving mechanical compliance. This represents the primary innovation of the thesis, enabling intelligent manipulation without the compromises of traditional sensorization strategies.

5.2 Sensing Methodology

This vision-based sensing approach for Fin Ray-inspired soft robotic finger is capable of proprioceptive (Z-displacement) and exteroceptive (force estimation, slip detection, position estimation) sensing using a single embedded camera, Figure 5.1 shows the architectural flow from hardware development to real-time implementation. The multi-modal sensing solution overcomes the proprioceptive

and exteroceptive deformation disentanglement challenges inherent in elastomer-based sensors like GelSight. By integrating the camera into the structure of the self-adaptable finger and observing an embedded marker pattern, this approach neither influences the finger's structural properties nor alters its interaction with objects by eliminating the need for additional sensing layers that alter the structure's mechanical behavior. Furthermore, compared to external camera placement in the neuromorphic vision-based approach, the proposed system achieves structural deformation sensing by tracking the internal dot pattern, thereby avoiding possible occlusions that may occur during grasping.

The system is capable of accurate force estimation, Z-displacement measurement, and slip detection in a compact, adaptable gripping system. The design integrates a Pi module V3 image sensor within the framework of a soft finger. This camera, interfaced with a Raspberry Pi 5, captures images of a custom-designed dot pattern on the inner surface of the finger, which serves as the basis for the sensing ability. A CNN estimates the interaction forces and internal deformation (Z-displacement) and classifies the region of force application on the finger. Concurrently, an optical flow-based tracking algorithm was used to track the dot pattern embedded in the self-adaptable finger to detect object slip events. Custom test rigs were developed that work in conjunction with a UTM. The UTM enables precise control of the compressive and tensile forces as well as accurate force and displacement values, allowing creation of a reliable and comprehensive dataset and optimize the algorithm for detecting slip events. In addition, image processing and machine learning algorithms were deployed on an embedded Raspberry Pi, resulting in a modular and deployable sensing solution for various applications. The system was validated under operative conditions by integrating the sensorized self-adaptable finger into a hybrid industrial gripper. Moreover, this approach to force estimation, which utilizes the internal imaging of the self-adaptable finger deformation with corresponding force measurements from a UTM, allows direct mapping between the finger's internal structural changes and externally applied forces, thus providing a simpler approach compared to GelSight's indirect inference from the deformation of a silicon layer on the sensor's surface.

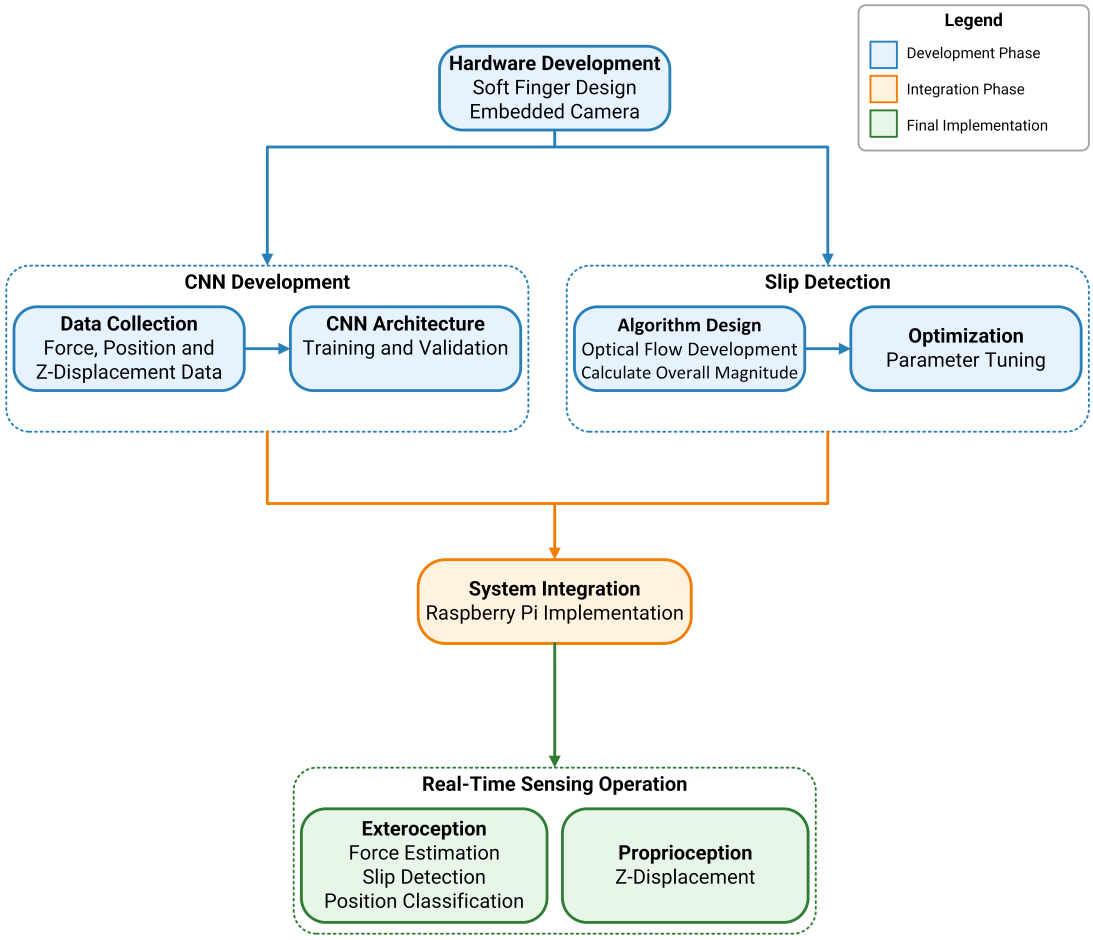


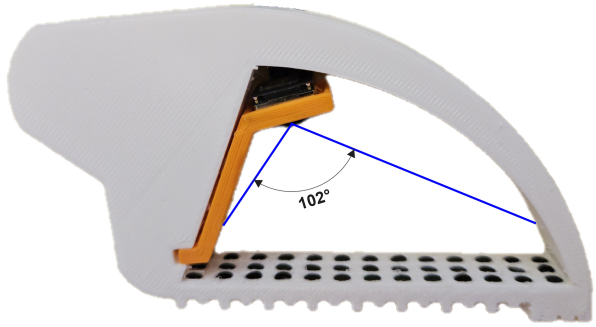
Figure 5.1: Methodology: Development and Integration for Vision Sensing

5.3 Self-Adaptable Finger Morphology, Materials, and Camera Integration

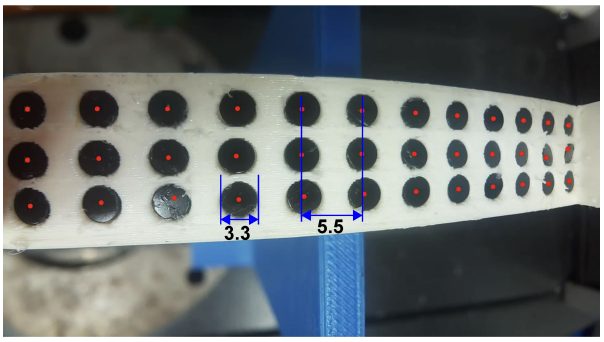
5.3.1 Finger Morphology

The finger morphology used in this study serves as an exemplary implementation for validating our sensorization methodology. The selected structure offers sufficient compliance and contact adaptation to evaluate our sensing approach across various object geometries. This validation shows the possible applicability of the proposed method to different morphologies and materials for soft fingers, although specific finger mechanical characteristics may vary with the design parameters. The finger structure is designed to accommodate a camera stand, allowing a

camera to be positioned for optimal coverage of a dot pattern, facilitating precise feedback and sensing during object interaction, as shown in Fig. 5.2.



(a)



(b)

Figure 5.2: Finger morphology and camera integration: (a) Camera mount with 102° FOV covering the dot pattern, (b) Visible pattern with camera FOV with circular 3.3 mm diameter dots spaced 5.5 mm from their center points

5.3.2 Finger Materials

The gripper fingers were manufactured using 3D printing, with Thermoplastic Polyurethane (TPU) 93A material for the finger (white), Polylactic Acid (PLA) for the circular dot pattern (black) and Acrylonitrile Butadiene Styrene (ABS) for the camera stand (orange). The inward surface features a circular vector pattern of 3.3 mm diameter dots spaced 5.5 mm center-to-center, which serves as a visual reference for proprioceptive and exteroceptive sensing. The pattern scale was empirically found to ensure reliable feature detection across the finger's operating range. PLA was chosen for the dots to ensure better printability and precision, as it maintains its dimensions even during finger deformation. The dot pattern plays a crucial role in providing sensory feedback.

5.3.3 FOV

The camera mount location and orientation were verified using FEA to maximize the camera's Field of View (FOV) while ensuring structural integrity. The embedded camera provides a 102° FOV and was mounted such that the entire internal dot pattern was visible, enabling tracking of pattern movements as shown in Fig. 5.2.

5.3.4 Camera Integration

The vision-based system uses a Raspberry Pi Camera Module V3 Wide, having a 12-megapixel Sony IMX708 CMOS sensor with a CSI interface connected to a Raspberry Pi 5. We used a Sony sensor extension cable [40], shown in Fig. 5.3, allowing the sensor to be separated from the camera PCB and further reducing the form factor for easier and compact integration. Moreover, this reduction in the sensor form factor allows the finger design to be scaled down if required, with careful consideration of the sensor focal length and FOV to ensure adequate visibility of the dot pattern.

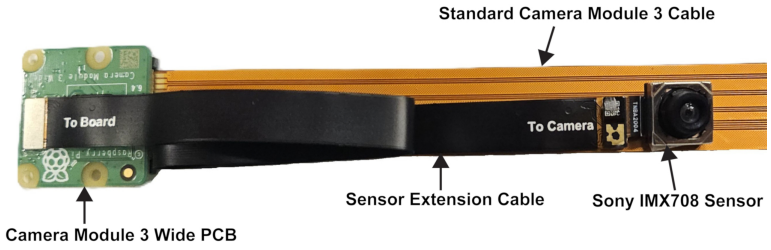


Figure 5.3: Sensor extension cable for camera module V3

The versatility of the vision-based sensing solution is shown by implementation across different soft finger designs with object interface lengths ranging from 65mm to 86mm and distinct curvature profiles, as illustrated in Fig. 5.4. The mounting solution requires only angular adjustment of the sensor mount to ensure a full view of the dot pattern, allowing the ability to transfer the sensing system across various finger designs without the need for significant modifications or redesigns. This enables rapid deployment across diverse object handling scenarios for industrial applications, where different manipulation tasks require customized soft finger designs.

(a) (b) (c)

Figure 5.4: Camera integration across a range of soft finger morphologies: (a) Triangular design, (b) Rigid outward curve with soft interface design, (c) Elongated soft finger design

5.4 Validation of Camera Support

The camera support was designed to provide mechanical stability and stable image acquisition. The mount also features an integrated anchor that partially extends into the finger structure to enhance the position stability. To verify that the camera stand does not interfere with the free deformation of the finger and ensures stable dot pattern visibility, the design was validated through FEM analysis.

The analysis was conducted using ANSYS Workbench, starting from the experimental tensile values. The mechanical properties of the TPU used for the finger structure were assessed using the Zwick Roell Z010 testing machine. The thermoplastic rubber material was tested in accordance with ASTM D412 standards, which outline the procedures for evaluating the tensile properties of both vulcanized thermoset rubber and thermoplastic elastomers. A total of five tensile tests at 5 mm/min were conducted. The experimental values were used as input for the ANSYS curve fitting tool to estimate the hyperelastic material law parameters. To describe the material behavior, a 5-parameter Mooney-Rivlin model Bergström (2015) was selected. The constant values are reported in Table 5.1.

Table 5.1: Material constants for five-parameter Mooney–Rivlin model

Parameter (units)	C_{10} (MPa)	C_{01} (MPa)	C_{11} (MPa)	C_{20} (MPa)	C_{02} (MPa)	d (-)
Values	-13,35	21,97	0,15	-0,01	3,52	0,00

The obtained material parameter was validated in ANSYS using a 3D, 10-node tetrahedral structural solid element and a mesh size of 1 mm. The simulation results of the tensile test on the dogbone were compared with the experimental results in order to validate the characterization of the material. The force-

displacement results of the experimental and simulated tests are shown in Fig. 5.5.

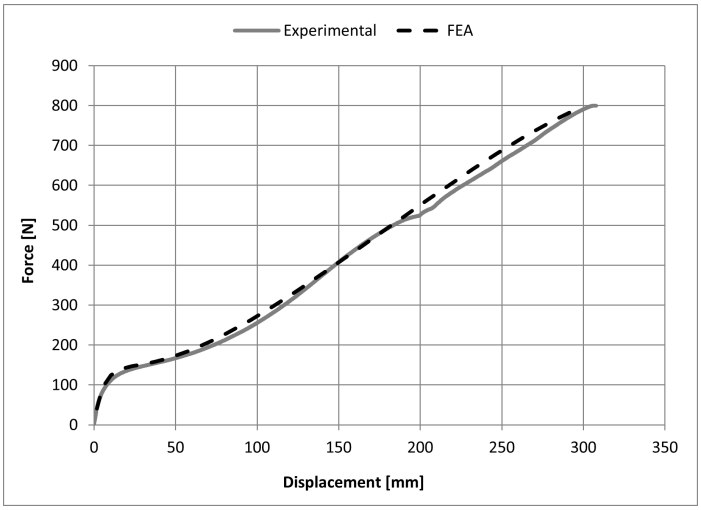


Figure 5.5: Tensile test Result, representative curve

The model of the finger test was developed including the CREO geometry using a 3D, 10-node tetrahedral structural solid element with a mesh size of 1.5 mm. The finger design contains two fixing holes (A) and (B); to constrain the soft finger, it is subjected to fixed constraints. For Camera support, additional constraints were applied to the virtual representation (C), as shown in Fig. 5.6

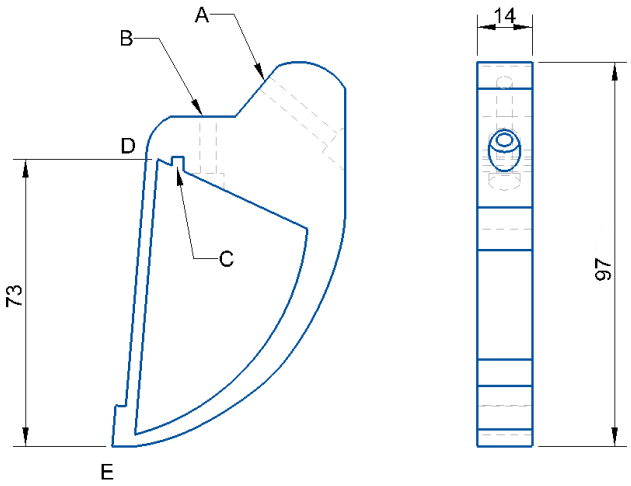
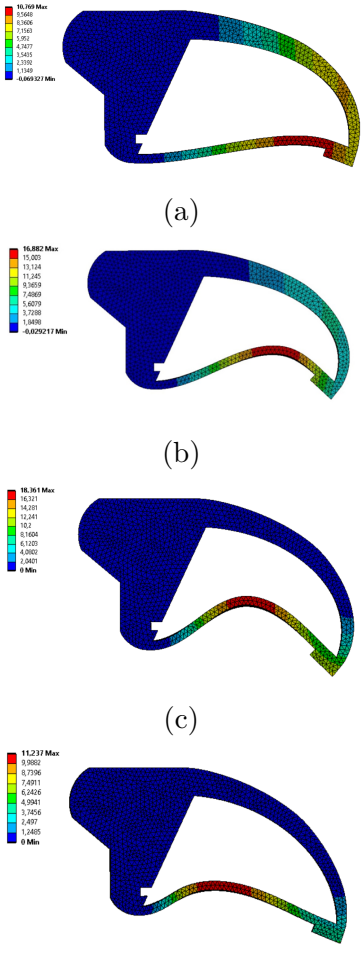


Figure 5.6: Finger constraint point and dimensions in mm

A force of 15 N was applied during the test at four specific points, as shown in Fig. 5.15. The application area was modeled as a 4 mm length, positioned at the middle of each point, to represent the contact area of the object used in the actual test.

The results of the FEA analysis show the deformation depicted in Fig. 5.7. A comparison between the deformation obtained from the experimental test and the virtual simulation under a 15 N load is presented in Table 5.2. The comparison demonstrates a reproduction of the movements, with errors below 5 percent, confirming the model's validity. During the test, there is no movement at the camera stand fixing point (blue color), and the deformation at the maximum payload force does not affect the camera area.



(a) (b) (c) (d)

Figure 5.7: FEA 15 N applied force: a) Point 1; b) Point 2; c) Point 3; d) Point 4 - Deformation in mm on the left

Table 5.2: Displacement at a force of 15 N, comparison between experimental and simulation tests

Point	Displacement [mm]		
-	FEA	Experimental	Error
P1	10.77	11.05	3
P2	16,88	16,33	5
P3	18,36	17,62	5
P4	11,22	11,24	0

5.5 Slip Detection

The development of slip-event detection systems for soft fingers is challenging because of their compliance and complex deformations. We address this issue with a scheme that incorporates optical flow tracking of a dot pattern integrated on the internal wall of the finger, statistical analysis of the tracking, and parameter optimization with a custom-designed experimental setup.

5.5.1 Slip Detection Algorithm

The slip detection algorithm presents a novel combination of internal vision sensing and robust statistical analysis with several innovations in the processing pipeline. The slip detection approach includes two main phases: a preprocessing phase to isolate the dot pattern within a Region Of Interest (ROI) using thresholding and morphological operations that provide feature points for tracking. The second phase is the core slip detection phase using Lucas-Kanade Baker & Matthews (2004) optical flow tracking combined with both Median Absolute Deviation (MAD) Avendaño *et al.* (2024) and standard deviation-based statistical analysis. The dual threshold approach minimizes the effects of outliers and sudden displacement fluctuations considering the varying levels of compliance and flexibility of the self-adaptable finger. Temporal validation is used to further enhance this, distinguishing genuine slip events from noise. Furthermore, we implemented a tracking loss compensation mechanism to maintain feature point consistency by reinitialization. The image processing and analysis pipeline allows accurate feature tracking and slip detection on Raspberry Pi 5.

5.5.1.1 Preprocessing Phase

The primary objectives of the preprocessing phase are to segment and identify the dotted pattern within the soft finger as shown in Algorithm 5.1. This ensures that feature points are consistently detected for tracking, irrespective of variations in lighting or interactions with objects.

1. **Region of Interest (ROI) Initialization):** To realize consistent feature tracking throughout, a fixed ROI containing the internal dot pattern is initialized, and detected features are tracked in consecutive frames.

2. **Frame Conversion and Mask Generation:** During preprocessing phase, the input frames are first converted to HSV color space, and then a binary mask is generated using thresholds identified by trial and error. These thresholds were fine-tuned to effectively isolate the black dotted pattern under different lighting conditions. The mask is then refined through morphological transformations through erosion and dilation to remove noise and for clear object separation, thus enhancing distinction and feature clarity.

3. **Contour Detection and Feature Extraction:** Contour detection is performed within the ROI using area-based filtering to identify relevant shapes. Then, the centroids of the contours are identified using the image moments. Then, the centroids of the contours are identified using the image moments.

5.5.1.2 Core Slip Detection Phase

To identify slip events, the primary slip detection phase observes finger deformation and tracks the identified features across different frames. It relies on statistical measures such as the Median Absolute Deviation (MAD) and standard deviation, complementing these with motion tracking through the Lucas-Kanade optical flow to enhance detection accuracy as shown in Algorithm 5.2.

The mathematical formulation for the second phase is detailed below.

1. **Feature Tracking with Lucas–Kanade Optical Flow:**
The motion of feature points between successive frames is tracked using the pyramidal Lucas–Kanade (LK) method.The displacement vectors are calculated by estimating the displacement of each tracked feature by calculating the difference in the position of points between two consecutive frames. The magnitude of each displacement vector is determined using the Euclidean norm, which represents the distance a tracked point has moved between frames as a scalar. Using these individual displacement magnitudes, the mean magnitude is calculated.

For each feature point i, the displacement vector is defined as:

$$\boldsymbol{d}_i = \boldsymbol{x}_i^{t+1} - \boldsymbol{x}_i^t$$

where $\boldsymbol{x}_i^t = (x_i^t, y_i^t)$ and $\boldsymbol{x}_i^{t+1} = (x_i^{t+1}, y_i^{t+1})$ denote the feature coordinates at frames t and $t+1$, respectively. The displacement magnitude is computed using the Euclidean norm:

$$\|\boldsymbol{d}_i\| = \sqrt{(x_i^{t+1} - x_i^t)^2 + (y_i^{t+1} - y_i^t)^2}$$

A window size of 200x200 was selected for sufficient pattern coverage after testing various window sizes from 50×50 to 400×400 pixels. The pyramid level was set to 10, and the termination of the iterative search was set to an accuracy level of 0.03 or 20 iterations, established after analyzing tracking stability and feature point consistency across rapid slip events. A tracking loss approach was applied with a threshold value of 0.5 after evaluating the feature reinitialization requirement in different scenarios to provide a balance between stable tracking and computational load. The tracking loss is checked after each iteration of the optical flow, and the number of successfully tracked points in the current frame is compared to the total number of tracked points in the previous frame. If the ratio of tracked points falls below 50% it is considered a tracking loss condition, which indicates a tracking degradation scenario, and a feature reinitialization process is triggered. This adaptive tacking loss monitoring strategy ensures more robust tracking.

2. **Displacement Magnitude and Sliding Window Analysis:**
   The average displacement magnitude across N valid feature points is given by:

$$M_{\text{overall}} = \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{d}_i\|$$

   To mitigate frame-to-frame variability, a fixed-length buffer of 20 frames is used as a sliding window for statistical analysis.

3. **Dual-Threshold Scheme:**
   To ensure robust slip detection against noise and transient spikes, a dual-threshold scheme is adopted:

   - **Standard Deviation Threshold:**

$$T_{\text{std}} = \mu_B + \tau_s \cdot \sigma_B$$

     where μ_B and σ_B denote the mean and standard deviation of the buffer, and $\tau_s = 1.5$ is a sensitivity factor.

- **Median Absolute Deviation (MAD) Threshold:**

$$T_{\text{mad}} = \tilde{m} + 2 \cdot \text{MAD}_B$$

where $\tilde{m}$ is the median of buffer values and MAD_B is the median absolute deviation.

The final adaptive threshold is defined as:

$$T_{\text{combined}} = \max(T_{\text{std}}, T_{\text{mad}}, T_{\text{min}})$$

where $T_{\text{min}} = 0.01$ provides a lower bound to maintain baseline sensitivity.

4. **Temporal Constraints for Event Validation:**
To avoid false positives, slip detection is constrained by duration:

- A valid slip must persist for at least $\delta_{\text{min}} = 0.1$ s.
- Events exceeding $\delta_{\text{max}} = 1.0$ s are segmented as distinct slips.

Thus, slip is flagged if:

$$\text{Slip detected if } M_{\text{overall}} > T_{\text{combined}} \quad \text{and} \quad \delta_{\text{min}} \leq t \leq \delta_{\text{max}}.$$

Algorithm 5.1 Slip Detection – Part 1: Preprocessing and Feature Initialization

Require:
F_t: Current frame at time t
R: Fixed Region of Interest for dot pattern tracking
function InitializeTracking(F_0)
    $F_{ROI} \leftarrow$ CropToROI(F_0, R)
    $F_{HSV} \leftarrow$ ConvertToHSV(F_{ROI})
    $M \leftarrow$ BinaryMask(F_{HSV}, HSVthresholds)
    $M' \leftarrow$ Morphology(M)
    $C \leftarrow$ FilterContours(M', [800, 12000])
    $P_0 \leftarrow$ CalculateCentroids(C)
    return P_0 ▷ Initial tracking points
end function
function PreprocessFrame(F_t)
    $F_{ROI} \leftarrow$ CropToROI(F_t, R)
    $F_{HSV} \leftarrow$ ConvertToHSV(F_{ROI})
    $M \leftarrow$ BinaryMask(F_{HSV}, HSVthresholds)
    $M' \leftarrow$ Morphology(M)
    $C \leftarrow$ FilterContours(M', [800, 12000])
    return CalculateCentroids(C)
end function

End of Part 1. → Proceed to Algorithm 5.2 for core slip detection.

Algorithm 5.2 Slip Detection – Part 2: Optical Flow and Dual-Threshold Evaluation

Require:
F_t: Current frame at time t
P_{t-1}: Previous feature points
ω: Sliding window size
τ_s: Slip sensitivity factor
$\delta_{min}, \delta_{max}$: Min/max slip duration
τ_{min}: Minimum allowable threshold
λ: Tracking loss threshold
function CalculateMAD($\mathbf{X}$)
    $\tilde{x} \leftarrow \text{Median}(\mathbf{X})$
    $\text{MAD} \leftarrow \text{Median}(|\mathbf{X} - \tilde{x}|)$
    return $\tilde{x}, \text{MAD}$
end function
function DetectSlip(F_t)
    $G_t \leftarrow \text{ConvertToGrayscale}(F_t)$
    $G_{ROI} \leftarrow \text{CropToROI}(G_t, R)$
    $P_t, \text{status} \leftarrow \text{OpticalFlowPyrLK}(G_{ROI,t-1}, G_{ROI,t}, P_{t-1})$
    if $|P_t| < \lambda |P_{t-1}|$ **then**
        $P_t \leftarrow \text{PreprocessFrame}(F_t)$
    end if
    $\Delta\mathbf{d} \leftarrow P_t - P_{t-1}$
    $\mathbf{m} \leftarrow \|\Delta\mathbf{d}\|_2$ ▷ Displacement magnitudes
    $\bar{m} \leftarrow \text{Mean}(\mathbf{m})$
    $\mathbf{B} \leftarrow \text{UpdateBuffer}(\mathbf{B}, \bar{m}, \omega)$
    if $|\mathbf{B}| = \omega$ **then**
        $\mu_B \leftarrow \text{Mean}(\mathbf{B})$
        $\sigma_B \leftarrow \text{StdDev}(\mathbf{B})$
        $\tilde{m}, \text{MAD}_B \leftarrow \text{CalculateMAD}(\mathbf{B})$
        $\tau_{std} \leftarrow \mu_B + \tau_s \sigma_B$
        $\tau_{mad} \leftarrow \tilde{m} + 2\text{MAD}_B$
        $\tau_{combined} \leftarrow \max(\tau_{std}, \tau_{mad}, \tau_{min})$
        if $\bar{m} > \tau_{combined}$ **then**
            if $t_{start} = \text{null}$ **then**
                $t_{start} \leftarrow t$
            else if $t - t_{start} \geq \delta_{min}$ **then**
                return TRUE ▷ Slip detected
            end if
        else
            if $t_{start} \neq \text{null}$ **then**
                $\delta_t \leftarrow t - t_{start}$
                if $\delta_{min} \leq \delta_t \leq \delta_{max}$ **then**
                    $t_{start} \leftarrow \text{null}$
                end if
            end if
        end if
    end if
    return FALSE
end function

where:

- F_t is the input frame at time t
- R is the fixed Region of Interest containing the dot pattern
- P_t represents the feature points at time t within ROI
- $\Delta\mathbf{d}$ is the displacement vector between consecutive frames
- $\mathbf{m}$ is the magnitude vector of displacements
- $\bar{m}$ is the mean magnitude of displacement
- μ_B, σ_B are the mean and standard deviation of buffer
- τ_s is the slip detection threshold
- τ_{min} is the minimum threshold for slip detection
- MAD represents the Median Absolute Deviation
- λ is the tracking loss threshold for feature reinitialization

5.5.2 Experimental Setup for Tensile Testing

A specialized pullout rig was developed to test the behavior of soft fingers under load to optimize the slip detection parameters, as shown in Fig. 5.8. One of the self-adaptable fingers was equipped with a Pi camera, and both fingers were mounted on the rig with an interface block mounted on linear sliders. The designed setup includes pneumatic cylinders (FESTO 1908253) secured via holders on the guide rails, allowing controlled lateral movement of the linear sliders that mount the self-adaptable fingers. Each cylinder is set to produce a force of 45 N at a pressure of 5.73 bar. This limit was set based on an analysis of both system constraints and operational requirements. The 45N threshold was derived based on the nominal pressure limitations of the cylinders (6 bar) used in the experimental setup and the typical manipulation forces used in an industrial scenario for objects in the size range of 25-37 mm. The pullout mechanism incorporates a Universal Testing Machine (UTM) that continuously applies a controlled force in the vertical position as it pulls the object against the gripping force of the two fingers. This setup allowed tuning the algorithm parameters like the slip detection threshold, to adjust the sensitivity of actual slips while minimizing false positives. The optimization was performed by correlating the actual slip events (verified by UTM measurements and self-observation) with the slips detected by our algorithm in experiments with different object shapes and sizes.

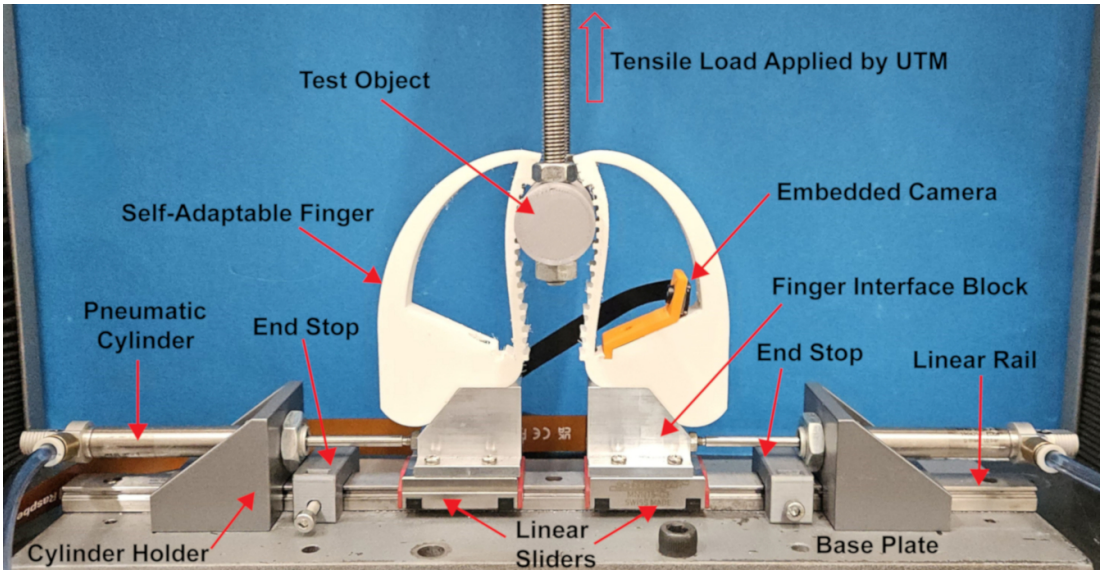


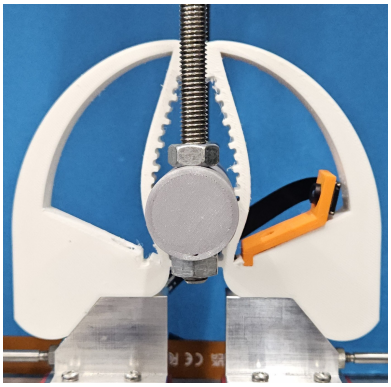
Figure 5.8: Tensile Testing Rig Setup - mounted on the UTM to measure the pull-out forces and slip events

5.5.3 Slip Detection Experiments and Results

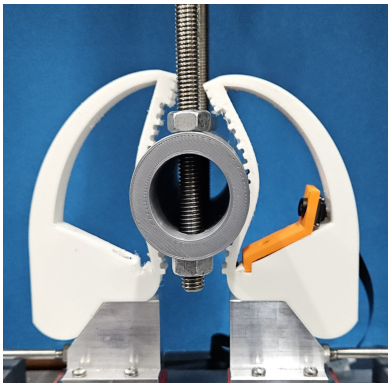
The slip detection algorithm performance was tested using a customized rig integrated with the UTM. We performed pullout tests with four distinct objects:

- Two circular objects with diameters of 25 mm and 37 mm
- Two square objects with diagonals of 33.3 mm and 53 mm

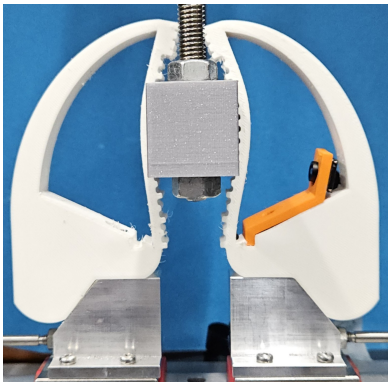
Throughout the testing phase, key algorithmic parameters such as the tracking loss threshold, slip threshold, and temporal constraints were finely tuned. The slip threshold was meticulously defined to distinguish real slip events from typical item movement. Temporal constraints filtered out short-term deformations, enforcing that slip conditions persist for a minimum duration before being detected, while the tracking loss threshold ensured ongoing feature monitoring. As illustrated in Figure 5.9, the diverse array of object sizes and shapes enabled the algorithm to accommodate a wide spectrum of deformation patterns, showcasing its precision and dependability. These initial tests confirmed the algorithm's proficiency in identifying slip events within controlled environments, providing a solid base for subsequent assessments and refinements.



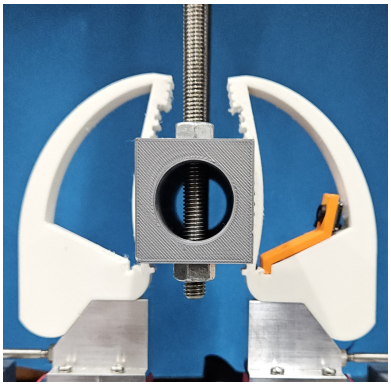
(a) Testing of 25mm diameter circular object



(b) Testing of 37mm diameter circular object



(c) Testing of square object with a 33.3mm diagonal



(d) Testing of square object with a 53mm diagonal

Figure 5.9: Tensile testing for slip detection algorithm optimization using different objects. The objects are pulled at a constant rate while monitoring force behavior

The results obtained through these experiments using the optimized parameters are presented next.

5.5.3.1 Slip Detection Experiment: 25mm Circular Object

The graph in Fig. 5.10 presents the force-time curve of 25 mm diameter circular object. The detected slip events are highlighted by vertical red dashed lines. Due to the nature of the experiment, numerous slips are detected. As seen in the initial region of the graph at t = 5 s, the first slip is detected, where the force rises from near zero to a significant peak of 10 N, which corresponds to the onset of motion between the object and the gripping surface. Following this phase until t = 200 s, a series of periodic slip events occurred at intervals that were observed while exhibiting a baseline force of approximately 10 N. Continuous slip events

demonstrate the finger's adaptive response while grasping. At t = 273 s, a high force exceeding 20 N was observed showing prominent slip events as the object moved toward the fingertip. Post this peak at t = 300-350 s a high frequency cluster of slip events was observed, highlighting critical grasp instability. By t = 400 s, a sharp decrease in the force was observed, signifying the near complete escape of the object from the finger grasp, correlating to the rapid detection of slip events.

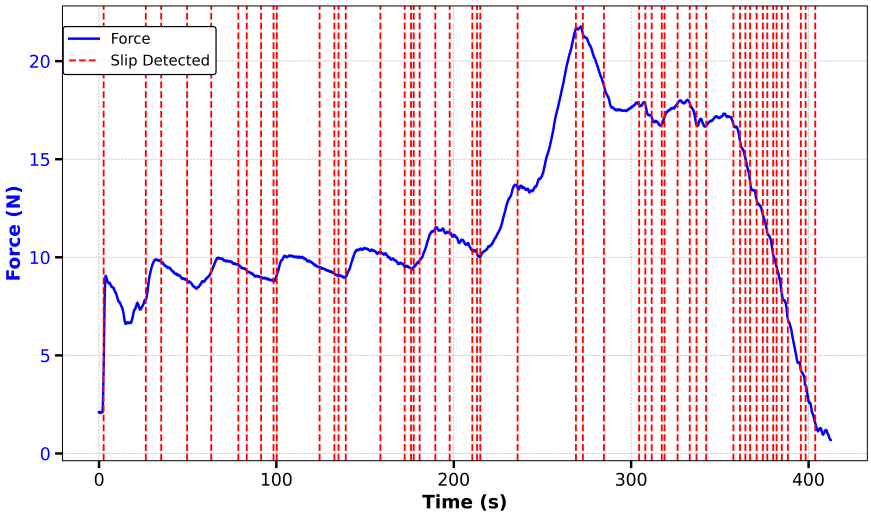


Figure 5.10: Force response during tensile testing of 25mm diameter circular object showing detected slip events overlaid on UTM force measurements

5.5.3.2 Slip Detection Experiment: 37mm Circular Object

The graph in Fig. 5.11 represents the force-time curve of a 37 mm diameter circular object. As seen in the initial region of the graph at t = 5 s, the first slip is detected, where a small decrement in force is observed from 2.4 N to 2.2 N. Following this phase until t = 170 s, a series of periodic slip events occurred at intervals that were observed while exhibiting a steady increase in force till 20 N. At t = 180 s, a high force of 23 N was observed for prominent slip events as the object moved toward the tip. Post to this peak at t = 200-250 s a cluster of slip events was captured. By t = 270 s, a sharp decrease in the force was observed along with rapid slip events, signifying near complete object slippage from the finger.

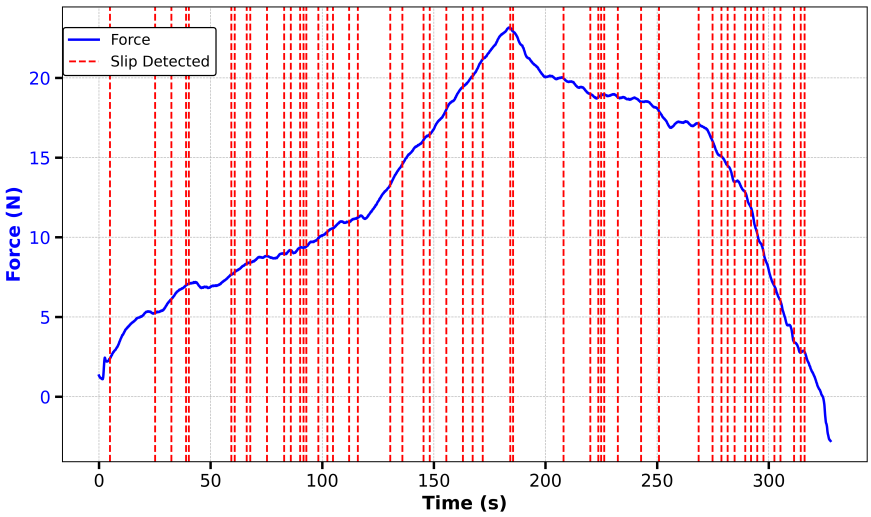


Figure 5.11: Tensile test results for 37mm circular object showing correlation between UTM force measurements and detected slip events

5.5.4 Slip Detection Experiment: 33.3 mm Square Object

The graph in Fig. 5.12 presents the force-time curve of a square object 33.3 mm diagonal. As seen in the initial region of the graph at t = 5 s, where the force rises from near zero to a peak of 11 N, a potential slip is missed by the algorithm. The system captures its first significant slip at t = 12 s in response to a force peak of 13 N. At t = 25 s, a sudden drop in the force was observed mainly due to the geometry of the object. The intermediate phase t = 50-200 s demonstrates the accurate response of the algorithm to changing grip conditions by detecting slip events that are closely correlated to force fluctuations. These force variations indicate the adjusting grip response, resulting in clusters of slip events. At t = 260 s the largest peak of force 106 N was observed, with prominent slip events as the object moves towards the tip of the finger. Afterward, the continuous slip events were captured, highlighting a decrease in the force, signifying the near slippage of the square object from finger grasp.

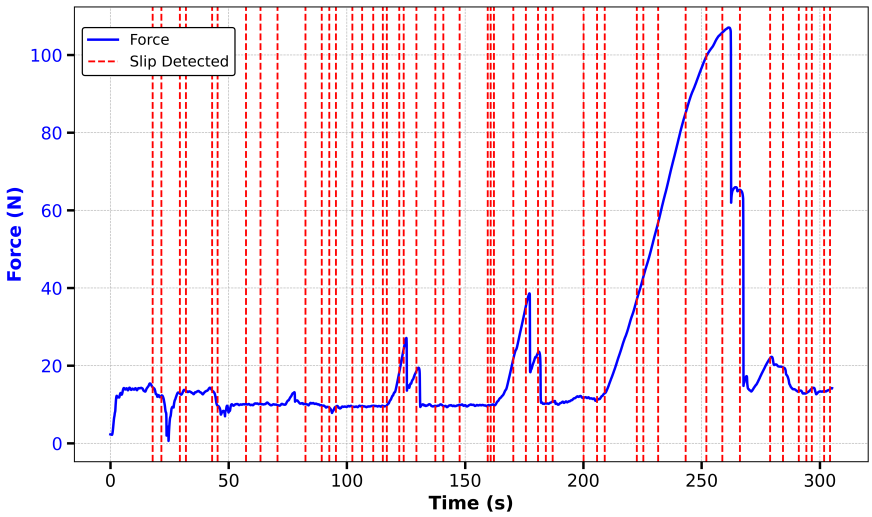


Figure 5.12: Force response during tensile testing of 25mm square object showing detected slip events overlaid on UTM force measurements

5.5.5 Slip Detection Experiment: 53 mm Square Object

The graph in Fig. 5.13 presents the force-time curve of a square object 53 mm diagonal. As seen in the initial region of the graph at t = 5 s, where the force gradually rises to a peak of 30 N, the first slip event is observed. At t = 43 and t = 94 s, a sudden drop in the force was observed, mainly due to the finger adapting to the shape of the object. In the phase t = 40-178 s, various slip events are observed, which are caused by the fluctuating force response as the self-adaptable finger adjusts to the shape of the object. At t = 232 s, the largest peak of force 130 N was observed, followed by major slips with a drastic decrement in force as the object was now escaping the grasp of the finger.

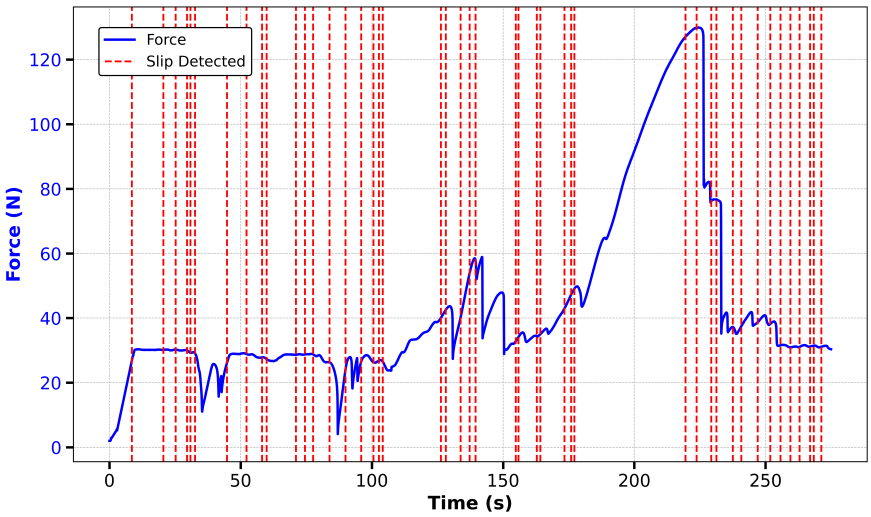


Figure 5.13: Tensile test results for 37mm square object showing correlation between UTM force measurements and detected slip events

In summary, the experimental results underscore the effectiveness of the slip detection algorithm in real-time applications. The custom-designed pullout mechanism served the dual purpose of algorithm validation and parameter optimization while providing precise movement control and ground truth measurements of the applied forces. This allowed us to optimize the algorithm parameters, such as the slip detection threshold and the optimal multiplication factor for standard deviation, to balance the sensitivity and minimize false positives. The integration of UTM data, combined with the continuous pull-up testing, confirms the system's capability to reliably detect and respond to slip events, even in scenarios involving high compliance and dynamic variations. The adaptive thresholding mechanism, in conjunction with a robust data processing pipeline, makes this system highly suitable for practical, real-world deployment.

5.6 Force, Z-displacement and position Estimation using CNN

This section explores an approach to achieve the estimation of force, z-displacement, and position using a Convolutional Neural Network (CNN).

5.6.1 CNN Architecture

The proposed CNN architecture to predict the force, internal deformation (Z-displacement), and force application position for a soft finger is shown in Fig

5.14.

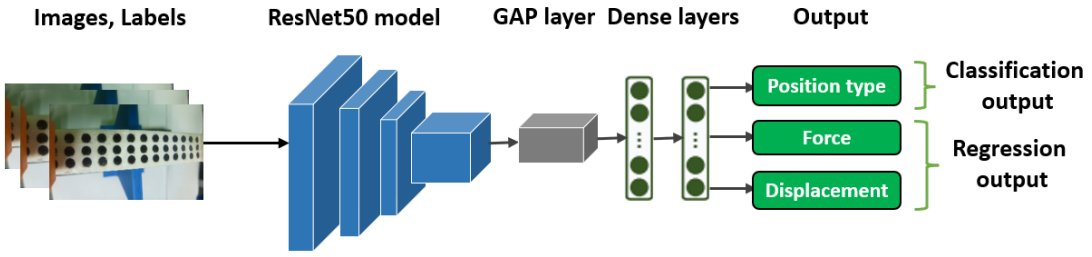


Figure 5.14: CNN architecture to estimate force, z-displacement, and position of the soft finger. GAP: Global Average Pooling.

The force and displacement predictions are carried out as a regression task, whereas the position estimation is conducted as a classification task. The proposed CNN architecture leverages the ResNet-50 model, pre-trained on the ImageNet dataset, acting as a feature extractor that processes input images and produces high-level features. Input images are downscaled to 224×224 RGB images of the finger's internal wall with the embedded dot pattern. Following ResNet-50, a Global Average Pooling (GAP) layer is employed for reducing the spatial dimensions of the output feature maps, averaging each feature map to produce a one-dimensional vector, and optimizing the features for the Fully Connected (FC) dense layers. These FC layers integrate and process the retrieved features for subsequent tasks, enabling the model to learn complex representations and efficiently combine the features. The model predicts two regression variables, force and displacement, processed by a regression head, and one classification head to categorize the force application position (e.g., Tip, Mid, One-Third, Two-Third).

The model is developed using TensorFlow Abadi *et al.* (2015), with the primary objective of deploying the designed CNN architecture on a Raspberry Pi to enable a functional embedded system application. The TensorFlow model is converted into a TensorFlow Lite (TFLite) model, optimized for resource-constrained embedded platforms such as Raspberry Pi. This conversion decreases the model's size and computational demands while preserving accuracy, rendering it appropriate for embedded-system inference. This method enables the system to classify finger positions and estimate force and displacement values efficiently, facilitating a practical and portable solution.

5.6.2 Data Acquisition

In this section, focus is given on the acquisition of video and image data from the soft finger, with the objective of training a CNN model for the estimation of force,

displacement, and position. The data acquisition protocol consists of a systematic approach that includes the experimental setup and the structural collection of data. The process entails setting up the robotic soft finger along with its camera system, exerting forces at four different positions, and diligently capturing video and image data while maintaining consistent precision and environmental conditions to guarantee the dataset's reliability and robustness.

The experimental setup to acquire the required data is shown in Fig. 5.15. This setup is a custom-built jig designed for compression testing of Self-adaptable fingers using a UTM. The jig encompasses a soft finger mounting base attached to a sliding mechanism for vertical linear motion. UTM exerts the compression force on the soft finger and a stop block is positioned beneath the finger to make contact with it at four different test positions: P1:Tip, P2:Two-Third, P3:Middle, and P4:One-Third. The proprietor software of UTM records the force and displacement data generating quasi-static force-displacement characteristic curves, providing insights into the mechanical behavior of the soft finger under various compression positions.

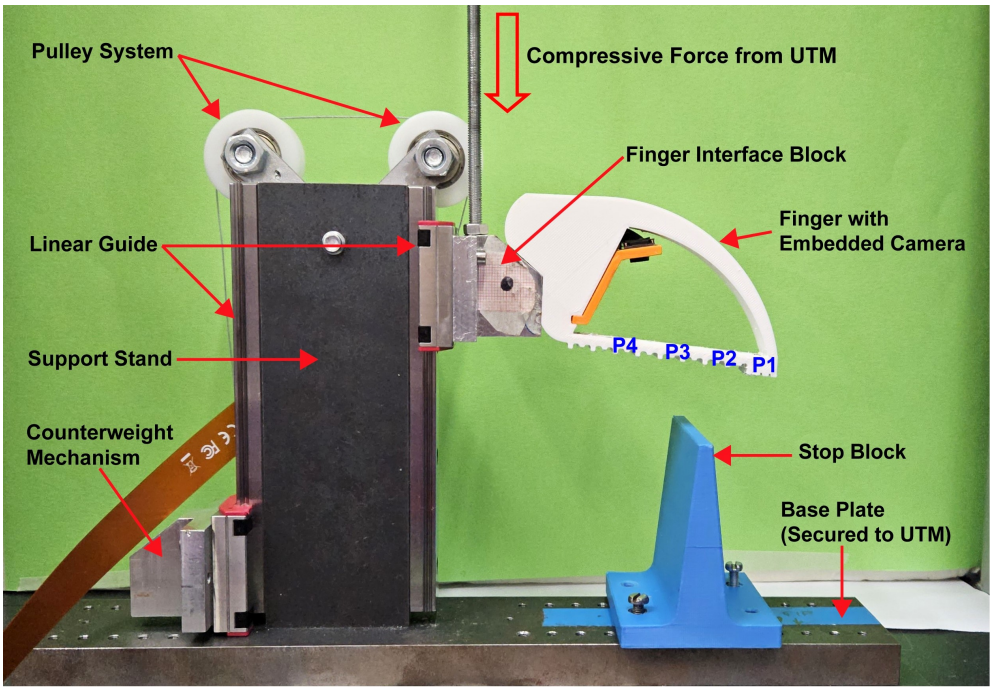


Figure 5.15: Compression Testing Rig Setup with UTM Integration for CNN data collection - The soft finger, equipped with the embedded camera, was subjected to controlled forces at four distinct positions (P1-P4), enabling correlation between ground truth UTM measurements and internal deformation patterns captured by the embedded camera

A systematic approach was implemented for data collection to develop and evolve an accurate predictive model. We used Raspberry Pi Camera V3 embedded within a soft finger, oriented to observe and capture the deformation of the internal wall, which featured a black-dot pattern serving as visual markers for tracking deformation during object interactions. As the UTM applies compression forces to the finger at four predetermined positions, the embedded camera simultaneously records high-resolution Full-HD images of the internal wall, while the UTM logs the data of applied force (N) and z-displacement (mm). Moreover, to generalize across different environmental conditions and varying backgrounds and to focus mainly on the dot pattern features, we deliberately varied background colors and visible elements in the camera FOV. This setup allowed us to obtain a diverse and extensive dataset of 20,778 image-label pairs, illustrating different soft finger positions. The dataset is divided into 80% (16622 frames) for training and 20% (4156 frames) for validation sets, respectively. In addition, unseen test data is created by recording the videos of complete sessions in which the soft finger is fully deformed in all four positions in order to assess the generalizability and diversity of the proposed CNN model in estimating the force, z-displacement, and force application position during grasping and manipulation tasks.

5.6.3 Results: Force, Z-Displacement and Position Prediction

In this section, the application of a CNN-based approach for determining position through classification tasks, as well as force and Z-displacement via regression tasks are discussed. A comprehensive evaluation of the model's predictive accuracy is provided, backed by quantitative measurements and visual depictions.

The proposed CNN model is trained for 50 epochs using a learning rate of 0.001 and a batch size of 32, incorporating learning rate reduction and early stopping strategies to avoid the model overfitting. During the model training, data augmentation techniques, including random alterations to brightness, contrast, and hue, are used to enhance the model's resilience and generalizability. Fig. 5.16 illustrates the training and validation loss of a model throughout 50 epochs. Training loss (blue) and validation loss (orange) initially decrease consistently, signifying learning. However, a minor increase in validation loss occurs at epoch 22, indicating temporary instability. Finally, both losses converge towards zero by the end.

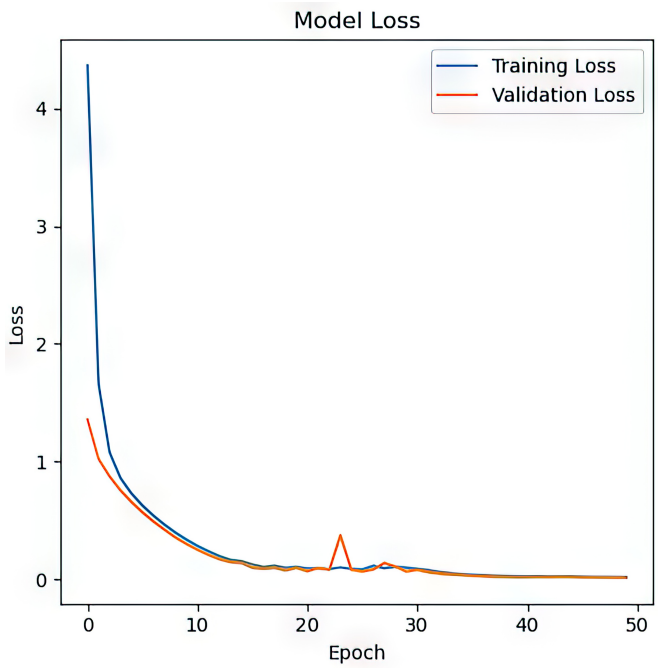


Figure 5.16: Training vs validation loss

5.6.3.1 Regression Results: Force and Z-Displacement

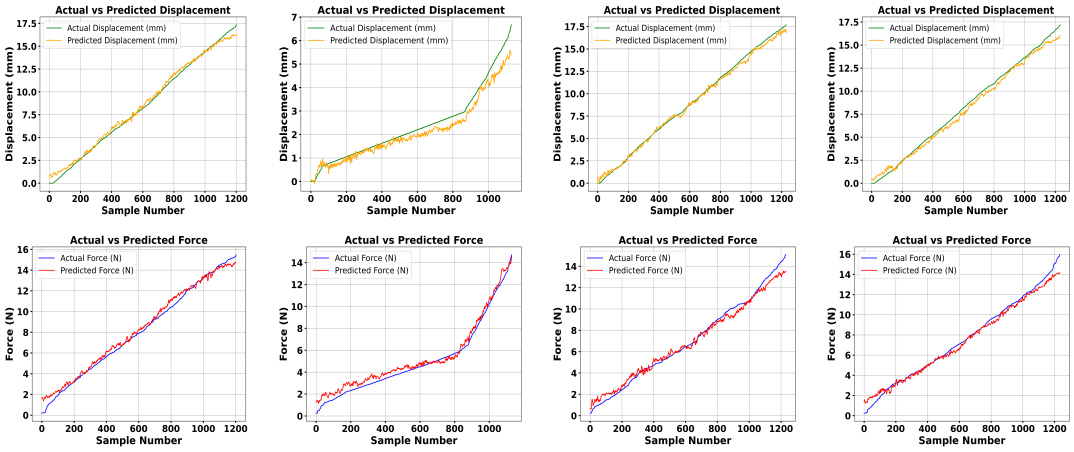
The performance evaluation to estimate force and displacement values for the testing and validation datasets is presented in Table 5.3. The evaluation metrics used include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), which quantify the performance of the regression task. The smaller values of these metrics indicate that the methodology for constructing the dataset and model is effective in estimating force and displacement. The RMSE values for force and displacement at the four positions of the soft finger indicate differences in the model's predictive performance. The tip position exhibits the lowest RMSE for force, indicating improved accuracy in this context, presumably attributable to a more consistent data representation for this particular position. The one-third position exhibits the highest RMSE for force. The mid position demonstrates optimal displacement performance, yielding the lowest RMSE, while the two-third position exhibits the highest error rate. The increased error values in force and displacement estimations across various positions result from subtle deformation changes of the finger that are more challenging to detect. For all positions in both validation and testing datasets, a nearly identical trend is observed across all evaluation metrics.

Furthermore, Fig. 5.17 and Fig. 5.18 illustrate the estimated force and displacement values for each position in the validation and testing datasets, re-

Table 5.3: Performance evaluation of CNN architecture at predicting the force, z-displacement, and position outputs on the custom-built validation and test datasets.

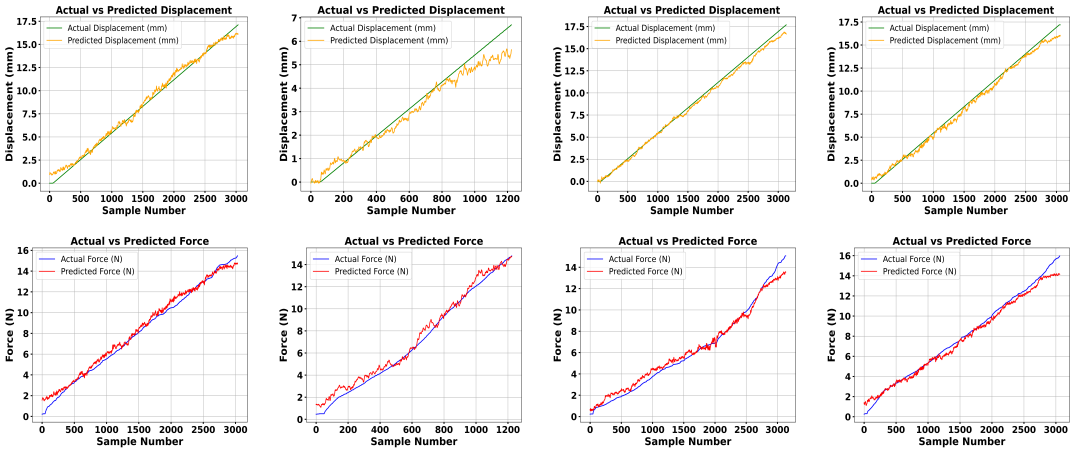
Dataset	Position	Force			Z-Displacement		
		MAE	MSE	RMSE	MAE	MSE	RMSE
Validation	Tip	0.38	0.23	0.48	0.37	0.20	0.45
	One-third	0.45	0.28	0.53	0.30	0.15	0.38
	Mid	0.41	0.27	0.52	0.28	0.13	0.36
	Two-third	0.36	0.24	0.49	0.40	0.24	0.49
Test	Tip	0.40	0.24	0.49	0.37	0.21	0.46
	One-third	0.45	0.29	0.54	0.32	0.19	0.43
	Mid	0.42	0.27	0.52	0.29	0.13	0.37
	Two-third	0.39	0.28	0.52	0.38	0.21	0.46

spectively, alongside the ground truth values, providing a more comprehensive illustration of model performance. The model accurately adheres to the deformation trends of force and displacement values across all positions of the soft finger. The model accurately predicts force and displacement values in accordance with the finger deformation trend across four distinct positions. There are very minor variations in the initial and final values of both force and displacement graphs with respect to their ground truth values, attributed to very slight changes in finger deformation in those regions. In predicting the displacement values, the trend is more consistent at the tip, one-third, and two-third positions, with slight variation observed at the mid position. However, in the case of force trends, the prediction aligns more closely at the tip and one-third positions, while exhibiting slight variation at the mid and two-third positions. Comparison of the data in Fig. 5.17 and Fig. 5.18 reveals similar variations in the trends of force and displacement relative to their ground truth trends. The model demonstrates superior performance in both testing and validation datasets, indicating its potential for extension to additional unseen datasets for practical applications.



(a) Tip position (b) One-third position (c) Mid position (d) Two-third position

Figure 5.17: Qualitative results of force and displacmenet estimation using our proposed CNN model on the custom built validation dataset.



(a) Tip position (b) One-third position (c) Mid position (d) Two-third position

Figure 5.18: Qualitative results of force and displacmenet estimation using our proposed CNN model on the custom built test dataset.

5.6.3.2 Classification Results: Position Estimation

Concerning the position classification results, Fig. 5.19 illustrates prediction for four positions of the soft finger through a confusion matrix. The confusion matrix assesses the performance of a classification model by comparing actual labels with predicted labels, offering an intuitive way to evaluate the model's effectiveness

and identify challenging instances. The model demonstrates high performance for one-third, two-third, and tip positions, exhibiting minimal misclassifications.

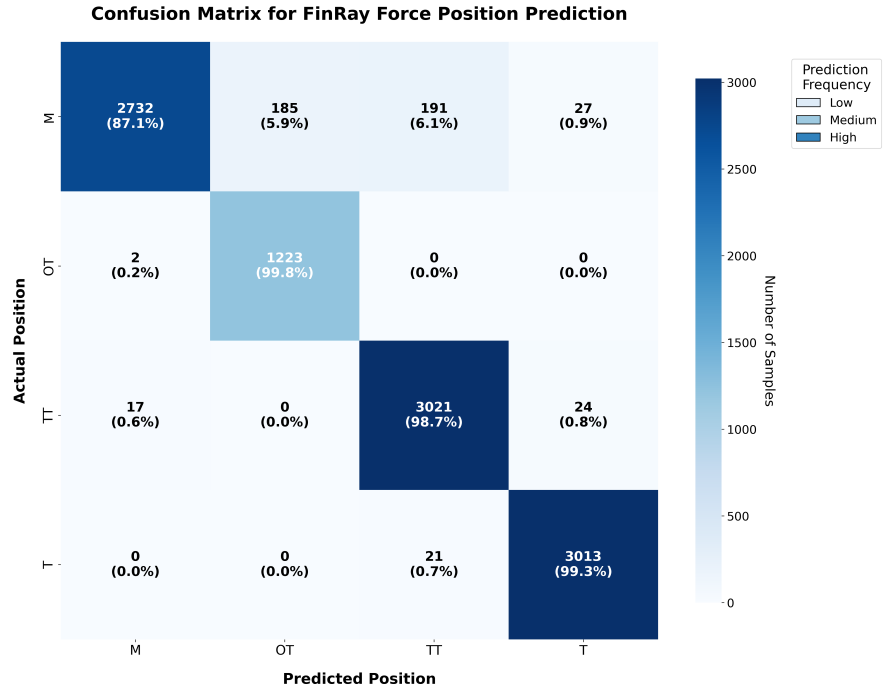


Figure 5.19: Confusion matrix for the position classification.

The model's lower recall and accuracy values for the mid position, on the other hand, show a higher rate of misclassifications, indicating that this class is more difficult for it to classify.

5.7 Integration with Industrial Gripper

The complete sensing pipeline, including CNN-based estimation and optical flow-based slip detection, was deployed on the Raspberry Pi 5. The sensorized soft finger was mounted as part of the universal gripper on a 6-DOF COMAU industrial robot for experimental validation.

5.7.1 Experimental Sequence

An experimental sequence was designed involving dynamic object manipulation under different orientations and controlled slips to assess both exteroceptive and proprioceptive ability. Fig. 5.20. The gripper was controlled in an open-loop configuration with constant motor commands. The experimental sequence began

with the robot in a vertical orientation, configured to grasp an object from a table surface, and then establishing a grasp. Following a secure initial grasp, the robot's wrist joint is rotated 90 degrees, transitioning the gripper with the grasped object to a horizontal orientation. In this position, a manual slip was induced by disturbing the position of the object within the self-adaptable finger grasp. The six-axis robot was then reorientated to its original vertical orientation, where a second manual slip event was performed, thereby completing the experiment. Manual slip induction was used to apply controlled perturbations to the object in the open-loop grasp configuration, thereby enabling temporal markers of slip events and a clear validation of the detection methodology.

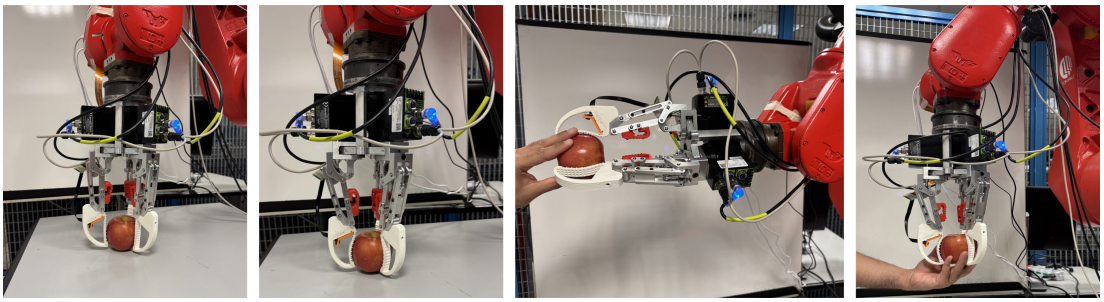


Figure 5.20: Sequential demonstration of the experimental procedure for proprioceptive and exteroceptive sensing: (a) Pre-grasp configuration showing gripper positioning, (b) Establishment of secure grasp in vertical orientation, (c) Horizontal transition with manual slip induction, and (d) vertical re-orientation with final slip validation. This sequence validates the slip detection capability across changing orientations and external perturbations.

5.7.2 Experimental Findings

Fig. 5.21 shows the result of the experimental sequence test, where an apple was used as the test object. The test begins with the pre-grasp period (t = 0-1 s), where the gripper self-adaptable fingers barely touch the apple, and both the displacement and force values are around zero. The initial grasp occurs from t = 1-8 s with a steady increase in force and displacement measurements reaching approximately 8 N and 15mm, respectively, indicating a secure grasp. In the period t = 8-16 s, the robot transitioned to a horizontal orientation, causing a gradual decrease in the displacement to reach 8.5 mm with slowly increasing force values of 8–8.8 N. The displacement shows an increasing trend due to the settling of the finger structure in the horizontal configuration after the transition effects. At t = 17 s while in the same horizontal position, the first manual slip was caused by the movement of the apple within the secure grasp and was accurately detected,

as shown in Fig. 5.21 as the first vertical red dashed line. Subsequently, as the gripper began reorienting back to its starting vertical position, the displacement started to settle at 17mm, and a decreasing trend in the force values was observed. This occurs during the transition as the gravitational and inertial loads are re-distributed and due to the finger's structural change as it adapts. A second slip was detected at t = 20.7 s, resulting from a combination of finger self-adaptation after the disturbance caused by manual slip and reorientation dynamics. During the transition back to the vertical position, a gradual decrease in displacement to 10.1 mm corresponding to increasing force is observed, similar to the behavior observed in the first orientation change. After reorientation, as the adaptation continues, the final manual slip is induced at t = 25.8 s. Following this slip, the force exhibits an increasing trend and reaches 8.4 N, whereas displacement oscillations between 10-12.5mm are observed, indicating the dynamic response to the disturbance.

Throughout the experiment, the sensing approach demonstrated good performance for force, displacement, and slip detection, while the position classification remained consistent as mid-position, thus establishing sensing capabilities in both proprioceptive and exteroceptive sensing modalities.

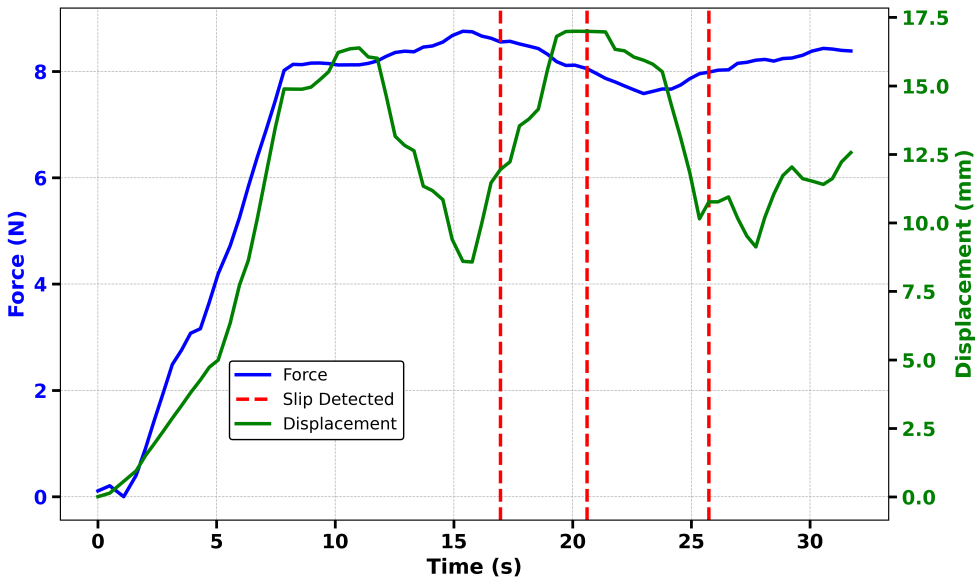


Figure 5.21: Integrated sensor measurements during apple manipulation

5.7.3 Validation and Testing Analysis

The detailed use case of apple has been discussed; however, the analysis was performed using 10 objects of varying weights, sizes, and physical properties that were selected from the Amazon Picking Challenge and YCB datasets. The slip detection and position classification metrics are shown in Table 5.4. The physical characteristics of objects such as apple and tennis ball correlate with the success rate of slip detection exceeding 95%. In contrast, soft and semi-rigid objects like a sponge and black tube demonstrated a lower success rate of 85% and 88% with more false positives and negatives due to their varying physical characteristics during the experimental sequence.

Position classification showed a distinct correlation with the object geometry and grasping orientation. Elongated objects such as the black tube and scissors demonstrated exceptional accuracy of 97.2% and 98.5% in their grip positions of two-third and tip respectively. The cereal box and mug showed good classification at their grasp position of one-third 86.5% and 97.5%, with lower accuracy of the cereal box due to its semi-rigid nature causing confusion. Objects with a spherical shape like an apple and tennis ball, grasped from the middle position demonstrated an accuracy of 87.5% and 89.2%. Likewise, the water bottle grasped from the mid showed 88.4% accuracy, which was mainly influenced by the change in weight distribution during manipulation. Despite its deformable structure, the scrub sponge demonstrated a reasonable accuracy of 85.4% at the mid-position.

The detailed analysis shows that our system can maintain reliable slip detection and position classification accuracy across different object geometries during dynamic manipulation with changing orientation and manually induced object slippage events, further validating our integrated approach.

The full iteration of the algorithm takes 160 ms (6.25 Hz), with the CNN requiring a processing time of 125 ms (8 Hz) and the slip detection algorithm response time of 35 ms (28.5 Hz). The higher processing time of CNN inference represents the major processing bottleneck; however, slip detection achieves true real-time performance. The complete processing cycle of 160 ms (6.25 Hz) allows practical feedback manipulation, considering typical industrial grip adjustments response time is 300-500 ms Hugo (2013). These performance characteristics achieved on an embedded platform establish the system's potential for implementation in real-world applications where proprioceptive and exteroceptive sensing is vital for maintaining secure and stable grasps.

The proposed vision-based sensorization approach addresses several longstanding challenges in soft robotic sensing:

- Preserves compliance by embedding sensing without rigid components.
- Provides multi-modal feedback from a single compact architecture.

Table 5.4: Performance evaluation metrics across ten daily objects used in the experimental sequence

		Apple	Mug	Scissor	Water Bottle	Cereal Box	Tennis Ball	Toy Duck	Screw Driver	Scrub Sponges	Black Tube
Object Image											
Weight (g)		120	200	75	550	365	58	47	150	15	180
Physical Nature		Rigid	Rigid	Rigid	Rigid	Semi-Rigid	Semi-Rigid	Semi-Rigid	Rigid	Soft	Semi-Rigid
Slip	Success Rate	100.0%	92.5%	95.0%	97.0%	90.0%	98.0%	93.0%	91.0%	85.0%	88.0%
	False Positives	0.0%	5.0%	3.0%	2.0%	6.0%	1.5%	4.0%	5.0%	8.0%	7.0%
	False Negatives	0.0%	2.5%	2.0%	1.0%	4.0%	0.5%	3.0%	4.0%	7.0%	5.0%
Position Accuracy		87.5% (M)	97.5% (OT)	96.8% (T)	88.4% (M)	86.5% (OT)	89.2% (M)	96.4% (M)	98.5% (T)	85.4% (M)	97.2% (TT)

- Demonstrates real-time performance on embedded hardware.

Compared to traditional resistive or capacitive sensors, the camera-based approach avoids mechanical interference and achieves higher fidelity across multiple sensing modalities. Its integration with the universal gripper platform validates its potential for industrial applications requiring adaptability and intelligent feedback.

5.8 Chapter Summary

This chapter introduced a novel vision-based sensing paradigm for FinRay-inspired fingers, embedding a compact camera system. The integration of vision-based sensing within the additively manufactured soft finger structure establishes the feasibility of achieving multiple sensing modalities with a single compact embedded system while retaining the characteristic properties of the fingers. The proposed system successfully estimates normal interaction forces, measures internal deformation (Z-displacement), classifies the position of the applied force, and detects slip events with the complete sensing pipeline processed on an embedded platform while avoiding complex signal disambiguation challenges and occlusion issues. The strategy of using adaptive thresholds combined with robust statistical schemes resulted in reliable sensing performance, with the slip detection component operating at 28.5 Hz and the CNN inference at a practical update rate of 8 Hz. Moreover, the implementation on a modular embedded platform enables easy integration into existing soft robotic systems. The open-side design of the finger presents challenges due to environmental variations; However, it allows simplified finger interchangeability and hardware reuse in different finger designs. Although we have integrated techniques to compensate for environmental variations in our

algorithm, extreme changes in ambient conditions can impact sensing reliability. In such cases, the addition of a finger enclosure and constant internal illumination may improve the sensing results. These considerations were carefully balanced against the system's accessibility and deployment requirements.

Chapter 6

Resistive Sensorization of FinRay-Inspired Fingers via Additive Manufacturing

6.1 Introduction

The development of compliant robotic fingers requires embedded sensing methods that preserve mechanical compliance while providing reliable feedback. Traditional sensor integration often involves attaching pre-fabricated resistive or capacitive sensors onto soft structures, which can compromise flexibility, introduce stress concentrations, and reduce long-term durability. Additive manufacturing presents a promising alternative, enabling the **co-fabrication of mechanical structures and sensing elements** in a single process.

This chapter introduces a resistive sensing strategy for FinRay-inspired soft fingers, fabricated entirely through multi-material fused deposition modeling (FDM). The co-fabrication approach enhances durability, and reduces assembly complexity. The design, fabrication methodology, and experimental validation of this resistive sensorization approach are presented. The results demonstrate that additive manufacturing can deliver a cost-effective, integrated sensing solution for industrially relevant compliant fingers.

6.2 Resistive Sensorization via Additive Manufacturing

6.2.1 Motivation and Concept

Resistive sensors provide a lightweight and economical way to integrate feedback into flexible fingers. Traditional designs are typically fabricated independently and then mounted onto soft structures, which can compromise both flexibility and longevity. In this study, a resistive sensor was 3D-printed and **integrated directly within the soft finger** through multi-material FDM printing. This technique results in adaptable and effective sensors that address challenges in soft robotics. The sensors' enhanced durability and sensitivity allow them to perform excellently even with frequent use. Conductive polylactic acid (PLA) is another crucial material within soft robotics. Its popularity in additive manufacturing, especially 3D printing, stems from its low melting point and favorable mechanical properties, making it ideal for developing complex sensor designs. It combines robustness with conductivity, perfect for crafting complex sensors that easily integrate into soft robotic frameworks. Conductive PLA sensors deliver consistent performance and are essential for tasks requiring precise control and feedback, such as soft robotic fingers. The described method utilizes multi-material FDM printing to fabricate conductive TPU and PLA based sensors, which are inexpensive and entirely 3D-printable materials, contrasting with former methods that relied on intricate fabrication approaches like lithography or inkjet printing of silver nanowires, carbon nanotubes, or liquid metals. This technique enables the straightforward creation and integration of sensors into robotic frameworks. Distinct from past studies often centered on single-track or flat sensors, this research examines the impact of both track multiplicity and thickness variation within a systematic design-to-performance framework. A fundamental component of these sensors is the application of Wheatstone bridge circuits to detect minor resistance changes during robot operation. The circuit, with a configuration of four resistive elements, heightens the sensors' sensitivity and precision by translating resistance fluctuations into detectable voltage variationsSaeedi & Effatnejad (2021). This precise feedback enables soft robotic components to recognize even slight pressure and strain fluctuations, facilitating the safe and precise execution of meticulous tasks, such as handling delicate items and performing intricate surgical operationsGeorgopoulou *et al.* (2021). The progress of soft robotics owes much to these easily integrable sensors, whose reliable performance is ensured by the combination of cutting-edge materials and sophisticated circuitryWhitesides (2018).

6.3 Materials and Methods

6.3.1 Material Selection

6.3.1.1 TPU-Based Sensors

Two commercially available thermoplastic polyurethane materials were selected:

Conductive Component:

- Material: Filaflex TPU conductive filament
- Shore hardness: 92A
- Tensile modulus: 90 MPa
- Elongation at break: <100%
- Electrical resistivity: ∼3.9 Ω/cm
- Function: Core sensing element

Insulating Component:

- Material: Non-conductive flex filament
- Shore hardness: 93A
- Tensile modulus: 40 MPa
- Function: Electrical insulation between conductive tracks

6.3.1.2 PLA-Based Sensors

A second sensor variant utilized:

- **Conductive PLA:** Resistivity of 15 Ω/cm along printed layers
- **Standard non-conductive PLA:** For insulation and structural support

6.3.2 Fabrication Process

The fabrication process employed a Prusa i3 MK3 3D printer equipped with the MMU2S multi-material upgrade kit, enabling the use of up to five different filaments in a single printing cycle through automatic filament switching via a single nozzle. This capability was essential for creating the integrated sensor structures with alternating conductive and insulating materials. The printing parameters

were carefully optimized, utilizing a rectilinear infill pattern with 100% infill density to ensure consistent electrical properties throughout the sensor structure. Particular attention was paid to temperature control during material transitions to prevent cross-contamination between conductive and non-conductive materials while maintaining adequate inter-material adhesion. The 3D models were prepared and sliced using the open-source PrusaSlicer 2.7.4 software, which allowed precise control over multi-material printing parameters. Temperature profiles were individually adjusted for each material transition to balance the competing requirements of preventing thermal degradation, ensuring proper layer adhesion, and avoiding material mixing at the nozzle during filament changes.

6.4 Sensor Design

This section describes the design of resistive sensors developed for high-precision measurements in our study of soft robotics. The design process required a careful examination of several elements, including sensing systems and geometric configurations. Additionally, we designed robotic fingers with integrated sensors to ensure simple performance and functionality. The resistance R of the sensors is given by: $R = \rho \cdot \frac{l}{A}$, where ρ is the resistivity of the material used in the sensor, l is the length of the sensor, and A is the cross-sectional area of the sensor trace. As the sensors are deformed, R changes accordingly.

6.4.1 Sensor Tracks

Using CAD modeling, four different resistive sensors with various track designs were created to examine differences in electrical resistance and enhance strain sensitivity. The dimensions of these models were kept constant at 25 mm in length, and 1 mm in thickness to ensure measurement consistency across designs. After conducting thorough measurements across the various track configurations, a design showing eight tracks for the resistive sensors was selected. This decision was based on the better performance observed under various load conditions, with the eight-track arrangement proving to be the most durable and dependable design for managing a range of mechanical loads. Following this selection, optimization CAD design focused on improving the sensor's thickness to enhance its structural performance and consistency.

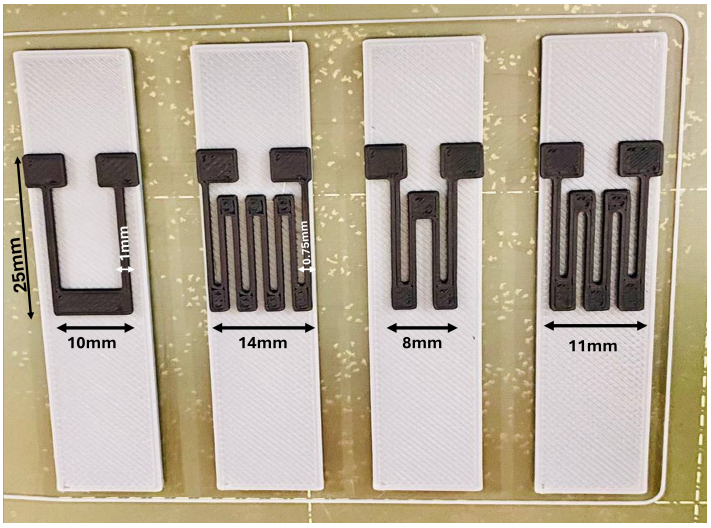


Figure 6.1: 3D-printed version of the designed sensors

6.4.2 Sensor Thickness

Four resistive sensors with varied thicknesses (0.2 mm, 0.4 mm, 0.6 mm, and 0.8 mm) were designed using CAD to investigate their effect on material resistance and durability while maintaining the constant eight-track setup. Each model was precisely constructed with a uniform length of 25 mm and a track width of 14 mm to ensure consistent measurement capabilities across all designs. This approach allowed us to understand how variation in the thickness affects the sensor's performance, helping to optimize the sensor's structure and functional accuracy.

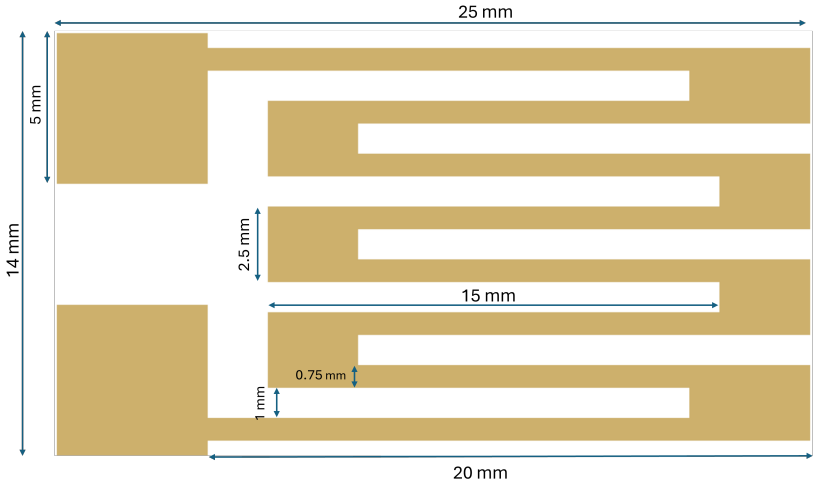


Figure 6.2: 3D Model of the designed sensor with 8-track

6.5 Robotic Finger Design with Integrated Sensor

In the design of advanced robotic fingers (Figure 6.3 and Figure 6.4), precise strain measurement is essential for accurate and responsive movements. Our design was significantly inspired by the work presented in Suder *et al.* (2021). This study evaluated various FinRay finger designs with particular emphasis on internal structural configurations. Their findings showed that fingers without internal structure demonstrated superior performance overall in object grasping.

Two different robotic finger variations measuring 82 mm in length, 14 mm in width, 44 mm in height, and 3.6 mm in wall thickness were created. A soft resistive sensor is incorporated into both designs to improve functionality.

6.5.1 Open Configuration

This design features an open configuration where the sensor is placed in the space under the finger, allowing more direct contact between the sensor and the object being grasped(Figure 3). The size of the sensor and the base is 1.4 mm. This ensures easy access and maintenance and allows for accurate strain measurement when flexing the finger. To minimize possible misalignments or movements during the grasping task, the resistive sensor was integrated into the soft finger using embedded sensor channels.

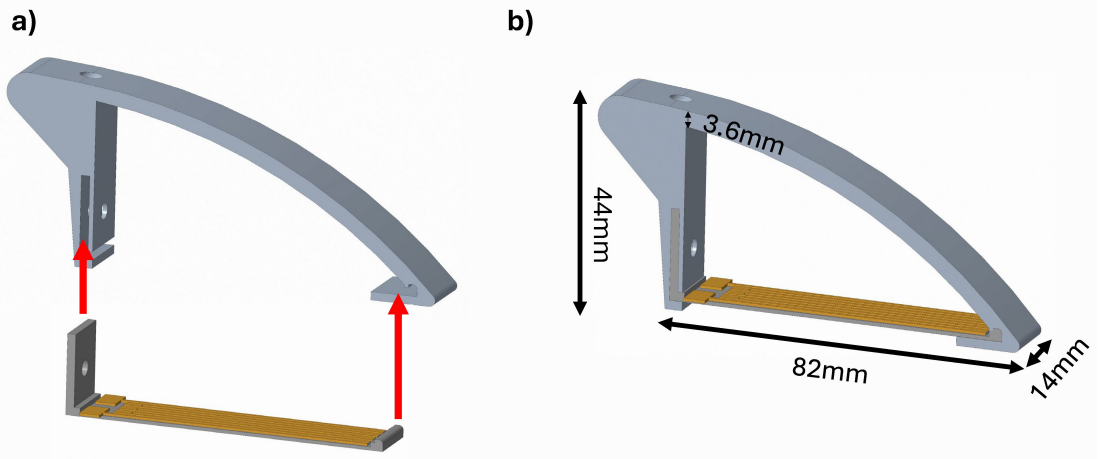


Figure 6.3: Open Configuration Robotic Finger a) View showing the assembly of the open configuration finger with the resistive sensor positioned underneath b) Assembled open configuration with complete dimensional specifications

6.5.2 Closed Configuration

With a continuous frame structure, the finger completely encapsulates the resistive sensor within the finger structure, ensuring seamless and protected integration. The closed design provides a uniform grip surface, which improves the grasping task performance. By fully embedding the sensor along the entire length of the finger, this configuration enables continuous strain measurements throughout the finger structure and provides comprehensive feedback on finger deformation and interaction with objects.

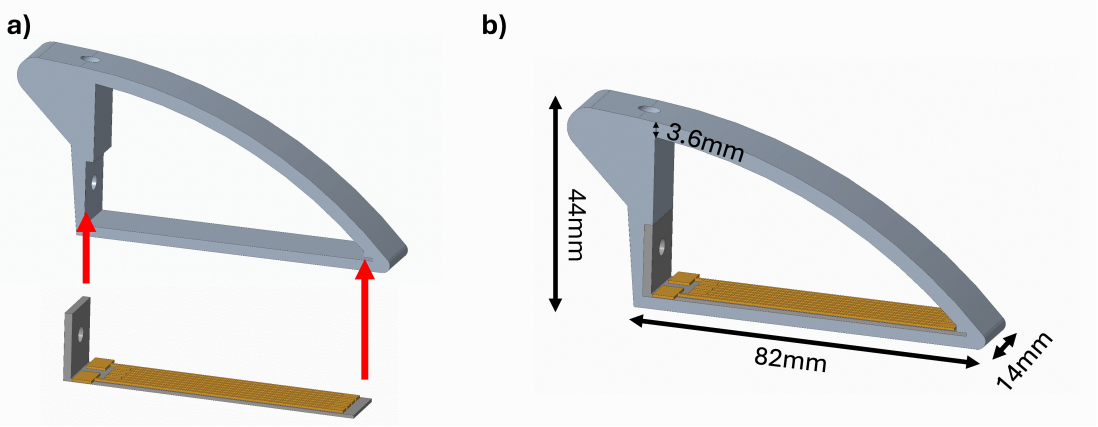


Figure 6.4: Closed Configuration Robotic Finger a) View of the closed configuration design showing the internal support structure and sensor integration b) Assembled closed configuration with complete dimensional specifications

6.6 Mechanical Integration

Several sensors and fingers were printed using a standard FDM printer (Prusa i3 MK3) equipped with an MMU2S multi-material upgrade kit. Various sensors were printed using conductive and non-conductive PLA filaments to test and optimize the design. The primary goal was to ensure the dimensions of the sensing tracks matched the printer specifications, such as nozzle diameter, to avoid over or under-extrusion problems. After the simulation, in the first stage of the final print, the optimized sensor design was printed using both conductive and non-conductive filaments simultaneously. Because of this, we were able to create the thinnest employed sensor with exact track dimensions. In the second stage, the robotic finger was printed using flex filament with a Shore hardness of 93A.

The integration of the sensor with the robotic finger was conducted with two different configurations: open and closed. For the open configuration (a), the finger's body was printed with TPU, and the sensor was printed separately using PLA and PLA conductive. The PLA sensor was then attached below the TPU finger, resulting in what we refer to as the "TPU-PLA sensor base."

For the closed configuration (b), the finger was printed totally with TPU. The sensor, composed of both conductive (Filaflex TPU) and non-conductive above-mentioned TPU filaments, was printed separately and then positioned on top of the TPU finger. This assembly is referred to as the "fully TPU finger." These configurations allow us to evaluate the performance and suitability of the integrated sensors within the robotic fingers for various applications.

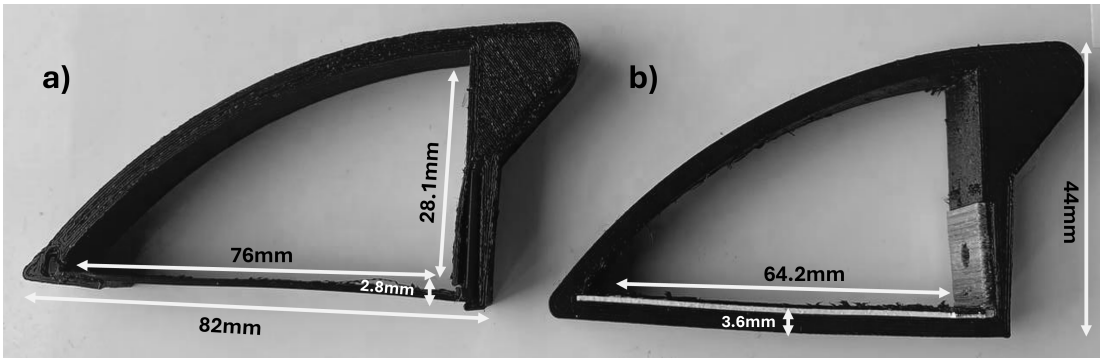


Figure 6.5: 3D Printing Finger and Sensor

A limitation of our methodology that becomes more obvious with extended use is the drift in the printed sensor's readings. This drift often occurs with resistive-based sensors caused by the heating of the sensor when electricity passes through its conductive tracks. The drift due to heating continues to increase the longer the sensor is used, making it difficult to accurately characterize the sensor response. To address this issue, we implemented a Wheatstone bridge circuit designed to convert the sensor's change in resistance into voltage, amplify the output voltage across the 0–5 V range, and minimize drift by reducing the current through the sensor. This circuit, the Wheatstone bridge, is balanced at the sensor's base resistance value, with R1 and R2 set to 15 kΩ and a potentiometer used to zero the output voltage when the sensor is flat.

6.7 Experimental Results

6.7.1 Sensor's Evaluation

To evaluate the sensor's response, we systematically tested and characterized the sensors with different tracks. The initial resistance values of each sensor after printing varied, with the 2-track sensor measuring 6.26 kΩ, the 4-track sensor at 7.46 kΩ, the 6-track sensor at 10.5 kΩ, and the 8-track sensor at 17.70 kΩ. In the first tests, resistance variations under various applied forces were measured to examine the behavior of the resistive sensors. A support structure for mounting the sensor to a bar was part of the measurement setup. An Arduino setup combined with a Wheatstone bridge circuit was used to record the corresponding voltage changes when forces of (0, 20, 40, 50, 100, and 150) grams were applied successively. This gave us important information on the sensitivity and performance of the various track configurations and enabled us to precisely record the sensor's response to changing force levels. To apply a consistent and controlled

load during testing, calibrated weights (ranging from 0 g to 150 g) were carefully placed on the sensor surface in a point loading behavior using a small platform to ensure alignment and minimize bending or uneven distribution. To keep the sensors flat and unmoved during the experiment, they were positioned horizontally on a rigid bar and attached with spring clamps. With little external noise or error, this configuration made it possible to repeat loading at the same identity. These measures were taken to ensure the accuracy and consistency of the results across different tracks and thickness variations.

We collected and analyzed the resistivity plots for sensors with 2-tracks, 4-tracks, 6-tracks, and 8-tracks under applied forces of (0, 20, 40, 50, 100, and 150) grams. The 8-track design has the highest total resistance (18.2-20.0 kΩ and the smoothest slope with load, indicating good sensitivity and stability. The 6-track design has minimal fluctuations (10.0-10.4 kΩ, indicating high reliability. The 4-track design exhibits irregular behavior, particularly at higher loads, with significant drift above 50 g. The 2-track design shows the lowest fluctuation (5.9-6.0 kΩ, but also the lowest sensitivity to load changes. The plot below illustrates the resistance vs. load for different track designs, showing these performance trends. Table 6.1 presents the sensitivity for different track designs calculated from the resistance changes observed under the applied force range of 0-150 g. While the 4-track design shows high numerical sensitivity, its irregular behavior and drift make it unreliable for practical applications. The 8-track design offers 4.4 times higher sensitivity than the 6-track design, confirming that the 8-track configuration provides the optimal balance between sensitivity and stability for our application.

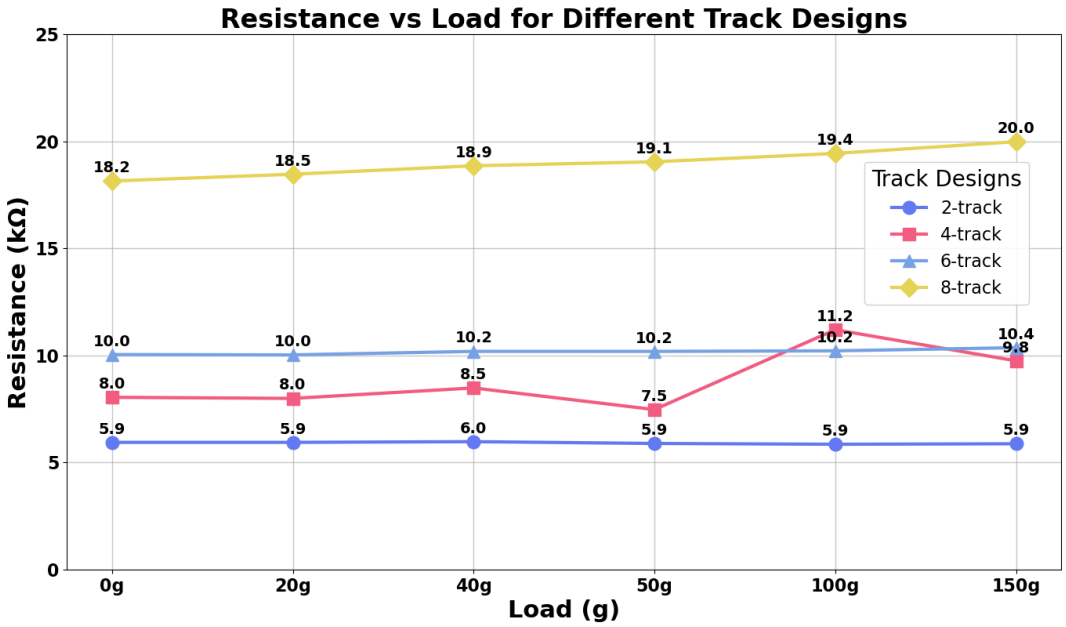


Figure 6.6: Plots of Tracks Sensors

Table 6.1: Sensitivity comparison between track configurations

Sensor Configuration	Resistance Range (kΩ)	Resistance Change (kΩ)	Sensitivity ($\Delta R/\Delta F$) (kΩ/g)
2-track	5.9–6.0	0.1	0.0007
4-track	7.5–11.2*	3.7*	0.0247*
6-track	10.0–10.4	0.4	0.0027
8-track	18.2–20.0	1.8	0.0120

*Note: The 4-track design exhibited irregular behavior with significant drift.

After selecting the 8-track sensor for its good performance, the next phase involved optimizing the sensor by adjusting the thickness of the conductive material to maintain consistent and accurate track performance. Sensors with different thicknesses were printed, and their performance was evaluated under applied forces of (0, 20, 40, 50, 60, and 100) grams. The plot below illustrates the performance of each sensor across these applied forces.

- Sensor I (0.2 mm) demonstrated high sensitivity but showed no change in resistance at 60 and 100 grams, indicating instability.
- Sensor II (0.4 mm) showed noticeable drift at no load and decreased response at 100 grams.

- Sensor III (0.6 mm) demonstrated the best performance with optimal sensitivity and stability, making it highly suitable for precise measurements.
- Sensor IV (0.8 mm) displayed good stability but had a limited range of resistance changes.

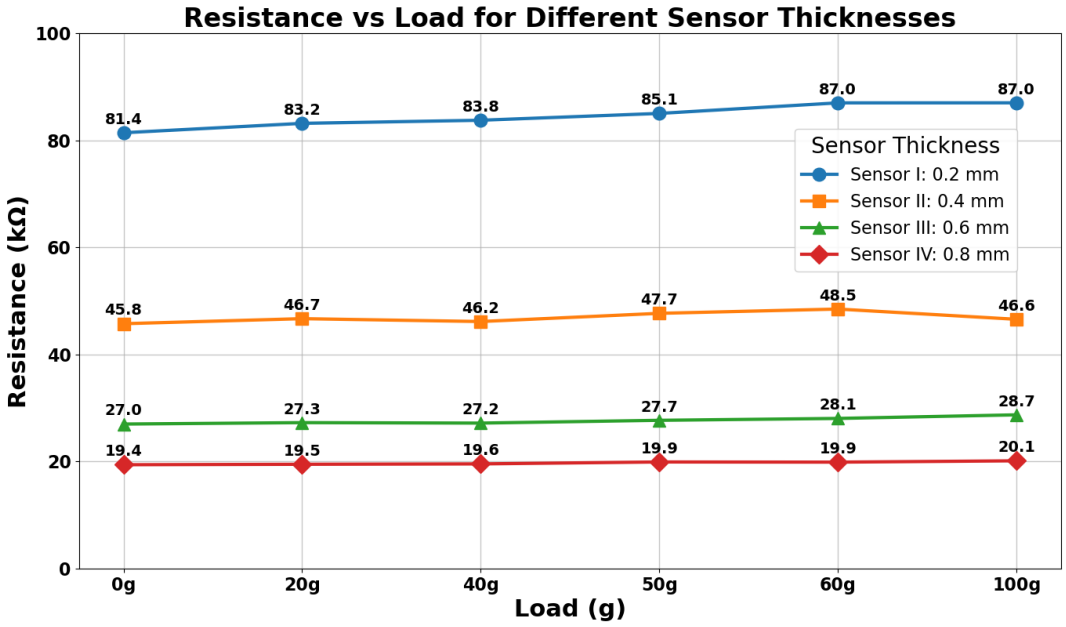


Figure 6.7: Plots of Resistance vs Load for different sensor thickness

Based on these evaluations, Sensor III (0.6 mm) was identified as the most effective configuration, offering a balanced amount of sensitivity and stability. It was therefore selected for further development and integration into soft finger robotics. The following table summarizes the performance of each sensor:

Table 6.2: Sensor response to applied loads

Metric	**Sensor I**	**Sensor II**	**Sensor III**	**Sensor IV**
Sensitivity to Load	High	Medium	High	High
Resistivity Range	Widest	Medium	Moderate	Narrowest
Signal Stability	Low	Low	High/Medium	High

With the successful optimization of the 8-track sensor with a thickness of 0.6 mm, the next step is the robotic finger and integration with sensors

6.8 Finger Evaluation

For the bending testing the TPU finger and the TPU/PLA base finger were mounted onto the machine and subjected to various mechanical tests using a UTM. These tests aimed to measure the finger's strength, flexibility, and durability while assessing the sensor's performance in real-world conditions. The sensor on the finger was connected to a microcontroller (Arduino) to collect real-time data, while the UTM machine simultaneously collected mechanical data on load, deformation, and stress.

The collected data from both the Arduino and UTM machines were imported into MATLAB for analysis. The datasets were synchronized to align force and resistance readings. Multiple fitting approaches were evaluated, and a 4th-degree polynomial was selected for its superior performance. The fitting was performed using MATLAB's polyfit() and polyval() functions. The general form of the polynomial model is:

$$R = a_4F_4 + a_3F_3 + a_2F_2 + a_1F + a_0$$

where R is the sensor resistance, F is the applied force, and through are the polynomial coefficients obtained through regression. The process involved cleaning and preprocessing the data to ensure accuracy, aligning the datasets for organized analysis, and applying polynomial fitting to model the relationship between the applied force and the sensor response. This comprehensive approach enabled a thorough evaluation of both the sensor's performance and the gripper finger's mechanical behavior.

The fingers were further tested in different positions—head, middle, and end—for both the fully TPU finger and the TPU/PLA base finger. This additional testing aimed to understand how the fingers operated and behaved in different conditions and force application points. We could evaluate the overall strength of the fingers under various stress points as well as the consistency and dependability of the sensor readings by assessing the fingers in these particular positions.

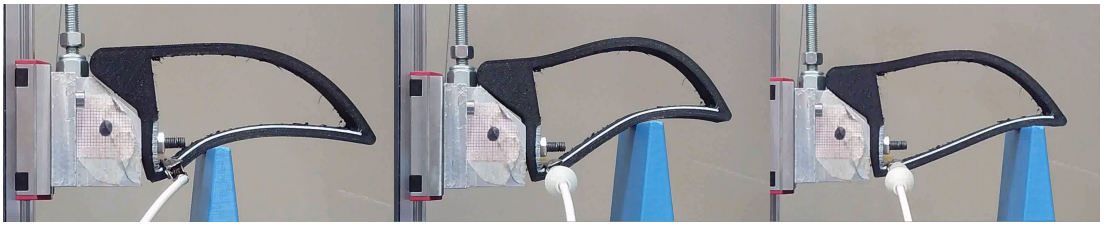


Figure 6.8: bending test for fully TPU finger

The plots illustrate the resistance changes of the fully TPU finger tested at the head, middle, and end positions. The blue dots represent experimental data, while the red line is a 4th-degree polynomial fit. At each position, the resistance increases smoothly with applied force, and the polynomial fit closely matches the experimental data.

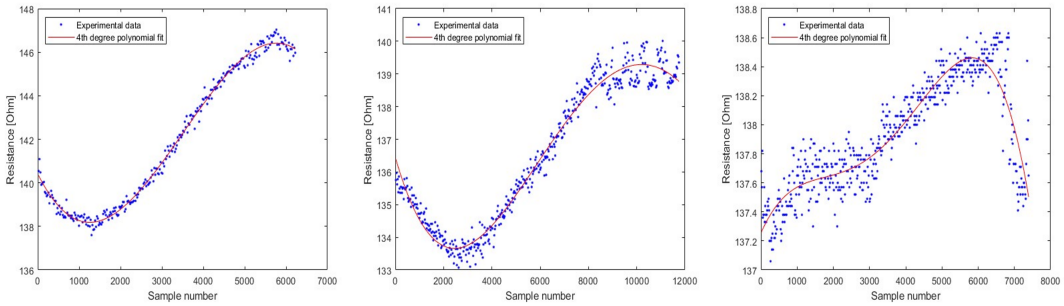


Figure 6.9: Plots of TPU finger

Figure 6.10: Bending test for fully TPU/PLA Base finger

The plots show the experimental data and the 4th-degree polynomial fit for the TPU/PLA base finger. The first plot represents the head position, the second plot represents the middle position, and the third plot represents the end position of the finger, illustrating the resistance changes in response to applied forces and demonstrating the sensor's performance across different stress points. A result of the differences, the fully TPU finger, and the TPU/PLA base finger behave differently mechanically and react differently when forces are applied. Because of its uniformity, and flexibility, the fully TPU finger shows more uniform flexibility and consistent sensor response across all tested positions—head, middle, and end—producing smooth resistance changes that closely resemble the 4th-degree polynomial fit.

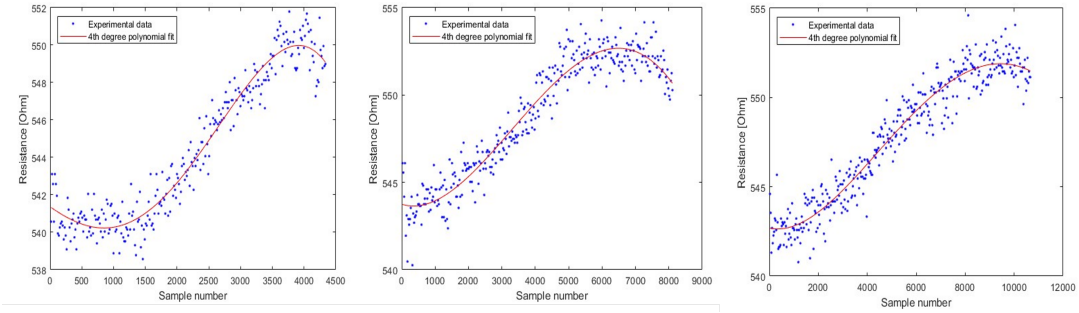


Figure 6.11: plots of TPU/ PLA Base Finger

The assembly of flexible TPU and rigid PLA in the TPU/PLA base finger, on the other hand, demonstrates more variation in its mechanical response. Because the stiffness of the PLA influences the finger's deformation and strain distribution, resulting in less uniform resistance changes compared to the fully TPU finger, this causes more position-dependent sensor readings. In the final analysis, the fully TPU finger provides more mechanical flexibility and consistent sensor performance than the TPU/PLA base finger, considering the former's increased structural rigidity.

6.9 Conclusion

This work successfully developed and tested flexible TPU sensors using FDM printing for soft robotic fingers, focusing on the limitations of typical, rigid sensors. The fully TPU finger sensor is perfect for tasks requiring precise feedback, like handling delicate objects, as it showed fewer fluctuations in experimental data and more consistent and predictable resistance changes. The TPU finger with a PLA base, on the other hand, was able to resist higher forces but displayed more variability in the plots and less consistency in resistance changes. The advantages and disadvantages of each design are compared in this comparison, which helps choose the right materials and configurations for various soft robotics applications.

This chapter presented a comprehensive approach to developing fully 3D-printable flexible resistive sensors for soft robotic applications. Through systematic optimization of track configurations and material thickness, we identified an 8-track, 0.6 mm thick sensor design offering optimal sensitivity (0.0120 kΩ/g) and stability. Integration into self-adaptable fingers demonstrated successful sensing capabilities in both open and closed configurations, with the fully TPU variant showing superior consistency in sensor response. The key contributions of this work include: Systematic evaluation framework for resistive sensor geometry op-

timization Accessible fabrication, Demonstration of complete sensor-finger fabrication using standard FDM technology, and Ready-to-use designs for soft robotic sensing applications. The resistance-force relationship following a fourth-degree polynomial enables predictable sensor behavior suitable for closed-loop control applications. While thermal drift remains a challenge for extended operation.

The approach highlights additive manufacturing as a scalable and accessible method for integrating sensing into compliant robotic systems. While resistive sensing is limited in spatial resolution and signal linearity, it provides a practical foundation for real-time feedback in industrial manipulation. The next chapter extends this work by exploring distributed tactile sensing through light-angle sensor arrays.

Chapter 7

Light-Angle Sensor Array Integration for Distributed Tactile Sensing in Self-Adaptable Fingers

7.1 Introduction

Resistive and capacitive sensors provide basic contact and deformation feedback, but their resolution and robustness remain insufficient for applications requiring detailed spatial information. Achieving more advanced tactile robust dexterous manipulation capabilities with soft, self-adaptable fingers requires tactile sensing that is compact, integrable, and sensitive to both static and dynamic contact phenomena.

This chapter reports the development and in-progress integration of *LiVec* Leslie *et al.* (2023a) light-angle tactile sensors into FinRay-inspired soft fingers. We first present the characterization of three pre-prototype LiVec sensors provided by University College Dublin (UCD), using a custom UTM rig to quantify 3D displacement/force response and repeatability. We then describe the design and fabrication of a four-layer rigid-flex PCB LiVec sensor array tailored to the soft finger geometry, together with a readout architecture and a silicone skin interface designed to preserve the soft fingers adaptability. Finally, we outline the test plans for array-level calibration, grasp sensing, and slip/vibration detection within our universal gripper platform. Results to date confirm promising linearity and repeatability on the demo sensors and demonstrate a manufacturable pathway toward dense, arrayed, non-camera optical tactile sensing within soft robotic fingers.

This chapter documents our path from *demo LiVec sensors* to a *custom array* integrated into a self-adaptable finger:

1. **Characterization of demo sensors (3 units):** custom UTM rig, Teensy-based readout, center/offset loading, and cyclic repeatability.
2. **Integration concept and design:** rigid-flex PCB array; readout electronics; silicone skin and housing.
3. **Implementation status & plan:** PCB array finalized and fabricated; finger/housing and firmware in progress; forthcoming array-level calibration and manipulation experiments.

7.2 Background

7.2.1 LiVec sensing principle

LiVec employs a black silicone pillar with a white reflector near the tip. Two IR LEDs illuminate the reflector; a light-angle sensor (Analog Devices ADPD2140) measures photon incidence angles and intensity. Pillar deformation (X/Y bending, Z compression) alters the reflector pose, modulating four photocurrents; multivariate regression maps these to 3D displacement/force. Two LEDs mitigate signal asymmetry and low SNR when the reflector moves away from a single emitter. Reported prototype precision is $\mathcal{O}(10\text{–}40)\,\mu\text{m}$ in displacement and $\mathcal{O}(30\text{–}70)\,\text{mN}$ in force, with sensitivity to vibrations up to $\sim 950\,\text{Hz}$ via pulsed illumination and ambient-light rejection Leslie *et al.* (2023a).

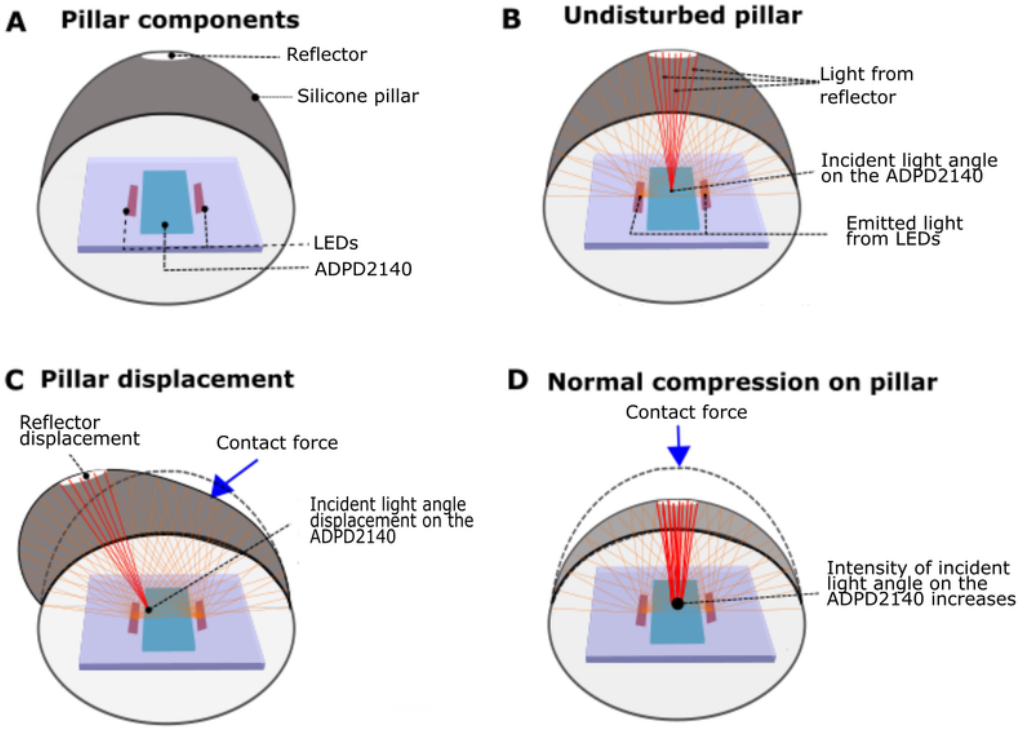


Figure 7.1: Sensing principle of the LiVec sensor. **A)** 3D cutaway view showing key components. **B)** In the neutral position, LEDs illuminate the pillar cavity. Light is either absorbed by black silicone or diffusely reflected by the disk-shaped reflector, with some scattered light reaching the ADPD2140 sensor. The average incident angle is 0° (normal to PCB); only select rays are shown for clarity. **C)** Lateral displacement of the pillar in the X direction alters the reflector position, changing the angle and intensity of light reaching the sensor. **D)** Z compression moves the reflector closer to the sensor, increasing light intensity—useful for inferring Z displacement.

ADPD2140 Angle Computation Principle

The ADPD2140 uses four photodiode channels (x_L, x_R, y_T, y_B) to compute normalized ratios that correspond to the horizontal and vertical light angles. The

basic equations are:

$$x = \frac{x_L - x_R}{x_L + x_R}, \tag{7.1}$$

$$y = \frac{y_T - y_B}{y_T + y_B}. \tag{7.2}$$

The resulting ratios are converted to angles using a proportionality constant M (angular slope):

$$\theta_x = \frac{x}{M}, \qquad \theta_y = \frac{y}{M}, \qquad M \approx 0.00631 \text{ (ratio/°)}. \tag{7.3}$$

Additionally, the overall light intensity is computed as:

$$L = x_L + x_R + y_T + y_B, \tag{7.4}$$

This sums all four photodiode currents to get the total incident light intensity. which in our application correlates with Z-axis displacement (compression of the silicone pillar).
Note: The proportionality constant $M \approx 0.00631$ and the linear mapping $\theta_x = \frac{x}{M}$, $\theta_y = \frac{y}{M}$ are valid only within the linear region of approximately $\pm 5°$ around the optical axis. Beyond this range, the sensor response becomes nonlinear, and accurate angle estimation up to the full $\pm 35°$ field of view requires a calibration curve or polynomial mapping. In the LiVec implementation, this limitation is addressed by using *multivariate regression models* trained on experimental data, which compensate for nonlinearities and allow accurate 3D displacement and force estimation across the entire operating range.

7.2.2 Demo Sensor Construction

The LiVec prototype sensor consists of a soft black silicone pillar with an internal hollow cavity housing a white silicone reflector. This reflector scatters infrared light emitted by two LEDs mounted on the printed circuit board (PCB). The PCB also hosts the ADPD2140 light-angle sensor and the ADPD1080 photometric front-end, which performs analog-to-digital conversion and LED control. The pillar is secured to the PCB using a 3D-printed top plate and four M2 screws, ensuring stable alignment during testing. A Teensy 4.1 microcontroller interfaces with the ADPD1080 via I^2C and streams data to a host PC over USB for real-time visualization and logging. This compact assembly enables precise measurement of 3D displacement and force through optical angle and intensity sensing.

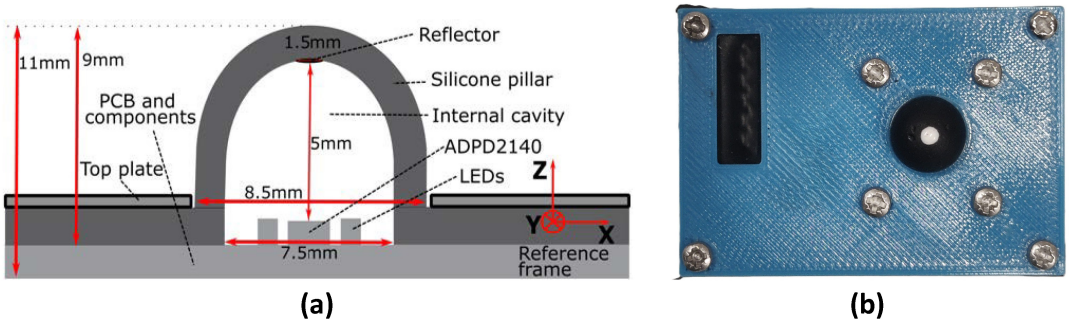


Figure 7.2: (a) Photograph of the LiVec sensor prototype. (b) Cross-sectional schematic illustrating key components: PCB with ADPD2140 light-angle sensor and two infrared LEDs, black silicone skin forming a pillar with an internal hollow cavity, embedded white silicone reflector, and a top plate securing the assembly. *(Image adapted from Leslie *et al.* (2023a)).*

7.2.3 Why LiVec for soft FinRay-inspired fingers

The FinRay effect depends on compliant structural coupling to conform around objects. Tactile instrumentation must therefore be thin, lightweight, conformable, and non-obstructive to passive adaptation. LiVec's compact height, PCB-integrated optoelectronics, and scalability to arrays align with these constraints and provide high temporal bandwidth for slip/vibration.

7.3 Part I — Demo-Sensor Characterization

7.3.1 Sensors and electronics

Three UCD LiVec prototypes included a silicone pillar/reflector assembly on a PCB with ADPD2140, ADPD1080 (14-bit ADC, LED control), and two IR LEDs; a Teensy 4.1 microcontroller acquired data and streamed to a PC GUI as shown in Figure 7.3.

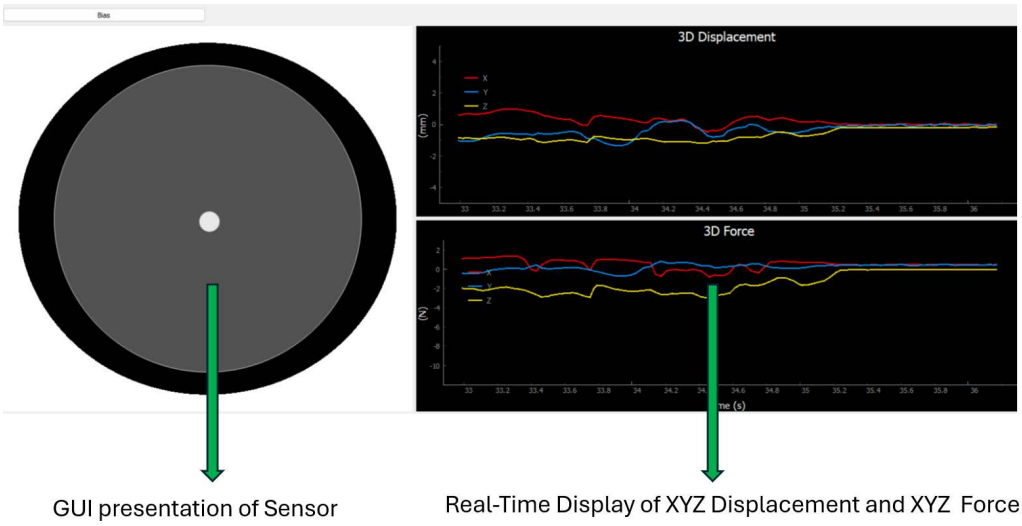


Figure 7.3: Graphical interface for demo sensor

7.3.2 UTM rig and protocol

A custom fixture for a Zwick/Roell UTM with a rigid 4 mm needle probe was designed to enable clean Z-axis load application. An XY plate targeted center and ±2.5 mm offset points. Tests: center Z-loading, offsets, and 5 N cyclic loading. TestXpert III logged UTM force/displacement.

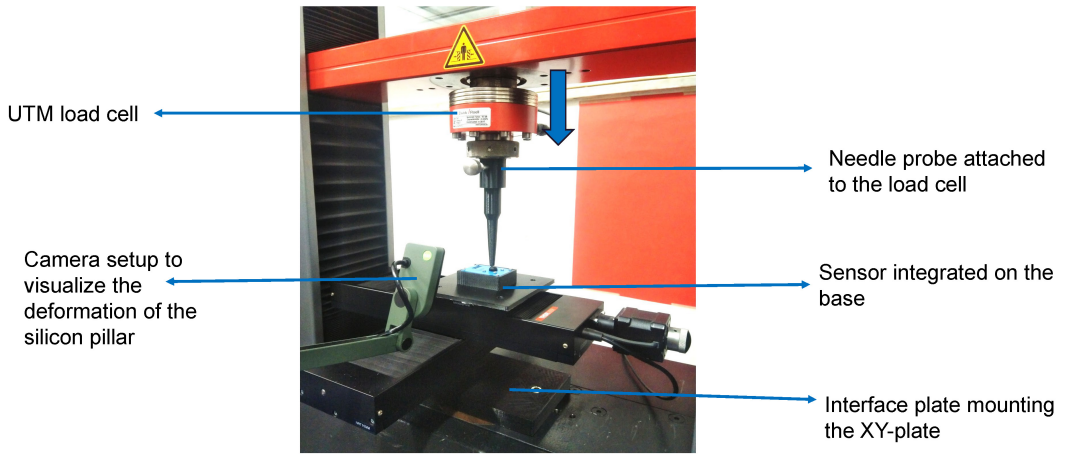


Figure 7.4: UTM-based test rig

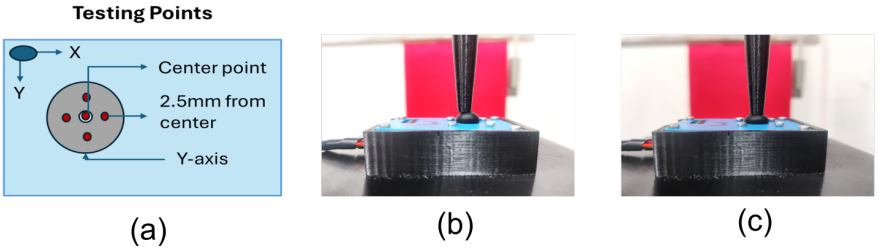


Figure 7.5: (a) Testing points on the sensor pillar. (b) Rigid needle compressing at 2.5 mm from the center. (c) Rigid needle compressing at center.

Table 7.1: LiVec sensors: Sensing ranges and resolutions

		LiVec #1			**LiVec #2**			**LiVec #3**		
		X	Y	Z	X	Y	Z	X	Y	Z
Disp. (mm)	Range	± 1	± 1	-2.4	± 1	± 1	-2.4	± 1	± 1	-2.4
	Res.	0.04	0.03	0.02	0.04	0.04	0.03	0.04	0.03	0.02
Force (N)	Range	± 1	± 1	-8	± 1	± 1	-8.5	± 1	± 1	-10
	Res.	0.03	0.04	0.09	0.04	0.04	0.30	0.05	0.08	0.04

7.3.3 Performance under Representative Loading Conditions

Only the results for Sensor #1 are presented here, as the other two sensors exhibited similar behavior and remained within their specified sensing ranges and resolutions (Table 7.1). The selected plots illustrate three representative cases: 6 N applied at the center, 5 N applied with a 2.5 mm lateral offset, and cyclic loading up to 5 N at the center. These tests were performed using the UTM setup, with acquisition of ground-truth force and displacement.

Center Loading (6 N). The sensor exhibits consistent and repeatable estimates of D_X, D_Y, and D_Z across three iterations, with minimal drift and low inter-trial variability. The corresponding force estimates (F_X, F_Y, F_Z) follow the expected trend, with F_Z dominating under axial compression and lateral components remaining near zero, confirming proper decoupling of axes.

Offset Loading (6 N at 2.5 mm). When the load is applied off-center, the sensor correctly detects combined axial and lateral deformation, reflected in D_X/D_Y

components and corresponding force estimates. This demonstrates the sensor's ability to capture multi-axis interactions, which is critical for grasping scenarios involving asymmetric contact.

Cyclic Loading (0–5 N). Under repeated loading cycles, the sensor output tracks the UTM reference with negligible hysteresis and stable baseline, indicating good repeatability and robustness against drift. The displacement and force curves maintain shape fidelity across cycles, supporting the suitability of the regression model for dynamic tasks.

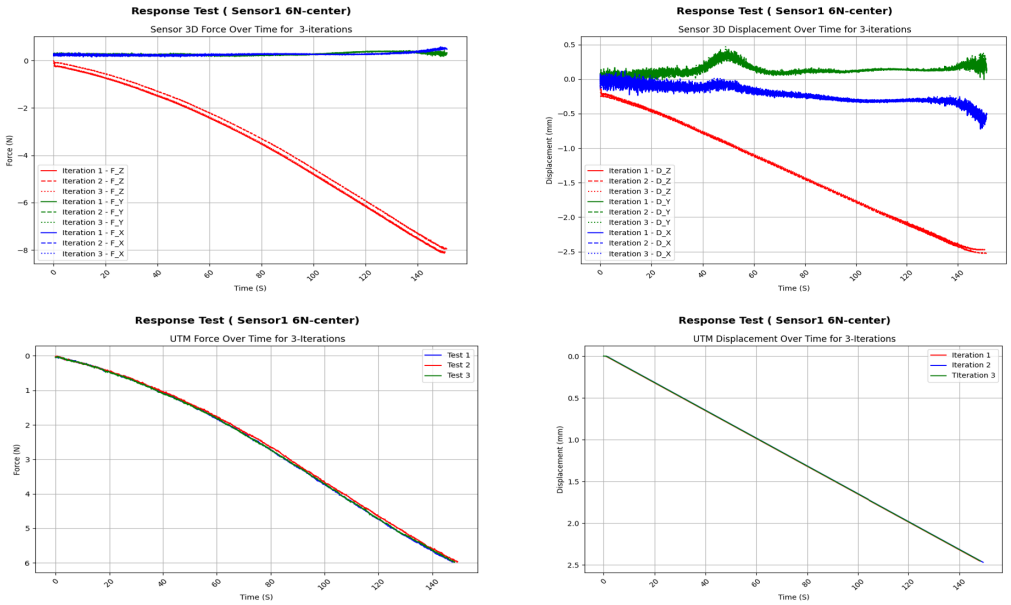


Figure 7.6: LiVec Sensor #1 response under 6 N center loading: estimated 3D displacement and force compared with UTM reference.

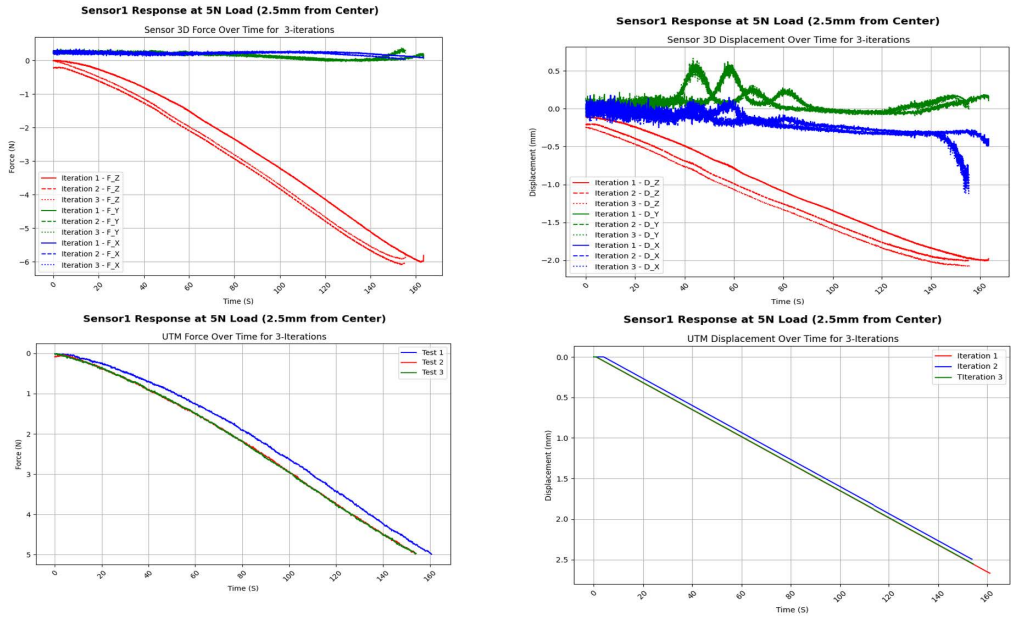


Figure 7.7: LiVec Sensor #1 response under 5 N loading at 2.5 mm lateral offset: estimated 3D displacement and force compared with UTM reference.

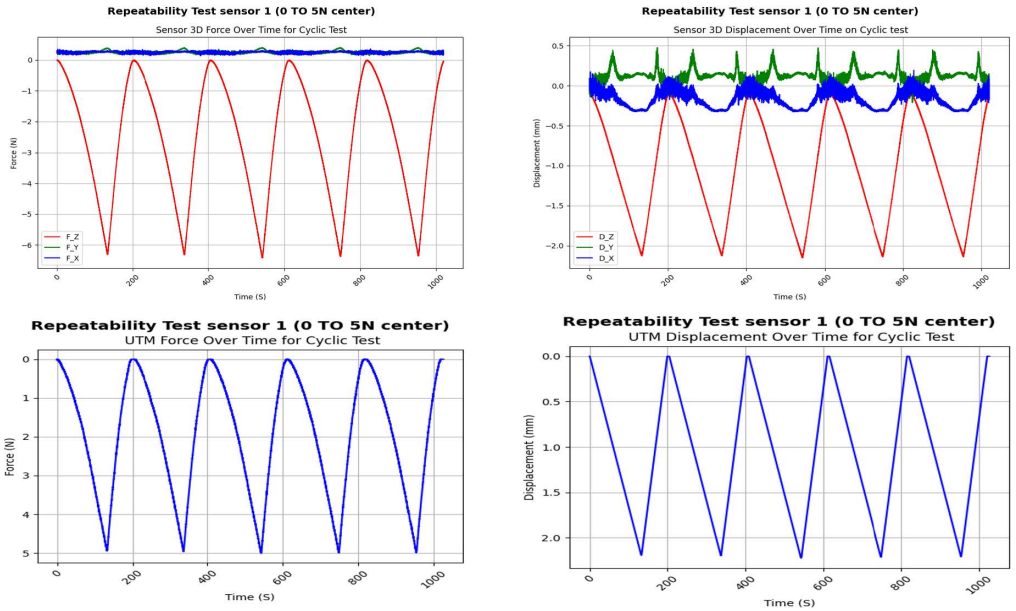


Figure 7.8: LiVec Sensor #1 response under cyclic loading (0–5 N): estimated 3D displacement and force compared with UTM reference.

Overall, Sensor #1 operates well within its specified sensing range (Table 7.1)

and achieves sub-0.1 N force resolution and sub-0.05 mm displacement resolution under practical loading conditions.

7.4 Part II —Integration of Rigid-Flex PCB into Soft Finger

To enable tactile sensing within the soft finger structure, a custom rigid-flex PCB was designed and fabricated, with the initial design of the PCB coming from UCD, Ireland. The rigid sections host the ADPD2140 light-angle sensors and IR LEDs, while the flexible interconnects allow the PCB to conform to the finger's curvature without inducing stiffness. This design ensures that the sensing elements remain aligned while preserving the adaptive mechanics of the self-adaptable finger.

7.4.1 PCB Design and Fabrication

Figure 7.9 shows the KiCad layout of the rigid-flex PCB, including top and bottom views. The design integrates multiple LiVec sensing units, each consisting of an ADPD2140 sensor and two IR LEDs, connected via flexible polyimide traces to minimize bending stress.

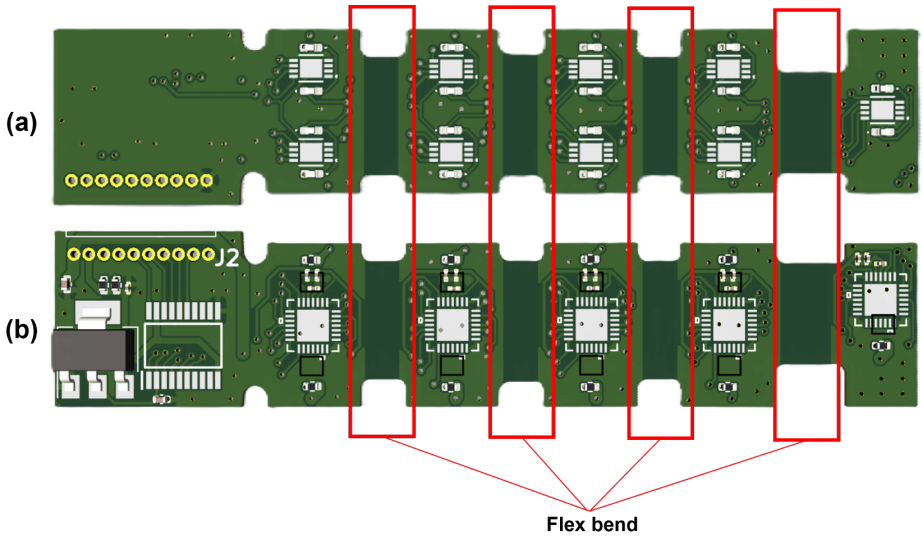


Figure 7.9: KiCad design of the rigid-flex PCB for LiVec sensor array: (a) top view and (b) bottom view. The rigid sections host sensors and LEDs, while the flex sections provide compliance for embedding into the soft finger.

The fabricated PCB is shown in Figure 7.10, illustrating the top and bottom

sides. The rigid islands contain the optoelectronic components, and the flexible interconnects enable folding and routing through the finger's internal channels.

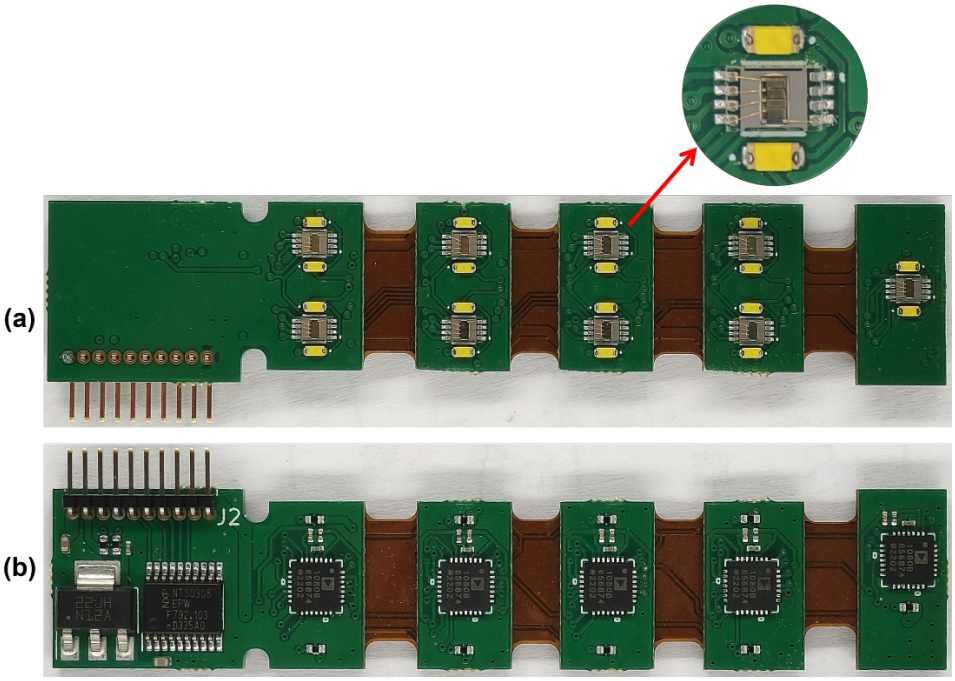


Figure 7.10: Fabricated rigid-flex PCB: (a) top side with ADPD2140 sensors and LEDs with a zoomed view of one unit, (b) bottom side showing flex interconnects.

7.4.2 Flexibility and Integration

To demonstrate mechanical compliance, the PCB was bent to check if it could match the curvature of the FinRay finger, as shown in Figure 7.11. The flex sections allow repeated bending without damage, ensuring durability during finger actuation.

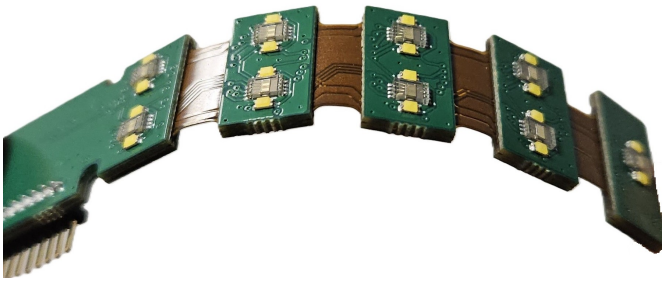


Figure 7.11: Rigid-flex PCB in bent configuration, demonstrating compliance for integration into soft FinRay fingers.

Finally, Figure 7.12 illustrates the integration of the rigid-flex PCB into a soft FinRay finger. The PCB is embedded along the inner surface of the finger, enabling multi-axis tactile sensing without compromising the finger's passive adaptability. The silicone skin (Mold Star 20A) provides a reflective cavity and external contact interface; the housing registers the skin over each island while minimizing added stiffness. Array positions align with high-probability contact zones; flex-region bend radii meet copper/polyimide limits. Mechanical interference between silicon pillars was also considered during the finalization of the array layout.

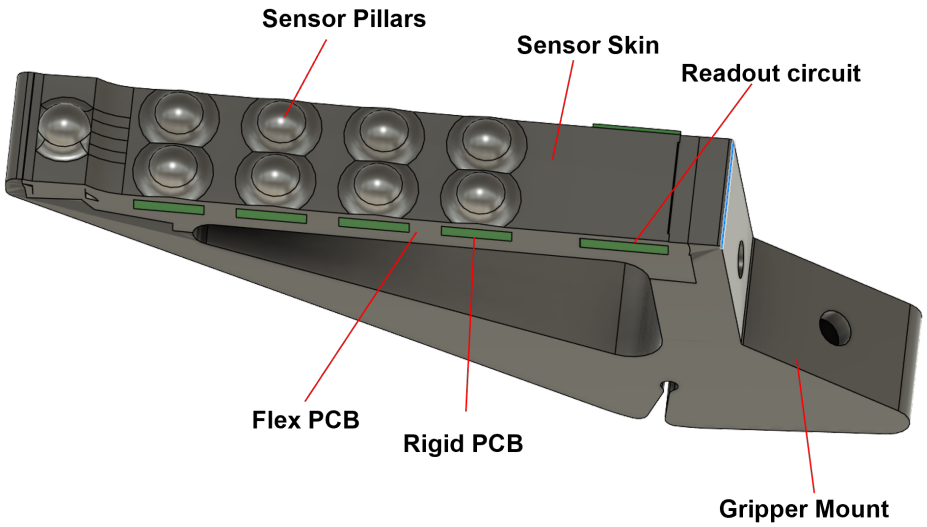


Figure 7.12: Soft FinRay finger with integrated rigid-flex PCB and LiVec sensors with readout, enabling multi-axis tactile sensing while preserving structural compliance.

7.5 Part III — Electronics, Firmware, and Data Pipeline

7.5.1 Readout Architecture

Each LiVec element comprises an ADPD2140 light-angle sensor with two IR LEDs, read by an ADPD1080 photometric front end. Multiple front ends are addressed over I^2C to a microcontroller (Teensy 4.1 in our prototype). The readout supports LED pulsing with synchronous demodulation to improve SNR and ambient-light rejection. The rigid islands on the PCB host the optoelectronics, while polyimide flex interconnects route I^2C, LED drive, and power along the

finger (see 7.10.

7.5.2 Timing and Sampling

We adopt a time-slotted acquisition across sensor elements. Within each slot, LEDs are pulsed and the ADPD1080 samples the four ADPD2140 channels (x_L, x_R, y_T, y_B); a dark (LED-off) sample may be inserted for ambient subtraction. On the MCU, samples are timestamped, minimally filtered (median or moving-average if needed), then streamed via USB to a host for logging and visualization. The GUI refresh runs at human-friendly rates (e.g., 25–60 Hz), whereas acquisition can run substantially faster; data logging remains decoupled from plotting to avoid I/O back-pressure.

7.5.3 Host Software and Visualization

A Python GUI (PyQt/pyqtgraph) renders time-series of D_X, D_Y, D_Z and F_X, F_Y, F_Z and a live 2-D contact map in the (D_X, D_Y) plane; intensity L scales the marker size. A bias-capture button computes and subtracts steady offsets online (see Algorithm 6). Data are buffered to CSV asynchronously to prevent UI stalls. The GUI currently supports individual sensors and will be extended to handle the full array, with data displayed using separate threads.

Algorithm 6 Online Bias Capture and Subtraction (per channel)

Require:
$\mathbf{s}(t)$: Streamed sensor data $(D_X, D_Y, D_Z, F_X, F_Y, F_Z)$
N: Window size for bias computation
`bias_active`: Boolean flag indicating if bias subtraction is enabled
function UpdateSensorStream($\mathbf{s}(t)$)
    if Bias button is pressed **then**
        Accumulate last N samples for each channel
        Compute mean $\boldsymbol{\mu}$ over window
        Store $\boldsymbol{\mu}$ as bias vector
        `bias_active` $\leftarrow$ `True`
        Log("Bias captured and activated")
    end if
    if `bias_active` **then**
        $\tilde{\mathbf{s}}(t) \leftarrow \mathbf{s}(t) - \boldsymbol{\mu}$ ▷ Subtract bias
        Output $\tilde{\mathbf{s}}(t)$
    else
        Output raw $\mathbf{s}(t)$
    end if
end function

7.6 Part IV — Calibration and Modeling

7.6.1 Planned Calibration Steps

The following steps outline the calibration and modeling procedure, which is currently in progress. These steps ensure accurate mapping from raw photocurrent signals to displacement and force estimates across the full operating range:

1. **Warm-up and ambient check:** Acquire LED-off baselines to verify dark current stability and ambient-light rejection.
2. **Baseline intensity:** Collect $\bar{L}$ over an initial window ($N \approx 200$ samples) to normalize subsequent readings.
3. **Planar contact tests:** Apply center and offset indentations along X and Y axes, stepped to peak loads within safe limits.
4. **Cyclic loading:** Perform multi-cycle loading to assess repeatability and drift characteristics.
5. **Model fitting and validation:** Fit multivariate polynomial regressors for (D_X, D_Y, D_Z) and (F_X, F_Y, F_Z) using normalized features, then validate on held-out data.

7.6.2 Feature Normalization and Angle Ratios

For each sample, normalized features are computed as:

$$x = \frac{x_L - x_R}{x_L + x_R}, \quad y = \frac{y_T - y_B}{y_T + y_B}, \quad L = x_L + x_R + y_T + y_B.$$

x and y approximate horizontal and vertical light angles only within a small linear region, while L correlates with axial compression. Normalizing channel values by $\bar{L}$ mitigates LED intensity variations and slow drifts.

7.6.3 Regression Models

Six independent polynomial models are planned:

$$(x_1, x_2, x_3, x_4, x_5) = \left(\frac{x_L}{\bar{L}}, \frac{x_R}{\bar{L}}, \frac{y_T}{\bar{L}}, \frac{y_B}{\bar{L}}, \frac{L}{\bar{L}}\right) \mapsto (D_X, D_Y, D_Z, F_X, F_Y, F_Z).$$

Model orders (3rd–4th) will be selected based on residual analysis to balance accuracy and complexity. Coefficients will be embedded in firmware or host software for real-time inference.

7.6.4 Quality Metrics

Performance will be evaluated using:

- Root-mean-square error (RMSE) and mean absolute error (MAE) against UTM references.
- Repeatability across cycles and offsets.
- Baseline stability and drift over time.

7.7 Discussion

7.7.1 Integration Benefits

The rigid-flex PCB approach enables embedding LiVec sensors into soft fingers without compromising compliance. Compared to camera-based tactile sensors, this solution offers:

- High temporal bandwidth for slip and vibration detection.
- Scalable architecture with low per-element compute cost.

7.7.2 Current Limitations

- **Nonlinearity:** Raw angle ratios are linear only within $\pm 5°$; regression mitigates this but requires dense calibration.
- **High-compression saturation:** High Z compression may saturate optical response; mechanical design and model clipping are needed.
- **Generalization:** Models trained on planar contacts may underperform on complex geometries without additional data.

7.8 Conclusion

We presented the design and integration of LiVec light-angle sensing into soft fingers using a custom rigid-flex PCB. The approach preserves finger compliance while enabling multi-axis tactile sensing. Calibration and modeling steps are underway to finalize displacement and force estimation across the full operating range. This architecture is compact, arrayable, and suitable for high-rate acquisition, making it promising for slip-aware control and adaptive grasping in industrial grippers.

Future Work

Future research will focus on the following identified directions:

1. **Sensor Calibration and Modeling:** Complete the calibration of all integrated sensors and validate regression models under diverse loading conditions. This includes refining the mapping between optical signals and force/angle estimates to improve accuracy and robustness.

2. **Optimization of Silicone Pillars:** Conduct a systematic study of the mechanical and optical behavior of silicone pillars under different loading conditions. Design parameters such as geometry, material hardness, and reflector placement will be optimized to minimize variability and enable generalizable regression models across multiple units.

3. **Finalization of Integrated Finger Prototype:** Complete the fabrication and assembly of the soft finger with the embedded rigid-flex PCB, ensuring robust mechanical integration, reliable electrical connections, and repeatable sensor performance under realistic grasping scenarios.

4. **Enhanced Interaction Scenarios:** Extend the training dataset to include non-planar contacts and dynamic interactions. Evaluate the system's ability to detect slip and vibration during realistic manipulation tasks, enabling more adaptive and responsive control strategies.

5. **Electronics and Layout Improvements:** Optimize the PCB layout for higher sensor density and improved readout solutions, reducing noise and latency while maintaining compact form factors.

6. **Closed-Loop Control Integration:** Incorporate array-level sensing into the universal gripper's closed-loop control framework. This will enable real-time adaptation of grasp strategies based on tactile feedback, improving performance in unstructured environments.

Chapter 8

Real-Time Path Correction System for Industrial Robot Coordination in Human-Collaborative Automotive Assembly

8.1 Introduction

This chapter presents the development and implementation of a real-time robot control system for the KUKA KR 150 R3100-2 industrial manipulator, developed within the context of the EU Horizon project SESTOSENSO. The work specifically addresses the automotive roof assembly task in confined cockpits, a scenario characterized by occlusions, restricted workspaces, and high collision risk.

The solution leverages the Robot Sensor Interface (RSI) protocol to enable online trajectory corrections, introduces velocity-based control strategies, enforces interrupt-based stops and integrates with a UR10e collaborative robot mounted on the KUKA flange. The UR10e, equipped with time-of-flight (ToF) sensors and perception capabilities, contributes distributed proximity sensing. The two robots are coordinated through ROS2 middleware, while human operators remain actively in the loop.

My contribution focused on the KUKA side of the system: developing the RSI-based Python control client, velocity regulation, safety stop protocols, and operator GUI. This ensured the KR150 could adaptively coordinate with the UR10e and human operator, validating SESTOSENSO's vision of distributed sensory systems for safe and efficient human–robot cooperative assembly.

8.1.1 Motivation and Industrial Context

Modern automotive assembly operations increasingly require flexible automation solutions that can operate safely alongside human workers while maintaining productivity. The assembly of interior components, particularly roof panels within vehicle cockpits, presents significant challenges: workers must maintain awkward postures in confined spaces, coordinate with robotic systems that provide positioning assistance, and avoid collisions in visually occluded environments. Traditional safety systems based on external monitoring (safety scanners, light curtains) prove inadequate in such confined spaces where line-of-sight is frequently obstructed.

The SESTOSENSO project addresses these challenges through distributed proximity and tactile sensing (ProxySKIN technology) embedded directly on robotic systems. This chapter focuses on the control architecture developed for the primary manipulator—a KUKA KR 150 R3100-2—which supports a UR10e collaborative robot equipped with time-of-flight (ToF) proximity sensors. The system must coordinate both robots, respond to obstacle detection in real-time, and provide intuitive operator control while maintaining industrial safety standards.

8.1.2 Technical Objectives

The control system development targeted the following technical objectives:

1. **Real-time path correction**: Implement RSI-based corrections to pre-programmed robot trajectories with real-time response times
2. **Dual-mode control**: Support both absolute target positioning and relative correction modes for different operational phases
3. **Velocity control**: Implement proportional control with adjustable gains and saturation limits to ensure smooth movement
4. **Emergency response**: Develop interrupt-driven safety mechanisms triggered by proximity sensor detection of obstacles
5. **Intuitive interface**: Develop a graphical user interface supporting manual corrections, routine scheduling, logging, and system monitoring for KUKA robot
6. **Middleware integration**: Enable coordination with the UR10e through ROS2 communication architecture
7. Validate the system in authentic automotive assembly scenarios with human workers present

8.1.3 Application Context: Automotive Roof Interior Assembly

The automotive assembly process involves the installation of a soft roof interior component into a Maserati Levante (body-in-white). This operation consists of six distinct phases:

1. **Roof Pickup**: The KUKA robot, with UR10e cobot mounted on its flange, moves to an overhead storage structure where soft roof is staged. The KUKA's vacuum gripper engages the roof component.

2. **Roof Transport**: The KUKA transports the gripped roof from the storage structure toward the vehicle's windscreen area, positioning for insertion.

3. **Roof Insertion**: The KUKA slowly inserts the roof through the windscreen opening into the vehicle interior, moving to a predetermined insertion position inside the car.

4. **Worker Verification and Clipping**: With the roof held in position by the KUKA, a human worker enters the vehicle to verify correct positioning. The RSI correction system enables real-time fine adjustments based on worker feedback or sensor data. Once verified, the worker clips the roof to temporary mounting points to prevent it from falling.

5. **Tool Change**: After the roof is clipped, the KUKA releases the vacuum gripper and exits the vehicle. The robot moves to a tool changer station where the UR10e's end-effector is automatically changed from a gripper to a screwdriver-like fastening tool.

6. **Fastening Assistance**: The KUKA re-enters the vehicle with the screwdriver tool, and the UR10e assists the worker in fully fastening the roof to its permanent mounting points. After completion, the KUKA returns to the tool changer to switch back to the vacuum gripper and returns to the roof storage structure, ready for the next assembly cycle.

The critical challenge addressed by this work is Phase 4, where the robot must hold a precise position inside the confined vehicle interior while performing real-time corrections for position corrections and obstacle avoidance with Human-in-the-loop (HITL). Traditional fixed-trajectory programming cannot handle this variability.

8.1.4 Technology Selection: RSI vs Ethernet-KRL

KUKA provides two primary interfaces for external communication with the KRC5 controller: Ethernet-KRL (E-KRL) and RSI. The selection between these technologies significantly impacts system capabilities, complexity, and performance.

8.1.4.1 Ethernet-KRL Overview

Ethernet-KRL extends the KRL programming language with TCP/IP socket communication capabilities, enabling the robot program to exchange data with external systems. Key characteristics include:

- **Communication paradigm**: Message-based, asynchronous socket communication (client or server mode)
- **Cycle time**: Variable, typically 50–200 ms depending on network conditions and message complexity
- **Programming model**: Messages handled within KRL program logic using blocking or non-blocking socket operations
- **Data format**: Flexible (text, binary, custom protocols)
- **Control authority**: Low-level - external system sends high-level commands (e.g., "move to position X"), but trajectory generation and execution remain in KRL

8.1.4.2 Robot Sensor Interface Overview

RSI provides deterministic, low-latency coupling between external sensors or control systems and the robot's interpolation cycle. Key characteristics include:

- **Communication paradigm**: Synchronous, cyclic exchange tightly coupled to robot interpolator
- **Cycle time**: Deterministic cycle time depending on the sensor mode. **#IPO** sensor mode: signal processing at sensor cycle rate of 12 ms or **#IPO_FAST** sensor mode: signal processing at sensor cycle rate of 4 ms.
- **Programming model**: KRL activates RSI mode; external system sends corrections every cycle via XML over UDP
- **Data format**: Standardized XML schema defined by KUKA
- **Control authority**: High-level - external system directly modifies Cartesian or joint space trajectories in real-time

8.1.4.3 Decision Rationale

RSI was selected for due to three critical requirements:

1. **Latency**: The 12 ms deterministic cycle provides responsive corrections. Ethernet-KRL's 50–200 ms latency would create noticeable lag where the robot continues moving after correction commands are issued.

2. **Smoothness**: RSI applies corrections incrementally at the interpolation rate (83 Hz), resulting in smooth motion. Ethernet-KRL's variable timing causes discrete position jumps and visible jerkiness.

3. **Direct trajectory control**: RSI modifies the robot's path every cycle, enabling fine adjustments of ±1–2 mm. Ethernet-KRL can only send target positions, relying on the robot's internal motion planner which cannot achieve comparable precision.

8.1.4.4 Selected Architecture

A **hybrid architecture** leverages the strengths of multiple technologies:

- **RSI**: Real-time trajectory corrections and targets during Phase 4 (12 ms cycle)
- **KRL**: Traditional programming for Phases 1, 2, 3, 5, 6, 7 with ability for small corrections during waypoint trajectory motion using RSI
- **ROS2**: Multi-robot coordination and sensor integration

This approach uses RSI where its unique real-time capabilities are essential while relying on proven KRL programming for standard motion sequences. Pure RSI implementation would require reimplementing safety-certified motion planning, collision avoidance, and singularity handling - negating the benefits of using an industrial robot controller. Pure Ethernet-KRL would result in unacceptable latency and motion quality during operator-guided corrections.

8.1.5 System Architecture Overview

The implemented system employs a split-control architecture where the KUKA manipulator and UR10e cobot maintain independent control stacks coordinated through a centralized task management layer. This approach was selected over unified multi-robot control for several reasons:

- **Safety preservation**: Native safety features of each robot remain fully functional
- **Reliability**: Proprietary real-time interfaces provide deterministic performance
- **Modularity**: Independent control stacks simplify development, testing, and maintenance
- **Industrial compliance**: Maintains compatibility with manufacturer-approved control methods

The KUKA control architecture comprises three primary layers:

1. **Low-level real-time control**: RSI interface running on the KRC5 controller (12 ms cycle time, 83 Hz)
2. **Mid-level coordination**: Python client managing RSI communication, control, and safety
3. **High-level supervision**: GUI application providing interface for manual corrections, routine scheduling, logging, and monitoring

The Overall system architecture is shown in 8.1. The architecture deliberately separates concerns:

- **KUKA controller**: Executes programmed motion sequences, manages safety systems, applies RSI corrections
- **Python RSI client**: Computes corrections based on operator commands or sensor inputs, maintains 83 Hz communication with KUKA
- **ROS2 nodes**: Coordinate KUKA and UR10e behaviors, distribute proximity sensor data, manage high-level task sequencing
- **UR control box**: Controls UR10e cobot, manages tool changing, interfaces with proximity sensors

- **GUI**: Translates intent into RSI commands, displays system state, enables manual corrections

The Python Control Client implements the RSI communication loop with the KUKA KRC5 controller, operating at 83 Hz. While this client can be integrated into a ROS2 node for multi-robot coordination, it is conceptually distinguished in the system architecture because of its cycle-critical, real-time communication role with the robot controller.

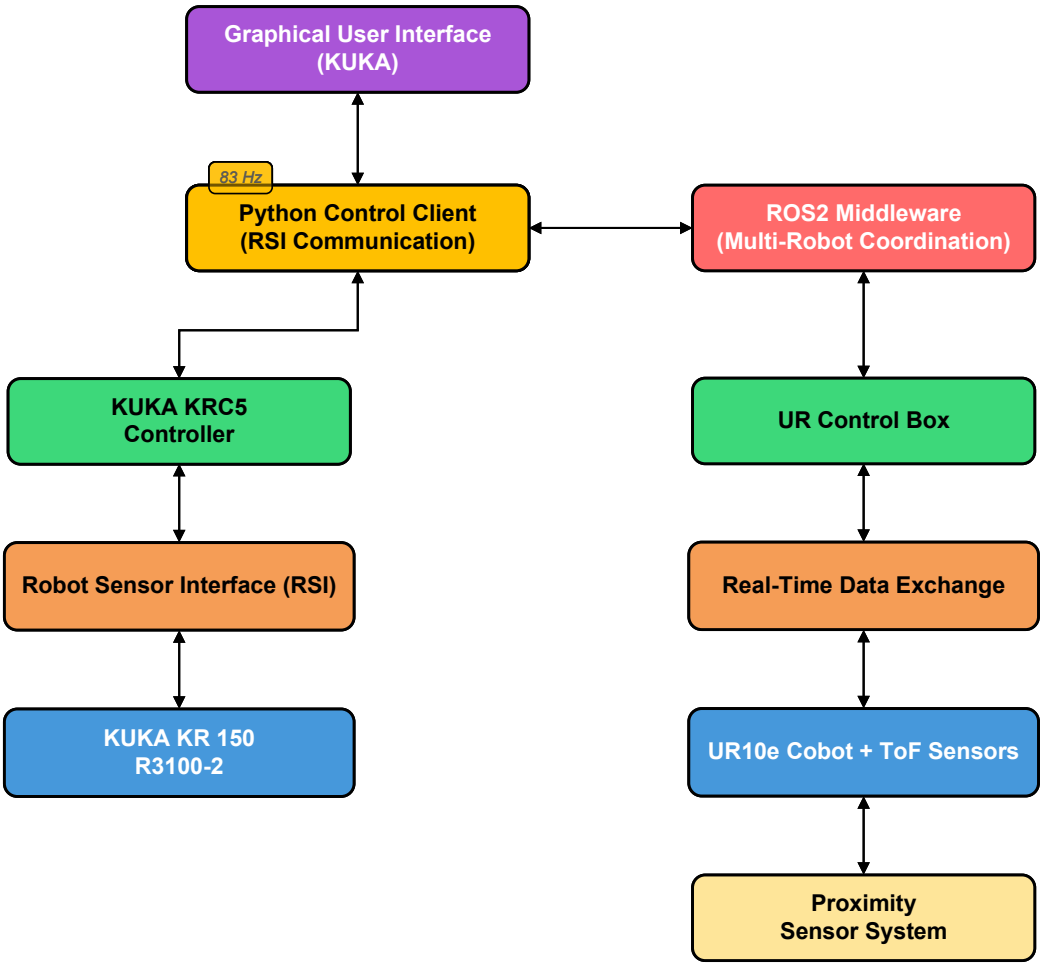


Figure 8.1: Overall system architecture showing the hierarchical control structure with communication frequencies. The split-control approach maintains independent robot controllers coordinated through ROS2 middleware.

8.2 RSI Protocol and Configuration

8.2.1 RSI Protocol

The KUKA Robot Sensor Interface enables external systems to modify robot motion in real-time by providing correction values superimposed on the commanded trajectory. The real-time sensor processing logic is developed using the graphical editor RSIVisual within the KUKA WorkVisual engineering suite. This process involves creating RSI contexts, which are configurations of interconnected RSI objects to define the required signal flow and correctional algorithms.

The RSI Ethernet object supports:

- **Cyclical transmission (Controller → External System)**: Position data, axis angles, operating mode, step counter, and custom signals transmitted parallel to program execution
- **Cyclical transmission (External System → Controller)**: Sensor data, correction commands, and control signals received parallel to program execution
- **Signal capacity**: Up to 64 inputs and 64 outputs can be defined for the ETHERNET RSI object. Input signals are sent to the external system; data received from the external system are available at the output signals within the KRL program.

Data exchange via Ethernet is implemented using the RSI object `ETHERNET` defined in the robot controller's configuration. This object handles UDP socket communication, XML parsing/generation, and synchronization with the interpolation cycle, abstracting low-level networking details from the KRL programmer. The functional mechanism of data exchange through Ethernet is shown in Figure 8.2.

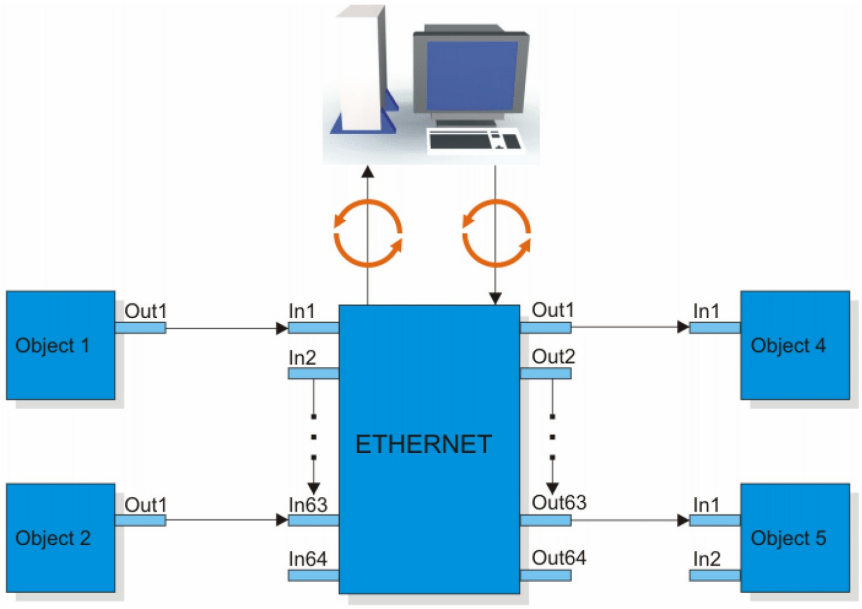


Figure 8.2: Ethernet data transmission and reception functional mechanism

The KRC5 controller executes KRL (KUKA Robot Language) programs defining nominal waypoint sequences while simultaneously receiving correction commands via UDP/IP communication at 83 Hz (12 ms interpolation cycle).
Each RSI cycle involves the following operations, as illustrated in Figure 8.3:

1. **Controller transmission**: Current robot state transmitted as XML packet
2. **External computation**: Client calculates corrections based on sensor data and control objectives
3. **Controller reception**: Correction values received and applied to motion interpolation
4. **Motion execution**: Corrected trajectory executed with built-in safety monitoring

The client must complete steps 2–5 within approximately 10 ms to ensure the response arrives before the next cycle begins. This tight timing constraint necessitates efficient computation and precludes complex algorithms requiring iterative optimization.

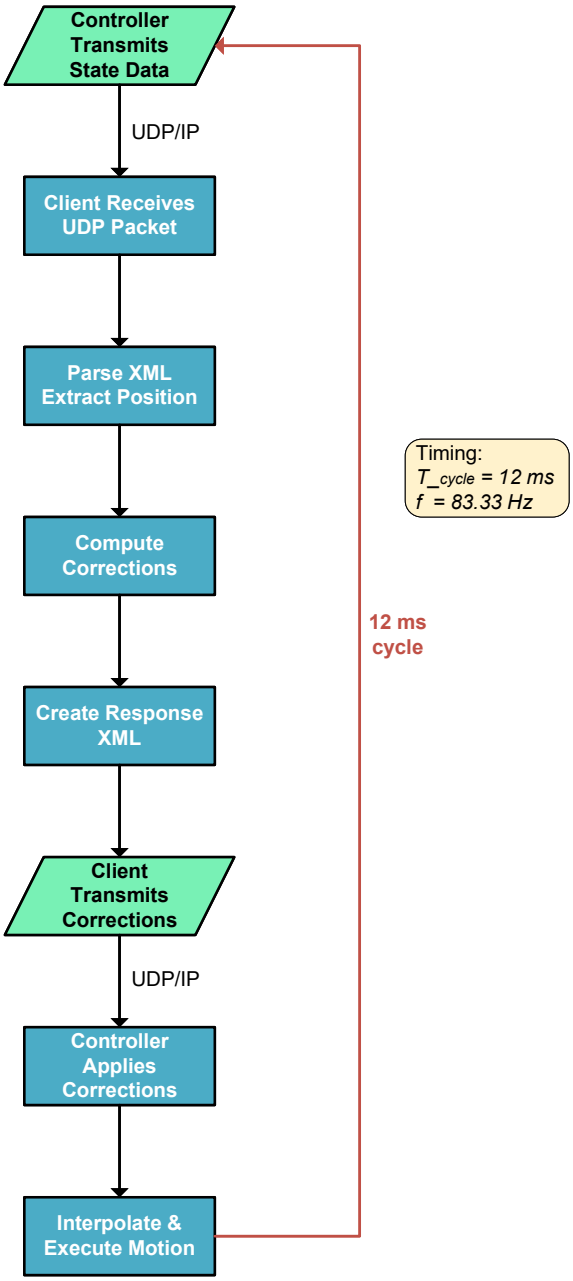


Figure 8.3: RSI communication cycle showing the 12 ms synchronous exchange between KUKA controller and Python client. This cycle operates continuously during both superposed correction mode (Phases 1–3, 5–7) where corrections modify the programmed path, and sensor-guided mode (`RSI_MOVECORR()`) where corrections are the sole source of motion commands. The deterministic timing ensures responsive control with total round-trip latency under 12 ms.

The RSI `ETHERNET` object handles UDP socket communication, XML parsing/generation, and interpolator synchronization, with data transmission occurring parallel to KRL program execution, the flow shown in Figure 8.4.

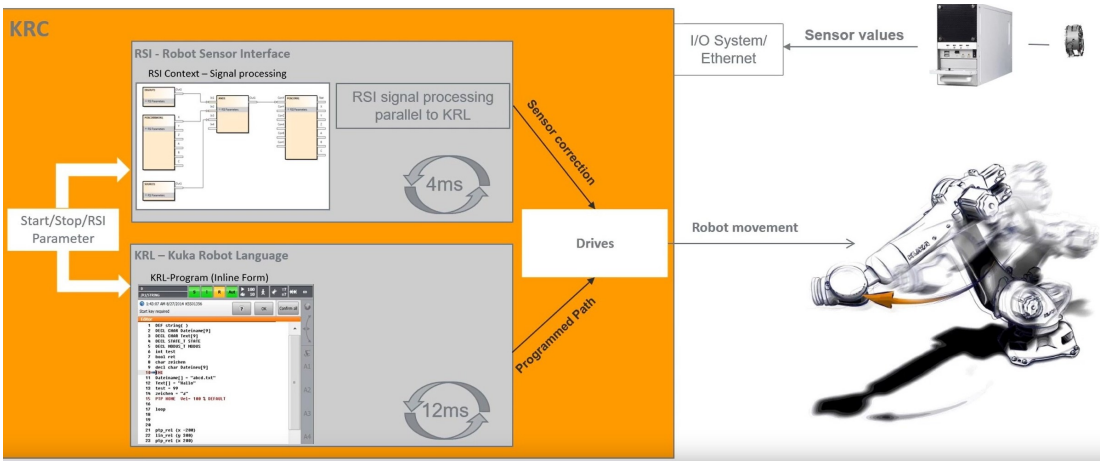


Figure 8.4: Parallel execution flow of RSI and KRL

8.2.2 RSI Operating Modes and Configuration

The system utilizes RSI in two distinct operating modes, selected based on the operational phase requirements. This dual-mode architecture enables both safe pre-programmed motion and flexible operator-guided positioning within a single continuous RSI session.

8.2.2.1 RSI Configuration Parameters

The RSI system is configured with the following parameters in the KRL initialization:

- **Correction mode**: `#RELATIVE` - corrections are incrementally integrated into the robot position
- **Correction type**: `POSCORR` - Cartesian correction (not axis-specific)
- **Correction coordinate system**: `#BASE` - corrections interpreted in the base coordinate frame
- **Integration coordinate system**: `#BASE` - relative corrections accumulated in base frame

The relative correction mode was selected because corrections must persist across motion segments. In absolute mode, corrections would reset when transitioning between waypoints, causing discontinuous motion. The BASE coordinate system provides a stable, world-fixed reference suitable for vehicle body positioning tasks. In relative mode, correction values are integrated into an overall correction vector that modifies the robot's position throughout the program execution. If a +10 mm correction is applied in the X direction, this offset persists through subsequent motion segments until explicitly modified or cleared by new correction commands.

8.2.2.2 Superposed Correction Mode (Phases 1, 2, 3, 5, 6, 7)

During programmed KRL motion sequences, RSI operates in **superposed correction mode**. In this mode, the robot follows pre-programmed teach points while the external system can apply small trajectory adjustments.

```
DEF RoofInsertion()
    INT ret

    ; Activate RSI with relative corrections in BASE frame
    ret = RSI_ON(#RELATIVE, , ,#BASE)

    ; Phase 1: Programmed motion with superposed
       corrections
    PTP HOME
    LIN RoofPickup
    ; vacuum gripper engages roof

    ; Phase 2-3: Transport and insertion
    LIN TransportStart
    LIN WindscreenApproach
    LIN InsertionPoint
    WAIT SEC 0.5  ; Settle before sensor-guided mode

    ; [Phase 4 code - see Listing~\ref{lst:movecorr}]

    ; Phase 5-7: Continue with superposed corrections
    LIN ExitPosition
    ; ... additional waypoints ...

    ret = RSI_OFF()
END
```

Listing 8.1: RSI initialization in KRL program

In superposed correction mode:

- The robot follows the **programmed path** defined by teach points
- Correction values from the Python client are **superposed on the planned trajectory**
- Small adjustments (±5–10 mm typically) can compensate for:
  - Vehicle body positioning variations
  - Roof dimensional tolerances
  - Dynamic obstacle avoidance (proximity sensor inputs)
- Motion continues to the programmed end point even if corrections are applied

The listing above uses canonical KRL motion commands (`PTP` for point-to-point, `LIN` for linear Cartesian motion) to illustrate the program structure. The actual implementation mostly employs `SPTP` (Spline Point-to-Point) commands generated by KUKA WorkVisual, which provide smooth, dynamically optimized trajectories with advanced blending between waypoints. The motion type does not affect RSI operation, corrections are applied identically regardless of whether the programmed path uses PTP, LIN, CIRC, or SPTP commands.

The superposed mode provides a safety advantage: if the external system fails or sends zero corrections, the robot continues executing its programmed path. This fail-safe behavior is critical for industrial applications where communication faults must not cause unpredictable robot behavior.

8.2.2.3 Sensor-Guided Motion Mode (Phase 4)

Phase 4 requires fundamentally different behavior, the robot must hold position without advancing to a programmed waypoint while allowing commanded adjustments. This is achieved using the `RSI_MOVECORR()` command, which activates sensor-guided motion mode.

```
; Phase 4: Worker verification and clipping
LIN InsertionPoint
WAIT SEC 0.5  ; Settle before switching modes

; Enter sensor-guided mode - no programmed endpoint
RSI_MOVECORR()

; Robot now controlled entirely by RSI corrections
; Python client sends target positions or incremental corrections
; $MOVECORR flag is TRUE, visible in RSI XML as Movecorr_flag
; Motion continues until client sets stop_flag to exit

WAIT FOR NOT $MOVECORR  ; Block until client signals completion

; $MOVECORR is now FALSE, continue to Phase 5
WAIT SEC 0.3

; Phase 5: Tool change sequence
; vacuum gripper release
LIN ExitPosition
```

Listing 8.2: Sensor-guided motion in Phase 4

In sensor-guided mode (`RSI_MOVECORR()`):

- The robot has **no programmed destination** and it moves purely based on RSI commands
- The system variable `$MOVECORR` is set to `TRUE`, which is transmitted to the Python client via the `Movecorr_flag` field in the RSI XML
- The Python client can implement two control strategies:
  1. **Target mode**: Client computes corrections as $u = K_p(x_{tgt} - x_{curr})$ to drive the robot toward an absolute target position

2. **Correction mode**: Client sends relative incremental corrections directly (e.g., "+5 mm in Y") based on operator input

- Motion continues indefinitely until the client sets the `stop_flag` in the RSI response XML, which causes `$MOVECORR` to transition to `FALSE`
- This mode is ideal for Phase 4 where the operator dictates position through commands or where sensor feedback requires continuous position adjustment

The `RSI_MOVECORR()` command transitions the robot into sensor-guided mode where it has no programmed destination and moves purely based on RSI corrections from the Python client. The `WAIT FOR NOT $MOVECORR` statement blocks program execution until the external system sets the stop flag, at which point the controller clears `$MOVECORR` to `FALSE` and execution continues to Phase 5. As with Listing 8.1, the actual implementation uses `SPTP` commands, but `LIN` is shown here for clarity. The motion type is irrelevant during `RSI_MOVECORR()`, the robot's position is entirely determined by external corrections regardless of the preceding or following motion commands.

The distinction between these modes is critical:

- **Superposed corrections** adjust a **planned path**—the robot has a destination
- **Sensor-guided motion** has **no path**—the robot goes wherever the RSI corrections direct it

8.2.2.4 Mode Transition Behavior

The transition between superposed and sensor-guided modes is handled automatically by the KUKA controller with state synchronization to the Python client:

1. The robot executes programmed motion in superposed mode
2. Robot reaches the waypoint `InsertionPoint` (end of Phase 3)
3. KRL program calls `RSI_MOVECORR()`
4. The controller sets `$MOVECORR` system variable to `TRUE`
5. This state is transmitted to the Python client via `Movecorr_flag` in the next RSI cycle (within 12 ms)
6. The Python client detects the transition and activates appropriate control logic:

- Operator correction controls are enabled (GUI/Terminal/ROS Interface)
- Control system switches from "passive monitoring" to "active positioning"
- Initial target position is captured for target mode operation

7. Worker performs verification and commands position adjustments
8. When Phase 4 is complete (worker clips roof), operator signals completion through GUI
9. Python client sets `stop_flag` in the RSI response XML
10. Controller clears `$MOVECORR` to `FALSE` (within 12 ms)
11. `WAIT FOR NOT $MOVECORR` statement in KRL unblocks
12. Robot continues to Phase 5 with programmed motion (superposed mode resumes)

This seamless transition enables the hybrid control architecture where most motion follows safe, pre-programmed paths, but Phase 4 provides full operator authority over positioning. The 12 ms state transition time ensures responsive mode switching without delay.

8.3 Control Algorithms

8.3.1 Control Modes Within Sensor-Guided Operation

During sensor-guided motion (`RSI_MOVECORR()`), the Python client implements two control modes selectable by the operator.

8.3.1.1 Target Mode

In target mode, the operator specifies an absolute Cartesian target position $x_{tgt} \in \mathbb{R}^6$ in BASE coordinates. The Python client implements a proportional controller that computes corrections to drive the robot toward this target:

$$u_i = \text{sat}_{[-u_{\max,i},\, u_{\max,i}]}\Big(K_p(x_{\text{tgt},i} - x_{\text{curr},i})\Big) \tag{8.1}$$

where:

- x_{curr}: current robot position from RSI state data

- x_{tgt}: target position set by operator
- $K_p = 0.005$: proportional gain
- $u_{\max,i}$: velocity saturation limits per axis:

$$u_{\max,i} = \begin{cases} 0.05 \text{ mm/cycle} & i \in \{X, Y, Z\} \quad \rightarrow \quad v_{\max} = 4.17 \text{ mm/s} \\ 0.025 \text{ deg/cycle} & i \in \{A, B, C\} \quad \rightarrow \quad \omega_{\max} = 2.08 \text{ deg/s} \end{cases} \tag{8.2}$$

Algorithm 8.1 details the target mode control law.

Algorithm 8.1 Proportional Control with Axis-Dependent Saturation (Target Mode)

Require:
x_{tgt}: Target pose (6-DOF, BASE frame)
x_{curr}: Current pose from RSI
$K_p = 0.005$
Linear saturation: $u_{\max} = 0.05$ mm/cycle
Angular saturation: $u_{\max} = 0.025$ deg/cycle
Precondition: Active when a manual target is set (i.e., at least one component of x_{tgt} is nonzero).
function ComputeCorrections(x_{tgt}, x_{curr})
    $u \leftarrow [0, 0, 0, 0, 0, 0]^T$
    if NoManualTarget(x_{tgt}) **then**
        return u ▷ Matches code path that outputs zeros if no manual target
    end if
    for $i \in \{X, Y, Z\}$ **do** ▷ Linear axes (mm)
        $e_i \leftarrow x_{\text{tgt},i} - x_{\text{curr},i}$
        $u_{\text{tmp}} \leftarrow K_p \cdot e_i$
        $u_i \leftarrow \text{Sat}_{[-u_{\max}, u_{\max}]}(u_{\text{tmp}})$
    end for
    for $i \in \{A, B, C\}$ **do** ▷ Angular axes (deg)
        $e_i \leftarrow \text{ShortestAngleDiff}(x_{\text{tgt},i}, x_{\text{curr},i})$
        $u_{\text{tmp}} \leftarrow K_p \cdot e_i$
        $u_{\text{sat}} \leftarrow \text{Sat}_{[-u_{\max}, u_{\max}]}(u_{\text{tmp}})$
        if $i = A$ **then**
            $u_i \leftarrow u_{\text{sat}}$
        else ▷ $i \in \{B, C\}$ use inverted sign
            $u_i \leftarrow -u_{\text{sat}}$
        end if
    end for
    return u
end function

Sign Convention. For axes B and C, the correction sign is inverted to match KUKA RSI orientation handling.

Velocity-Limited Behavior. Although the control law is based on position error, the robot never receives absolute positions. Instead, the client sends small incremental corrections at each RSI cycle (83 Hz). This makes Target Mode effectively a *velocity-limited proportional controller*, where the maximum achievable speed is determined by the saturation limit and RSI frequency:

$$v_{\max} = u_{\max} \cdot f_{RSI} = 0.05 \text{ mm/cycle} \times 83.33 \text{ Hz} \approx 4.17 \text{ mm/s} \tag{8.3}$$

This conservative speed ensures safe operation in confined spaces, such as vehicle interiors.

8.3.1.2 Angular Correction Wrapping

Angular corrections require special handling to ensure the shortest rotational path. The Python client uses a normalization function to wrap angular errors into the range $[-180°, 180°]$. Algorithm 8.2 implements the angle normalization function.

Algorithm 8.2 Shortest-Angle Difference (Angular Error Normalization)

function ShortestAngleDiff($\theta_{\text{tgt}}, \theta_{\text{curr}}$)
    $\Delta \leftarrow \theta_{\text{tgt}} - \theta_{\text{curr}}$
    while $\Delta > 180$ **do**
        $\Delta \leftarrow \Delta - 360$
    end while
    while $\Delta < -180$ **do**
        $\Delta \leftarrow \Delta + 360$
    end while
    return Δ
end function
Example:
$\theta_{\text{curr}} = 30°$, $\theta_{\text{tgt}} = -10°$ (equivalently $350°$)
Raw difference: $\Delta = -10 - 30 = -40°$ (already in range)
Normalized: $-40°$ (correct shortest path)
Without normalization: using $350°$ instead of $-10°$ would give $+320°$ (wrong direction)

Figure 8.5 illustrates the angular wrapping concept.

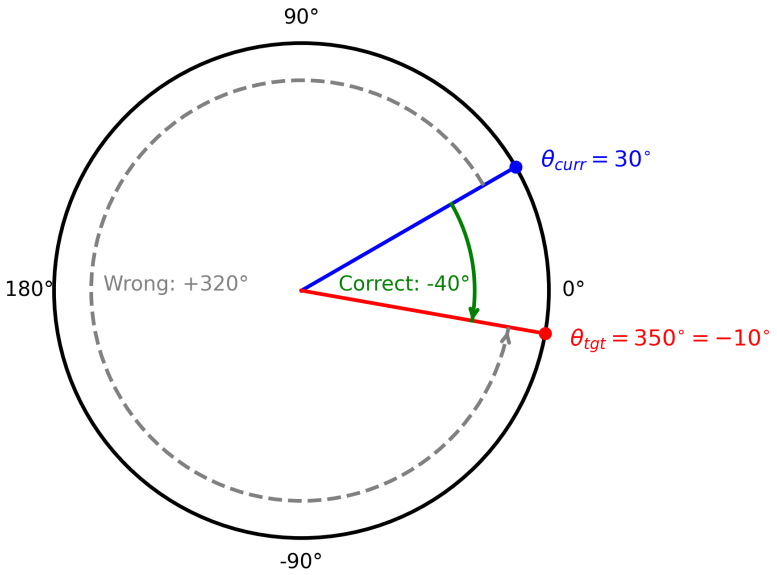


Figure 8.5: Angular wrapping and shortest path calculation. When the current angle is 30° and the target is -10° (equivalently 350°), the shortest rotational path is -40° (counterclockwise) rather than +320° (clockwise). The normalization function ensures the controller always selects the minimum rotation.

8.3.1.3 Correction Mode

In correction mode, the operator directly commands incremental position adjustments through the GUI (e.g., "move +10 mm in Y direction"). This mode is activated via the `correction AXIS VALUE` command in the GUI/command line interface (CLI)/ROS2 Interface. The Python client computes a temporary target position relative to a reference captured when the correction is first commanded:

his mode is activated via the `correction AXIS VALUE` command. Internally, the client computes a temporary target:

$$x_{\text{tgt},i} = x_{\text{curr},i}^{\text{initial}} + \Delta_i \tag{8.4}$$

where Δ_i is the user-specified correction for axis i and $x_{\text{curr},i}^{\text{initial}}$ is the pose at the moment the correction command was issued.

The control law remains proportional (as in target mode) but incorporates a ramp-up factor λ for smooth motion onset:

$$u_i = \lambda \cdot \text{sat}_{[-u_{\max,i},\, u_{\max,i}]}\Big(K_p(x_{\text{tgt},i} - x_{\text{curr},i})\Big), \quad \lambda \in [0, 1] \tag{8.5}$$

where λ prevents sudden motion when corrections are initiated. The ramp-up factor evolves according to:

$$\lambda(k+1) = \min\left(1.0, \lambda(k) + \Delta\lambda\right), \quad \lambda(0) = 0.001, \quad \Delta\lambda = 0.001 \tag{8.6}$$

This gradual ramp reaches full magnitude after approximately:

$$\text{Cycles} = \frac{1.0 - 0.001}{0.001} \approx 999, \qquad \text{Duration} = \frac{999}{83.33 \text{ Hz}} \approx 12 \text{ s}.$$

However, if the commanded correction is small, the robot will typically reach the target before λ reaches 1.0, so the full 12 s ramp is rarely observed.

Velocity Limits in Correction Mode To provide more responsive operator control while maintaining safety during manual adjustments, correction mode uses higher velocity limits than target mode:

- **Linear axes** (X, Y, Z): $u_{\max} = 0.7$ mm/cycle $\rightarrow v_{\max} = 58.3$ mm/s
- **Angular axes** (A, B, C): $u_{\max} = 0.35$ deg/cycle $\rightarrow \omega_{\max} = 29.2$ deg/s

These limits are 14× higher than target mode for both linear (0.05 mm/cycle) and angular (0.025 deg/cycle) corrections, allowing responsive operator control. Despite this increase, the system operates with significant safety margins: only 1.9% of maximum Cartesian velocity (3000 mm/s) and 11–28% of maximum joint velocities (105–260 deg/s for axes A1–A6). The ramp-up factor further ensures smooth motion onset, preventing abrupt movements when corrections are initiated.

Ramp-Up Effect on Motion. If the correction target is far, the error remains large for many cycles, so the ramp-up factor λ continues to increase until it reaches 1.0. At that point, the controller applies the full proportional correction (up to the saturation limit), achieving the maximum allowed speed. Conversely, if the correction target is near, the error decreases quickly. Even with $\lambda < 1.0$, the proportional term $K_p \cdot e$ becomes small, so the commanded correction drops toward zero before the ramp finishes. The robot reaches the target without ever hitting full speed.

Algorithm 8.3 formalizes the correction mode implementation.

Algorithm 8.3 Correction Mode with Ramp-Up (Aligned with Implementation)

Require:
Δ: Operator-commanded correction (per-axis; zero if inactive)
x_{curr}: Current robot pose (from RSI)
x_{ref}: Reference pose (captured when a new correction command is issued)
$\lambda \in [0, 1]$: Ramp-up factor
Context: Active during Phase 4 while `RSI_MOVECORR()` is running.
function ApplyCorrectionWithRampUp(Δ, x_{curr}, x_{ref}, λ)
  if a new `correction AXIS VALUE` command is received **then**
    $x_{ref} \leftarrow x_{curr}$ ▷ Capture reference at command time
    $\lambda \leftarrow 0.001$ ▷ Initialize ramp factor
  end if
  $u \leftarrow [0, 0, 0, 0, 0, 0]^T$
  for $i \in \{X, Y, Z, A, B, C\}$ **do**
    if $\Delta_i = 0$ **then**
      continue ▷ Axis inactive; no correction
    end if
    if $i \in \{A, B, C\}$ **then** ▷ Angular axes (deg)
      $x_{\text{tgt},i} \leftarrow \text{WrapTo}_{\pm 180}(x_{\text{ref},i} + \Delta_i)$
      $e_i \leftarrow \text{ShortestAngleDiff}(x_{\text{tgt},i}, x_{\text{curr},i})$
      $u_{\max} \leftarrow 0.35$ ▷ deg/cycle → 29.2 deg/s at 83 Hz
      $u_{\text{tmp}} \leftarrow K_p \cdot e_i$ where $K_p = 0.005$
      $u_{\text{sat}} \leftarrow \text{Sat}_{[-u_{\max}, u_{\max}]}(u_{\text{tmp}})$
      if $i = A$ **then**
        $u_i \leftarrow \lambda \cdot u_{\text{sat}}$
      else ▷ $i \in \{B, C\}$ use inverted sign
        $u_i \leftarrow -\lambda \cdot u_{\text{sat}}$
      end if
      if $|u_i| < 0.001$ **then** ▷ Optional convergence check
        $\Delta_i \leftarrow 0$ ▷ Clear completed correction
      end if
    else ▷ Linear axes (mm)
      $x_{\text{tgt},i} \leftarrow x_{\text{ref},i} + \Delta_i$
      $e_i \leftarrow x_{\text{tgt},i} - x_{\text{curr},i}$
      $u_{\max} \leftarrow 0.7$ ▷ mm/cycle → 58.3 mm/s at 83 Hz
      $u_{\text{tmp}} \leftarrow K_p \cdot e_i$ where $K_p = 0.005$
      $u_i \leftarrow \lambda \cdot \text{Sat}_{[-u_{\max}, u_{\max}]}(u_{\text{tmp}})$
      if $|u_i| < 0.0001$ **then** ▷ Optional convergence check
        $\Delta_i \leftarrow 0$ ▷ Clear completed correction
      end if
    end if
  end for
  if $\lambda < 1.0$ **then**
    $\lambda \leftarrow \min(1.0, \lambda + 0.001)$ ▷ Increment per cycle at 83 Hz
  end if
  return u, λ, Δ ▷ Updated corrections and ramp factor
end function

Figure 8.6 shows the ramp-up factor evolution.

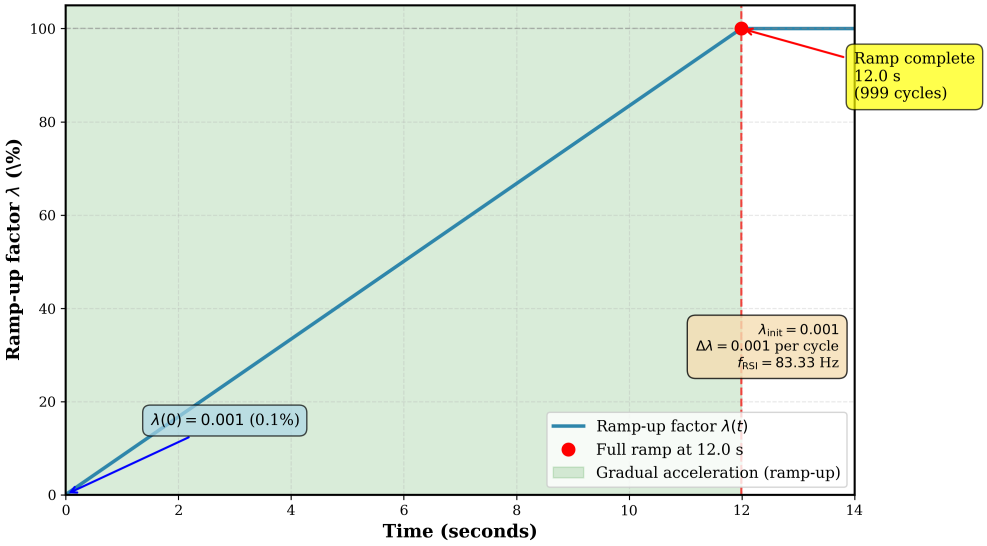


Figure 8.6: Ramp-up factor λ evolution in correction mode, showing gradual increase from 0.1% to 100% over approximately 12 seconds (1000 cycles at 83 Hz). The ramp-up prevents sudden motion onset when commanded corrections are initiated, ensuring smooth and safe robot behavior when a worker is present inside the vehicle. Note that target mode does not use ramp-up—corrections are applied immediately at full magnitude.

8.3.2 Saturation Characteristics

The velocity saturation function limits the maximum correction applied per cycle, ensuring predictable motion regardless of position error magnitude. This mechanism serves two critical purposes: (1) safety: preventing excessive velocities that could endanger workers in the confined vehicle interior, and (2) smooth convergence: automatically reducing speed as the robot approaches the target position.

Figure 8.7 illustrates the saturation behavior for both control modes, along with example trajectories demonstrating the motion profiles resulting from large position commands.

8.3.2.1 Saturation Threshold Analysis

The saturation threshold is the error magnitude at which the controller transitions from proportional to constant-velocity behavior is determined by the ratio of velocity limit to proportional gain:

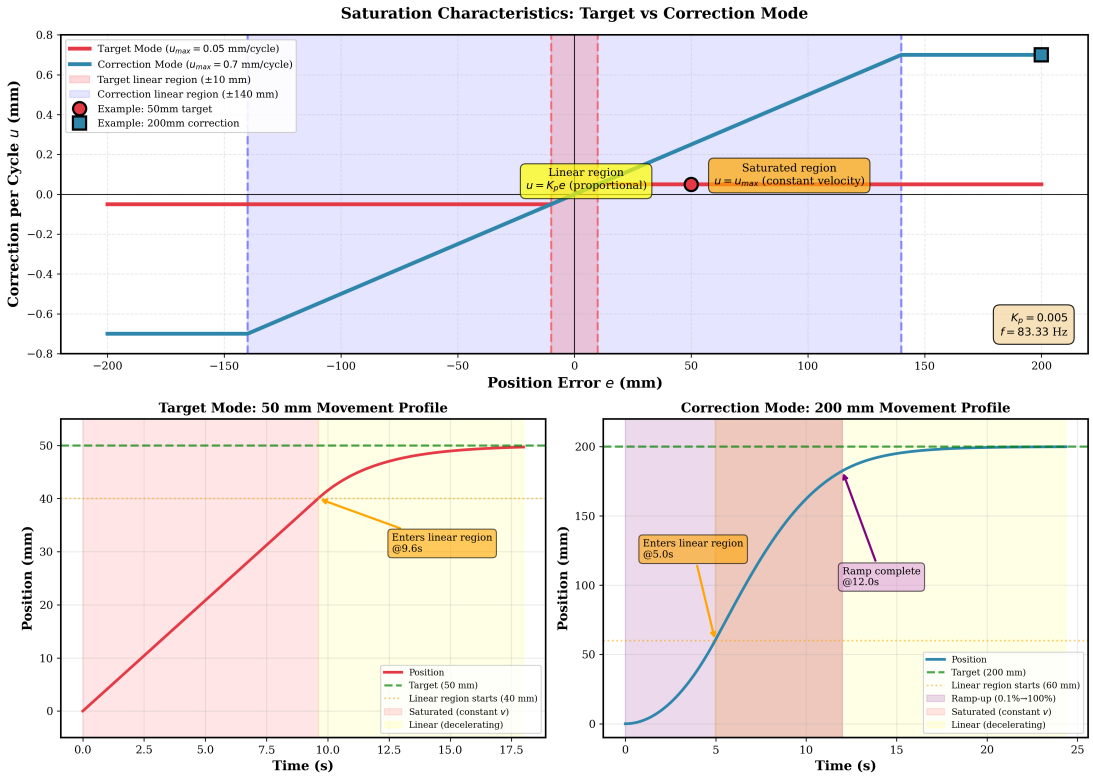


Figure 8.7: Saturation characteristics and motion profiles for both control modes. **Top:** Saturation curves showing the relationship between position error and applied correction. The linear region (shaded) exhibits proportional control ($u = K_p e$), while the saturated region enforces maximum velocity limits. Target mode saturates beyond ±10 mm error; correction mode saturates beyond ±140 mm. **Bottom-left:** Target mode trajectory for a 50 mm position command, showing constant velocity (4.17 mm/s) during saturation, followed by proportional deceleration in the final 10 mm. **Bottom-right:** Correction mode trajectory for a 200 mm position command, showing gradual ramp-up (first 12 s), constant maximum velocity (58.3 mm/s), and proportional deceleration over the final 140 mm.

$$e_{\text{sat}} = \frac{u_{\max}}{K_p} \tag{8.7}$$

For the implemented control parameters:

Target Mode:

- Velocity limit: $u_{\max} = 0.05$ mm/cycle $\rightarrow v_{\max} = 4.17$ mm/s
- Saturation threshold: $e_{\text{sat}} = 0.05/0.005 = 10$ mm
- **Behavior:** Errors beyond ±10 mm result in constant 4.17 mm/s motion. Within ±10 mm, velocity decreases proportionally, reaching near-zero as the robot approaches the target.

Correction Mode:

- Velocity limit: $u_{\max} = 0.7$ mm/cycle $\rightarrow v_{\max} = 58.3$ mm/s
- Saturation threshold: $e_{\text{sat}} = 0.7/0.005 = 140$ mm
- **Behavior:** After ramp-up completes, errors beyond ±140 mm result in constant 58.3 mm/s motion. Within ±140 mm, velocity decreases proportionally over a much longer approach distance than target mode.

8.3.2.2 Motion Profile Characteristics

The trajectory examples in Figure 8.7 demonstrate the practical implications of saturation:

Target Mode (50 mm command):

1. **Constant velocity phase** (0–10 s, 0–40 mm): Robot moves at maximum velocity (4.17 mm/s) while error exceeds 10 mm. The proportional controller commands higher velocities ($u_{raw} = K_p \cdot 50 = 0.25$ mm/cycle), but saturation limits the actual correction to 0.05 mm/cycle.
2. **Proportional deceleration** (10–25 s, 40–50 mm): As error drops below 10 mm, the robot enters the linear region where $u = K_p e$ without saturation. Velocity decreases smoothly: at 5 mm error $\rightarrow$ 2.08 mm/s; at 1 mm error $\rightarrow$ 0.42 mm/s.
3. **Convergence** (25 s, 50 mm): Robot settles to within ±0.5 mm of target with negligible residual motion. No overshoot occurs due to the purely proportional control law.

Correction Mode (200 mm command):

1. **Ramp-up phase** (0–12 s, 0–12 mm): The ramp factor λ increases from 0.1% to 100%, gradually accelerating the robot despite the controller commanding saturated corrections. This prevents sudden motion onset when the operator initiates a large correction.

2. **Constant velocity phase** (12–18 s, 12–60 mm): With $\lambda = 1.0$ and error exceeding 140 mm, the robot moves at maximum velocity (58.3 mm/s). The 14× higher velocity limit compared to target mode enables responsive operator control.

3. **Proportional deceleration** (18–23 s, 60–200 mm): As error drops below 140 mm, velocity decreases proportionally over a much longer distance than target mode. This extended deceleration zone (140 mm vs. 10 mm) provides smoother motion at higher speeds.

4. **Convergence** (23 s, 200 mm): Robot settles smoothly without overshoot.

8.3.2.3 Design Rationale

The divergent saturation thresholds reflect the distinct operational contexts:

Target mode (conservative, $e_{sat} = 10$ mm):

- Intended for automatic convergence to user-verified positions
- Low maximum velocity (4.17 mm/s) prioritizes safety when worker is verifying position
- Tight linear region (±10 mm) enables precise final positioning
- No ramp-up required—target is already validated before motion begins

Correction mode (responsive, $e_{sat} = 140$ mm):

- Intended for direct operator manipulation via GUI
- Higher maximum velocity (58.3 mm/s) provides responsive motion
- Wide linear region (±140 mm) allows smooth deceleration from higher speeds
- Ramp-up compensates for higher velocity potential, preventing jerky motion onset
- Operator maintains control authority—can stop motion at any time

The saturation mechanism ensures that even with large position errors (e.g., 200 mm), the robot never exceeds safe velocity limits. This bounded behavior is critical for human-robot collaboration in confined spaces where unpredictable high-speed motion could endanger workers. The proportional deceleration in the linear region provides smooth, predictable approach to the target position without requiring explicit velocity profiling or trajectory planning.

8.3.3 Mode Relationship Summary

Table 8.1 summarizes the relationship between RSI operating modes and control system modes across the assembly cycle.

Table 8.1: Relationship between RSI modes, KRL program phases, and control system modes.

KRL Phase	**RSI Mode**	**Control System Behavior**
Phases 1, 2, 3, 5, 6, 7	Superposed correction	Python client sends zero or minimal corrections. Robot follows programmed path. Small corrections applied only for obstacle avoidance or drift compensation.
Phase 4 (target positioning)	Sensor-guided (`MOVECORR`)	**Target mode**: Operator specifies absolute target position. System drives robot to target using proportional control.
Phase 4 (incremental adjustment)	Sensor-guided (`MOVECORR`)	**Correction mode**: Operator commands incremental adjustments. Ramp-up ensures smooth motion onset.

During Phases 1–3 and 5–7, the Python client typically sends zero or near-zero corrections, allowing the robot to follow its programmed path with full safety certification active. However, the superposed correction capability provides a valuable safety layer: if proximity sensors detect obstacles or if minor path adjustments are needed, the client can send corrections without interrupting the motion sequence or triggering a safety stop. These corrections are automatically integrated into the BASE coordinate system and persist through subsequent waypoints due to the `#RELATIVE` mode configuration.

Phase 4 is the only phase where `RSI_MOVECORR()` is active, giving the Python client full authority over robot positioning for the duration of worker verification and clipping operations. This authority operates in two distinct operator-controlled sub-modes:

- **Target mode** (absolute positioning): The operator specifies a complete 6-DOF target position x_{tgt}. The proportional controller then drives the robot smoothly toward this target with conservative velocity limits, suitable for precise final positioning. This mode is ideal when the operator knows the desired final position and wants the system to handle the trajectory.
- **Correction mode** (incremental adjustment): The operator commands individual axis adjustments (e.g., +10 mm in Y, -5 deg in C). Each correction is applied relative to a reference position captured when the correction is initiated. Higher velocity limits provide responsive control, while the ramp-up mechanism prevents sudden motion onset by gradually increasing control authority from 0.1% to 100% over approximately 12 seconds. This mode is ideal for iterative fine-tuning where the operator adjusts individual axes based on visual feedback.

The key distinction is control paradigm: target mode requires the operator to specify where to go, while correction mode allows the operator to specify how much to move. Both are bounded by velocity saturation and the RSI configuration's maximum correction range (typically ± 50 mm per axis), ensuring that even with full external control authority, the robot cannot make dangerous high-speed movements or exceed safe position limits. The combination of velocity saturation, ramp-up (in correction mode), and correction range limits provides defense-in-depth safety for human-robot collaboration in the confined vehicle interior.

8.4 Communication Protocol Implementation

8.4.1 XML Data Exchange Format

RSI communication uses XML-formatted messages exchanged via UDP. The controller transmits robot state data, and the client responds with correction commands. Both messages include an IPOC (Interpolation Cycle) timestamp for synchronization.

8.4.1.1 Controller to Client (State Data)

```
<Rob Type="KUKA">
  <RIst X="1523.45" Y="834.12" Z="1205.33"
        A="45.2" B="-12.8" C="90.5" />
  <AIPos A1="23.4" A2="45.6" A3="-30.2"
         A4="12.1" A5="67.8" A6="-15.3" />
  <Step>12458</Step>
  <Movecorr_flag>1</Movecorr_flag>
  <AnOut_1>99</AnOut_1>
  <IPOC>1678234567890</IPOC>
</Rob>
```

Listing 8.3: RSI state transmission from controller

Key fields:

- `RIst`: Cartesian position (X, Y, Z in mm; A, B, C in degrees)
- `AIPos`: Axis positions A1–A6 (joint angles in degrees)
- `Step`: Cycle counter (increments each 12 ms)
- `Movecorr_flag`: State of `$MOVECORR` (0=superposed mode, 1=sensor-guided mode)
- `AnOut_1`: Routine number for coordination (99=idle, 1–n=active routines)
- `IPOC`: Timestamp for synchronization (microseconds)

8.4.1.2 Client to Controller (Corrections)

```
<Sen Type="ImFree">
  <RKorr X="0.025" Y="-0.012" Z="0.008"
         A="0.0" B="0.0" C="0.0" />
  <Routine>99</Routine>
  <DiO>0</DiO>
  <IPOC>1678234567890</IPOC>
</Sen>
```

Listing 8.4: RSI correction response from client

Key fields:

- `RKorr`: Cartesian corrections (mm and degrees per cycle)
- `Routine`: Routine number echo for coordination
- `DiO`: Digital output (0=continue `MOVECORR`, 1=stop `MOVECORR`)
- `IPOC`: Echo of received timestamp

8.4.2 Main Control Loop

The Python client implements a continuous control loop that maintains the 83 Hz communication cycle. Algorithm 8.4 presents the main loop structure.

Algorithm 8.4 Main RSI Control Loop

Require:
UDP socket bound to port 59152
Connection timeout: 3 seconds
Packet timeout: 0.5 seconds
function MainControlLoop
    $connected \leftarrow$ `False`
    while running **do**
        $data, addr \leftarrow$ ReceiveUDP(timeout=0.5s)
        if $data =$ `None` **then**
            if time since last packet $>$ 3s **then**
                $connected \leftarrow$ `False`
                LogWarning("Connection lost")
            end if
            continue
        end if
        $connected \leftarrow$ `True`
        $state \leftarrow$ ParseXML($data$)
        Extract: x_{curr}, $step$, $movecorr\_flag$, $routine$, $IPOC$
        if $movecorr\_flag = 1$ **then** ▷ Sensor-guided mode active
            if target mode enabled **then**
                $u \leftarrow$ ComputeTargetCorrections(x_{tgt}, x_{curr}) ▷ Alg. 8.1
            else ▷ Correction mode enabled
                $u, \lambda \leftarrow$ ApplyCorrectionWithRampUp(Δ, x_{curr}, x_{ref}, λ) ▷ Alg. 8.3
            end if
        else ▷ Superposed mode
            $u \leftarrow$ ComputeSuperposedCorrections()
        end if
        $routine\_num \leftarrow$ UpdateRoutineState($routine$) ▷ Alg. 8.5
        $xml_{response} \leftarrow$ CreateResponseXML(u, $routine\_num$, $stop\_flag$, $IPOC$)
        SendUDP($xml_{response}$, $addr$)
        UpdateGUI(x_{curr}, u, $movecorr\_flag$, $connected$, $step$)
    end while
end function

The main loop continuously monitors for incoming RSI packets from the KUKA controller. Upon receiving a packet, the loop:

1. **Parses state data**: Extracts current position, mode flags, and routine status
2. **Determines control mode**: Checks `movecorr_flag` to distinguish between superposed and sensor-guided modes

3. **Computes corrections**: Applies appropriate control algorithm (target mode, correction mode, or superposed)
4. **Manages routines**: Updates routine state machine for task coordination
5. **Sends response**: Packages corrections into XML and transmits via UDP
6. **Updates GUI**: Refreshes operator interface with current state

The 0.5-second packet timeout ensures responsive detection of communication failures, while the 3-second connection timeout prevents false alarms from momentary network jitter. If no packets are received for 3 seconds, the system transitions to a disconnected state and logs a warning, but does not halt and the KUKA controller continues executing its programmed motion in superposed mode with zero corrections.

8.4.3 Routine Management

The routine management system coordinates trajectory sequencing between the controller and client through a handshaking protocol. Algorithm 8.5 details this coordination.

Algorithm 8.5 Routine Management Protocol

Require:
R_{num}: Current routine number (client state)
R_{queue}: Pending routine queue (operator commands)
R_{flag}: User-initiated flag (0=acknowledged, 1=new request)
$AnOut_1$: Controller's routine status (from RSI XML)
function UpdateRoutineState($AnOut_1$)
    if $AnOut_1 = 99$ **and** $R_{num} \neq 99$ **and** $R_{flag} = 0$ **then**
        $R_{num} \leftarrow 99$ ▷ Controller completed routine, return to idle
        Log("Routine completed, controller idle")
    else if $|R_{queue}| > 0$ **and** $R_{num} = 99$ **then**
        $R_{num} \leftarrow R_{queue}$.pop() ▷ Load next routine from queue
        $R_{flag} \leftarrow 1$ ▷ Mark as new user-initiated request
        Log("Starting routine", R_{num})
    else if $R_{flag} = 1$ **and** $R_{num} = AnOut_1$ **then**
        $R_{flag} \leftarrow 0$ ▷ Controller acknowledged routine
        Log("Routine", R_{num}, "acknowledged by controller")
    end if
    return R_{num} ▷ Routine number to send in XML response
end function

The routine management protocol implements a three-state handshaking mechanism:

1. **Idle state** ($R_{num} = 99$, $AnOut_1 = 99$): Both client and controller are idle. Client can initiate new routines from the queue.
2. **Request state** ($R_{num} \neq 99$, $R_{flag} = 1$): Client has requested a new routine but controller has not yet acknowledged. Client continues sending the routine number in every RSI response until acknowledgment.
3. **Active state** ($R_{num} \neq 99$, $R_{flag} = 0$, $AnOut_1 = R_{num}$): Controller is executing the requested routine. Client monitors for completion (controller returns to $AnOut_1 = 99$).

This protocol ensures reliable routine sequencing. The handshaking prevents scenarios where:

- The client issues a new routine before the controller completes the previous one
- The controller misses a routine request due to packet loss

Figure 8.8 illustrates the state transitions.

8.4.4 Error Handling and Robustness

The communication protocol includes several mechanisms for handling failures:

Packet Loss Handling UDP is inherently unreliable, but RSI tolerates occasional packet loss due to:

- **High frequency**: At 83 Hz, losing 1–2 packets per second results in <2% data loss
- **Stateless corrections**: Each correction is independent; missing one packet does not corrupt subsequent cycles
- **Controller interpolation**: The KUKA controller interpolates between received corrections, smoothing over brief gaps

Experimental validation showed packet loss rates below 0.01% on the dedicated Ethernet network, well within acceptable bounds.

Timeout and Reconnection If no packets are received for 3 seconds:

- Client transitions to disconnected state
- Upon receiving next packet, connection automatically reestablishes
- No manual intervention required for transient network issues

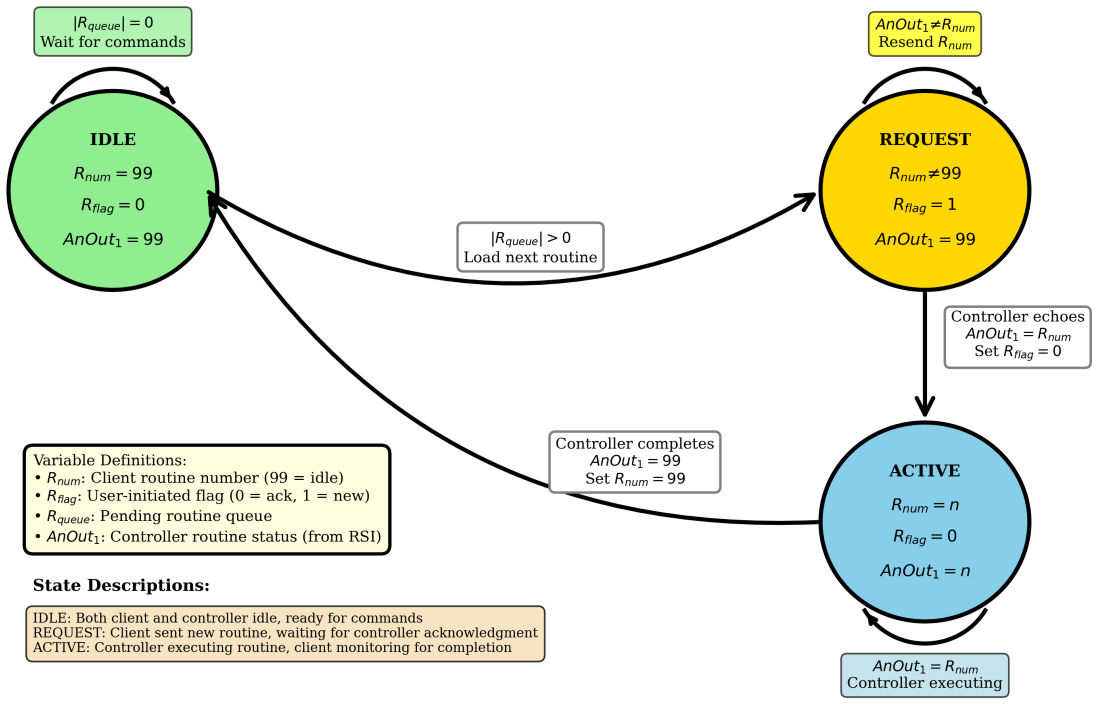


Figure 8.8: Routine management state machine implementing the handshaking protocol described in Algorithm 8.5. The system transitions between three states: **IDLE** (both client and controller idle), **REQUEST** (client sent routine, awaiting acknowledgment), and **ACTIVE** (controller executing routine). The client variable R_{num} represents the current routine (99 = idle), R_{flag} indicates whether a routine is user-initiated (1) or acknowledged (0), and the controller's $AnOut_1$ variable reflects its routine status. Self-loops show conditions under which states persist: IDLE waits for queued commands, REQUEST repeatedly sends routine number until acknowledged, and ACTIVE monitors for completion. The handshaking ensures reliable sequencing without race conditions, preventing scenarios where the client issues new routines before the controller completes previous ones.

Invalid Data Handling The XML parser validates incoming data:

- Malformed XML: Log error, skip packet, continue
- Missing fields: Use previous valid values
- Out-of-range values: Clamp to safe limits before processing

This defensive approach prevents single corrupted packets from crashing the control system.

8.5 Graphical User Interface

The GUI provides comprehensive control and monitoring capabilities for the RSI correction system. The interface is organized into functional tabs: Target Mode, Correction Mode, Routine Control, and Status.

8.5.1 Interface Architecture

The GUI employs a multi-threaded architecture to maintain responsiveness during high-frequency RSI communication (83 Hz). A dedicated UI update thread refreshes displays at 5 Hz, decoupled from the main RSI control loop to prevent blocking. This separation ensures that GUI operations which may take 50–100 ms for complex rendering do not interfere with the deterministic 12 ms RSI cycle.

8.5.2 Control Modes

8.5.2.1 Target Mode Interface

Target mode allows operators to specify absolute Cartesian positions. The interface (Figure 8.9) provides:

- Individual axis input fields (X, Y, Z, A, B, C) with unit indicators
- "Use Current" button to capture the robot's current position as the target
- Real-time position error display ($x_{tgt} - x_{curr}$)
- "Set Target" action button to activate proportional control

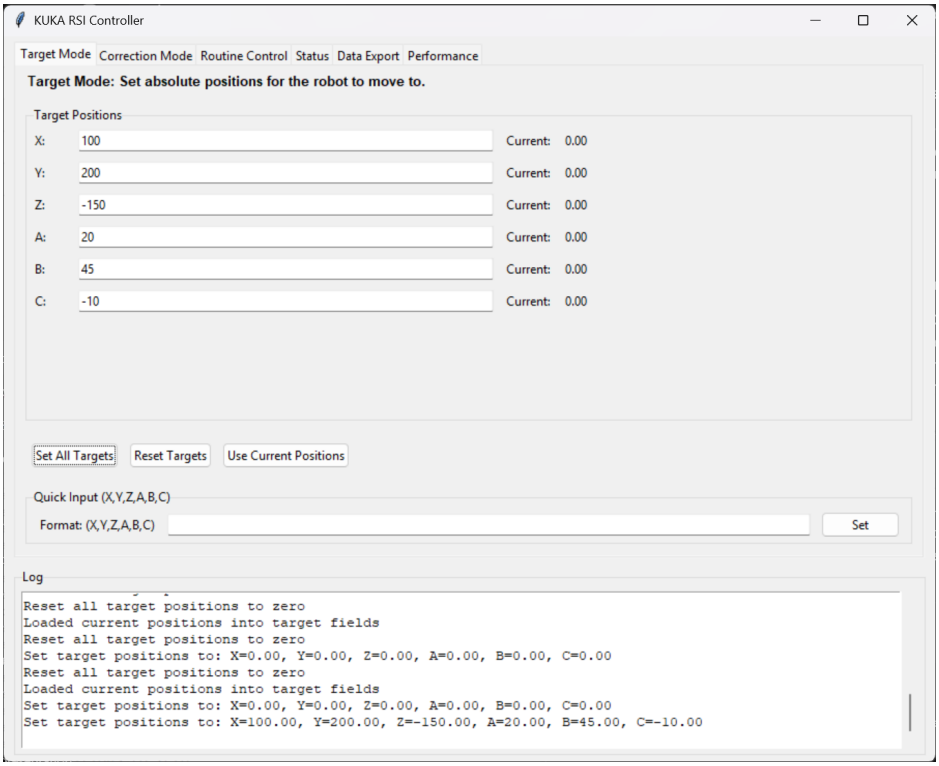


Figure 8.9: The developed Python GUI operating in *Target Mode*. This interface enables users to define absolute target coordinates for the KUKA manipulator. Upon target confirmation, the proportional controller (Algorithm 8.1) computes real-time correction commands that continuously drive the end-effector toward the desired position.

8.5.2.2 Correction Mode Interface

Correction mode provides incremental position adjustment controls:

- Incremental adjustments for each axis
- Real-time display of cumulative corrections relative to reference position
- "Apply Correction" commits pending adjustments with ramp-up
- "Reset" clears all corrections and returns to reference position
- "Stop MOVECORR" exits sensor-guided mode, returning to programmed motion

The interface abstracts the underlying control algorithms, presenting only application-relevant controls. User think in terms of "move the roof 5 mm to the left" rather than "apply +5 mm correction in negative Y direction,".

8.5.3 Status Monitoring

The Status tab provides real-time system monitoring:

- **Connection indicator**: Visual status (green = connected, red = disconnected, yellow = degraded)
- **Current position table**: Live 6-DOF pose with 0.1 mm / 0.01° resolution
- **Mode indicators**: Current RSI mode (superposed / sensor-guided) and control sub-mode (target / correction)
- **MOVECORR flag**: Boolean indicator synchronized with controller's `$MOVECORR` variable
- **Ramp-up factor**: Current value of $\lambda \in [0, 1]$ during correction mode operation
- **Cycle statistics**: Packet count, average cycle time, packet loss rate

The status display updates at 5 Hz, providing operators with sufficient feedback for manual control tasks while avoiding the visual fatigue that would result from 83 Hz updates. Critical events (connection loss, mode transitions, routine completion) trigger immediate GUI updates outside the regular refresh cycle.

8.5.4 Routine Management Interface

The Routine Control tab enables task sequencing through a simple interface to set the routine relevant number (e.g., "1: Roof Pickup", "2: Roof Insertion", "3: Tool Change"). Selecting a routine queues it for execution, with the handshaking protocol (Algorithm 8.5) ensuring reliable communication with the KUKA controller. A status indicator shows the current routine state: Idle (99), Requested (pending acknowledgment), or Active (executing).

8.5.5 Design Rationale

The GUI design prioritizes **simplicity over flexibility**. Rather than exposing low-level parameters (gains, saturation limits, timeout values), the interface presents task-oriented controls that allows easy operation and debugging. Advanced parameters are configured in the Python control client's initialization and are not modifiable through this interface, preventing inadvertent misconfiguration that could compromise safety or performance.

The multi-threaded architecture ensures that even during computationally intensive operations (e.g., exporting data logs), the RSI control loop maintains its 83 Hz cycle without jitter.

8.6 Integration with ROS2 Middleware

8.6.1 Multi-Robot Coordination Architecture

The system employs ROS2 for coordinating the KUKA KR 150 and UR10e robots while maintaining independent low-level control loops. The Python RSI client for the KUKA operates as a ROS2 node, publishing robot state and subscribing to coordination commands, while the UR10e operates through its native ROS2 driver. This distributed architecture enables coordinated task execution without tight coupling between robot controllers.

8.6.2 ROS2 Node Implementation

The KUKA RSI interface is implemented as the `KukaRsiNode` using a multi-threaded architecture with ROS2's `MultiThreadedExecutor`:

- **Main thread**: Handles ROS2 callbacks and service requests
- **Communication thread**: Manages 83 Hz UDP communication with KUKA controller

- **Thread synchronization**: Mutex-protected shared state prevents race conditions

The node publishes robot state information and provides services for external coordination. Table 8.2 summarizes the key ROS2 interfaces.

Table 8.2: ROS2 interfaces provided by the KUKA RSI node.

Interface	Type	Description
Published Topics		
`/current_pose`	`PoseStamped`	Current TCP position/orientation
`/connection_status`	`Bool`	RSI connection state
`/current_step`	`Int32`	Active waypoint number
Services		
`/routine_send`	`RoutineSend`	Queue numbered routines
`/corr_send`	`CorrSend`	Send axis corrections
`/stop_send`	`StopSend`	Trigger emergency stop

Custom message and service definitions are provided in the `test_package_interfaces` package, with robot descriptions adapted from the `kuka_experimental` repository (StoglRobotics-forks, ROS2 Humble branch).

8.6.3 Tool Change Coordination

Automatic tool changing between the vacuum gripper and screwdriver (Phase 5) is coordinated through routine-based sequencing:

1. KUKA moves to tool changer position (teach point) and stops (Routine 5 completes)
2. ROS2 coordinator monitors KUKA's `current_step` topic, detecting arrival at tool changer
3. UR10e executes tool change sequence:
   - Align end-effector with tool interface
   - Mechanical push-in engagement/release (passive, no digital I/O)
   - Retract to safe position

4. Upon UR10e completion, coordinator sends next routine to KUKA via `routine_send` service

5. KUKA resumes motion (Routine 6 begins)

Tool changing between the vacuum gripper and screwdriver is fully mechanical (passive push-in). Coordination relies on routine sequencing: KUKA moves to a predefined tool-change position and stops; ROS2 detects routine completion and triggers the UR10e tool-change sequence. After UR10e finishes, the next KUKA routine is sent via `routine_send`

8.6.4 Proximity-Based Emergency Response

The UR10e's ToF proximity sensors trigger graduated emergency responses through hardwired digital I/O connections to the KUKA controller. Two digital outputs from the UR controller connect to digital inputs 1 and 2 of the KUKA KR C5:

Warning Zone (Input 1): When an obstacle is detected at a safe distance (typically 200–500 mm):

- UR10e sets digital output 1 HIGH → KUKA input 1 triggered
- KUKA reduces velocity to 5% through the `$OV_PRO` system variable (program override)
- Robot continues executing programmed trajectory at reduced speed
- When obstacle clears, UR10e sets output 1 LOW → `$OV_PRO` returns to 100% (within 12 ms)

Danger Zone (Input 2): When an obstacle is detected at critical proximity (typically <200 mm):

- UR10e sets digital output 2 HIGH → KUKA input 2 triggered
- KUKA immediately brakes and holds position
- All motion ceases until obstacle clears
- When safe, UR10e sets output 2 LOW → KUKA resumes motion from stopped position
- Programmed trajectory continues from interruption point (no re-teaching required)

This graduated response strategy (slow → stop) provides defense-in-depth safety. The hardwired digital I/O connection ensures sub-50 ms response latency, independent of ROS2 communication timing. Both warning and danger responses preserve the programmed trajectory—the robot does not abort or require manual repositioning after an emergency event.

Figure 8.10 shows the timing diagram for a typical emergency sequence.

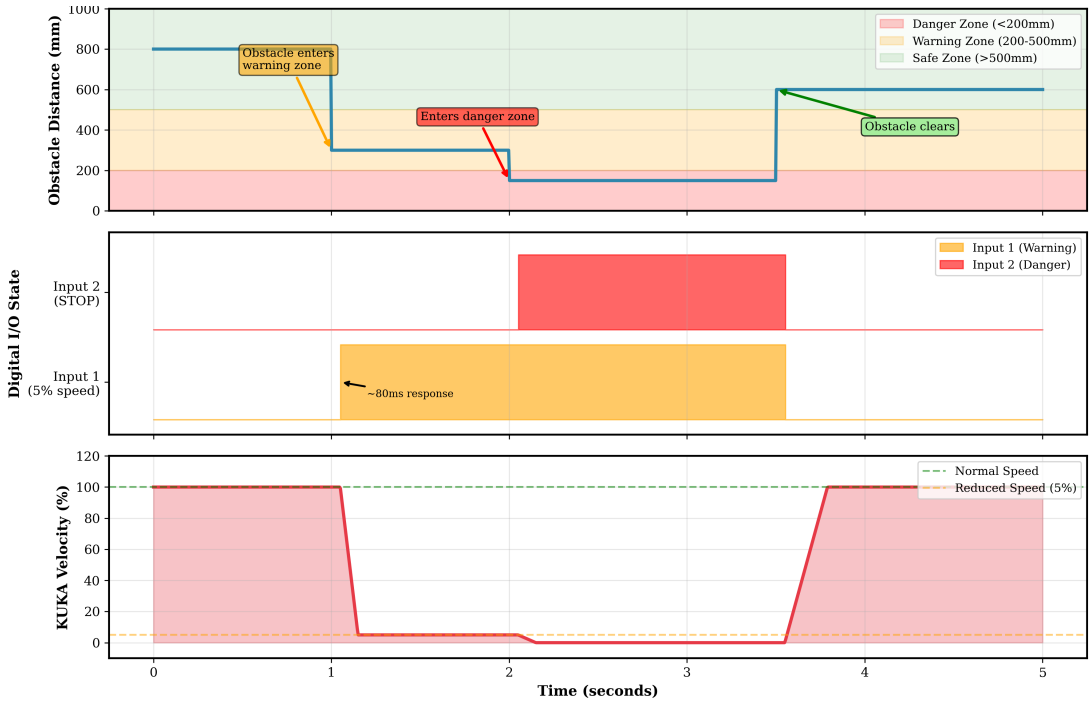


Figure 8.10: Emergency response sequence timing showing graduated response to obstacle detection. When an obstacle enters the warning zone, Input 1 triggers velocity reduction (5% speed). If the obstacle continues into the danger zone, Input 2 triggers complete stop. Total response time from detection to velocity reduction is approximately 50 ms (sensor processing) + 15 ms (digital I/O propagation) + 12 ms (KUKA cycle) approximately 80 ms. The robot resumes normal operation when the obstacle clears, continuing its programmed trajectory without requiring manual intervention.

When the obstacle clears (Input 1 returns to LOW), the KUKA controller restores `$OV_PRO` to 100% within one control cycle (12 ms). The robot then accelerates smoothly back to full programmed velocity according to its configured acceleration limits (typically 1–3 m/s^2), reaching 100% speed in approximately 200–300 ms. This gradual acceleration is handled entirely by the KUKA's internal motion planner and requires no external trajectory profiling.

8.6.5 Task Coordination Timeline

The complete six-phase assembly operation is coordinated through a combination of routine-based sequencing and state monitoring. Table 8.3 summarizes the coordination mechanisms for each phase.

Table 8.3: ROS2 coordination mechanisms across assembly phases.

Phase	KUKA Activity	ROS2 Coordination
1–3	Pickup, transport, insertion	KUKA executes Routines 1–3 autonomously. ROS2 monitors `current_step` for progress tracking.
4	Hold position for worker	KUKA in `RSI_MOVECORR()` mode. Python client (ROS2 node) sends corrections based on GUI commands.
5	Tool change	KUKA stops at tool changer (Routine 5 complete). ROS2 detects completion, triggers UR10e tool change sequence, then sends Routine 6 to KUKA.
6–7	Fastening, return home	KUKA executes Routines 6–7. Proximity sensors (Input 1/2) provide real-time safety via hardwired I/O.

8.6.6 Performance Characteristics

The ROS2 integration adds minimal overhead to the underlying RSI communication:

- **RSI cycle time**: 12 ms (unchanged from standalone Python client)
- **ROS2 publishing frequency**: 10 Hz for state topics (sufficient for coordination)

- **Service call latency**: 15–35 ms (non-critical path, does not affect RSI timing)
- **Emergency I/O latency**: <80 ms (sensor detection to velocity change)
- **Thread synchronization overhead**: <1 ms per cycle (mutex operations)

The multi-threaded architecture ensures that ROS2 callback processing (which may take 20–50 ms for complex operations) does not interfere with the deterministic 12 ms RSI communication cycle. This separation of concerns is critical for maintaining real-time performance while providing standard ROS2 interfaces.

8.6.7 Safety Architecture

The distributed safety architecture provides multiple layers of protection:

1. **Hardware layer**: Hardwired digital I/O for immediate emergency response (Input 1/2)
2. **Controller layer**: KUKA's native safety systems remain active (workspace limits, singularity avoidance, cell limits-configured)
3. **Middleware layer**: ROS2 watchdog monitors connection status, triggers safe stop if communication lost
4. **Application layer**: RSI correction limits (±50 mm) and velocity saturation prevent excessive motion

8.7 Experimental Validation

8.7.1 Test Environment and Setup

Validation experiments were conducted at the University of Genoa robotics laboratory in collaboration with Centro Ricerche Fiat (CRF). The test environment consisted of an authentic Maserati Levante chassis mounted on a stationary fixture, providing realistic geometric constraints and interior dimensions for the roof installation task.

Figure 8.11 shows the complete system configuration. The KUKA KR 150 R3100-2 robot (150 kg payload, 3100 mm reach) is equipped with a UR10e collaborative robot (12.5 kg payload) mounted on its flange via a custom fixture. Moreover, a vacuum gripper is mounted on the UR10e for grasping the soft roof.

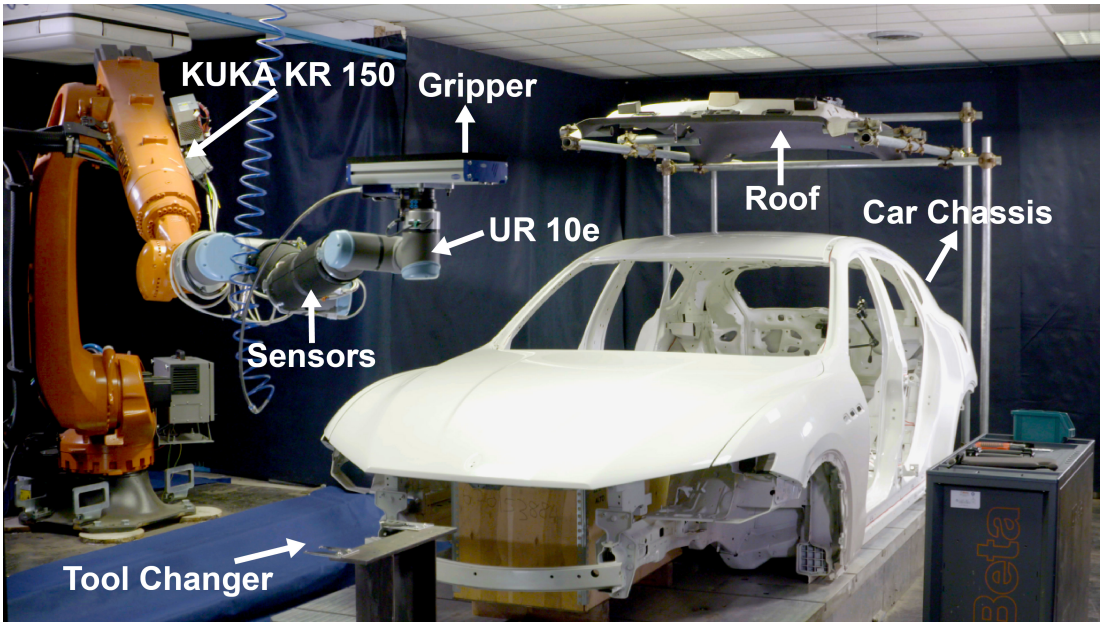


Figure 8.11: Experimental test setup at University of Genoa DIME laboratory. The KUKA KR 150 robot with flange-mounted UR10e operates on a Maserati Levante chassis. The ToF proximity sensors mounted on the UR10e enable obstacle detection. Testing was conducted in collaboration with Centro Ricerche Fiat (CRF) as part of the SESTOSENSO project industrial validation campaign.

The combined system provides the dexterity required for manipulation within the confined vehicle interior while maintaining sufficient reach and payload capacity for handling the roof component.

Table 8.4 summarizes the experimental configuration.

Table 8.4: Experimental test environment configuration.

Component	**Specification**
Primary robot	KUKA KR 150 R3100-2 (150 kg payload, 3100 mm reach)
Secondary robot	Universal Robots UR10e Cobot (12.5 kg payload, 1300 mm reach)
Workpiece	Maserati Levante interior soft roof
Vehicle chassis	Maserati Levante body (white chassis on fixed fixture)
Vacuum Gripper	Schmalz FXP series of vacuum area gripping systems
Tool Changer	Triple A robotics- Wingman tool changer kit
Sensors	ProxySKIN technology based network integrating 60 ToF sensors and over 4,400 pressure sensors on UR10e
Environment	University of Genoa robotics lab, controlled lighting
Collaboration	Centro Ricerche Fiat (CRF), SESTOSENSO project

8.7.2 System Operation Demonstration

The complete six-phase assembly operation was demonstrated successfully through multiple test cycles. Figure 8.12 shows the seven-phase temporal coordination diagram for the collaborative automotive roof insertion task, showing synchronized activities of the KUKA KR 150, UR10e cobot, and human worker from initial pickup through transport, insertion, worker verification with RSI corrections, tool changes, collaborative fastening, and return to home position.

The system successfully demonstrated:

- **RSI dual-mode operation**: Seamless transitions between superposed correction (Phases 1–3, 5–7) and sensor-guided motion (Phase 4) with sub-12 ms mode switching latency

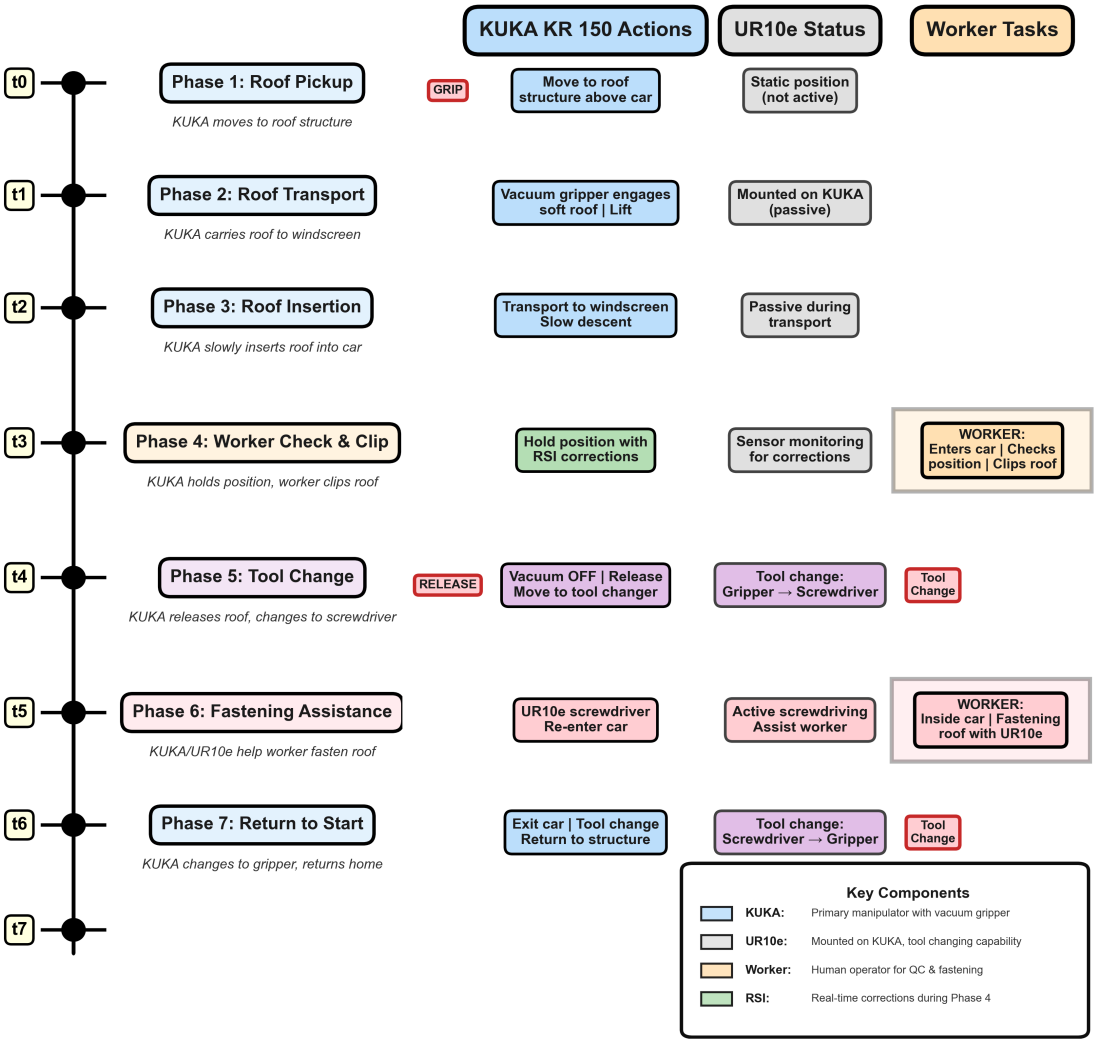


Figure 8.12: Timeline diagram illustrating the seven-phase coordination sequence for the collaborative roof insertion task in automotive assembly. **(a)** Phase 1: KUKA retrieves roof component from overhead storage with vacuum gripper engaged. **(b)** Phase 3: KUKA inserts roof through windscreen opening into vehicle interior, navigating the confined 50 mm clearances. **(c)** Phase 4: KUKA holds position in sensor-guided mode (`RSI_MOVECORR()`) while operator verifies positioning and applies temporary clips. **(d)** Phase 5: Automatic tool change at tool changer station—UR10e switches from gripper to screwdriver. **(e)** Phase 6: KUKA with screwdriver tool assists operator in permanent fastening operations. **(f)** Phase 7: KUKA returns to home position after tool change back to gripper.

- **Operator-guided positioning**: Workers commanded six-degree-of-freedom corrections during Phase 4, achieving positioning accuracy sufficient for manual clipping operations
- **Automatic tool changing**: The UR10e successfully executed gripper-to-screwdriver transitions through mechanical push-in engagement, coordinated via ROS2 routine sequencing
- **Human-robot collaboration**: Workers safely entered the vehicle interior during Phase 4 (holding) and Phase 6 (fastening assistance) without requiring safety cages or light curtains
- **Trajectory persistence**: The `#RELATIVE` correction mode in BASE frame maintained position continuity across all waypoint transitions throughout the 100+ second cycle

8.7.3 Proximity-Based Obstacle Avoidance

The graduated emergency response system (warning zone → danger zone) was validated through controlled obstacle introduction experiments. Figure 8.13 shows the proximity sensor response during operation.

The proximity detection system successfully:

- Detected obstacles in the warning zone (200–500 mm), triggering automatic velocity reduction to 5% through hardwired digital I/O (Input 1)
- Detected critical proximity (<200 mm), triggering immediate stop through Input 2
- Resumed programmed motion automatically upon obstacle clearance without operator intervention
- Maintained trajectory continuity—no waypoint re-teaching or program restart required after emergency events

Total response latency from obstacle detection to velocity change was measured at approximately 80 ms (sensor processing + I/O propagation + controller cycle), well within acceptable bounds for human-collaborative operations where typical human reaction time exceeds 200 ms.

8.7.4 SESTOSENSO Project Demonstrations

The complete system was demonstrated successfully as part of the SESTOSENSO project's final validation campaign, which included:

Figure 8.13: Proximity-based obstacle avoidance demonstration. A worker comes into the KUKA's workspace during programmed motion (Phase 3). The ToF sensors on the UR10e detect the intrusion, triggering graduated response: warning zone detection (200–500 mm) reduces velocity to 5% via Input 1, while danger zone detection (<200 mm) triggers complete stop via Input 2. The robot continues its programmed trajectory after obstacle removal (person leaves the warning zone) without requiring manual repositioning or program restart.

- **Major Demonstration (Project Milestone)**: Full assembly cycle demonstrated to project consortium reviewers, industrial partners (CRF, automotive OEMs), and EU project evaluators. The system completed multiple consecutive cycles showcasing all six phases, RSI mode transitions, automatic tool changing, and human-robot collaboration.
- **Mini Demonstrations**: Focused demonstrations highlighting specific capabilities:
  - RSI correction modes (target and correction) with operator GUI control
  - Proximity sensor integration and graduated emergency response
  - Tool change sequencing and routine management

The demonstrations validated the system's readiness for industrial deployment, with particular emphasis on:

- Robustness to dimensional variations (vehicle body positioning, roof component tolerances)
- Safe human-robot collaboration without physical barriers
- Integration compatibility with existing automotive assembly workflows

Feedback from automotive industry representatives (CRF engineers, assembly line supervisors) confirmed the practical applicability of the approach for variant-rich production with HITL scenarios where traditional fixed-trajectory programming is insufficient.

8.7.5 Observed System Performance

While comprehensive quantitative metrics were not collected during the demonstration phase, the following qualitative performance characteristics were observed:

- **Cycle time**: Approximately 100–120 seconds for complete six-phase operation (varies with native robot speed and operator verification duration in Phase 4)
- **Positioning accuracy**: Operator-commanded corrections in Phase 4 achieved positioning tolerance sufficient for manual clipping operations
- **Communication reliability**: No RSI communication failures observed during demonstration sessions (each 2–3 hours duration)

- **Mode transition smoothness**: No perceptible discontinuities observed during superposed and sensor-guided mode transitions
- **Tool change reliability**: 100% success rate for automatic tool changing across all demonstration cycles (>30 tool changes total)

These observations confirm the system's technical feasibility for the targeted application, though rigorous statistical validation would be required for production deployment.

8.8 Conclusions

This work demonstrates that industrial robots with real-time external interfaces can be adapted for dynamic, human-collaborative assembly tasks through a hybrid control architecture combining programmed motion with sensor-driven corrections.

This work demonstrates that industrial robots equipped with real-time external interfaces can be effectively adapted for dynamic, human–collaborative assembly tasks through a hybrid control architecture that integrates programmed motion with continuous, sensor-driven corrections.

8.8.1 Key Achievements

1. Dual-Mode RSI Control Architecture The developed system employs KUKA RSI in two complementary modes: superposed corrections during programmed motion (Phases 1–3 and 5–7) and sensor-guided control for operator-commanded positioning (Phase 4). A key design insight was maintaining RSI continuously active, rather than enabling it only for discrete phases. This allowed incremental compensation of dimensional deviations throughout the task cycle, enhancing robustness in variable assembly conditions.

2. Practical Implementation for Industrial Deployment The proportional velocity control with saturation ensures smooth, predictable motion, safe for operation in human-shared workspaces. A ramp-up mechanism prevents abrupt starts, enabling operator trust and compliance with collaborative safety standards.

3. Intuitive User Interface The developed Python GUI abstracts the complexity of six-degree-of-freedom control into intuitive, task-level commands. This design prioritizes usability for non-specialist operators, aligning with the goals of Industry 5.0 human-centered automation.

4. Multi-Robot Coordination Architecture Through ROS2-based middleware, seamless coordination was achieved between heterogeneous robotic systems—a KUKA KR150 R3100-2 and a UR10e mounted on its flange while preserving their independent safety and control layers. This architecture enables synchronized, cooperative behavior essential for hybrid assembly tasks.

5. Industrial Validation The full system was successfully validated on an authentic Maserati Levante assembly setup at the University of Genoa in collaboration with CRF as part of the EU Horizon SESTOSENSO project. These demonstrations confirmed both technical feasibility and industrial applicability of the approach in realistic automotive scenarios.

8.8.2 Open Questions and Future Directions

This work focused on feasibility demonstration rather than comprehensive evaluation. Key open questions for future research:

- **Statistical validation**: What are the true success rates, positioning accuracy distributions, and cycle time variability across hundreds of cycles with production-level conditions (worker fatigue, environmental variations, component quality variations)?
- **Generalization limits**: At what point do dimensional variations exceed RSI's correction range, necessitating vision-based localization? How does performance degrade as uncertainty increases?
- **Learning potential**: Can correction profiles be learned from historical data to improve performance on specific vehicle variants or compensate for systematic errors?

8.8.3 Future Enhancements

Several extensions would enhance system capabilities:

Algorithmic Improvements

- Model Predictive Control for multi-step trajectory optimization, particularly during Phase 3 insertion where current reactive control could benefit from predictive path planning
- Adaptive gain scheduling based on distance-to-target or phase context
- Learning-based correction profiles from historical data

Sensor Integration

- Vision-based vehicle body localization for automatic registration, eliminating manual fixture alignment
- Force/torque sensing for contact detection and compliant insertion during Phase 3
- Additional proximity sensors on KUKA for more sensing coverage and added accuracy

Generalization Studies

- Application to non-automotive domains (aerospace assembly, shipbuilding, construction)

Production Readiness

- Statistical validation with 100+ cycles per vehicle variant
- Failure mode analysis and recovery procedures item Establish performance benchmarks

8.8.4 Closing Remarks

The SESTOSENSO demonstration validates that sophisticated adaptive behavior can be achieved with industrial robots while maintaining the safety, reliability, and determinism expected in production environments. The key insight is that real-time external control (RSI) enables flexible automation without requiring complete reimplementation of robot control, the existing safety-certified motion planning, , and workspace monitoring remain active while external corrections compensate for any geometric or environmental uncertainty.

For practitioners seeking to implement similar systems, this work establishes that:

- Standard industrial robots with real-time interfaces (like RSI) are sufficient for adaptive automation
- Conservative control strategies (proportional, velocity-saturated) provide adequate performance for human-collaborative tasks
- Maintaining continuous RSI operation throughout the task cycle maximizes correction capability

- Operator interface design is critical—abstraction and task-oriented controls enable adoption by non-experts
- Reliable multi-robot coordination can be achieved through standard middleware (ROS2)

The SESTOSENSO project demonstrations confirmed industrial interest in this approach for variant-rich production scenarios. While further validation is required before progressing towards production deployment, the research establishes a strong foundation for next-generation, flexible automation architectures that combine safety, adaptability, and human-centric design.

Chapter 9

Conclusions

9.1 Summary of Findings

This thesis addressed the growing demand for robotic manipulation systems that combine mechanical precision, adaptive compliance, and intelligent sensing to meet Industry 4.0 and 5.0 requirements. Two research streams were pursued:

1. The design, enhancement, and sensorization of a universal industrial gripper with hybrid rigid–soft architecture.
2. The development of real-time control strategies for human–robot collaborative assembly within the EU Horizon *SESTOSENSO* project.

The universal gripper integrates a Chebyshev–parallelogram linkage with self-adaptable TPU fingers, achieving near-perfect linear motion (± 0.033 mm) and a mechanical advantage of 6.06:1. Independent actuation via JVL stepper motors and sensorless control strategies enabled reliable grasping without dedicated force sensors. Systematic enhancements transitioning from a four-bar to a six-bar linkage with compliant pads eliminated rigid link interference, improved wrapping capability by 74%, and increased payload capacity from 3 kg to 7.5 kg.

9.2 Key Contributions

- **Hybrid Gripper Architecture Study:** Modular design combining rigid precision and soft adaptability, validated on a COMAU 6-DOF robot.
- **Mechanical Enhancement Study:** Six-bar linkage with compliant pads for adaptive envelope grasps and expanded grasping diameter.

- **Multi-Modal Sensorization:**
  1. **Vision-based sensing with embedded camera** for force estimation (RMSE < 0.54 N), Z-displacement tracking (RMSE < 0.46 mm), slip detection (28.5 Hz), and position classification (95% accuracy).
  2. Resistive sensors via multi-material FDM for contact and bending feedback.
  3. Light-angle sensor array integrated through rigid-flex PCB for distributed tactile sensing.
- **Industrial Control Integration:** Real-time coordination of KUKA KR150 and UR10e robots using KUKA RSI and ROS2 within the EU Horizon SESTOSENSO project, enabling safe human–robot collaboration in automotive assembly.

9.3 Industrial Relevance

The outcomes directly support Industry 4.0/5.0 objectives by enabling flexible, perception-driven automation. The universal gripper demonstrated reliable manipulation of objects ranging from fragile items to heavy automotive components, while sensorization strategies provide a pathway toward intelligent, adaptive manufacturing systems.

9.4 Limitations and Challenges

Despite significant advancements, certain limitations remain:

- Vision-based sensing performance is sensitive to environmental lighting variations.
- Resistive sensors exhibit thermal drift during extended operation.
- Light-angle sensor array integration requires further calibration for dynamic tasks.

9.5 Future Research Directions

Building on this work, future research should focus on:

- Developing robust illumination and enclosure strategies for vision-based sensing.
- Exploring advanced materials and compensation algorithms to mitigate thermal drift in resistive sensors.
- Completing calibration and modeling of light-angle sensor arrays for high-resolution tactile feedback.
- Extending real-time control strategies by incorporating imitation learning approaches and AI-driven adaptive grasp planning.

9.6 Closing Remarks

This research advances the state of the art in hybrid gripper systems by integrating rigid precision, soft adaptability, and intelligent sensing within an industrially validated framework. The thesis contributions pave the way for safer, more efficient human–robot collaboration and flexible automation in next-generation manufacturing environments.

References

(1998). Manipulating industrial robots – Performance criteria and related test methods. Available at: https://www.iso.org/standard/22244.html [Accessed: 2025-06-10]. 49

Abadi, M., Agarwal, A., Barham, P. & et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. tensorflow.org. 122

ABB Robotics (2024). *Application manual: Externally Guided Motion*. ABB AB. 24

Abeywickrama, D., Le, N., Chance, G., Winter, P., Manzini, A., Partridge, A., Ives, J., Downer, J., Deacon, G., Rossiter, J., Eder, K. & Windsor, S. (2023). Soft gripping: Specifying for trustworthiness. 5

AboZaid, Y.A., Aboelrayat, M.T., Fahim, I. & Radwan, A.G. (2024). Soft robotic grippers: A review on technologies, materials, and applications. *Sensors and Actuators A: Physical*. 5

Al Abeach, L.A., Nefti-Meziani, S. & Davis, S. (2017). Design of a variable stiffness soft dexterous gripper. *Soft Robotics*, **4**, 274–284. 14

Ali, M.H., Zhanabayev, A., Khamzhin, S. *et al.* (2019). Biologically inspired gripper based on the fin ray effect. In *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, 865–869. 18

Ambaye, G., Boldsaikhan, E. & Krishnan, K. (2024). Soft robot design, manufacturing, and operation challenges: A review. *Journal of Manufacturing and Materials Processing*, **8**. 8

Avendaño, J.C., Leander, J. & Karoumi, R. (2024). Image-based concrete crack detection method using the median absolute deviation. *Sensors*, **24**. 110

Baker, S. & Matthews, I. (2004). Lucas-kanade 20 years on: A unifying framework part 1: The quantity approximated, the warp update rule, and the gradient descent approximation. *International Journal of Computer Vision - IJCV*. 110

Bannasch, R. & LK (2012). Manipulator tool and holding and/or expanding tool with at least one manipulator tool. U.S. Patent. 17, 30

Bergström, J. (2015). Elasticity/hyperelasticity. In J. Bergström, ed., *Mechanics of Solid Polymers*, 209–307, William Andrew Publishing, Boston. 71, 107

Bhirangi, R., Hellebrekers, T., Majidi, C. & Gupta, A. (2021). Reskin: versatile, replaceable, lasting tactile skins. In *5th Annual Conference on Robot Learning*. 20

Birglen, L. (2015). Enhancing versatility and safety of industrial grippers with adaptive robotic fingers. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2911–2916. 18, 31

Birglen, L. & Schlicht, T. (2018). A statistical review of industrial robotic grippers. *Robotics and Computer-Integrated Manufacturing*, **49**, 88–97. 3

Brown, E., Rodenberg, N., Amend, J. *et al.* (2010). Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences*, **107**, 18809–18814. 16

Chang, C.M., Gerez, L., Elangovan, N. *et al.* (2019). On alternative uses of structural compliance for the development of adaptive robot grippers and hands. *Frontiers in Neurorobotics*, **13**. 16

Collins, J.A., Grady, P. & Kemp, C.C. (2022). Force/torque sensing for soft grippers using an external camera. *arXiv*. 8

Crooks, W., Vukasin, G., O'Sullivan, M. *et al.* (2016). Fin ray® effect inspired soft robotic gripper: From the robosoft grand challenge toward optimization. *Frontiers in Robotics and AI*, **3**. 16

Crooks, W., Rozen-Levy, S., Trimmer, B., Rogers, C. & Messner, W. (2017). Passive gripper inspired by *Manduca sexta* and the fin ray effect. *International Journal of Advanced Robotic Systems*, **14**, 1729881417721155. 18

D'Avella, S., Sundaram, A., Friedl, W., Tripicchio, P. & Roa, M. (2023). Multimodal grasp planner for hybrid grippers in cluttered scenes. *IEEE Robotics and Automation Letters*, **8**, 2030–2037. 7

Demirsoy, M., Atalay, M.E., Bilurdagi, A. & Kutlu, M. (2024a). A low-cost soft gripper for automated pick-and-place systems. *Journal of Smart Systems Research*. 1

Demirsoy, M.S., Atalay, M.E., Bilurdagi, A. & Kutlu, M. (2024b). A low-cost soft gripper for automated pick-and-place systems. *Journal of Smart Systems Research*, **5**, 33–39. 16

Deng, Z. & Li, M. (2021). Learning optimal fin-ray finger design for soft grasping. *Frontiers in Robotics and AI*, **7**. 30

Dinakaran, V.P., Balasubramaniyan, M.P., Le, Q.H. *et al.* (2023). A novel multi objective constraints based industrial gripper design with optimized stiffness for object grasping. *Robotics and Autonomous Systems*, **160**, 104303. 16

Donato, E., Thuruthel, T.G. & Falotico, E. (2024). Towards interpretable visuo-tactile predictive models for soft robot interactions. *arXiv.org*. 19

Dong, S., Yuan, W. & Adelson, E.H. (2017). Improved gelsight tactile sensor for measuring geometry and slip. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 137–144, IEEE. 21

El-Atab, N., Mishra, R.B., Al-Modaf, F., Joharji, L., Alsharif, A.A., Alamoudi, H., Diaz, M., Qaiser, N. & Hussain, M. (2020). Soft actuators for soft robotic applications: A review. *Advanced Intelligent Systems*. 5

Elgeneidy, K., Fansa, A., Hussain, I. & Goher, K. (2020). Structural optimization of adaptive soft fin ray fingers with variable stiffening capability. In *3rd IEEE International Conference on Soft Robotics (RoboSoft)*, 779–784. 30

European Commission (2022). Physical Cognition for Intelligent Control and Safe Human-Robot Interaction | Sestosenso. https://cordis.europa.eu/project/id/101070310, horizon Europe research and innovation programme under grant agreement No 101070310. Start date: 01/10/2022. End date: 30/09/2025. 1

Faris, O., Muthusamy, R., Renda, F., Hussain, I., Gan, D., Seneviratne, L. & Zweiri, Y. (2023). Proprioception and exteroception of a soft robotic finger using neuromorphic vision-based sensing. *Soft Robotics*, **10**, 467–481. 21

Festo (2025). *Adaptive Gripper Finger*. Festo, Esslingen, Germany, product Catalog. 17

Fillamentum (2019). *Flexfill TPU 98A Technical Data Sheet*. Fillamentum, Fillamentum, ver. August 2019. [Online; accessed 24-Jan-2025]. 30

Gasparetto, A. & Scalera, L. (2019). A brief history of industrial robotics in the 20th century. *Advances in Historical Studies*, **8**, 24–35. 23

Georgopoulou, A., Vanderborght, B. & Clemens, F. (2021). Fabrication of a soft robotic gripper with integrated strain sensing elements using multi-material additive manufacturing. *Frontiers in Robotics and AI*, **8**, 615991. 135

Gerez, L., Chang, C.M. & Liarokapis, M. (2020). Employing pneumatic, telescopic actuators for the development of soft and hybrid robotic grippers. *Frontiers in Robotics and AI*, **7**. 15

Govindan, N. & Krishna, K. (2023). A novel hybrid gripper capable of grasping and throwing manipulation. *IEEE-ASME Transactions on Mechatronics*, 1–12. 7

Guerra-Zubiaga, D.A., Franklin, A., Escobar-Escobar, D., Lemley, T., Hariri, N., Plattel, J. & Ham, C. (2022). Computer vision and machine learning to create an advanced pick-and-place robotic operation using industry 4.0 trends. *Volume 2A: Advanced Manufacturing*. 1, 4

Hao, Y., Zhang, S., Fang, B., Sun, F., Liu, H. & Li, H. (2022). A review of smart materials for the boost of soft actuators, soft sensors, and robotics applications. *Chinese Journal of Mechanical Engineering*, **35**, 37. 20

Hegde, C., Su, J., Tan, J.M.R., He, K., Chen, X. & Magdassi, S. (2023). Sensing in soft robotics. *ACS Nano*. 8

Hernandez, J., Sunny, M., Sanjuan, J. *et al.* (2023). Current designs of robotic arm grippers: A comprehensive systematic review. *Robotics*, **12**. 1, 4, 14, 15

Hugo, P.O. (2013). Industrial grippers: State-of-the-art and main design characteristics. 131

Jamil, B., Yoo, G., Choi, Y. & Rodrigue, H. (2022). Hybrid robotic manipulator using sensorized articulated segment joints with soft inflatable rubber bellows. *IEEE Transactions on Industrial Electronics*, **69**, 10259–10269. 7

Ju, Z., Wu, R., Guo, D. & Xu, Y. (2024). Development of a load-bearing, terrain-adaptive hexapod robot with chebyshev-linkage legs. *Journal of Field Robotics*, early Access. 43

Jumet, B., Bell, M.D., Sanchez, V. *et al.* (2022). A data-driven review of soft robotics. *Advanced Intelligent Systems*, **4**, 2100163. 15

Kattan Urrutia, A.A. & Carrasco Bardales, A.M. (2024). Design and development of a soft pneumatic gripper for precise grasping of fragile objects. *Engineering Headway*, **12**, 69–81. 14

Khin, P.M., Yeow, C.H. & Ang, M.H.J. (2024). Hyper-versatile gripping: Synergizing mechanical and machine intelligence of a hybrid robotic gripper. *Advanced Intelligent Systems*, **6**, 2300533. 15

Kim, D., Kim, S.H., Kim, T., Kang, B.B., Lee, M., Park, W., Ku, S., Kim, D., Kwon, J., Lee, H., Bae, J., Park, Y.L., Cho, K.J. & Jo, S. (2021). Review of machine learning methods in soft robotics. *PLOS ONE*, **16**, 1–24. 19, 20

KUKA GmbH (2025). *KUKA.RobotSensorInterface*. KUKA GmbH, for KUKA System Software 5.4, 5.5, 5.6, 7.0. 24

Kumar, V., Fontul, M., Neves, C. & Coelho, P. (2025). Prototyping and characterization of gripper technologies for stiff fabric material. *IEEE Access*, **13**, 107890–107916. 13, 14

Lalegani Dezaki, M., Sales, R., Zolfagharian, A., Yazdani Nezhad, H. & Bodaghi, M. (2023). Soft pneumatic actuators with integrated resistive sensors enabled by multi-material 3d printing. *The International Journal of Advanced Manufacturing Technology*, **128**, 4207–4221. 20

Lambeta, M., Chou, P.W., Tian, S., Yang, B., Maloon, B., Most, V.R., Stroud, D., Santos, R., Byagowi, A., Kammerer, G. *et al.* (2020). Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. *IEEE Robotics and Automation Letters*, **5**, 3838–3845. 21

Lange, R., Erol, S. & Meissner, P. (2021). The digital transformation in manufacturing: How industry 4.0 and ai drive a paradigm shift. In A. Dengel & C. Schlick, eds., *Digital Entrepreneurship*, 257–278, Springer, Cham, this chapter discusses the impact of AI and advanced sensing on manufacturing processes, enabling a transition to more flexible and adaptive systems. 23

Lee, C., Kim, M., Kim, Y.J., Hong, N., Ryu, S., Kim, H.J. & Kim, S. (2017). Soft robot review. *International Journal of Control, Automation and Systems*, **15**, 3–15. 5, 15

Lee, L.Y., Tan, C.P. & Nurzaman, S. (2019). Design and analysis of a gripper with interchangeable soft fingers for ungrounded mobile robots. In *Proceedings of the IEEE International Conference on Cybernetics and Intelligent Systems and Robotics, Automation and Mechatronics (CIS-RAM)*. 30

Lepora, N.F. (2021). Soft biomimetic optical tactile sensing with the tactip: A review. *IEEE Sensors Journal*, **21**, 21131–21143. 21

Leslie, O., Bulens, D.C., Ulloa, P.M. & Redmond, S.J. (2023a). A tactile sensing concept for 3-d displacement and 3-d force measurement using light angle and intensity sensing. *IEEE Sensors Journal*, **23**, 21172–21188. xiv, 150, 151, 154

Leslie, O., Bulens, D.C., Ulloa, P.M. & Redmond, S.J. (2023b). A tactile sensing concept for 3-d displacement and 3-d force measurement using light angle and intensity sensing. *IEEE Sensors Journal*, **23**, 21172–21188. 20

Li, L., Crosby, D., Shuttleworth, M., Argin, O.F., Chen, A.S., Hermann, G., Kay, R. & Weightman, A. (2024). A comparative analysis and scoping review of soft–rigid and industrial parallel rigid grippers. *Advanced Intelligent Systems*, **7**. 15

Li, S., Zhao, H. & Shepherd, R.F. (2017). Flexible and stretchable sensors for fluidic elastomer actuated soft robots. *Mrs Bulletin*, **42**, 138–142. 19

Li, X., Chen, W., Lin, W. *et al.* (2018). A variable stiffness robotic gripper based on structure-controlled principle. *IEEE Transactions on Automation Science and Engineering*, **15**, 1104–1113. 16

Li, Y., Chen, Y., Ren, T. *et al.* (2021). A dual-mode actuator for soft robotic hand. *IEEE Robotics and Automation Letters*, **6**, 1144–1151. 3

Liu, F., Sun, F., Fang, B., Li, X., Sun, S. & Liu, H. (2023a). Hybrid robotic grasping with a soft multimodal gripper and a deep multistage learning scheme. *IEEE Transactions on Robotics*, **39**, 2379–2399. 7

Liu, S.Q. & Adelson, E.H. (2022). Gelsight fin ray: Incorporating tactile sensing into a soft compliant robotic gripper. In *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, 925–931, IEEE. 21

Liu, S.Q., Ma, Y. & Adelson, E.H. (2023b). Gelsight baby fin ray: A compact, compliant, flexible finger with high-resolution tactile sensing. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, 1–8, IEEE. 21

Loo, J.Y., Ding, Z.Y., Baskaran, V.M., Nurzaman, S.G. & Tan, C.P. (2022). Robust multimodal indirect sensing for soft robots via neural network-aided filter-based estimation. *Soft Robotics*, **9**, 591–612, pMID: 34171965. 20

Low, J.H., Khin, P.M. & Han, Q.Q. (2021). Sensorized reconfigurable soft robotic gripper system for automated food handling. *IEEE/ASME Transactions on Mechatronics*, **27**, 3232–3243. 8

Lu, S., Zlatanov, D., Ding, X. *et al.* (2014). A new family of deployable mechanisms based on the hoekens linkage. *Mechanism and Machine Theory*, **73**, 130–153. 42, 43

Luo, Z., Cheng, W., Zhao, T. & Xiang, N. (2024). Intelligent sensory systems toward soft robotics. *Applied Materials Today*. 8

Matthias, B. & Smarra, P. (2022). Collaborative robots in manufacturing: A review of recent developments. *Journal of Advanced Robotics*, **36**, 25–40, this article reviews the development of collaborative robots and their integration into hybrid human-robot workcells to enhance productivity, flexibility, and worker safety. 23

Monkman, G.J., Hesse, S., Steinmann, R. *et al.* (2007). *Robot grippers*. John Wiley & Sons. 13, 14

Muller, A., Aydemir, M., Glodde, A. *et al.* (2020). Design approach for heavy-duty soft-robotic-gripper. *Procedia CIRP*, **91**, 301–305, enhancing design through the 4th Industrial Revolution Thinking. 15, 17

Nof, S.Y. (1999). *Handbook of Industrial Robotics*. John Wiley & Sons, 2nd edn. 23

Norton, R.L. (2014). *Design of Machinery: An Introduction to Synthesis and Analysis of Mechanisms and Machines*. McGraw-Hill, New York, NY, USA, 5th edn. 42, 45

OnRobot A/S (2025). *RG2/RG6 Grippers*. OnRobot A/S, Odense, Denmark, product Manual, ver. 2.1. 16

(OSP), O.S.P. (2024). Tracker video analysis and modeling tool for physics education. https://physlets.org/tracker/. 75

Otti, M., Monsalve, D., Chapelle, F., Bouzgarrou, C. & Lapusta, Y. (2025). Recent improvements in the development of soft grippers capable of dexterous manipulation. *Applied Sciences*, **15**, 275. 15

Pagoli, A., Chapelle, F., Corrales-Ramon, J.A., Mezouar, Y. & Lapusta, Y. (2022). Large-area and low-cost force/tactile capacitive sensor for soft robotic applications. *Sensors*, **22**. 19, 20

Pang, C., Mak, K., Zhang, Y., Yang, Y., Tse, Y.A. & Wang, M.Y. (2021). Viko: An adaptive gecko gripper with vision-based tactile sensor. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 736–742. 21

Park, J., Lee, S., Lee, J. & Um, J. (2020). Gadgetarm—automatic grasp generation and manipulation of 4-dof robot arm for arbitrary objects through reinforcement learning. *Sensors*, **20**. 1

Park, W., Seo, S. & Bae, J. (2019). A hybrid gripper with soft material and rigid structures. *IEEE Robotics and Automation Letters*, **4**, 65–72. 15, 16

Pasupuleti, M.K. (2025). Ai-driven automation: Transforming industry 5.0 withmachine learning and advanced technologies. 4

Prusa Research (2023). Original Prusa i3 MK3S+ 3D printer. Product information and specifications. 30

Qu, J., Cui, G., Li, Z., Fang, S., Zhang, X., Liu, A., Han, M., Liu, H., Wang, X. & Wang, X. (2024). Advanced flexible sensing technologies for soft robots. *Advanced Functional Materials*, 2401311. 20

Raval, S. & Patel, B.K. (2016). A review on grasping principle and robotic grippers. *International Journal of Engineering Development and Research*, **4**, 483–490. 13, 14

Romanov, A.M., Gyrichidi, N. & Romanov, M.P. (2022). A novel gripper with integrated rotary unit and force control for pick and place applications. *Robotics*, **11**. 16

Saeedi, M. & Effatnejad, R. (2021). A new design of dual-axis solar tracking system with ldr sensors by using the wheatstone bridge circuit. *IEEE sensors journal*, **21**, 14915–14922. 135

Schmitt, F., Piccin, O., Barbé, L. & Bayle, B. (2018). Soft robots manufacturing: A review. *Frontiers Robotics AI*, **5**. 5

SCHUNK GmbH & Co. KG (2025). *EGS 25-N-N-B Rotary Gripping Module*. SCHUNK GmbH & Co. KG, accessed: 2025-06-10. 49

Shan, X. & Birglen, L. (2020). Modeling and analysis of soft robotic fingers using the fin ray effect. *The International Journal of Robotics Research*, **39**, 1686–1705. 16, 31

She, Y., Liu, S.Q., Yu, P. & Adelson, E. (2020). Exoskeleton-covered soft finger with vision-based proprioception and tactile sensing. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 10075–10081. 21

Shin, J.H., Park, J.G., Kim, D.I. *et al.* (2021). A universal soft gripper with the optimized fin ray finger. *International Journal of Precision Engineering and Manufacturing-Green Technology*, **8**, 889–899. 17, 31

Shintake, J., Cacucciolo, V., Floreano, D. & Shea, H. (2018a). Soft robotic grippers. *Advanced Materials*, **30**, 1707035. 18

Shintake, J., Cacucciolo, V., Floreano, D. *et al.* (2018b). Soft robotic grippers. *Advanced Materials*, **30**, 1707035. 14, 15

Siciliano, B. & Khatib, O. (2016). *Springer Handbook of Robotics*. Springer, Cham, 2nd edn., this comprehensive handbook includes dedicated sections on the diverse applications of robots beyond industrial settings, such as in logistics, medicine, and service industries, highlighting their increasing presence in unstructured environments. 23

Sirithunge, C., Wang, H. & Iida, F. (2024). Soft touchless sensors and touchless sensing for soft robots. *Frontiers in Robotics and AI*, **11**, 1224216. 20

Suder, J., Bobovský, Z., Mlotek, J., Vocetka, M., Oščádal, P. & Zeman, Z. (2021). Structural optimization method of a finray finger for the best wrapping of object. *Applied Sciences*, **11**, 3858. 73, 74, 139

Sun, T., Chen, W., Li, J., Li, X., Li, X., Meng, Y. & Tian, Y. (2023). A versatile and high-load soft gripper enabled by vacuum-assisted bio-inspired interfacial adhesion. *Smart Materials and Structures*, **33**, 015034. 16

Taylor, I.H., Dong, S. & Rodriguez, A. (2022). Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger. In *2022 International Conference on Robotics and Automation (ICRA)*, 10781–10787. 21

Tennakoon, C.L., Kulasekera, A.L., Gopura, R.A.R.C. & Chathuranga, D.S. (2024). Pneunet based hybrid soft gripper for multi-shape object handling. *IEEE Access*, **12**, 190158–190168. 16

Tran, N.D., Ly, H.H., Nguyen, X.T., Mac, T.T., Nguyen, A. & Ta, T.D. (2024). Hybrid gripper with passive pneumatic soft joints for grasping deformable thin objects. *arXiv preprint arXiv:2410.05789*. 15

Trinh, H.X., Nguyen, H.H., Pham, T.D. & My, C.A. (2024). A novel rigid-soft gripper for safe and reliable object handling. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, **46**, 176. 15

Truby, R.L., Wehner, M.F., Grosskopf, A.K., Vogt, D.M., Uzel, S., Wood, R.J. & Lewis, J.A. (2018). Soft somatosensitive actuators via embedded 3d printing. *Advanced Materials*, **30**. 19

Truby, R.L., Katzschmann, R.K., Lewis, J.A. & Rus, D. (2019). Soft robotic fingers with embedded ionogel sensors and discrete actuation modes for somatosensitive manipulation. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, 322–329. 19

Wang, H., Gao, B., Hu, A. & He, J. (2023). A variable stiffness gripper with reconfigurable finger joint for versatile manipulations. *Soft robotics*. 15

Wang, J. & Chortos, A. (2022). Control strategies for soft robot systems. *Advanced Intelligent Systems*, **4**, 2100165. 14, 15

Wang, Y., Wang, Y., Mushtaq, R.T. & Wei, Q. (2024). Advancements in soft robotics: A comprehensive review on actuation methods, materials, and applications. *Polymers*, **16**. 5

Wang, Z., Furuta, H., Hirai, S. *et al.* (2021). A scooping-binding robotic gripper for handling various food products. *Frontiers in Robotics and AI*, **8**. 4, 13, 16

Welle, M.C., Lippi, M., Lu, H., Lundell, J., Gasparri, A. & Kragic, D. (2023). Enabling robot manipulation of soft and rigid objects with vision-based tactile sensors. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, 1–7. 18, 21

White, E.L., Yuen, M.C., Case, J.C. & Kramer, R.K. (2017). Low-cost, facile, and scalable manufacturing of capacitive sensors for soft systems. *Advanced Materials Technologies*, **2**, 1700072. 19

Whitesides, G.M. (2018). Soft robotics. *Angewandte Chemie International Edition*, **57**, 4258–4273. 135

Wolf, A. & Schunk, H.A. (2018). *Grippers in motion: the fascination of automated handling tasks*. Carl Hanser Verlag GmbH Co KG. 4

Xu, X., Lu, Y., Vogel-Heuser, B. & Wang, L. (2021). Industry 4.0 and industry 5.0—inception, conception and perception. *Journal of Manufacturing Systems*. 4

Yamaguchi, A. & Atkeson, C.G. (2016). Combining finger vision and optical tactile sensing: Reducing and handling errors while cutting vegetables. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 1045–1051, IEEE. 21

Yang, H., Ding, S., Wang, J., Sun, S., Swaminathan, R. & Ho, G.W. (2024). 4. computational design of ultra-robust strain sensors for soft robot perception and autonomy. *Nature Communications*. 19

Yang, Y., Zhu, H., Liu, J., Lu, H., Ren, Y. & Wang, M. (2022). A proprioceptive soft robot module based on supercoiled polymer artificial muscle strings. *Polymers*, **14**, 2265. 19

Yao, J., Fang, Y. & Li, L. (2023a). Research on effects of different internal structures on the grasping performance of fin ray soft grippers. *Robotica*, **41**, 1762–1777. 17

Yao, J., Fang, Y. & Li, L. (2023b). Research on effects of different internal structures on the grasping performance of fin ray soft grippers. *Robotica*, **41**, 1762–1777. 18

Yao, J., Fang, Y., Yang, X., Wang, P. & Li, L. (2024a). Design optimization of soft robotic fingers biologically inspired by the fin ray effect with intrinsic force sensing. *Mechanism and Machine Theory*, **191**, 105472. 17

Yao, J., Wang, P., Guo, S. & Fang, Y. (2024b). Modeling and performance analysis of a trapezoidal section beam for soft robotic fingers using the fin ray effect. *Mechanism and Machine Theory*, **199**, 105673. 30

Yap, H.K., Ng, H.Y. & Yeow, C.H. (2016). High-force soft printable pneumatics for soft robotic applications. *Soft Robotics*, **3**, 144–158. 15

Yoon, J., Jeong, H., Park, J., Gong, Y., Shin, D., Seo, H., Moon, S. & Choi, H. (2025). A three-finger adaptive gripper with finger-embedded suction

cups for enhanced object grasping mechanism. *IEEE Robotics and Automation Letters*, **10**, 915–922. 15

Yoshimoto, S., Sakamoto, K., Takeda, R. & Yamamoto, A. (2024). Design of a high-performance tomographic tactile sensor by manipulating the detector conductivity. *IEEE Transactions on Industrial Electronics*. 20

Yuan, W., Dong, S. & Adelson, E.H. (2017). Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, **17**, 2762. 21

Zhang, B., Xie, Y., Zhou, J. *et al.* (2020). State-of-the-art robotic grippers, grasping and control strategies, as well as their applications in agricultural robots: A review. *Computers and Electronics in Agriculture*, **177**, 105694. 16

Zhang, Z., Yang, H. & Yin, Z. (2023). Gelflow: Self supervised learning of optical flow for vision-based tactile sensor displacement measurement. 21

Zhao, Y., Khaw, C.K. & Wang, Y. (2023). Measuring a soft resistive strain sensor array by solving the resistor network inverse problem. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, 1–7, IEEE. 19

Zhou, S., Li, Y., Wang, Q. & Lyu, Z. (2024). Integrated actuation and sensing toward intelligent soft robots. *Cyborg and Bionic Systems*. 8

Zhu, W., Lu, C., Zheng, Q., Fang, Z., Che, H., Tang, K., Zhu, M., Liu, S. & Wang, Z. (2023). A soft-rigid hybrid gripper with lateral compliance and dexterous in-hand manipulation. *IEEE/ASME Transactions on Mechatronics*, **28**, 104–115. 16