Imperial College London

Sensing and Machine Learning in Popup Soft Robotics

Authors: Aaron Ram

Student ID: 02367356

"A thesis submitted in partial fulfilment of the requirements for the degree of MRes in Medical Robotics and Image Guided Intervention and for the Diploma of Imperial College"

September 15, 2023

Supervisors: Mark S Runciman, James Avery, Jianlin Yang and George Mylonas

Acknowledgements

I would like to thank my project supervisors, George Mylonas, Mark Runciman, James Avery and Jianling Yang, for their guidance and support through each stage of the Individual Project .

Contents

Abbreviations

OFMS Optical Fibre Misalignment Sensor

DOF Degrees of Freedom

EIT Electrical Impedence Tomography

RMSE Root Mean Square Error

MSE Mean Square Error

MAE Mean Absolute Value

OOB Out-Of-Bag

PMMA Polymethyl methacrylate

MIS Minimally Invasive Surgery

PAM Pneumatic Artificial Muscles

LSTM Long-Short-Term-Memory

Abstract

Positional accuracy remains a key challenge that restricts the broad use of soft inflatable actuators in minimally invasive surgery (MIS). To mitigate this, we present a novel Optical Fiber Misalignment Sensor (OFMS), an innovation in Fibre Optic Intensity Modulation (FOIM) technology. Diverging from traditional FOIM methods that use waveguides or fiber Bragg gratings (FBGs), our OFMS exploits the misalignment of two sets of colinear optical fibers attached to the actuator which offers an effective yet simple and cost-efficient methodology for acquiring a contraction-based signal.

Employing biocompatible Polymethyl methacrylate (PMMA) fibers, this low-hysteresis sensing mechanism presents lower safety risks compared to metal-based or capacitive sensors. Our evaluation, conducted in the 50 g – 200 g weight range, revealed notable improvements. This weight range was chosen to encapsulate the range of forces that may be encountered during MIS. For pneumatic actuators, the OFMS, when integrated into a closed-loop system, reduced the average positioning error from 0.507 mm with a standard deviation of 0.441 mm to an error of 0.154 mm with a standard deviation of 0.092 mm. In hydraulic actuators, sensor fusion via a Random Forest algorithm led to a reduction in average contraction error to 0.377 mm with a standard deviation of 0.234 mm, compared to a previous error of 0.590 mm with a standard deviation of 0.361 mm using an open-loop system.

This simple yet effective sensor can be effortlessly incorporated into existing soft robotic systems, offering a pathway to more accurate and cost-efficient MIS and beyond. The low-hysteresis and safety aspects of this system not only fill a crucial gap in the existing literature but also pave the way for more reliable and safer MIS technologies.

8606 words

Introduction

1.1 Background and Significance of the Area

Minimally Invasive Surgery (MIS) has revolutionized the medical field by reducing patient trauma and speeding up recovery times. Soft inflatable actuators has the potential to reduce costs of surgical robots while improving patient care, however one of the enduring challenges that restrict its potential is the issue of positional accuracy in soft inflatable actuators. The challenge stems from the intrinsic compliance of soft materials, which, while beneficial for interacting safely with biological tissues, makes precise control a complex endeavor.[28]

1.2 Actuation Methodologies

In the realm of soft robotics, multiple unique actuation techniques since many rigid-robotic actuation mechanisms cannot be carried over. For instance, Pouch Motors are a category of fluidic soft actuation mechanisms that aim to simplify robot mass production using easily available materials and manufacturing techniques. In order for actuation to occur, the motors must be hermetically sealed with the chambers being formed through heat-sealing. This type of actuator can be used for both linear and rotational motion.

Analytical models, like the one put forth by Niiyama et al.[23], can predict their force and torque

behaviors when regulated by pressure.

Another proposed actuation method[16] for pouch motors allows for wirelessly controlled soft actuators through the use of lasers which heat liquid within the actuator in order to facilitate contraction. While this eliminates the integration of electronics within the actuators themselves, this methodology still contains safety risks for MIS due to the use of such a high energy method of actuation. Moreover, the methodology still has energy efficiency concerns which need to be resolved.

However, high-force application remains a hurdle in the soft robotics, especially for robots made of silicone materials. To address this, researchers have taken inspiration from the antagonistic muscle pairings to create inflatable robotic structures using rigid yet pliable textiles.[5] These structures can modify their rigidity and form to better interact with their surroundings.

Pneumatic Artificial Muscles (PAMs) have the potential to add value to the field of wearable technology and rehabilitative devices due to their light weight and minimized misalignment issues.[11] Despite this, the challenge of creating PAMs that are both highly compact and responsive with low hysteresis remains. Kim et al.[18] tackled this problem with the introduction of a flat fabric PAM (ffPAM), which uses non-stretchable rip-stop nylon material. This design allows for inexpensive large-scale manufacturing while incorporating a fabric sensor that measures the muscle's contraction, displaying linear behavior across varied pressure levels.

1.3 Manufacturing Techniques

Various techniques are available for manufacturing soft robots as well, again with their benefits and shortcomings. Soft robotic systems typically employ malleable materials like elastomers or hydrogels to build actuators designed for gentle interaction with objects. Soft pneumatic actuators are particularly appealing due to their affordability, simplicity, and longevity. Multiple production

techniques for smaller actuators exist, such as photolithography and dip-coating.

Molding remains a popular manufacturing choice, albeit a very delicate process. During molding of catalyzed polymers such as silicone rubbers, unwanted air bubbles can form, weakening the final product. Techniques like vacuum degassing and centrifugal forces can be employed to rectify this. Advanced molding approaches, like rotational or infusion molding, offer possibilities for crafting intricate internal structures and compartments.[32]

A cutting-edge fabrication method involves using laser technology to simultaneously slice and weld thermoplastic polyurethane (TPU) films.[6] This technique uses readily available and cost-effective materials and equipment, resulting in actuators that can be as slim as 70 microns. This is a departure from the design principles of pouch motors, which have two distinct layers—an actuator layer and a structural layer.

Additive manufacturing has grown significantly since its inception in the early '90s and offers unmatched design freedom. Techniques like stereolithography and selective laser sintering allow for complex structures without the need for additional machining of support pieces. Advanced forms of additive manufacturing even allow for multi-material structures, which enable soft-rigid hybrid robotic systems. The emergence of 'smart materials' such as shape memory polymers has led to innovations like 4D-printed or shape-morphing systems.[32]

1.4 Previous Sensing Approaches and Limitations

Several approaches to positional control of soft actuators in MIS have been explored. Hydraulic inflatable robots show great potential for use in MIS due to their higher force-output per size ratio. However, research on open-loop control has shown limitations on achieving higher positional accuracy.[1] Moreover closed-loop positional control methods that have been explored for hydraulic actuators involve Electrical Impedence Tomography (EIT)[8][26][31][] which utilises

metal electrodes. This introduces safety risks and biocompatibility concerns that metal-free solutions would be able to overcome.

Many emerging systems in soft robotics integrate both sensing and actuating functionalities together. Helps and Rossiter[15] introduced a novel concept that involves using a conductive fluid for dual-purpose action—both as an actuating medium and as a strain sensor. Researchers have created two types of these Flexible Fluidic Actuators (FFAs) with proprioceptive capabilities. Actuators such as these can aid in the creation of cost-effective bionic healthcare devices, such as prosthetic limbs.

Lindenroth et al.[20] suggests that a hydraulic, parallel soft robot can be used for instrinsic contact sensing. During testing with 1-DOF actuators, they were able to showcase force-sensing capabilities using volume and pressure data. This approach is similar to the one used in this paper to achieve accurate positional control.

One biocompatible methodology uses a combination of saline and gold electrodes in order to improve the safety of the EIT methodology.[30]. While the results show low hysteresis, there are challenges with corrosion on the electrodes, and the complex and expensive manufacturing process. Both of which we hope to avoid in this research.

Additionally, many piezoelectric sensors have been researched for tactile sensing within soft robots, and the development of soft robot skin.[17][40][25] These promise to give accurate sensing for shape deformation, but still require metal and electrical components, running into the same problem as with EIT.

1.5 Contribution of this Work

To address these challenges, we introduce a novel Optical Fiber Misalignment Sensor (OFMS) based on a low-hysteresis sensing mechanism. Our innovation leverages the misalignment of two sets of colinear optical fibers attached to the actuator and utilizes biocompatible Polymethyl methacrylate (PMMA) fibers to mitigate the risks associated with metal-based sensors.[21] Furthermore, our evaluation has shown significant improvements in both pneumatic and hydraulic actuators when using OFMS, especially within the weight ranges relevant to MIS.

1.6 Research Objectives and Scope

In this dissertation, we focus on achieving position control of soft actuators through an integrated sensing approach. The initial part of the research will concentrate on integrating closed-loop control within pneumatic actuators. As we progress, we aim to utilize machine learning techniques, specifically Random Forest algorithms, to enhance sensor fusion, thereby improving the system's overall performance, especially in hydraulic actuators. [Insert literature on machine learning and sensor fusion if applicable]

1.7 Organization of the Dissertation

The remainder of this dissertation is organized as follows: Chapter 2 offers a thorough review of the sensor development timeline, focusing on the experimentation and refinement of the sensor, as well as the theory behind its functionality. In Chapter 3, we delve into the methodology, detailing the design and fabrication processes of the OFMS, as well as the development of the data collection algorithms and closed-loop control strategies. Chapter 4 serves as the presentation of our exhaustive evaluation and results. Chapter 5 discusses the significance and implications of these research findings, comparing it to similar research, identifying the limitations and future research. Finally, Chapter 6 concludes the research.

Sensor Development

2.8 Foundational Research

In the quest to find effective sensing methodologies for soft inflatable actuators, existing literature provided both insight and inspiration. One notable work utilized LEDs and phototransistors to ascertain the volume and end-effector position in bellows actuators.[10] The principle hinged on the reflection of light from the top of the bellows back to the phototransistors, thus enabling position estimation. This chapter chronicles the journey from this initial inspiration towards the development of an innovative optical fibre misalignment sensor tailored for pouch motors.

2.9 Validation and Learning

The first step was validating the research by conducting a simple version of the bellows experiment. In doing so, we are able to understand the nature of the relationship between light intensity and contraction, as well as the limitations and risks of the data collection.

Figure 2.1: Bellow Example

The experiment conducted first required the manufacturing of the bellow actuator, which was an experiment in multi-layered welding using the laser welding manufacturing method which will be explained in detail within the methodology. In this, I attempted to target an inner circle and outer circle one after the other to create the bellow shape. This task was difficult as the laser would penetrate the target layer and weld the layers underneath, thus creating unwanted welds between each layer. The only method that showed marginal success was to lower the power of the welding laser from 5.5 V to 2.5 V. However, this would still create weaker inter-layer welds. The experience would prove useful however, in utilising effective multi-layer welding in the final iteration of the OFMS system.

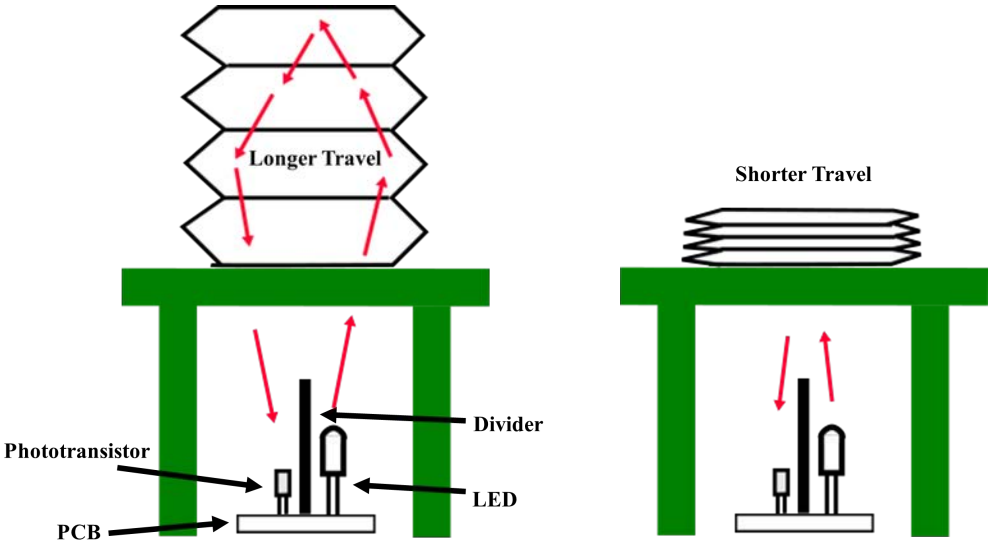


Figure 2.2: Bellows Experiment

The experiment placed the bellow actuator on a 3D-printed ”shelf” which had a hole in the middle. The PCB with the LED and phototransistor would be placed underneath with a divider to avoid non-experimental light detection. As the actuator inflates, the light would have to travel further, thus more was lost to diffusion and reflection. As the bellow deflates and shrinks, the light would travel a shorter distance, and more would be detected by the phototransistor.

While there is a noticeable change in light intensity during inflation and deflation of the bellow, the phototransistor was also heavily influenced by the ambient light. After this discovery, future experiments would have the phototransistor completely isolated to limit the effect of ambient light. This would be done using heat shrinks and a 3D printed cap with only one hole small enough for the experimental light source to enter.

2.10 The Initial OFMS Concept

The next step was an extrapolation of existing technologies into the realm of pouch motors. The hypothesis was straightforward: optical fibres, when connected to an LED and a phototransistor,

would allow light to internally reflect within the pouch motor and be captured by the receiving fibre, thereby providing a similar function as in bellows actuators. However, the empirical data defied this hypothesis, warranting a deeper investigation.

Subsequent research and experimentation revealed a fundamental attribute of optical fibres: co-linearity. The fibres needed to be aligned not just linearly but also at a specific angle to enable effective internal light reflection. This revelation led to a series of experiments aimed at manipulating the light intensity by adjusting the relative angles between the transmitting and receiving fibres while maintaining their ends colinear.

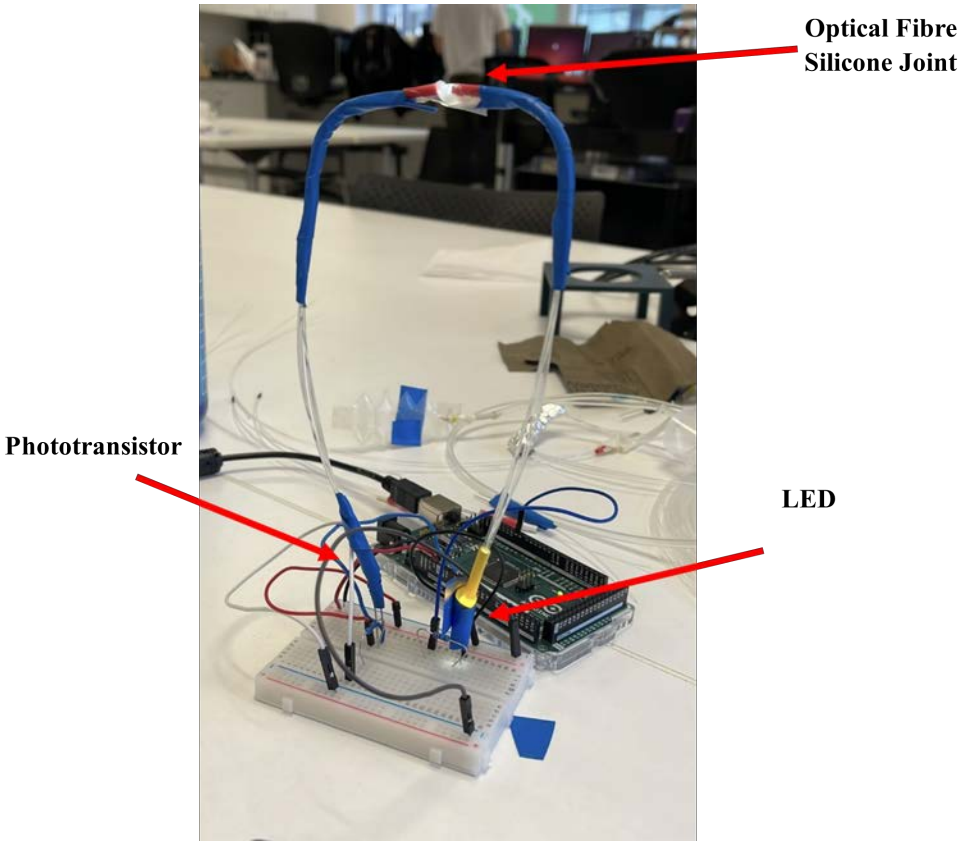


Figure 2.3: First OFMS Prototype

2.11 Material Selection and Challenges

The initial prototypes relied on elastic silicone to hold the fibres together, a choice that introduced hysteresis into the sensor readings. Multiple iterations led to a novel approach—substituting the silicone with a paper support and suspending it in a thin silicone layer moulded on a square paper. Although this reduced hysteresis, the need for further refinement was apparent.

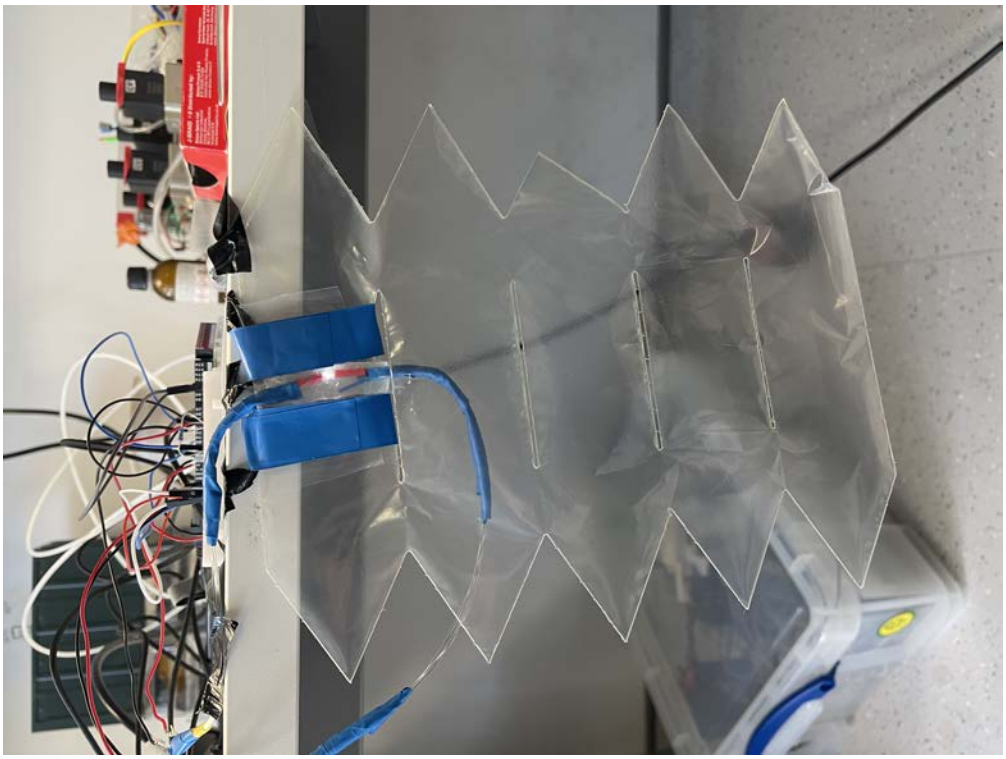


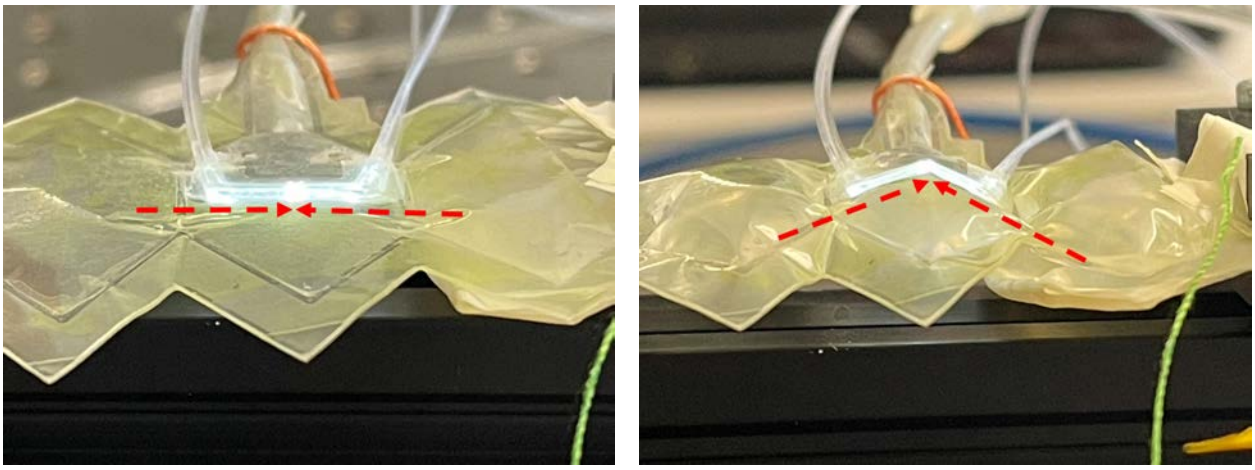
Figure 2.4: First Test Actuator

2.12 Manufacturing Methodology and Refinements

The initial sensor used too many optical fibres, and was too large. Further studies demonstrated that using polyethylene sheets in conjunction with polyolefin could directly attach the fibres to the actuator body. This advancement led to the elimination of silicone and paper from the manufacturing process, culminating in a sensor that was not only more compact but also exhibited increased reliability. The OFMS could now be utilised by smaller actuators than was previously possible.

2.13 Current Functionality of the OFMS

The current version of the OFMS utilises two pairs of optical fibres on both the light transmission and reception side of the system in order to double the amount of light in the system and increase the sensor sensitivity. As the actuator inflates, the point of contact between the fibres misaligns and causes a reduction of detected light for the phototransistor. The values read by the Arduino are voltage readings from 0 V - 5 V that are scaled from 0 - 1023. Maximum values that have been noticed during manufacturing go up to 975, however the OFMS is regularly and repeatably seen operating up to a value of 675-700 due to the inflatable body not returning to a perfect vacuum and remaining flat.



(a) Full Deflation allows Fibre Alignment (b) Inflation Causes Fibre Misalignment

Figure 2.5: OFMS Functionality

Methodology

This section outlines the materials and methods used in manufacturing the actuators and the OFMS. We describe the calibration and validation procedures for the sensors, followed by the data collection and analysis methods. Statistical techniques for evaluating the performance improvements achieved by integrating the OFMS into both pneumatic and hydraulic actuator systems are also discussed. Finally, the section concludes with an overview of the software and code employed in the experiments, as well as relevant validation tests.

3.14 Materials and Tools

We utilized a polyethylene-polyethylene terephthalate-polyethylene triple laminate thermoplastic sheet with a thickness of 30/60/30 μm to construct soft inflatable actuators with a length of 47 mm and a theoretical contractive limit of 11.75 mm. This material was specifically chosen for its unique combination of tensile strength and flexibility, essential properties for inflatable actuators in MIS applications. The OFMS was constructed using PMMA optical fibers with a diameter of 0.75 mm. PMMA was selected for its cost-effectiveness, flexibility, and biocompatibility, which reduces the risk of adverse biological reactions—a key consideration in MIS.

Sensor readings and input control were managed through an Arduino Mega2560 microcontroller. Data collection was facilitated through Python 3.9's CSV library for real-time data logging, while

the SKLearn library was employed for building and testing the Random Forest algorithm models. Ground truth contraction data was collected using OptiTrack's Motive software, chosen for its high accuracy and real-time tracking capabilities. This data was streamed to the Python data collector through a Python NatNet client. Statistical analyses, including t-tests and ANOVA, were performed using Origin 2023b software.

3.15 Actuator Manufacturing

The actuator manufacturing employs a specialized laser welding system, originally developed as part of previous research conducted in our lab. This methodology allows for the rapid production of thin, yet strong pouch motor actuators.[22] The system makes use of a Cartesian robot controlled by an Arduino Mega 2560 microcontroller. This robot focuses a 940 nm laser beam into a 0.8 mm spot to selectively weld thermoplastic sheets. Infrared-absorbing dye is applied to areas targeted for welding to facilitate the process. The system operates at a welding speed of 23 mm/s and a laser current of 6 A.

Before welding, we clean all sheets with isopropyl alcohol to remove debris that could weaken the weld lines. We then apply vacuum pressure of 15kPa to clamp the thermoplastic sheets together for welding.

We first weld the sensor housing mechanism onto one layer of thermoplastic. This housing mechanism is placed across the middle of a sub-actuator. Next, we weld a second layer of thermoplastic to create the actuator body. Finally, we laser-cut the actuators from the sur-

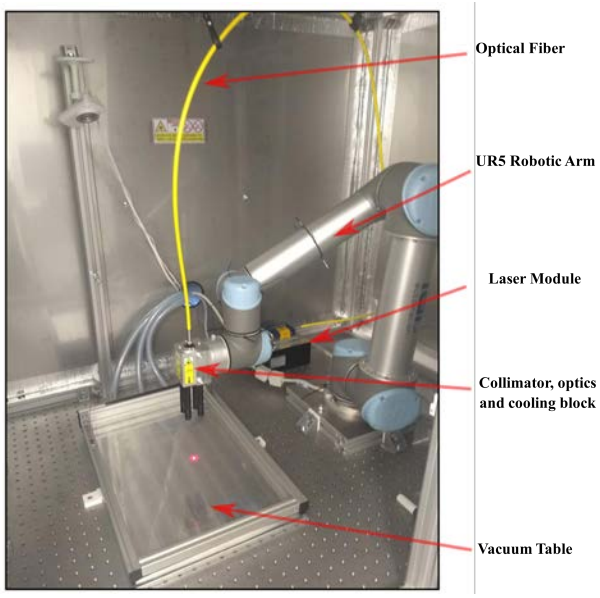


Figure 3.6: Laser Welding System

rounding material.

We verify the quality of each weld through visual inspection to identify any weak or broken weld lines. Weak weld lines will appear faded in comparison to full contact weld lines, and broken weld points will look similar to the surrounding unwelded thermoplastic material.

3.16 Sensor Manufacturing

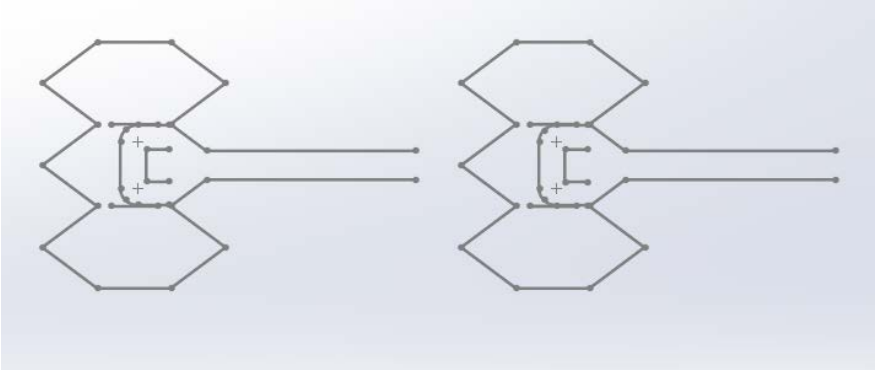


Figure 3.7: Actuator Design

First, we cut the desired length of PMMA optical fibers. Both the transmitting and receiving components of the sensor comprise two optical fibers to enhance light transmission within the system.

The ends of the fibers are meticulously shaved using a scalpel to create smooth, flat surfaces, optimizing contact and light transmission with all interacting components.

Two housing mechanisms for both a white LED and phototransistor are glued to the fibers to ensure optimal and consistent contact with the electrical components. The housing mechanisms also ensure isolation of the phototransistor from ambient light, allowing reliable light transmission across lighting conditions.

The ends of the optical fibers that will enter the actuator are then bent at 90° using a hot water bath of 75°C to fit the fibers within the actuator housing mechanism. To ensure that the fibers stay inside the housing mechanism, a surface primer is applied before superglue is used to stick the fibers to the polyethylene sheets. This is necessary as polyethylene is a low surface-energy plastic.

The fibers are placed so that each side of the housing mechanism contains one transmitting and one receiving fiber, with the opposite side containing the same in opposing format. This provides a more robust sensing of the actuation by placing reception and transmission sections on both sides of the line of deformation.

3.17 Verification of OFMS

In order to validate the performance of the OFMS, we affix it to a mechanical hinge system. The hinge system serves as a controlled environment to simulate a range of motions, from 0 ° to 90 °. The hinge is equipped with a series of 3D-printed wedges, each varying in angle by increments of 5°. These wedges facilitate precise angle adjustments and provide the hinge system with discrete positions for verification of sensor output.

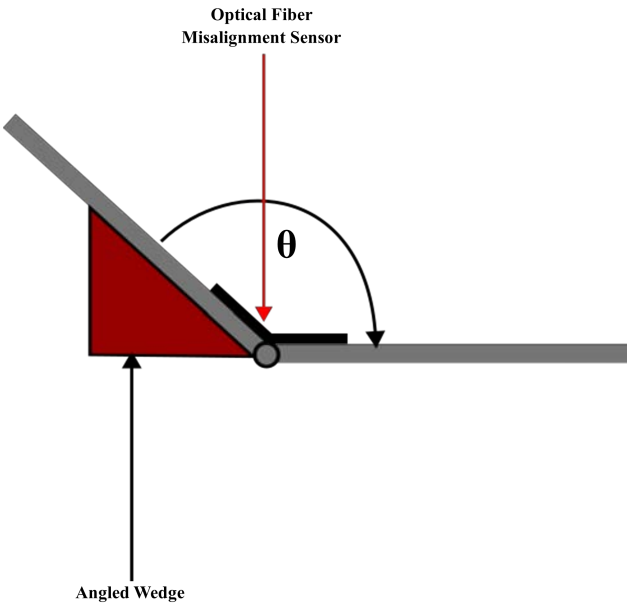


Figure 3.8: Hinge Mechanism

For the verification process, we conduct three independent tests while increasing the angle from 0° to 90° and another three tests while decreasing it back to the original position. Throughout the tests, we continuously record the sensor output values corresponding to each wedge placement. Our aim is to investigate how the light value changes over angular change and whether any hysteresis is observable in the sensor's performance during the angle change.

For data collection, we employ the Arduino Mega2560 with a baud rate of 9600. Data is sampled at 120 Hz for five minutes at each angle for all trials, providing a large and comprehensive dataset for accurate analysis.

To assess the reliability and consistency of the OFMS, we use Origin Pro 2023b to employ a paired t-test to investigate potential hysteresis effects. The pooled standard deviation is calculated to summarize the spread of sensor output across multiple trials. We also calculate 95% confi-

dence intervals to estimate the precision of the mean sensor output. These statistical measures are intended to estimate the reliability and precision of the OFMS.

3.18 Optitrack System Setup

To calibrate the Optitrack system, we utilize two cameras that provide binocular vision. These cameras maintain an exposure setting of 20 µs during the calibration phase. Using a calibration wand, we collect volumetric data points where the testing mechanism will reside, logging 5,000 data points to ensure accurate coordinate measurements down to 0.002 mm. Subsequently, we position a ground plate to establish the base coordinates for the experimental volume.

Our testing mechanism comprises a rail equipped with two markers; we affix one marker firmly to a fixed point on the rail while placing the other on a track mechanism. The actuator connects to both markers, and we mount the rail vertically within the predefined testing volume. In its fully deflated state, gravity draws the actuator down along with the variable marker. Upon inflation, the actuator contracts, pulling up the variable marker. Throughout the experiments, the cameras record the 3D Cartesian coordinates of both markers at a consistent exposure setting of 10 µs, minimizing the risk of false-

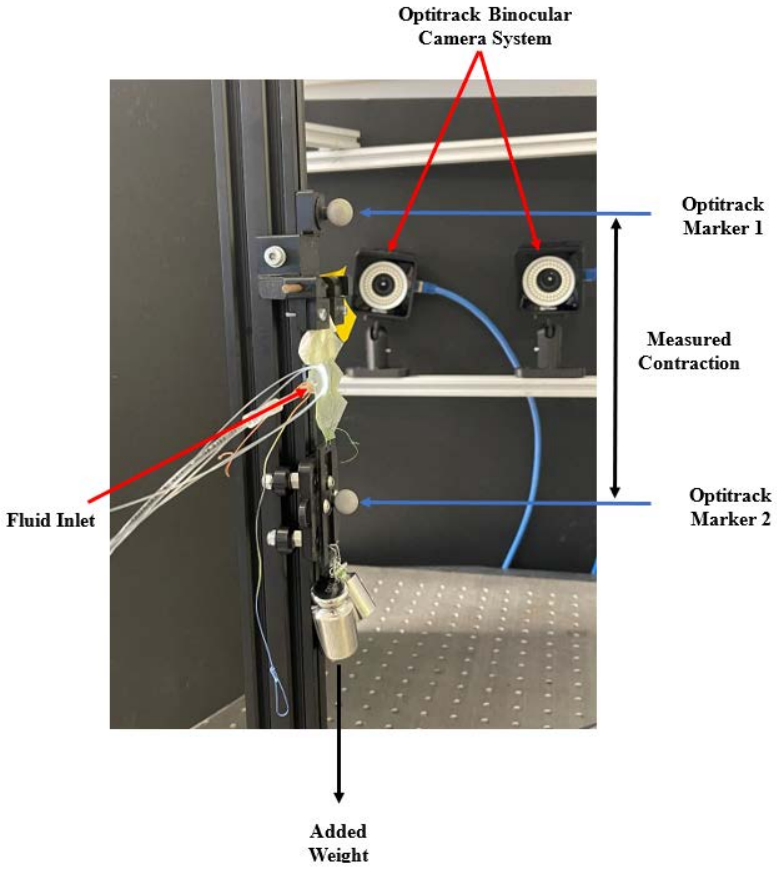


Figure 3.9: Annotated Image Explaining the Optitrack System

positive environmental marker identification.

3.19 Data Collection

We utilize a programmable pneumatic pressure regulator to manage the air pressure supplied to the actuator within an open-loop system. Prior to data collection, we establish a safe upper limit for air pressure at 40 kPa. This limit is set to reduce the likelihood of actuator rupture due to repeated high-pressure cycles. A control algorithm then modulates the air pressure sinusoidally between 0 kPa and the set maximum. Air pressure and OFMS data are captured using an Arduino Mega2560 and relayed to a custom Python-based data logger on a separate computer at a sampling rate of 120 Hz. Concurrently, we acquire marker data from the Optitrack system, also at 120 Hz. This marker data serves to compute the Euclidean distance between the markers, providing a measure of actuator contraction.

[H]

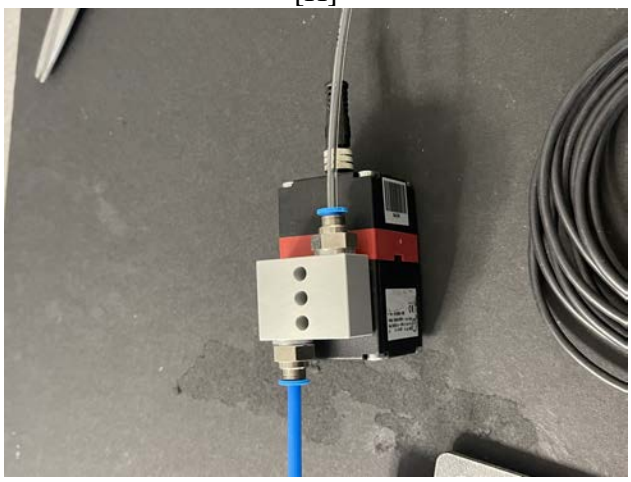


Figure 3.10: Air Pressure Regulator

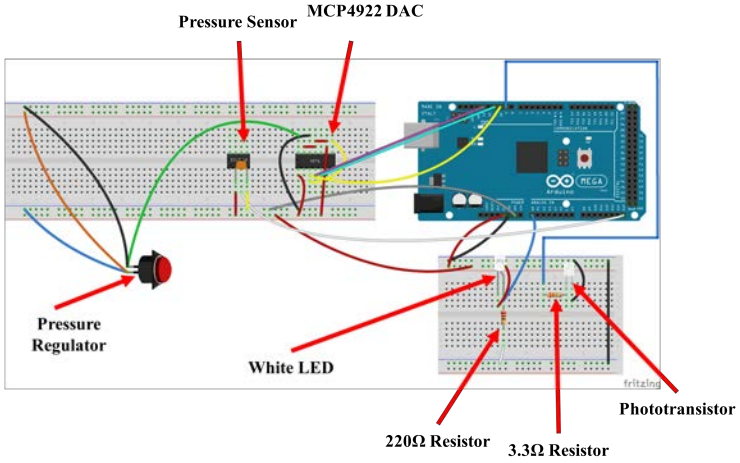


Figure 3.11: Pneumatic Pressure and OFMS Circuit[2]

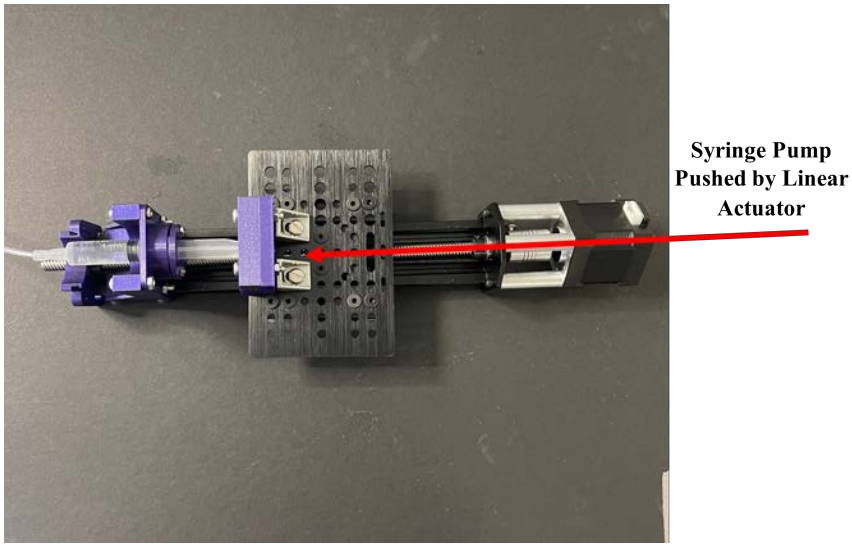


Figure 3.12: Syringe Pump and Linear Actuator

For the hydraulic setup, a syringe pump driven by a stepper motor controls the fluid pressure. Each step of the motor corresponds to a 1.8° rotation. We take meticulous measures to eliminate air bubbles within the hydraulic pathway, extending from the syringe to the water pressure sensor and then to the actuator. To determine the range of syringe volume, we manually rotate the linear actuator to user-defined upper and lower limits. In our study, the stepper motor operates within a range of 855 steps to maintain safe contraction limits.

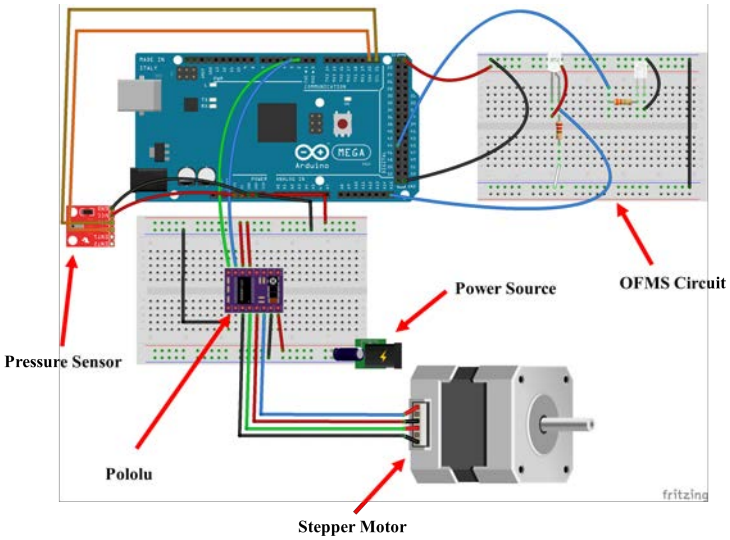


Figure 3.13: Hydraulic Pressure and OFMS Circuit[2]

The step count serves as a proxy for fluid volume and is concurrently sent with water pressure and OFMS data to the data logger. This data undergoes the same logging procedure as the pneumatic system data.

To calibrate the sensors for different actuator sizes, we perform five cycles of sinusoidal inflation and deflation. These cycles are conducted at various actuator loads of 50 g, 100 g, 150 g, and 200 g. The loads are set using physical weights attached to the variable marker rail.

3.20 Open-Loop Systems

3.20.1 Pneumatic System

In our pneumatic open-loop system, we employ an empirical approach to guide actuator contraction. A line-of-best-fit is generated using air pressure data collected across a diverse range of load conditions. This line forms the basis for our open-loop control scheme. We set specific contraction goals for the actuator and adjust the air pressure accordingly to meet them. The efficacy of this approach serves as a benchmark for evaluating the performance of our subsequently developed closed-loop system.

3.20.2 Hydraulic System

Similarly, our hydraulic open-loop system relies on an empirical strategy where stepper motor step counts correlate with actuator contraction levels. A line-of-best-fit is derived from step count data across various testing conditions, serving as the foundation for open-loop control. We establish contraction objectives and use this line to calculate the necessary stepper motor steps to achieve these goals. The observed actuator contractions serve as an additional performance benchmark for our closed-loop system.

3.21 Closed-Loop Systems

3.21.1 Pneumatic System: PID Control

Our goal is to develop a closed-loop model that surpasses the open-loop air pressure model in positional accuracy. To this end, we employ the OFMS to produce a contraction-based signal. We correlate this signal with contraction data across various weight ranges, using a line of best fit generated through OriginPro 2023b's simple fit tool.

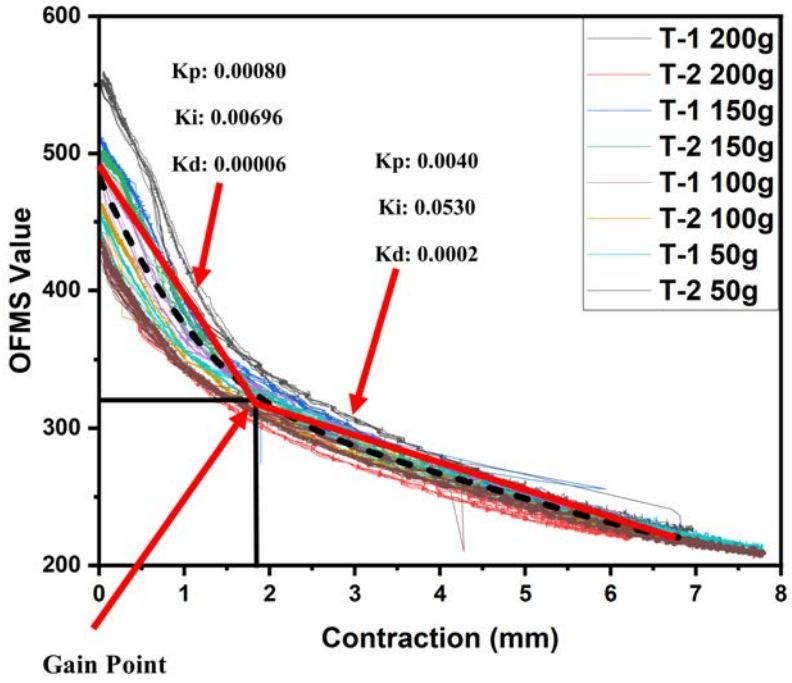


Figure 3.14: OFMS-Contraction with PID gain values

We then establish a PID control system, tuning the Kp, Ki, and Kd parameters with the Zeigler-Nichols method.[4] Recognizing that the relationship between contraction and sensor readings is non-linear, we divide the line of best fit into two linear segments. These segments are differentiated by a threshold OFMS value of 320, identified via user testing and tuning. The choice of this threshold aims to account for non-linearity and is a significant aspect of our methodology.

One limitation of this approach is that it relies on the assumption that the relationship between sensor values and contractions can be approximated as linear within the identified segments. This approximation may not hold under all circumstances, introducing a potential source of error.

In the PID control system, we use one set of Kp, Ki, and Kd values when the current OFMS value exceeds 320 and a different set when it is equal to or below 320. We finely tune these parameters to optimize the system's performance. Importantly, these two linear segments serve as approximate models, manually adjusted to navigate the inherent non-linear relationship between sensor values and contractions.[38]

3.21.2 Hydraulic System: Sensor Fusion

Our objective is to enhance positional accuracy within the hydraulic actuator by implementing a sensor fusion strategy. The advantages of sensor fusion include improved reliability through redundancy, a holistic view of system state, adaptability in prioritizing data from the most relevant sensors, noise reduction, and support for complex algorithms and decision-making.[35]

We adopt the Random Forest algorithm for its various benefits, such as handling large datasets efficiently, robustness to overfitting, minimal pre-processing requirements, high accuracy, and its ability to capture non-linear relationships. The algorithm's built-in cross-validation through Out-of-Bag (OOB) samples further justifies its selection over other potential sensor fusion models.[3]

One challenge in implementing sensor fusion with Random Forest is managing the complexity arising from the combination of various sensor data. This increases the computational power required and prevents the model from being loaded onto the Arduino. This can lead to data transfer errors between the Arduino and the PC that is running the Python code for the model, necessitating a reduction in the transfer rate of data and actuation speed.

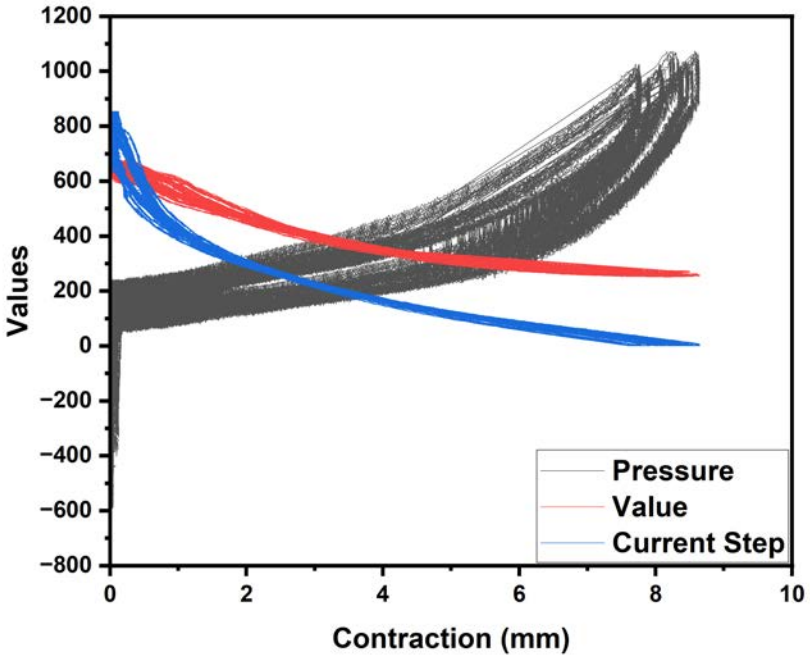
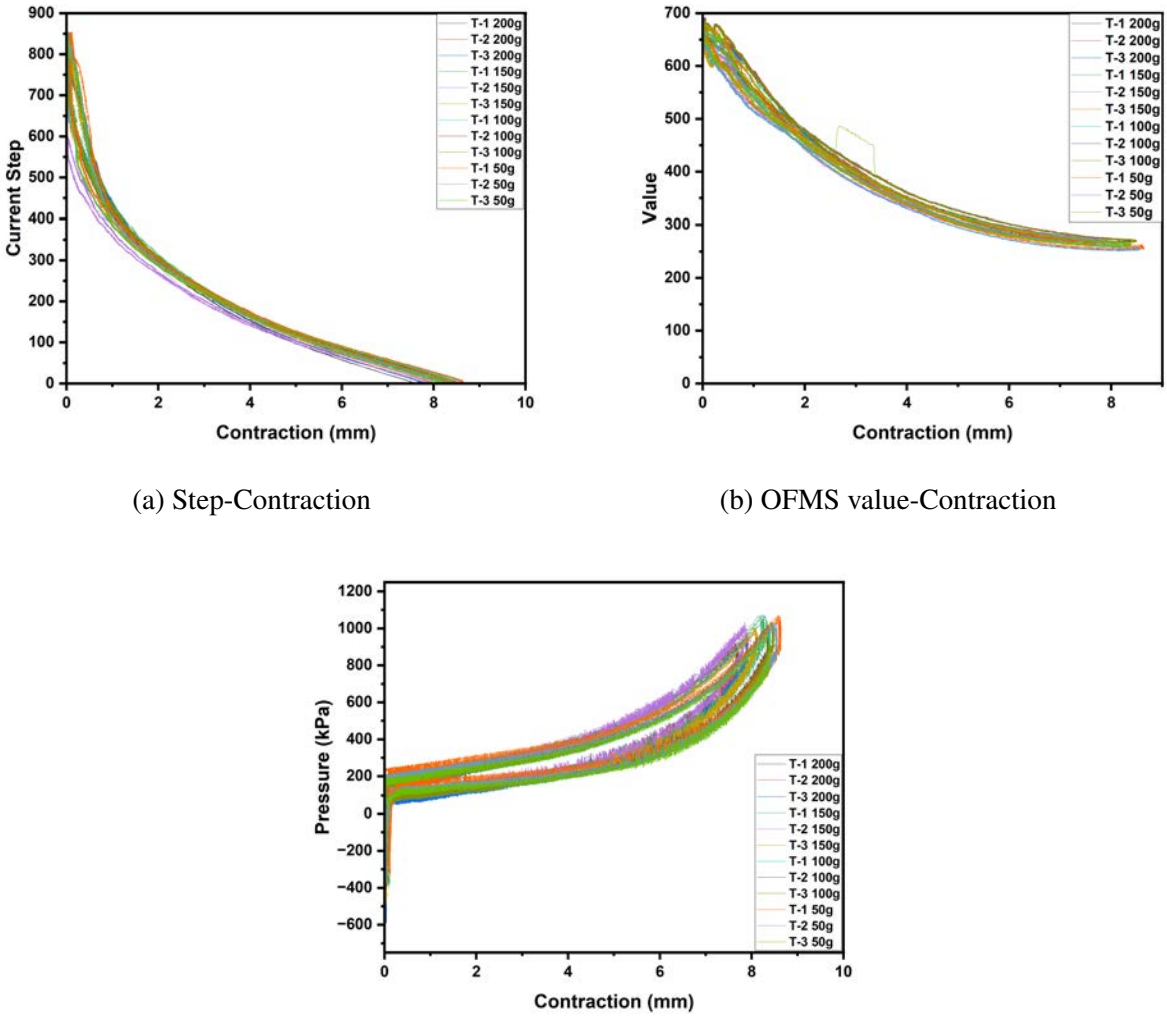


Figure 3.15: All Sensor Data

Using Random Forest, we develop a predictive model that amalgamates data across all weight ranges, incorporating OFMS sensor values, stepper count readings, and liquid pressure metrics. The model then predicts the current contraction level, enabling the control system to adjust the stepper motor steps until the predicted and desired contraction levels align.

3.21.3 Sensor Fusion Data



(a) Step-Contraction

(b) OFMS value-Contraction

(c) Pressure-Contraction

Figure 3.16: Individual Hydraulic Sensor Data

3.21.4 Random Forest Model Justification

Table 3.1: Mean Squared Error Values of Random Forest Models

Condition	MSE Value
No OFMS	0.0200
All Sensors	0.0042
No Pressure	0.0110

From the analysis, it becomes evident that both step count OFMS metrics exhibit robust behavior, evidenced by limited hysteresis during actuator contraction. On the other hand, pressure sensor readings demonstrate a significantly higher level of hysteresis. As shown in the table above, we evaluated the mean squared error (MSE) under three different modeling scenarios: one that excludes OFMS data, a second that incorporates all sensors, and a third that omits pressure sensor data. Notably, the exclusion of OFMS data leads to an almost five-fold increase in the MSE, emphasizing its critical role in the model. In contrast, removing pressure sensor data also results in a non-negligible MSE increment, from 0.0042 to 0.0110. This observation suggests that despite its variability, pressure sensor data contributes valuable information to the model's predictive capability and therefore merits its inclusion.

3.22 Pseudocode and Software

3.22.1 Overview

In our study, Python 3.9 served as the backbone for various computational tasks, including data collection, logging, and the development of the predictive control system for the hydraulic actuator. The Random Forest model was specifically crafted using the scikit-learn (sklearn) library. On the hardware control side, Arduino platforms were employed for multiple purposes. For the pneumatic actuator, an Arduino-based closed-loop control system was established to regulate air pressure and read OFMS values. Similarly, in the hydraulic system, Arduino was utilized to manage the stepper motor. This multi-platform approach allowed us to leverage the strengths of each programming environment: Python for its rich data processing and machine learning libraries, and Arduino for

its real-time control capabilities.

3.22.2 Pneumatic Data Collection

1. **Initialization:** Configure SPI settings, GPIO pins, and essential constants. Then initialize timer constants for loop execution timing.
2. **Setup:** Activate Serial communication for diagnostics and data monitoring. Establish pin modes for SPI and auxiliary controls and initialize the SPI communication protocol.
3. **Supporting Functions:**
    - `writeMCP4922_AB`: To write the desired pressure values to the DAC.
    - `readPressure`: To read current pressure from the sensor.
4. **Main Loop:**
    - Acquire current pressure and OFMS sensor data.
    - Generate a sinusoidal pressure profile within predefined limits.
    - Utilize the DAC function to set the desired pressure.
    - Transmit current time, desired pressure, measured pressure, and OFMS data over Serial for further analysis.

Algorithm 1: Pneumatic Data Collector

Data: Data input
Result: Result output

Initialization;
Initialize all SPI settings, pins, and constants;
Initialize all pressure sensor and DAC constants;
Initialize timer constants;
setup();
Begin Serial communication;
Set pin modes for SPI and other controls;
Initialize SPI;
writeMCP4922_AB(outSelect, desStructPress);
Calculate V_{DAC_BITS} based on desired pressure;
Convert V_{DAC_BITS} to 12-bit integer;
Send via SPI to MCP4922;
readPressure();
Read analog value from pressure sensor pin;
Convert to voltage and pressure;
Return pressure value;
loop();
if *time elapsed greater than loop period* **then**
    Read pressure using readPressure();
    Read and filter sensor value from analog pin A0;
    Generate sinusoidal desired pressure within limits;
    Call writeMCP4922_AB() to set desired pressure;
    Send desired pressure, measured pressure, and OFMS value over serial;
    Set LED pin to HIGH;

3.22.3 Pneumatic Open-Loop Controller

1. **Initialization:** Prepares the SPI settings, GPIO pins, and essential variables. Four distinct sets of predetermined pressure values (i.e., `values50, values100, values150, values200`) are defined and chosen to be looped through in the main loop.
2. **Setup:** Establishes the initial settings for serial communication, pin modes, and SPI.
3. **Supporting Functions:**
    - `writeMCP4922_AB`: To write the desired pressure values to the DAC.
    - `readPressure`: To read current pressure from the sensor.
    - `selectPressure`: To calculate the pressure based on the polynomial equation.
4. **Main Loop:**
    - The current pressure is read and filtered.
    - A target pressure (‘goalpress‘) is set based on one of the predefined value sets.
    - A proportional control approach is employed to reach the ‘goalpress‘.
    - Diagnostic data are outputted to serial for real-time monitoring.
5. **User-defined Sequencing:** The variable n steps through a sequence of predefined pressure values (e.g., `values100, values150`).

Algorithm 2: Pneumatic Open-Loop Controller

Data: Data input
Result: Result output

Initialization;
Initialize SPI, MCP4922 DAQ, GPIO pins, P value, and global variables;
Initialize arrays for predefined pressure values;
Function: setup;
    Initialize Serial Communication;
    Set Pin modes for SPI, MCP4922, and analog read;
    SPI Begin;
    Set initial pressure_baseline value;
Function: writeMCP4922_AB;
    Input: outSelect, desStructPress;
    Output: None;
    Calculate V_DAC_BITS;
    Convert to 12-bit unsigned integer;
    Set appropriate output bit for A or B channel;
    Transmit the value using SPI;
Function: readPressure;
    Input: None;
    Output: Pressure Value in kPa;
    Read from pressure sensor;
    Convert reading to pressure value;
Function: selectPressure;
    Input: x;
    Output: Selected Pressure Value;
    Calculate pressure using polynomial equation based on input x;
Main Loop;
    Track time at start of loop;
    Read current pressure;
    Read and filter sensorValue;
    Set goal pressure based on user-defined values (e.g., values100, values150, etc.);
    Calculate P controller output based on error;
    Cap the output within predefined limits;
    Update prior pressure with desired pressure;
    Write desired pressure to MCP4922;
    Print data to serial output for monitoring;
    If goal pressure is reached;
        If all values have been looped through, break loop;
        Else, increment n and set new goal pressure;
        EndIf;
    EndIf;
End Loop;

3.22.4 Pneumatic Closed-Loop Controller

1. **Initialization:** Sets up SPI communication configurations, general-purpose input/output (GPIO) pins, and fundamental variables. Four specific arrays of preset pressure values (i.e., ‘values50‘, ‘values100‘, ‘values150‘, ‘values200‘) are defined for experimental trials.
2. **Setup:** Initializes the microcontroller’s serial interface, configures pin modes for SPI, and activates the SPI library.
3. **Supporting Functions:**
    - **writeMCP4922_AB:** Sends the desired pressure data to the DAC in accordance with SPI specifications.
    - **readPressure:** Retrieves real-time pressure readings from the sensor.
    - **selectPressure:** Utilizes a polynomial equation to derive pressure based on the OFMS value.
4. **Main Loop:**
    - Pressure measurement and application of a simple low-pass filter.
    - Setting a pressure target (‘goalpress‘) as per one of the predefined value sets.
    - Implementation of a Proportional-Integral-Derivative (PID) algorithm to control pressure and reach ‘goalpress‘.
    - Serialization of performance metrics for monitoring.
5. **User-Defined Sequencing:** Utilizes the variable ‘n‘ to iterate through sequences of predetermined pressure values, such as ‘values100‘ and ‘values150‘, based on control needs.

Algorithm 3: Pneumatic Closed-Loop Control

Data: Data input
Result: Result output

Initialization;
Initialize SPI settings, GPIO pins, and global variables;
Initialize arrays for predefined pressure values;
Function: setup;
    Initialize Serial Communication at 19200 baud;
    Set Pin modes for SPI, MCP4922, and analog read;
    SPI Begin;
    // Set initial pressure_baseline value (optional);
Function: writeMCP4922_AB;
    Input: outSelect, desStructPress;
    Output: None;
    Calculate V_DAC_BITS based on desStructPress;
    Convert V_DAC_BITS to a 12-bit unsigned integer;
    Configure the bit for selecting A or B channel;
    Transmit these values over SPI;
Function: readPressure;
    Input: None;
    Output: Pressure Value in kPa;
    Read analog value from pressure sensor pin;
    Convert analog value to actual pressure in kPa;
    Return pressure value;
Function: selectPressure;
    Input: x;
    Output: Selected Pressure Value;
    Use polynomial equation to calculate and return pressure value based on input x;
Main Loop;
    Record current time in microseconds;
    Read current pressure via readPressure;
    Read and apply a low-pass filter on sensorValue;
    Calculate goal pressure based on a user-defined array (e.g., values100, values150, etc.);
    Perform PID control to calculate the new desired pressure;
    Constrain desired pressure within set limits;
    Update prior pressure with the newly calculated desired pressure;
    Write desired pressure to MCP4922 via writeMCP4922_AB;
    Send data to serial output for monitoring;
    If goal pressure is within a certain threshold;
        If reached the end of the user-defined array, break loop;
        Else, move to the next index in the user-defined array;
        EndIf;
    EndIf;
End Loop;

3.22.5 Hydraulic Data Collector

1. **Import Libraries:** Includes standard SPI, Wire, and custom SparkFun MS5803 libraries for interfacing hardware components.
2. **Variable Definitions:** Global variables for GPIO pins, pressure variables, stepper motor variables, and time variables are initialized.
3. **Setup:**
    - Initializes Serial Communication at 19200 baud rate.
    - Initializes I2C communication.
    - Configures GPIO pins for various purposes like stepper motor control and analog input.
    - Resets and initializes the MS5803 sensor.
4. **Main Loop:**
    - Calculates the time elapsed since the last cycle to determine if another iteration should occur.
    - Fetches pressure data from the MS5803 sensor.
    - Reads and filters analog sensor data.
    - Computes the desired stepper motor position using a sine function.
    - Determines the direction of stepper movement and triggers a single step.
    - Assembles and sends a serialized data packet through the serial port for monitoring.

Algorithm 4: Hydraulic Data Collector

Data: Sensor data, Time data
Result: Pressure regulation, Stepper motor control

Import Libraries;
Include SPI, Wire, and SparkFun MS5803 Libraries;
Variable Definitions;
    Initialize global variables for GPIO pins, time intervals, stepper positions, and pressure variables;
Function: setup;
    Initialize Serial and I2C Communications;
    Set GPIO pin modes;
    Reset and initialize MS5803 sensor;
Function: singleStep;
    Input: None;
    Output: Single stepper motor step;
    Toggle STP pin to execute a single step;
Main Loop;
    Calculate time elapsed since last cycle;
    If time for next iteration;
        Fetch pressure from MS5803 sensor;
        Read and apply filter on analog sensor value;
        Compute desired stepper position based on sine function;
        Determine stepper direction;
        Execute a single step;
        Prepare and send data packet via Serial port for monitoring;
    EndIf;
End Loop;

3.22.6 Hydraulic Open-Loop Control

1. **Initialization:** Import necessary libraries and declare all constants, flags, and variables required for the program. Initialize key parameters.
2. **Setup:** Commence serial communication and I2C bus. Initialize the pressure sensor and digital pin configurations for the stepper motor.
3. **Supporting Functions:**
    - **calculateStepperPosition:** Compute the stepper motor position from a given input using a quadratic equation.
    - **singleStep, increaseStep, decreaseStep:** Execute a single step in the stepper motor, either increasing or decreasing the step count based on the direction.
4. **Main Loop:**
    - Time-keeping for a controlled loop frequency.
    - Analog read and filtering of a sensor value, updating a setpoint during initial cycles.
    - Conditional branching to increase or decrease the stepper motor position based on desired position.
    - Iteration through predefined sequences of values to set desired positions for the stepper motor.
    - Serialization of performance metrics such as pressure, desired and current stepper position, and timing.

Algorithm 5: Hydraulic Open-Loop Control

Data: User defined pressure sequences

Result: Controlled stepper motor positions and recorded metrics

Initialization;
Import libraries; Declare constants, flags, and variables;
Function: setup;
    Begin Serial and I2C communication;
    Initialize pressure sensor and digital pins;
Function: calculateStepperPosition;
    Input: x;
    Output: Calculated stepper motor position;
    Use a quadratic equation to calculate motor position;
Function: singleStep, increaseStep, decreaseStep;
    Input: Direction for increaseStep and decreaseStep;
    Output: Increment or decrement in stepper position;
    Execute a step and update step count and related metrics;
Main Loop;
    Control loop frequency using micros();
    Perform sensor reading and filtering;
    If in initial cycles, update setpoint;
    Control stepper motor based on desired position;
    Iterate through predefined sequence to set desired positions;
    Serialize performance metrics;

3.22.7 Hydraulic Random Forest Algorithms

1. **Initialization:** Import requisite libraries and define essential variables such as ‘DIR‘, ‘STP‘, ‘M2‘, ‘M1‘, ‘M0‘, and PID coefficients (‘Kp‘, ‘Ki‘, ‘Kd‘).
2. **Setup:** Set up serial communication at 6400 baud rate and initialize pressure sensor and digital pins for stepper motor.
3. **Supporting Functions:**
    - **calculateLightValue and calculateStepperPosition:** Use polynomial equations to find light values and stepper motor positions from an input float.
    - **read_float_from_serial:** Read a 4-byte float from the serial buffer.
    - **is_checksum_valid:** Authenticate checksum for received serial data.
    - **singleStep, increaseStep, decreaseStep:** Execute stepper motor steps while updating the step count and related metrics.
4. **Main Loop:**
    - Manage loop frequency using ‘micros()‘.
    - Filter sensor readings for setpoint adjustments in the initial cycles.
    - Utilize conditional logic to control stepper motor based on target positions.
    - Serialize performance metrics such as pressure, stepper position, and time.

Algorithm 6: Hydraulic Random Forest (Arduino-Side)

Data: Sensor Values, Pressure Data, PID Coefficients
Result: Controlled Stepper Motor Positions, Recorded Metrics

Initialization;
    Import required libraries; Declare constants, flags, and variables;
Function: setup;
    Begin Serial and I2C communication;
    Initialize pressure sensor and digital pins;
Function: calculateLightValue;
    Input: Intensity, x;
    Output: Calculated light value;
    Use a quadratic equation to calculate light value based on x;
Function: calculateStepperPosition;
    Input: Pressure level, x;
    Output: Calculated stepper motor position;
    Utilize a quadratic equation to deduce the stepper motor position;
Function: read_float_from_serial;
    Input: None;
    Output: Float value from serial input;
    Read 4 bytes from the serial buffer and combine into a float;
Function: is_checksum_valid;
    Input: Float value, checksum;
    Output: Boolean for checksum validity;
    Validate the received float against the received checksum;
Function: singleStep;
    Input: None;
    Output: None;
    Pulse the stepper motor for one step;
    Update internal step count;
Function: increaseStep and decreaseStep;
    Input: Direction for increaseStep and decreaseStep;
    Output: Increment or decrement in stepper position;
    Change stepper motor position;
    Update sensor values and timing metrics;
    Serialize these data and send over serial;
Main Loop;
    Control loop frequency using micros();
    Perform sensor reading and filtering;
    If in initial cycles, update set point;
    Control stepper motor based on desired position;
    Iterate through predefined sequence to set desired positions;
    Serialize performance metrics;

1. **Initialization:** Import necessary Python libraries such as matplotlib, numpy, serial, and others. Create a queue for data storage.
2. **Setup:** Set up a client for NatNet tracking and initialize serial communication with 'COM3' at 6400 baud rate.
3. **Supporting Functions:**
    - **read_data:** Reads data from serial and NatNet tracking, validating and storing the data into a queue.
    - **Model Initialization:** Load the Random Forest model for predictive operations.
4. **Main Loop:**
    - Initialize plotting elements and list variables for storing sensor and predicted data.
    - Monitor and control the contraction and goal values based on the Random Forest model.
    - Serialize the collected data and plot real-time graphs.
    - Manage repeated data collection cycles and handle system termination after completion.

Algorithm 7: Hydraulic Random Forest (Python-Side)

Data: Sensor and NatNet Marker Data
Result: Real-time Plots, Predictions, and Collected Metrics

Initialization;
    Import Python libraries; Declare essential variables and initialize data queue;
Function: setup;
    Initialize NatNet Client; Begin Serial communication;
Function: read_data;
    Input: Queue, Serial object, NatNet object;
    Output: Filled data queue;
    Read and validate data from Serial and NatNet; Store in data queue;
Function: Main Loop;
    Initialize plotting variables; Load Random Forest model;
    Control contraction and goal values based on Random Forest model;
    Perform real-time plotting and serialize data for collection;
    Handle repeated cycles and system termination;

3.23 Statistical Methods and Validation

To establish the accuracy and reliability of our OFMS sensor, we conducted paired t-tests and calculated confidence intervals. For our closed-loop systems, we compared their performance with the open-loop versions by calculating the average error and its standard deviation. We also computed the relative improvement in error to specifically quantify the advantages of employing our closed-loop methodologies. The fidelity of both the closed-loop and open-loop systems was assessed using the Root Mean Squared Error (RMSE), offering an additional dimension of comparison.

3.24 Conclusion to Methodology

In summary, this study presents a comprehensive methodology that integrates robust actuator manufacturing techniques, advanced sensor development, and innovative control systems for both pneumatic and hydraulic actuators. Employing materials chosen for their specific qualities, the actuators and OFMS sensors were crafted to cater to MIS applications. The employment of statistical analyses and machine learning models contributes to the reliability and robustness of our system, offering insights into potential avenues for improvement. A detailed overview of both open-loop and closed-loop systems was provided, including the challenges and limitations encountered, setting the stage for future research. The multi-platform software environment, consisting of Python and Arduino, was pivotal in achieving the desired outcomes and stands as a testament to the interdisciplinary nature of this work. The methods presented herein form a strong foundation for upcoming evaluations and applications, potentially revolutionizing minimally invasive surgical procedures by enhancing accuracy and reliability.

Results

4.25 Sensor Performance

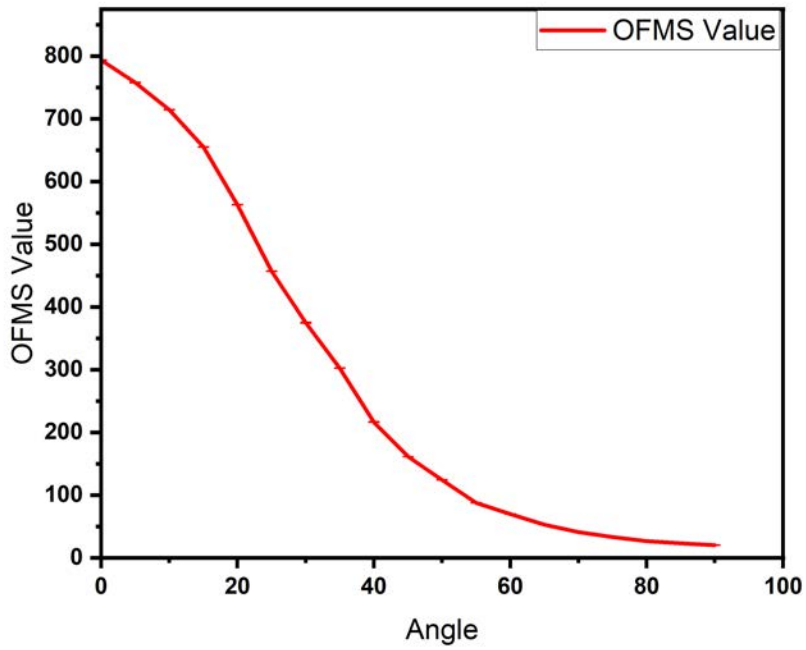


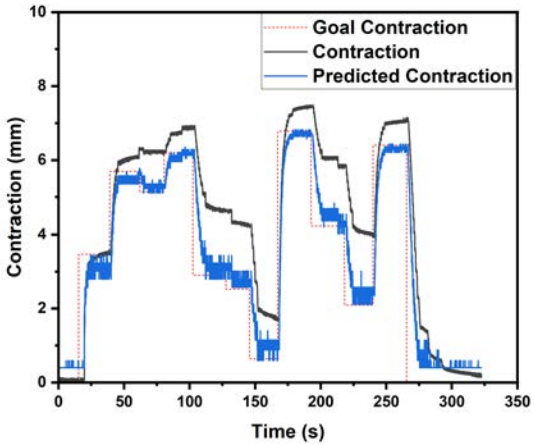
Figure 4.17: OFMS Value over Angular Change

In evaluating sensor performance, we conduct a paired t-test to assess potential hysteresis in the dataset —a critical parameter for ensuring reliable, repeatable control in medical robotics. The test compares sensor readings during angle increments from 0 ° to 90 ° and decrements back to

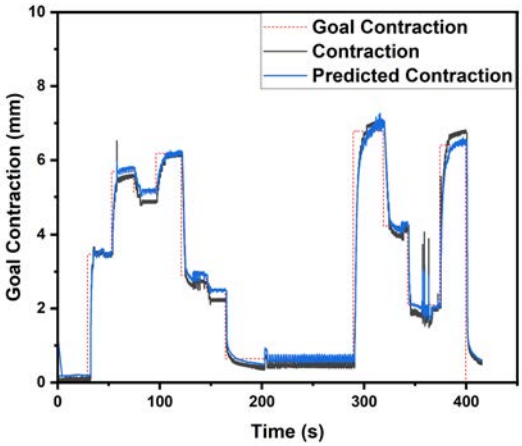
0 °. The resulting t-statistic is 1.041, with 56 degrees of freedom, and a p-value of 0.30235. These results confirm that, at a 0.05 significance level, no significant difference exists between the population means for upward and downward angle changes. This validates the consolidation of the data into a single dataset, affirming the absence of hysteresis. Further, we calculate confidence intervals at 5-degree increments, using the grand mean of all repeats and the pooled standard deviation. This analysis offers additional insight into the sensor's consistent performance across diverse operational conditions. As illustrated in 4.17, the value-angle graph of the OFMS reveals an inverse sigmoid function, with negligibly small error bars due to the confidence intervals being less than 1 on either side of the mean OFMS value at each angle.

4.26 Pneumatic Performance

4.26.1 50 g Test Data



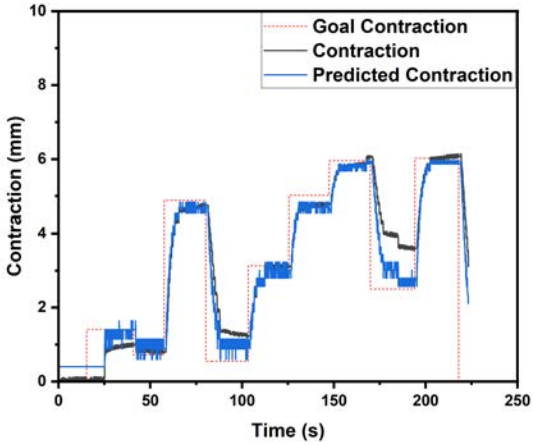
(a) Open Loop
Error: 0.857 ± 0.543 mm



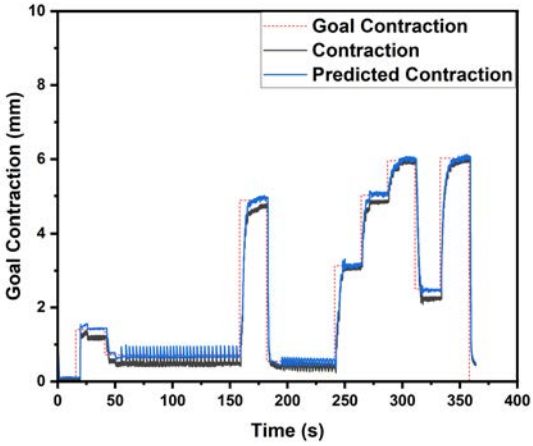
(b) Closed Loop
Error: 0.168 ± 0.087 mm

Figure 4.18: Test Data for 50 g

4.26.2 100 g Test Data



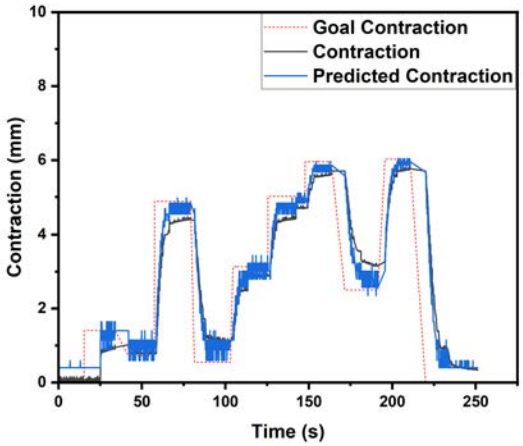
(a) Open Loop
Error: 0.284 ± 0.286 mm



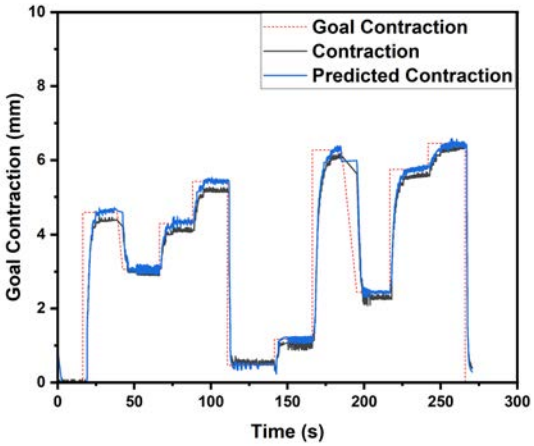
(b) Closed Loop
Error: 0.183 ± 0.080 mm

Figure 4.19: Test Data for 100 g

4.26.3 150 g Test Data



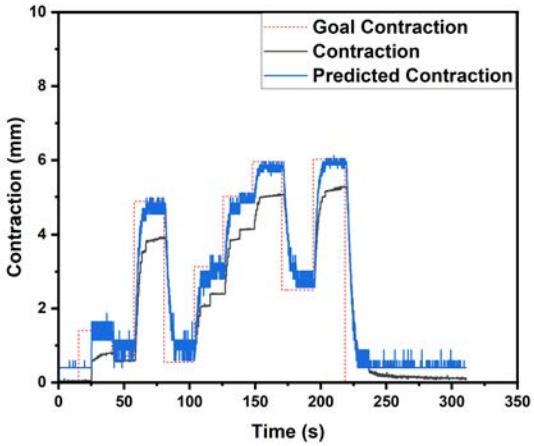
(a) Open Loop
Error: 0.276 ± 0.164 mm



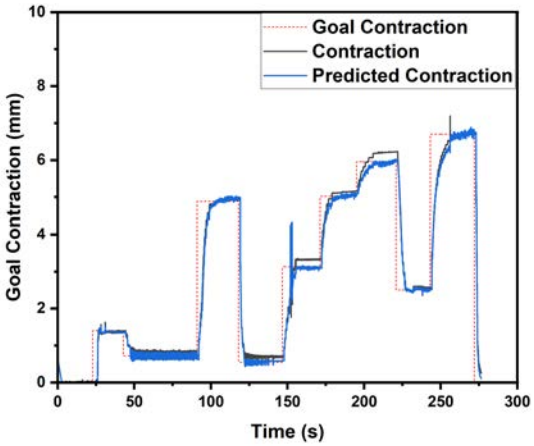
(b) Closed Loop
Error: 0.144 ± 0.092 mm

Figure 4.20: Test Data for 150 g

4.26.4 200 g Test Data



(a) Open Loop
Error: 0.451 ± 0.284 mm



(b) Closed Loop
Error: 0.104 ± 0.094 mm

Figure 4.21: Test Data for 200 g

4.26.5 RMSE

The following table summarizes the RMSE values for open-loop and closed-loop control for all tested weights in pneumatic actuators:

Table 4.2: RMSE

Weight	Open-Loop	Closed-Loop	Relative Improvement
50 g	1.014 mm	0.189 mm	81.4%
100 g	0.403 mm	0.200 mm	50.4%
150 g	0.321 mm	0.171 mm	46.8%
200 g	0.533 mm	0.140 mm	73.7%

4.26.6 Statistical Analysis

To further scrutinize the performance improvement, paired t-tests were conducted for the RSME of individual waypoints across all weights. The results show a statistically significant improvement in the closed-loop system compared to the open-loop system ($p < 0.05$), with a t-statistic of 5.81941, 37 degrees of freedom, and a p-value < 0.0001 strengthening the argument for its robustness.

4.26.7 Error Analysis

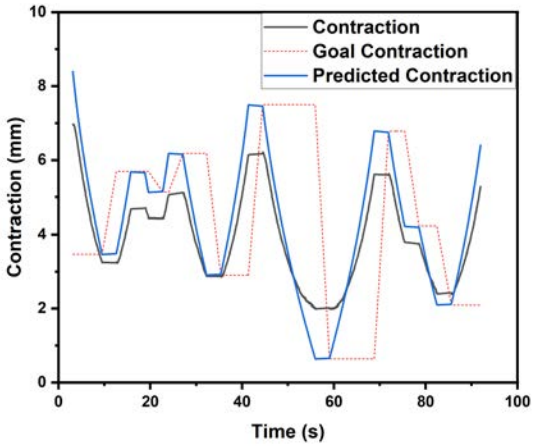
The mean absolute error (MAE) and Standard Deviation for each weight category are summarized in the table below for both open-loop and closed-loop control:

Table 4.3: Error Table

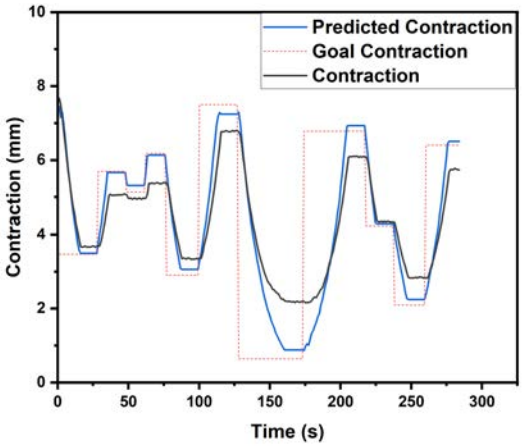
Weight	Open-Loop	Closed-Loop
50 g	0.857 ± 0.543 mm	0.168 ± 0.087 mm
100 g	0.284 ± 0.286 mm	0.183 ± 0.080 mm
150 g	0.276 ± 0.164 mm	0.144 ± 0.092 mm
200 g	0.451 ± 0.284 mm	0.104 ± 0.094 mm

4.27 Hydraulic Performance

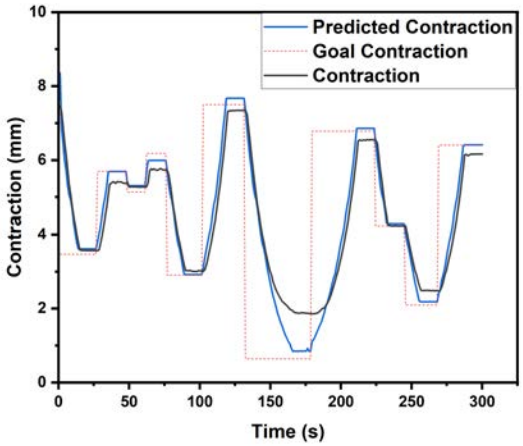
4.27.1 50 g Test Data



(a) Open Loop
Error: 0.608 ± 0.331 mm



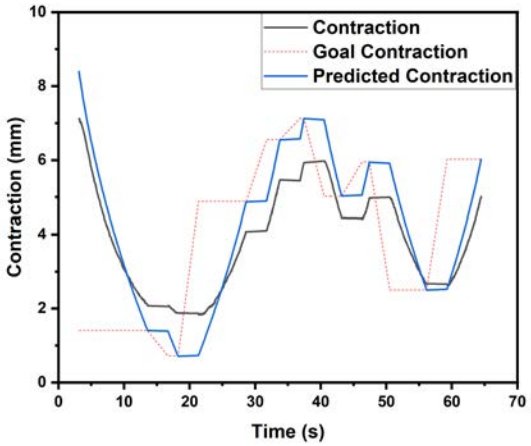
(b) Fusion (No OFMS)
Error: 0.530 ± 0.357 mm



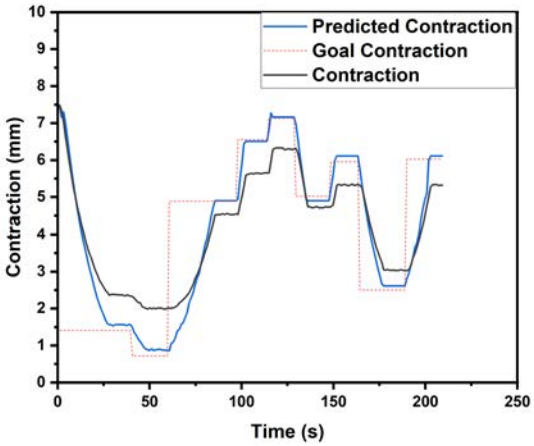
(c) Fusion (OFMS)
Error: 0.337 ± 0.250 mm

Figure 4.22: Test Data for 50 g

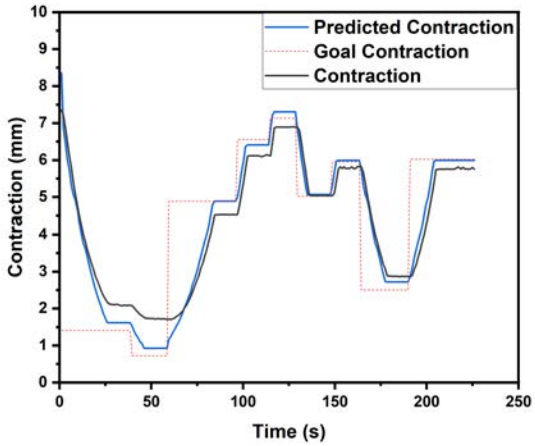
4.27.2 100 g Test Data



(a) Open Loop
Error: 0.555 ± 0.309 mm



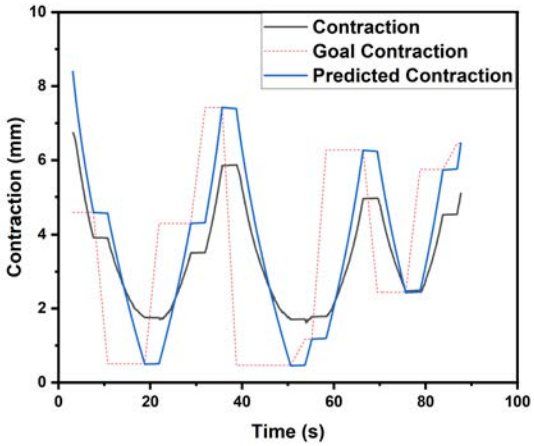
(b) Fusion (No OFMS)
Error: 0.574 ± 0.333 mm



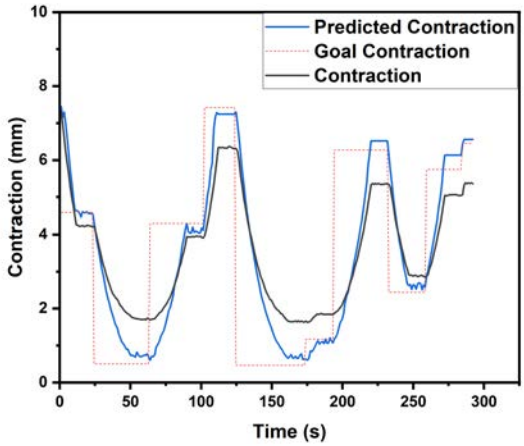
(c) Fusion (OFMS)
Error: 0.346 ± 0.204 mm

Figure 4.23: Test Data for 100 g

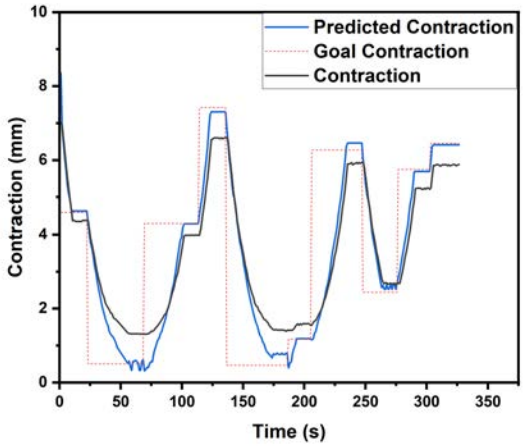
4.27.3 150 g Test Data



(a) Open Loop
Error: 0.664 ± 0.392 mm



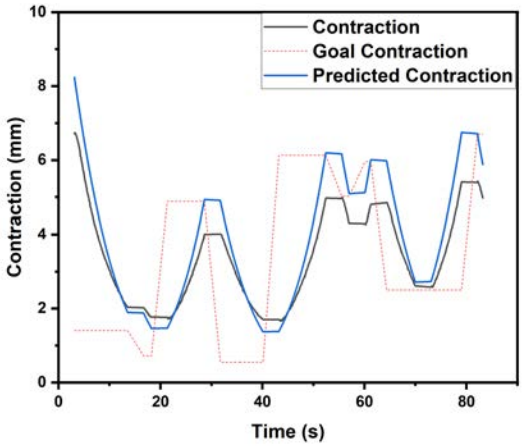
(b) Fusion (No OFMS)
Error: 0.633 ± 0.371 mm



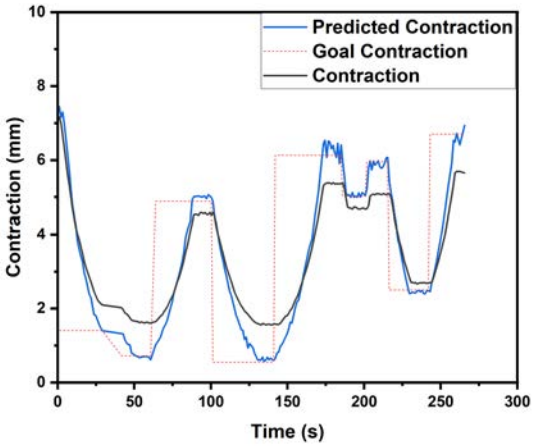
(c) Fusion (OFMS)
Error: 0.439 ± 0.255 mm

Figure 4.24: Test Data for 150 g

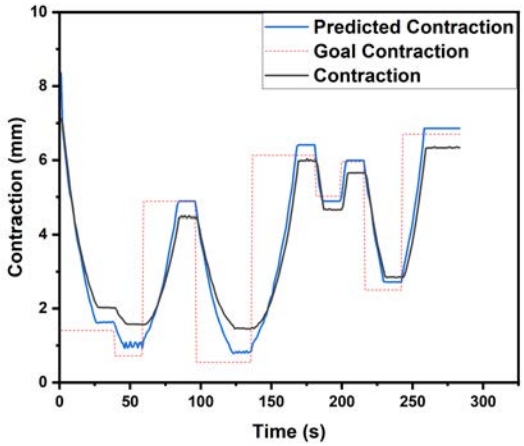
4.27.4 200 g Test Data



(a) Open Loop
Error: 0.513 ± 0.371 mm



(b) Fusion (No OFMS)
Error: 0.520 ± 0.314 mm



(c) Fusion (OFMS)
Error: 0.374 ± 0.197 mm

Figure 4.25: Test Data for 200 g

4.27.5 RMSE

The following table summarizes the RMSE values for open-loop and random forest sensor fusion control for all tested weights in hydraulic actuators. The relative improvement uses the open-loop results as the baseline:

Table 4.4: RMSE (No OFMS)

Weight	Open-Loop	Sensor-Fusion	Relative Improvement
50 g	0.692 mm	0.639 mm	7.66%
100 g	0.635 mm	0.664 mm	-4.57%
150 g	0.772 mm	0.733 mm	5.05%
200 g	0.633 mm	0.607 mm	4.11%

Table 4.5: RMSE (OFMS)

Weight	Open-Loop	Sensor-Fusion	Relative Improvement
50 g	0.692 mm	0.420 mm	39.3%
100 g	0.635 mm	0.401 mm	36.9%
150 g	0.772 mm	0.507 mm	34.3%
200 g	0.633 mm	0.423 mm	33.2%

4.27.6 Statistical Analysis

To critically examine the performance advantages of the sensor-fusion approaches, paired t-tests were executed for the RMSE values at individual waypoints, pooled across all weight classes. For the comparison between the Open-Loop system and the Sensor-Fusion without OFMS, a t-statistic of 2.42301 was recorded, with 37 degrees of freedom. The obtained p-value of 0.0204, being less than the 0.05 significance level, implies that the difference in population means is statistically significant. This lends credence to the efficacy of the Sensor-Fusion approach even without OFMS in enhancing the system's performance.

Furthering this analysis, the Open-Loop system was also compared to the Sensor-Fusion with OFMS. Here, the t-statistic shot up to 7.66948 with the same degrees of freedom (37). Even more compelling was the p-value, which was less than 0.0001. This extremely low p-value fortifies the argument for the robustness of the Sensor-Fusion system incorporating OFMS. The results not only show statistically significant improvements but strongly suggest that the sensor-fusion techniques, especially with OFMS, deliver a notably superior performance when contrasted with the Open-Loop system.

4.27.7 Error Analysis

The mean absolute error (MAE) and Standard Deviation for each weight category are summarized in the table below for both open-loop and closed-loop control:

Table 4.6: Error Table

Weight	**Open-Loop**	**Sensor Fusion (No OFMS)**	**Sensor Fusion (OFMS)**
50 g	0.608 ± 0.331 mm	0.530 ± 0.357 mm	0.337 ± 0.250 mm
100 g	0.555 ± 0.309 mm	0.574 ± 0.333 mm	0.346 ± 0.204 mm
150 g	0.664 ± 0.392 mm	0.633 ± 0.371 mm	0.439 ± 0.255 mm
200 g	0.513 ± 0.371 mm	0.520 ± 0.314 mm	0.374 ± 0.197 mm

Discussion

5.28 Integrated-Camera Sensing

The central thesis of this project rests on the OFMS sensor's ability to deliver high positional accuracy at a fraction of the cost of many existing sensing methods, while also offering a statistically significant reduction in errors and ensuring bio-compatibility.

In contrast, the paper proposed by Werner et al. [39] that we compare our results to employs a vision-based sensing approach that relies on an integrated camera. This method, though effective, is computationally intensive and requires more sophisticated hardware, leading to higher costs and potentially higher safety risks.

5.28.1 Results Comparison

When scrutinized through the lens of RMSE values, our closed-loop pneumatic system showed an overall 72.8% improvement in relative positioning accuracy across all tests from 0.672 mm to 0.189 mm. This is considerably better than the vision-based approach's performance, which had an RMSE of 1.37 mm for undisturbed trajectory.

For hydraulic actuators, our sensor fusion system integrated with OFMS exhibited an overall RMSE improvement of 35.8% across all tests to a value of 0.444 mm. This presents a compelling

case for OFMS over camera-based systems.

However, this research's sensing methodology has not been tested under lateral motion like was done under the camera-based system, only sensing deformation under contraction. Therefore, we cannot yet compare the OFMS performance over the camera-based sensing in this regard.

5.29 Resistive Sensing

Another paper utilizes Fluidic Elastomer Actuators with copper electrodes interpreted via Long Short-Term Memory (LSTM) neural networks.[33] Our OFMS offers a lower-risk, biocompatible sensing option that aligns well with the stringencies of medical settings. On the other hand, sponge resistive sensors present an intriguing low-error option for general robotic applications but might face challenges in biocompatibility and sterile environments.[14]

5.29.1 Results Comparison

In our research, we realized significant reductions in error metrics for soft actuator positioning. Specifically, the RMSE for pneumatic and hydraulic actuators under closed-loop control was refined to less than 0.5 mm across a range of loads. This improvement was corroborated by statistically significant paired t-tests, underscoring the robustness of our findings. The paper under discussion also achieved low error percentages in actuator tip positioning along the x, y, and z-axes, ranging from 0.37% to 2.38%. However, the RMSE value to predict end-effector position was 12.57 mm. This level of error won't be suitable for MIS where at least less than 1 mm error would be required.

5.29.2 Sensor Cost, Suitability and Biocompatibility

Our study employs PMMA fibers, presenting a safer alternative to metal-based sensors. In contrast, the comparative paper's research focused on the effects of sponge resistive sensors on the actuator's

working space. While the actuator material itself is biocompatible, the copper electrodes would be cause for concern. Aside from safety and biocompatibility, the material cost for the OFMS sensing mechanism will be much lower than the metal required for resistive sensing. Additionally, the conductive methodology would be unsuitable for the polyethylene body the actuators in this study are comprised of.

5.29.3 Calibration and Error Sources

While we utilized a Random Forest algorithm for sensor fusion, the paper in comparison adopted Long Short-Term Memory (LSTM) neural networks for handling the sensor drift. Both methods indicate the growing influence of machine learning in calibrating complex sensor signals, but a head-to-head comparison under similar conditions would offer a more substantive discussion on performance

5.29.4 Implications and Future Work

Our results clearly support the adoption of OFMS in medical contexts, specifically minimally invasive surgeries. The low RMSE values and high statistical robustness make it a compelling choice. However, it would be interesting to explore how our technology fares in other applications that the LSTM-based system is well-suited for.

5.30 Model Based Position Control System

Runciman et al.[29] proposes a model-based control system utilising open-loop control with the port-Hamiltonian formulation. This is in contrast to the closed-loop and sensor-fusion approaches integrated into position control methodology of our study.

The model-based approach achieves a high degree of positional accuracy, however this is approach is only suitable with external forces of up to 1 N due to the open-loop control system. The study's

results indicate higher degrees of positional error when higher payloads are used than the 50 g test weight.

5.30.1 Results Comparison

Our OFMS sensor results in lower overall positional accuracy than the model-based approach, with pneumatic positional error going up to 0.183 mm with a maximum standard deviation of 0.094 mm, and hydraulic actuators showing a maimum of 0.439 mm error and a maximum standard deviation of 0.255 mm in contrast to the model-based approach's results of 0.043 mm error with a standard deviation of 0.033 mm.

5.30.2 Implications and Future Work

It may be possible to integrate a more accurate model-hybrid approach in order to achieve higher positional accuracy using the OFMS sensing mechanism, though this will require more tests and a hysteresis analysis to identify the correlation between various weights and sensor values.

5.31 Force-Sensing and Shape-Sensing

Force perception continues to be a challenging hurdle in the advancement of soft robotics.[27] The sensor investigated in our study shows promise for improving control in activities that involve contractive forces, such as pulling or pushing. However, its efficacy in managing lateral forces is still questionable. Owing to the sensor's design, which is optimized for contraction-based movements, there is a possibility of inaccurate readings when exposed to lateral forces. Moving forward, it's critical to delve deeper into the sensor's limitations and to explore avenues for rectifying these issues in future research. Certain studies have been able to achieve low-cost force sensing with conductive ink.[24] This could be integrated with the OFMS mechanism, however this would involve putting electrical components inside the actuators themselves.

Optical-based methods of shape-sensing are being researched as well.[34] This paper has shown less than 2.1% error in measuring the curvature of pneumatic actuators. As this methodology is in line with the philosophy of the research in this paper, it could allow for metal-free shape sensing within the pouch motors.

5.32 Control System

The results indicate that the time required to conclude the tests was notably longer using PID control compared to open-loop control. One reason behind this could be the inherent limitations of PID as a linear control system, which might not be optimally suited for the complex actuation relationships in soft robotics, which often have hysteresis and non-linear responses. In searching for more efficient alternatives, we found Franco et al.'s[12] work compelling; they demonstrated that a non-linear energy-based control system outperformed traditional PID systems. As we aim for more robust and swift actuation in future iterations, our research will explore the adoption of non-linear control systems.

Additional research has been done to approximate the dynamic model of soft robots to the first-order, allowing for simpler control systems while keeping accuracy loss to a minimum.[13][37] Further work can be done to investigate these control systems with respect to the OFMS as well.

5.33 Sensor Fusion Optimisation

The sensor fusion model utilised in this research was Random Forest due to its ability to handle non-linearities, it's robustness to overfitting, and it's lack of required pre-processing. However, there may be more optimal algorithms and methodologies for refining sensor fusion such as proposed by Brena et al.[9] who utilised a meta-data methodology where the algorithm was trained on data where each row summarised the feature points of a sensor rather than the whole dataset, and saw a 91% accuracy rate with certain algorithms.

Research has been done on integrating soft strain sensors with a LSTM neural networks to achieve a positional accuracy of over 2 mm at 2-DOF.[36] Integrating strain sensing into the pouch motors could improve the accuracy of the sensing mechanism in place as well while remaining electronics-free.[7][19]

Conclusion

In this study, we introduced an OFMS that pioneers a new pathway in the realm of medical robotics, particularly in minimally invasive surgeries. Our OFMS not only aligns well with the stringent requirements of biocompatibility and sterility but also markedly improves the positional accuracy of soft actuators. Through rigorous testing, the RMSE for both pneumatic and hydraulic actuators under closed-loop control was refined to an impressive margin of around 0.154 mm accuracy for pneumatic systems and 0.441 mm for hydraulic systems. Statistically significant results from paired t-tests further corroborate the robustness of our findings, making OFMS a compelling alternative to existing sensing mechanisms. Further research will delve into shape-sensing and force-sensing methodologies, while integrating the current system into a complex robot. Moreover, the sensor fusion system can be developed further in order to create a more rapid response.

References

[1] Applied Sciences — Free Full-Text — Open Loop Position Control of Soft Hydraulic Actuators for Minimally Invasive Surgery. https://www.mdpi.com/2076-3417/11/16/7391.

[2] How to Use a Phototransistor with an Arduino. https://www.digikey.co.uk/en/maker/blogs/2022/how-to-use-a-phototransistor-with-an-arduino.

[3] What is Random Forest? — IBM. https://www.ibm.com/topics/random-forest.

[4] 9.3: PID Tuning via Classical Methods. https://eng.libretexts.org/Bookshelves/Industrial_and_Systems_Engi Integral-Derivative_(PID)_Control/9.03%3A_PID_Tuning_via_Classical_Methods, May 2020.

[5] ALTHOEFER, K. Antagonistic actuation and stiffness control in soft inflatable robots. *Nature Reviews Materials 3*, 6 (June 2018), 76–77.

[6] AMIRI MOGHADAM, A. A., ALAIE, S., DEB NATH, S., AGHASIZADE SHAARBAF, M., MIN, J. K., DUNHAM, S., AND MOSADEGH, B. Laser Cutting as a Rapid Method for Fabricating Thin Soft Pneumatic Actuators and Robots. *Soft Robotics 5*, 4 (Aug. 2018), 443–451.

[7] ARAROMI, O. A., GRAULE, M. A., DORSEY, K. L., CASTELLANOS, S., FOSTER, J. R., HSU, W.-H., PASSY, A. E., VLASSAK, J. J., WEAVER, J. C., WALSH, C. J., AND WOOD, R. J. Ultra-sensitive and resilient compliant strain gauges for soft machines. *Nature 587*,

7833 (Nov. 2020), 219–224.

[8] AVERY, J., RUNCIMAN, M., DARZI, A., AND MYLONAS, G. P. Shape Sensing of Variable Stiffness Soft Robots using Electrical Impedance Tomography. In *2019 International Conference on Robotics and Automation (ICRA)* (May 2019), pp. 9066–9072.

[9] BRENA, R. F., AGUILETA, A. A., TREJO, L. A., MOLINO-MINERO-RE, E., AND MAYORA, O. Choosing the Best Sensor Fusion Method: A Machine-Learning Approach. *Sensors (Basel, Switzerland) 20*, 8 (Apr. 2020), 2350.

[10] BUSO, A., SCHARFF, R., DOUBROVSKI, E., WU, J., WANG, C., AND VINK, P. Soft Robotic Module for Sensing and Controlling Contact Force. In *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)* (May 2020), pp. 70–75.

[11] DAERDEN, F., AND LEFEBER, D. The Concept and Design of Pleated Pneumatic Artificial Muscles. *International Journal of Fluid Power 2* (Jan. 2001).

[12] FRANCO, E., AYATULLAH, T., SUGIHARTO, A., GARRIGA-CASANOVAS, A., AND VIRDYAWAN, V. Nonlinear energy-based control of soft continuum pneumatic manipulators. *Nonlinear Dynamics 106* (Sept. 2021).

[13] GEORGE THURUTHEL, T., RENDA, F., AND IIDA, F. First-Order Dynamic Modeling and Control of Soft Robots. *Frontiers in Robotics and AI 7* (2020).

[14] HEGDE, C., SU, J., TAN, J. M. R., HE, K., CHEN, X., AND MAGDASSI, S. Sensing in Soft Robotics. *ACS Nano 17*, 16 (Aug. 2023), 15277–15307.

[15] HELPS, T., AND ROSSITER, J. Proprioceptive Flexible Fluidic Actuators Using Conductive Working Fluids. *Soft robotics 5* (Apr. 2018).

[16] HIRAKI, T., NAKAHARA, K., NARUMI, K., NIIYAMA, R., KIDA, N., TAKAMURA, N., OKAMOTO, H., AND KAWAHARA, Y. Laser Pouch Motors: Selective and Wireless Activa-

tion of Soft Actuators by Laser-Powered Liquid-to-Gas Phase Change. *IEEE Robotics and Automation Letters 5*, 3 (July 2020), 4180–4187.

[17] KHANBAREH, H., DE BOOM, K., SCHELEN, B., SCHARFF, R. B. N., WANG, C. C. L., VAN DER ZWAAG, S., AND GROEN, P. Large area and flexible micro-porous piezoelectric materials for soft robotic skin. *Sensors and Actuators A: Physical 263* (Aug. 2017), 554–562.

[18] KIM, W., PARK, H., AND KIM, J. Compact Flat Fabric Pneumatic Artificial Muscle (ffPAM) for Soft Wearable Robotic Devices. *IEEE Robotics and Automation Letters 6*, 2 (Apr. 2021), 2603–2610.

[19] KOIVIKKO, A., LAMPINEN, V., PIHLAJAMÄKI, M., YIANNACOU, K., SHARMA, V., AND SARIOLA, V. Integrated stretchable pneumatic strain gauges for electronics-free soft robots. *Communications Engineering 1*, 1 (June 2022), 1–10.

[20] LINDENROTH, L., STOYANOV, D., RHODE, K., AND LIU, H. Toward Intrinsic Force Sensing and Control in Parallel Soft Robots. *IEEE/ASME Transactions on Mechatronics 28*, 1 (Feb. 2023), 80–91.

[21] MIN, R., HU, X., PEREIRA, L., SIMONE SOARES, M., SILVA, L. C. B., WANG, G., MARTINS, L., QU, H., ANTUNES, P., MARQUES, C., AND LI, X. Polymer optical fiber for monitoring human physiological and body function: A comprehensive review on mechanisms, materials, and applications. *Optics & Laser Technology 147* (Mar. 2022), 107626.

[22] MOGHADAM, A. A. A., CAPRIO, A., ALAIE, S., MIN, J. K., DUNHAM, S., AND MOSADEGH, B. Rapid Manufacturing of Thin Soft Pneumatic Actuators and Robots. *JoVE (Journal of Visualized Experiments)*, 153 (Nov. 2019), e60595.

[23] NIIYAMA, R., RUS, D., AND KIM, S. Pouch Motors: Printable/inflatable soft actuators for robotics. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (May 2014), pp. 6332–6337.

[24] PAGOLI, A., CHAPELLE, F., CORRALES RAMON, J. A., MEZOUAR, Y., AND LAPUSTA, Y. Large-Area and Low-Cost Force/Tactile Capacitive Sensor for Soft Robotic Applications. *Sensors 22* (May 2022), 4083.

[25] PAN, M., YUAN, C., LIANG, X., ZOU, J., ZHANG, Y., AND BOWEN, C. Triboelectric and Piezoelectric Nanogenerators for Future Soft Robots and Machines. *iScience 23*, 11 (Nov. 2020), 101682.

[26] PARK, T., KIM, K., OH, S.-R., AND CHA, Y. Electrohydraulic Actuator for a Soft Gripper. *Soft Robotics 7*, 1 (Feb. 2020), 68–75.

[27] RUNCIMAN, M., AVERY, J., ZHAO, M., DARZI, A., AND MYLONAS, G. P. Deployable, Variable Stiffness, Cable Driven Robot for Minimally Invasive Surgery. *Frontiers in Robotics and AI 6* (2020).

[28] RUNCIMAN, M., DARZI, A., AND MYLONAS, G. Soft Robotics in Minimally Invasive Surgery. *Soft robotics 6* (Mar. 2019).

[29] RUNCIMAN, M., FRANCO, E., AVERY, J., RODRIGUEZ Y BAENA, F., AND MYLONAS, G. Model Based Position Control of Soft Hydraulic Actuators. In *2023 IEEE International Conference on Robotics and Automation (ICRA)* (May 2023), pp. 2676–2682.

[30] RUSSO, S., RANZANI, T., LIU, H., NEFTI-MEZIANI, S., ALTHOEFER, K., AND MENCIASSI, A. Soft and Stretchable Sensor Using Biocompatible Electrodes and Liquid for Medical Applications. *Soft Robotics 2*, 4 (Dec. 2015), 146–154.

[31] SCHARFF, R. B. N., FANG, G., TIAN, Y., WU, J., GERAEDTS, J. M. P., AND WANG, C. C. Sensing and Reconstruction of 3-D Deformation on Pneumatic Soft Robots. *IEEE/ASME Transactions on Mechatronics 26*, 4 (Aug. 2021), 1877–1885.

[32] SCHMITT, F., PICCIN, O., BARBÉ, L., AND BAYLE, B. Soft Robots Manufacturing: A Review. *Frontiers in Robotics and AI 5* (2018).

[33] SHU, J., WANG, J., CHENG, K. C.-C., YEUNG, L.-F., LI, Z., AND TONG, R. K.-Y. An End-to-End Dynamic Posture Perception Method for Soft Actuators Based on Distributed Thin Flexible Porous Piezoresistive Sensors. *Sensors 23*, 13 (Jan. 2023), 6189.

[34] SUN, G., HU, Y., DONG, M., HE, Y., YU, M., AND ZHU, L. Posture measurement of soft pneumatic bending actuator using optical fibre-based sensing membrane. *Industrial Robot: the international journal of robotics research and application 46*, 1 (Apr. 2019), 118–127.

[35] THURUTHEL, T. G., AND IIDA, F. Multimodel Sensor Fusion for Learning Rich Models for Interacting Soft Robots, May 2022.

[36] THURUTHEL, T. G., SHIH, B., LASCHI, C., AND TOLLEY, M. T. Soft robot perception using embedded soft sensors and recurrent neural networks. *Science Robotics 4*, 26 (Jan. 2019), eaav1488.

[37] TRUMIĆ, M., JOVANOVIĆ, K., AND FAGIOLINI, A. Decoupled nonlinear adaptive control of position and stiffness for pneumatic soft robots. *The International Journal of Robotics Research 40*, 1 (Jan. 2021), 277–295.

[38] WANG, J., AND CHORTOS, A. Control Strategies for Soft Robot Systems. *Advanced Intelligent Systems 4* (Feb. 2022).

[39] WERNER, P., HOFER, M., SFERRAZZA, C., AND D'ANDREA, R. Vision-Based Proprioceptive Sensing for Soft Inflatable Actuators, Sept. 2019.

[40] ZHOU, X., AND LEE, P. S. Three-dimensional printing of tactile sensors for soft robotics. *MRS Bulletin 46*, 4 (Apr. 2021), 330–336.