

# MIE498 - Thesis

## Development of an Integrated System for Characterization of Artificial Muscles

By:

P. Cuvin

Supervised by:

Mihai Duduta, PhD



Department of Mechanical and Industrial Engineering

Faculty of Applied Science and Engineering

University of Toronto

April 2023

# Development of an Integrated System for Characterization of Artificial Muscles

P. Cuvin

## Abstract

Dielectric elastomer actuators are constructed from an elastomer or other compliant material sandwiched between two electrodes, with high potential for applications as artificial muscles in soft robotics. However, the current use of artificial muscles (in particular, DEAs) in research and industry is limited by the difficulty in assessing the properties and quality of large batches of muscle samples, as well as the slow rate at which full tests of artificial muscle samples can take place and be processed (upwards of 2-3 hours for DEAs per sample when considering processing). This report aims to present a design for an integrated framework for characterizing the performance of DEAs, in order to reduce the time and cost of DEA testing and enable the wider application of DEAs. Requirements were developed from the necessary technical specifications of the new framework, and were used to guide the design process in selection of systems and solutions. The integrated framework underwent several design iterations and improvements to decrease test run time and improve its accuracy when operating under different environmental conditions and with different types of DEA samples, resulting in a final integrated framework consisting of the physical design, the hardware design, and the software design. Testing of this framework against the existing framework demonstrated greatly improved runtime for testing, data collection, and data processing, most significantly due to the CV strain measurement system that reduced the time required for strain measurement by a factor of 20. Testing of the framework across different types of DEA samples demonstrated that the system is compatible with different configurations of DEAs and different environmental lighting, indicating the robustness of the system. The results of this report demonstrate that the integrated testing framework is feasible for use for wide-scale testing of DEA samples, and can be used to widen the availability of DEA performance data for use in better control of DEA samples. Future work should be done to collect a dataset of DEA performance with the system, to investigate the feasibility of ML-based models of DEA performance prediction and control that are trained using the DEA dataset, and to extend the testing framework to other classes of artificial muscles.

## Acknowledgements

I acknowledge and thank this project's supervisor, Professor Mihai Duduta, for his support and expertise throughout all phases of this project, and whose guidance was instrumental to the success of this project. I also thank Ang Li of the Materials for Actuators, Robots & Batteries Lab (MARB-L) team for his guidance on the project and for his development of the electronic system of the integrated testing framework. Credit is additionally owed to the Department of Mechanical & Industrial Engineering (MIE) at the University of Toronto for supplying facilities and funding throughout this project, and to the MIE Undergraduate Office team for their administrative support.

## **Table of Contents**

<b>1. Introduction</b>	<b>4</b>
1.1 Background	4
1.2 State of the Art	5
1.3 Motivation	5
1.4 Objectives and Methodology	6
<b>2. Design of Testing Framework</b>	<b>7</b>
2.1 Requirements of Proposed Testing Framework	7
2.2 Hardware and Software Approaches to Achieving System Requirements	9
2.3 Hardware and Software Integration	14
<b>3. Development and Prototyping</b>	<b>14</b>
3.1 Development and Iteration of Physical System	14
3.2 Development and Iteration of Software and CV System	16
3.3 Prototyping and Integration of Framework	21
<b>4. Testing of System and Results</b>	<b>23</b>
4.1 Integrated Framework vs. Existing Framework Performance Test	24
4.2 Integrated System Live DEA Sample Test	26
<b>5. Discussion and Future Work</b>	<b>26</b>
<b>6. References</b>	<b>28</b>

## **List of Abbreviations**

1. Dielectric Elastomer Actuators (DEAs)
2. Extensile Fluidic Artificial Muscles (EFAMs)
3. Inductance, Capacitance, Resistance (LCR)
4. Computer Vision (CV)
5. Machine Learning (ML)
6. Hue, Saturation, Value (HSV)
7. Hough Circle Transform (HCT)
8. Long Short-Term Memory (LSTM)

## 1. Introduction

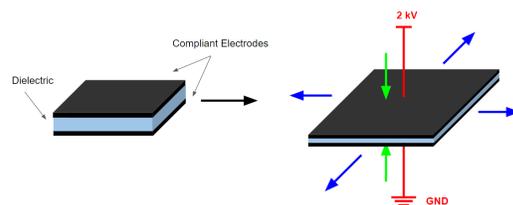
### 1.1 Background

Artificial muscles are a class of robotic actuators that have significant advantages compared to conventional rigid robotic actuators, including human-safe actuation forces, lower mass, lower material and manufacturing cost, and a greater degree of mobility and extensibility [1]. One class of artificial muscles that have demonstrated particularly high potential for use in soft robotic applications are Dielectric Elastomer Actuators, or DEAs [2]. DEAs comprise a diverse and highly controllable class of artificial muscle, being constructed from an elastomer or other compliant material sandwiched between two electrodes, offering significant cost savings compared to other soft robotic actuators due to the low cost of material. DEAs are effectively soft capacitors, composed of an elastomeric dielectric layer placed between two compliant electrodes. When a voltage is applied to the electrodes, the charge buildup on the opposite electrodes causes the electrodes to attract and compress the elastomeric dielectric layer, causing the entire DEA to expand in the planar direction, developing strains of up to 30-40% in certain compositions of DEAs [2]. Conventional single-film DEAs develop force outputs in the mN range and similarly small displacements, whereas novel multi-layer composite DEAs made with ultrathin carbon nanotube (CNT)-based percolative electrodes can develop forces of upwards of 10 N [2]. The extension and contraction cycle of a DEA is repeatable for several thousand cycles (depending on the DEA composition, applied voltage, and applied force) and can be leveraged by rolling a DEA sheet in a cylinder to create an artificial muscle strand that behaves similarly to human muscle fiber [2]. Consequently, the development of DEAs offers significant opportunities for applications in lightweight and affordable human

prosthetics and in high-performance biomimetic robotics.

However, the current use of artificial muscles (in particular, DEAs) in research and industry is limited by the low predictability of the response and limits of soft robotic actuators made from different materials as well as individual artificial muscle samples, presenting significant challenges in controlling and verifying the quality of individual actuators in large-scale applications of DEAs. Higher-accuracy and faster testing would significantly accelerate the rate at which individual and batch DEA samples can be tested, allowing for the collection of a representative dataset of DEA performance and behavior. As such, this report presents a novel and integrated framework for the large-scale characterization of artificial muscle properties, with the aim of significantly improving the speed and accuracy of DEA testing and building datasets for ML-based prediction of DEA performance.

The focus of this report is on developing a framework to characterize the performance of specifically DEAs, as they offer significant potential for a feasible artificial muscle compared to other soft robotic actuators, with the system presented being optimized to DEA testing in the form-factor of the hardware used, as well as the design of the image processing software. However, the presented mechanical, hardware, and software framework is highly adaptable to other artificial muscles testing setups, and can easily be extended to offer diverse testing capability to soft robotics labs.



**Fig. 1:** Diagram of DEA behavior under applied voltage

## 1.2 State of the Art

Current research in the field of artificial muscles and DEAs focuses primarily on the development and testing of small samples of actuators made via novel materials or methods, with little focus given to testing of larger-scale individual samples, large batches of artificial muscles, or investigating the feasibility of scale manufacturing methods for artificial muscles. Similarly, there is little research on scalable testing frameworks for artificial muscles, leaving a significant gap in translating the success of artificial muscles in individual literature to integrated robotic or prosthetic applications, which this report will seek to address through its presented integrated testing framework.

The few testing frameworks that exist are highly specialized to particular classes of artificial muscles, require significant in-lab work to operate, are poorly scalable, and do not have software integration for high-volume data collection and processing, significantly reducing their utility in accelerating the characterization of artificial muscles. J. Chipka et al. present a dynamometer-based testing system for testing of McKibben (pneumatic) artificial muscles, which captures the kinematic performance and energy usage of the actuator, but whose application is limited to high-force, low-frequency devices and cannot capture strain data, and thus being poorly adaptable to characterization of DEAs [3]. Similarly, J. Garbulinski et al. present a framework and model for testing extensile fluidic artificial muscles (EFAMs), using a pressure transducer and load cell connected to an Arduino board for collection of force data, which is converted into stress and strain data via use of a script, with the results being used to validate a previously-developed mathematical model to predict the stress-strain behavior of EFAMs [4]. This framework, while significantly more scalable

than the framework presented by J. Chipka, is specialized to testing only EFAMs, limiting its use in informing the design of DEA testing frameworks [4]. T. Henderson et al. present a data-driven approach for selection of artificial muscles for use in robotic and prosthetic applications based on the performance constraints of stress and bandwidth of different classes of artificial muscles - however, the presented literature is relevant to the broad selection of artificial muscle classes for specified application rather than the prediction of the performance of any particular class of artificial muscle, and does not present any new approaches for *collecting* artificial muscle data, instead relying upon scattered data points from existing literature on artificial muscles to form the dataset used to train the selecting model [5]. The low number of data points and wide variability in the source and consistency of the data significantly reduces the utility of the dataset in developing predictable models for DEAs (or any other artificial muscle), further obviating the need for a scalable framework for testing DEAs.

## 1.3 Motivation

The gap in sufficiently robust datasets of artificial muscle performance and testing frameworks to enable the compilation of such datasets, as well as the high potential for DEA use in soft robotics applications if challenges of sample performance predictability and quality are resolved, indicates a strong need for the development of an integrated and scalable DEA testing system.

To characterize the behavior of a DEA sample, the sample's resistance, capacitance, and strain response to an applied voltage has to be measured from zero applied voltage to near (or beyond) the breaking point of the sample (characterized by the electrical or mechanical breakdown of the dielectric elastomer, shorting the electrodes and causing the DEA to rupture and contract). It is

essential to maintain consistent testing conditions between trials to obtain an accurate characterization of sample behavior, which increases the time required between sample tests to ensure that testing conditions are maintained within acceptable limits. The rapid and accurate measurement of DEA strain presents an additional challenge, as traditional methods for strain measurement such as strain gauges would interfere with the results of the test, while manual measurement of strain by area is inaccurate and infeasible for a large number of data points. Advances in rapid manufacturing potentially allow for the design of modular, scalable, and dimensionally accurate structures that allow sample tests to be run in consistent conditions, while CV techniques allow for DEA strain to be measured automatically and accurately. An integrated system utilizing these features would have greater testing capacity, consistency, and speed, and would additionally enable more feasible large-scale testing of DEA samples. Such a system would additionally enable the development of machine learning models to predict the behavior of different classes of DEAs based on material properties, physical dimensions, manufacturing method, and defects, with LSTM and transformer-based systems being the most likely candidates for effective models in this context. As such, the development of a system that can accelerate the testing of artificial muscles and is able to robustly predict the behavior of batches of DEA samples would greatly increase the feasibility of large-scale DEA characterization and quality assessment, and allow the use of DEAs in a wider range of robotics and industrial applications.

#### **1.4 Objectives and Methodology**

This project will contribute a design for a novel artificial muscle testing platform and software package for use in research and industrial

contexts, and will demonstrate the platform's performance in characterizing the behavior of diverse DEA samples in single-sample and parallel tests compared to conventional testing platforms. The project will additionally discuss the feasibility of extending the framework to predicting the behavior of different classes of artificial muscles, and will discuss the next steps of developing and training a sample machine learning system on data collected from the testing framework to effectively predict the behavior of a sample batch of DEAs.

This report is divided into five sections: An introduction to the research problem, High-level design of the testing framework, Hardware and software development of the framework, Testing and discussion of results, and Investigation of future work. The design section discusses the engineering approach to the design of the framework, including:

1. Identification of major requirements and potential hardware and software approaches
2. Integration of hardware and software systems
3. Presentation of conceptual design and justification of design decisions

The development section discusses the realization of the integrated system, including:

1. The development of iterations of the mechanical and hardware system, including challenges faced and solutions
2. The development of the software and computer vision (CV) system, including challenges faced and solutions
3. Prototyping and integration of completed framework

The testing section discusses the results of testing the system on sample data and actual DEAs,

including an investigation of overall testing time of a DEA sample with and without the system, as well as the scalability of the design.

The future work section discusses the application and extension of the work presented in this report, including:

1. Application of the system to compile dataset on DEA performance
2. Extension of system to other artificial muscle classes
3. Discussion of feasibility of ML model to predict performance of DEAs

## 2. Design of Testing Framework

### 2.1 Requirements of Proposed Testing Framework

The engineering requirements of the DEA testing framework were collected via benchmarking of the performance of the existing DEA testing setup, identification of necessary parameters to measure during testing of DEAs, and scoping of specifications for acceptable time and accuracy performance of the testing framework to enable large-scale and accurate data collection and processing of DEA tests.

The existing setup consists of a plastic disc containing a single DEA sample, connected via leads to a variable-voltage power supply and a BK Precision 894 high-resolution LCR meter. The voltage output of the power supply is manually controlled by the lab technician, and can be varied from 0 V to 4 kV. The LCR meter records the applied voltage with a precision of 0.005V and the resistance, capacitance, and inductance (LCR) with a precision of 0.05%, and passes the data via USB to a computer running LabView 2022, where the data points are logged every 0.001 seconds and parsed into a .csv file when data collection is complete.

The existing setup uses a Spinel OV9281 720p USB camera mounted on a cardboard frame to capture video of the DEA during the test run, with the built-in webcam software of the connected computer being used to record footage of the DEA sample, the recording being manually initiated by the lab technician.

The procedure for a test of a DEA sample with the existing setup is as follows:

1. Place DEA sample in disc, connect leads to electrodes
2. Turn on power supply and LCR meter, set desired voltage ramp and data collection parameters
3. Manually turn on computer with LabView, open file with voltage and LCR collection, ensure LCR meter is passing data to computer
4. Place USB camera over DEA sample and turn on webcam recording software
5. Start voltage ramp and data collection on LabView, note time and start recording of video of DEA on computer
6. When DEA ruptures, identified by visual inspection of the DEA or increase in the measured LCR, turn off power supply and LCR meter and stop LabView recording, note time
7. Stop video recording and note time
8. Perform manual strain measurements of DEA at selected time intervals by drawing circles, measuring area, converting to area in  $\text{mm}^2$ , and converting to strain

The essential parameters required for characterization of the performance of a DEA sample (and required resolution) are as follows:

1. Voltage (V)
2. Resistance ( $\Omega$ )
3. Capacitance (F)
4. Strain ( $\text{mm}^2/\text{mm}^2$ )

## 5. Time (s)

Voltage is the input parameter of the DEA, and is used to control the response of the DEA, all other parameters being dependent on the supplied voltage. The LCR of the DEA varies over the testing cycle, and varies by the material composition of a DEA sample - characterizing the voltage-resistance curve of a DEA is essential to finely controlling the strain of the DEA without causing dielectric breakdown or rupture. Capacitance is additionally used to detect the rupture point of a DEA sample, with capacitance significantly decreasing when the DEA ruptures, as the electrodes contact each other and create a short, reducing the ability of the DEA to retain charge, thereby significantly reducing the system capacitance. The strain characterizes the extent to which the DEA sample expands over the course of the test, with the voltage-strain curve being required to control the extension and contraction of the DEA in a robotic or prosthetic application. The time of the results is used to synchronize the data points, and is thus required to be captured precisely to ensure the voltage-resistance and voltage-strain curves of DEA samples can be linked and used to characterize the LCR and strain performance of a DEA sample.

Collection of the above parameters across a single given DEA sample represent a single data point in a prospective DEA performance dataset, multiple tests of the same DEA composition and other DEA compositions, totalling upwards of thousands or tens of thousands of individual tests being required to form a useful corpus of data that can be used for training control and prediction models [6].

The weaknesses of the existing setup are clear upon review of the test procedure and required parameters for development of a useful DEA dataset:

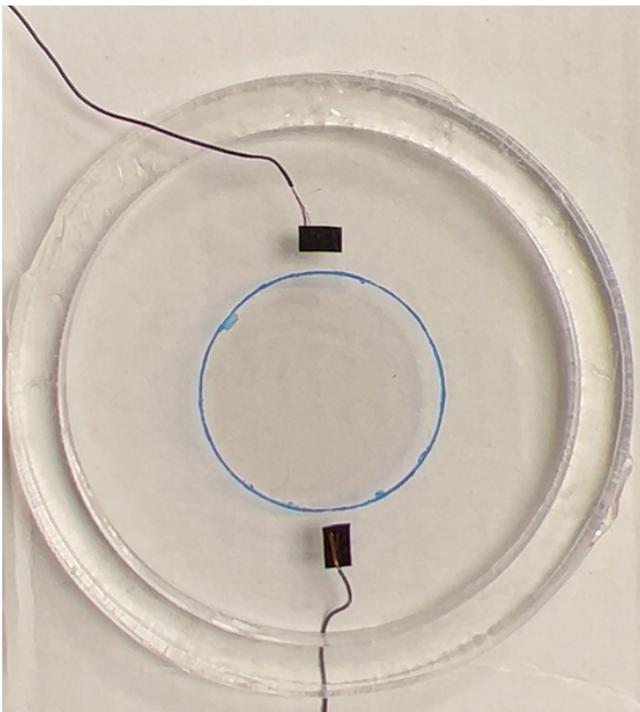
1. High cycle times with processing, upwards of 40 minutes, including manual strain measurement and data synchronization
2. Setup only allows testing of single DEA sample at a time - collection of large number of data points infeasible considering high cycle times
3. Significant extent manual setup, parameter control, and data collection and analysis - time-consuming and poorly optimized
4. Manual control of camera positioning, data synchronization, strain measurement creates significant risk of experimental and user error
5. Poor environmental control (open-air test, camera angle/positioning, lighting) creates risk of experimental variation and error

The DEA testing setup, as it exists in its current form, does not offer acceptable cycle times and data consistency for feasible development of a useful DEA performance dataset. A successful testing framework should be able to collect data on the required parameters with minimal turnaround, minimal manual input from the lab technician, and would minimize experimental variation between trials, and would resolve the weaknesses of the existing setup. As such, the requirements of the proposed DEA testing framework are as follows:

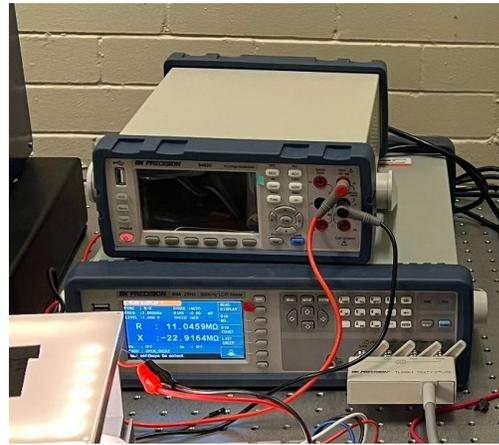
1. Measure voltage (V) to precision of 0.005 V
2. Measure resistance ( $\Omega$ ) to precision of 0.05%
3. Measure capacitance (F) to precision of 0.05%
4. Automatically measure strain ( $\text{mm}^2/\text{mm}^2$ ) to precision of 10%
5. Record data at frequency of 1000 Hz or better

6. Synchronize all data points to same clock cycle
7. Start and stop test and data collection with single operator input
8. Operate within single application container, output all data to .csv and video file
9. Maintain consistent voltage input, DEA sample loading, camera positioning, lighting across trials
10. Reduce testing cycle time to 20 minutes or better
11. Enable testing of multiple samples in parallel with expanded testing framework

The requirements listed above were used to guide the design and development of the DEA testing framework, and were used to assess the performance of the final prototype in Section 4.



**Fig. 2:** Image of contoured DEA sample in DEA sample ring, with electrodes connected to leads



**Fig. 3:** Image of BK Precision 894 high-resolution LCR meter, as used in the existing testing setup (below)

## 2.2 Hardware and Software Approaches to Achieving System Requirements

Requirements 1 and 2 were already achievable with the LCR meter and LabView components of the existing setup, but Requirements 3-10 presented significant engineering challenges in integrating appropriate hardware and software systems to achieve the required testing framework performance.

Requirement 3 was most significant to the successful performance of the DEA testing framework, and most challenging to achieve, as it required the development of a system capable of automatically and accurately measuring the strain of the DEA sample at a frequency of 1000 Hz from camera footage of the sample. This presents a classical image recognition problem, with the aim of consistently detecting and measuring a circular object with low contrast from noisy footage.

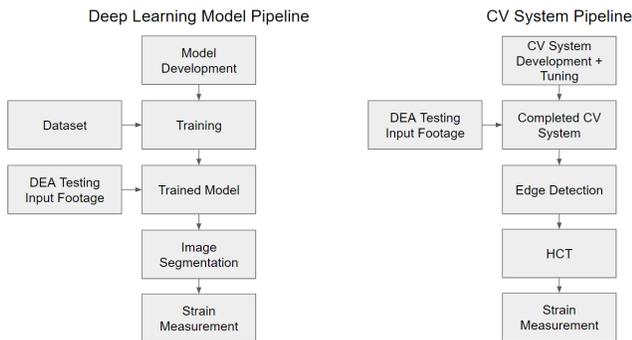
Comparable image recognition applications in literature have used computer vision (CV) or deep learning to automatically detect and measure shapes, with G. Li et al. using simple computer vision to enable measurement of strain of samples in tensile testing setups, while Z. Zhao et al. provide a review of the successful application of

deep learning models in diverse object detection and measurement tasks [7, 8]. Deep learning models for image recognition have been standard since the late 2010s and are highly adaptable to noisy and variable images, offering object identification accuracy of upwards of 98% and similarly high strain measurement accuracy if optimized and tuned to the input images, while also being capable of running detection and measurement operations in fractions of a second with relatively low computer hardware requirements [8]. However, a deep learning image detection solution (and all machine learning approaches) would be inappropriate for use in the DEA testing framework as it requires significant volumes of training data in the form of footage of the DEA sample during testing (either labelled or unlabeled, depending on the type of deep learning model selected) to train the model to detect the DEA and measure strain accurately, defeating the purpose of the development of the DEA testing framework. As such, a CV image detection system was selected for strain measurement, as it requires no training data to operate.

however CV image detection systems are most well-optimized for detection of simple shapes such as circles.

CV systems are “dumb” in the sense that they apply the same image filtration and simple detection algorithms (in effect, simple matrix transformations of pixel values and simple probabilistic algorithms for shape detection) to all input images to detect shapes, however, with appropriate tuning and effective input image control, CV systems have comparable or better performance to deep learning systems, without requiring training data or substantial computer hardware for training the model.

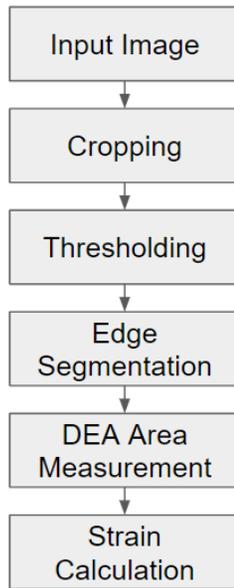
In a shape detection and area calculation application such as the DEA testing framework, a CV system will apply sequential image transformations to crop irrelevant portions of the image, apply contrast transformations to highlight the edges of a shape, apply a pixel value threshold to remove all pixels not part of the edge, and then applying backfilling of the enclosed edge perimeter or simple probabilistic circle detection algorithms to calculate the area. Given consistent positioning of the camera and control of the testing environment across trials, a CV system can accurately measure strain to a precision of 10%, achieving Requirement 3.



**Fig. 4:** Comparison of operation of deep learning model and CV system for strain measurement of DEA samples from video of DEA samples

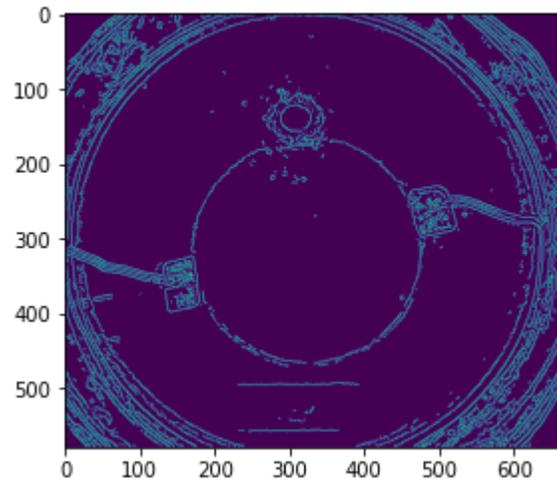
CV image detection systems operate by inputting image data and manipulating pixel values by means of image transformations to remove irrelevant features and detect desired features, whether these be edges, shapes, or complex images such as human faces or license plates,

## CV System Strain Calculation Pipeline



**Fig. 5:** CV system pipeline for strain calculation

Edge-based detection and backfill measurement of shape area, as described by G. Li et al. and B. Li et al., presents the simplest approach to strain measurement, with the detected edge contour being backfilled with white pixels and the number of white pixels being counted and converted to an area in  $\text{mm}^2$ , the  $\text{pixel}^2$  to  $\text{mm}^2$  conversion factor being known beforehand from the measurement in pixels of a ruler in the image frame and the consistent placement of the camera throughout different trials [8, 9]. However, while edge-based backfilling and area calculation is tolerant of non-circular shapes and is the quickest CV strain measurement technique (calculating strain in less than 0.2 seconds per frame, depending on the processing hardware available), it is inaccurate if a complete edge contour cannot be detected before backfilling and area measurement, which, with noisy and low-contrast input images as presented by the USB camera used in the testing setup, leads to unacceptable variation in strain measurement.

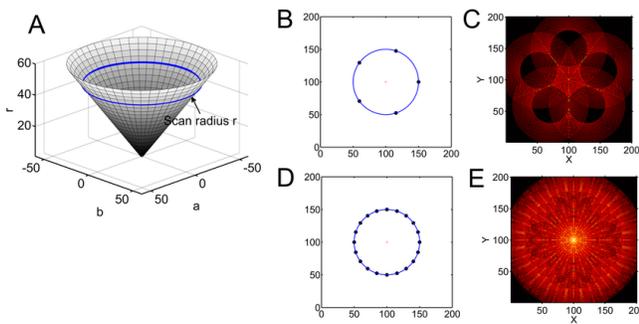


**Fig. 6:** Image of edge detection of DEA sample (Canny algorithm)

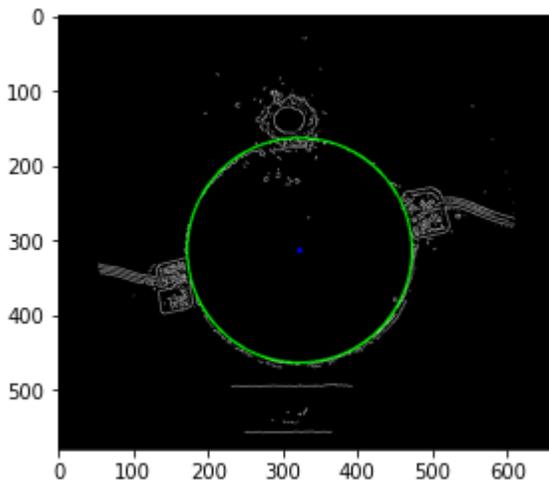
An alternative to edge-based detection and backfill measurement of shape area is Hough Circle Transform detection of circular shapes and calculation of circular area, as described by H.K. Yuen et al. [10]. HCT divides an input image into a grid and moves a circle of varying size across the grid, multiplying the pixel value of the circle with the image pixel value and updating an internal accumulator matrix (convolution). Given parameters of maximum and minimum circle size, number of expected circles within the image, and grid size, the HCT algorithm outputs the location and radius of detected circles on the image, with the area being calculated directly from the radius of the detected circle. The resultant location and radius of the detected circles is precise if the input HCT parameters are correctly tuned to the input image, with the algorithm being tolerant to incomplete edges and partially-circular shapes. HCT does result in relatively high processing times (upwards of 1 second per frame) if poorly optimized, therefore it is important to control the input parameters of the HCT to minimize processing time and maximize accuracy of detected circle radius. Given the circular shape of the DEA sample and the difficulty of obtaining complete edge contours from the noisy input

images, HCT was selected as the CV approach for strain measurement, as it presented accurate and robust strain detection with acceptable image processing times.

For the purposes of simplicity and integration with other software subsystems, the CV strain measurement system was coded in Python. The iteration of the CV strain measurement system is discussed in detail in Section 3.2.



**Fig. 7:** Illustration of operation of HCT - (A) Hough accumulator space for a circle  $(x,y,r)$  for unknown radius  $r$ . The scanning circles are on the cone surface. (B) 5 points on a circle  $(100,100,50)$ . (C) Circles in the Hough accumulator space corresponding to each of the input points in (B). (D) 20 points on a circle  $(100,100,50)$ . (E) Circles in the Hough accumulator space plotted, matching the input points in (D). The intersecting peak represents the center of the circle being detected by the HCT algorithm. [12]



**Fig. 8:** Example of HCT-based detection of a circular shape in an image of a DEA sample, which can be used to calculate the area and strain of the DEA sample

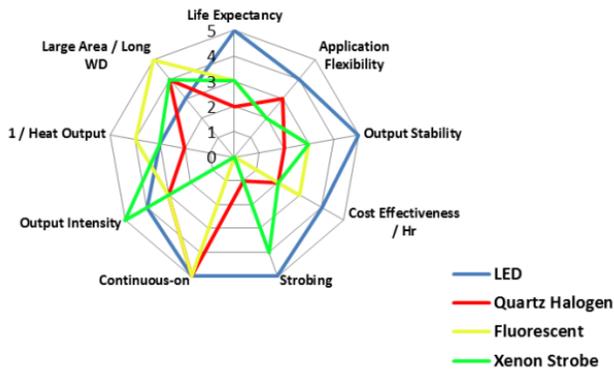
Requirements 5-7 relate to streamlining the experiment control and data management at the software level, by connecting all data inputs (voltage, LCR, strain, time) within a single table, synchronizing the voltage and LCR data collection with the strain data collection, while also enabling the operator to control the experiment, collect and review data, and output tabular and graphical results within a single application framework. This requires the use of a software framework that can be integrated with all data sources (Python CV strain measurement data and LabView voltage and LCR data), with the ability to accurately measure runtime globally, as well as the ability to create user interfaces for displaying and manipulating data, and the ability to write .csv and video files. Python was selected as the software framework to build the application for managing all data processing and user interfaces, as Python is easily integrated with LabView, has a global clock function, enables streamlined processing and recording of data, and can be used to create user interfaces, while also being capable of performing operations sufficiently quickly to meet data frequency requirements.

The same camera as in the existing framework (Spinel OV9281 720p USB camera) was selected as it provides sufficiently high-resolution video, and auto-adjusts focus, allowing its focus to be maintained across trials and ensuring that clear footage is captured in all testing contexts.

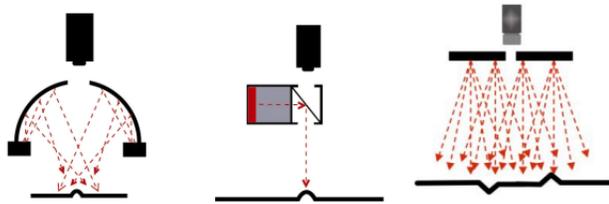
Requirements 8 and 10 relate to the consistent physical setup and scalability of the testing framework, requiring the design of a mechanical testing framework to house the camera, DEA sample, electronic components, and lighting system, while also ensuring consistent setup configuration between trials and minimizing environmental effects (undesired lighting, dust) on the experiment.

The overall physical setup was designed as a three-tier system, consisting of the camera frame, the DEA sample housing, and the electronic system housing. The camera frame mounts the camera with bolts to a rigid frame that does not move relative to the DEA sample between trials, ensuring consistency between experiments, consistent image capture, and accurate strain measurement results from the CV system. The DEA sample housing consists of a box with a circular ring for housing the DEA sample, including sliding doors to enable replacement of the DEA sample between trials, and with the lighting system attached around the perimeter of the top of the inner edge of the box. Holes at the bottom of the DEA sample housing enable the leads to be connected from the electronic system to the DEA electrodes. The electronic system housing consists of a simple box without a roof to house the electronic system, consisting of a simple circuit to enable fine voltage ramping during DEA sample tests. The camera frame bolts to the DEA sample housing, while the DEA sample housing connects to the electronic system housing via the use of integrated pins. The cubic shape and small form-factor of the design was selected to enable multiple setups to be used on a single lab bench, enabling parallel operation of DEA sample tests. Optimization of the physical design and material selection are discussed in Section 3.1.

The lighting system is intended to provide consistent lighting for the DEA sample across different trials, while also minimizing glare on the DEA sample and maximizing the contrast between the DEA and the DEA housing. A review of CV lighting by D. Martin indicates that white LEDs offer the best brightness, power efficiency, and form factor for the DEA testing framework, while also being the most cost-effective option of all the lighting alternatives [11]. Glare presents a major challenge to the functionality of the CV system, as it creates reflective artifacts in the image of the DEA that the CV system is not tuned to filter out - as such, the accuracy of the CV system is significantly impacted by the presence of glare, indicating that controlling glare is an important component of lighting design for the DEA testing framework. The diffuseness of light strongly controls the extent of glare developed on incident surfaces, especially reflective surfaces such as the surface of the DEA sample, with more diffuse light causing less glare. As LEDs are particularly prone to developing harsh point light, LEDs with diffusers were selected for use in the DEA sample housing, to reduce the glare created on the DEA sample and increase the accuracy of the CV strain measurement system. The light color temperature was selected as white (4000 K) to offer the greatest contrast across a range of DEA samples with unmarked, red-marked, and blue-marked borders (DEA sample marking discussed in more detail in Section 3.2). A flat diffuse lighting configuration was selected to further minimize the effect of glare on the DEA sample and to maximize the contrast of the DEA edge against the DEA sample housing.



**Fig. 9:** Comparison of properties of different CV lighting sources [11]



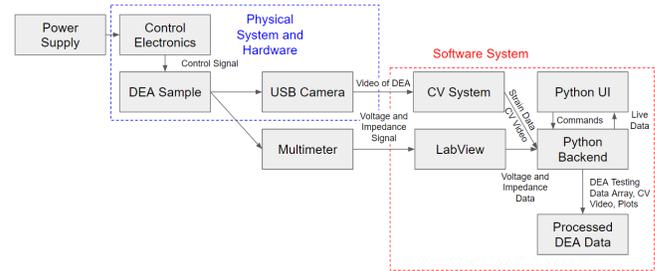
**Fig. 10:** Diagram of diffuse lighting configurations for a CV setup: Dome Diffuse (Left), On-Axis Diffuse (Center), Flat Diffuse (Right) [11]

The electronic system was fully designed by Ang Li, a lab collaborator, and was used to isolate the LCR meter from the high voltage signal supplied to the DEA sample from the power supply. Due to the acceptable performance of the system during testing, the electronic system design used in the integrated framework is the same as the electronic system design used in the existing framework.

### 2.3 Hardware and Software Integration

The unified DEA testing framework is composed of the physical system, consisting of the camera housing, the DEA sample housing, and the electronic system housing, the hardware system, consisting of the camera, the voltage isolation electronics, the power supply, the LCR meter, and the cables connecting the hardware to the computer, and the software system, consisting of the CV strain measurement system, the Python

data collection and synchronization backend, and the Python-based interface. The block diagram of the integrated system is shown in Figure 11 below.



**Fig. 11:** Block diagram of integrated DEA testing framework

## 3. Development and Prototyping

### 3.1 Development and Iteration of Physical System

The physical system underwent several stages of design iteration to optimize geometry, material use, lighting, and scalability, resulting in the development of an optimized physical system that enables a consistent DEA testing setup, robust environmental control, and accurate CV strain measurement.

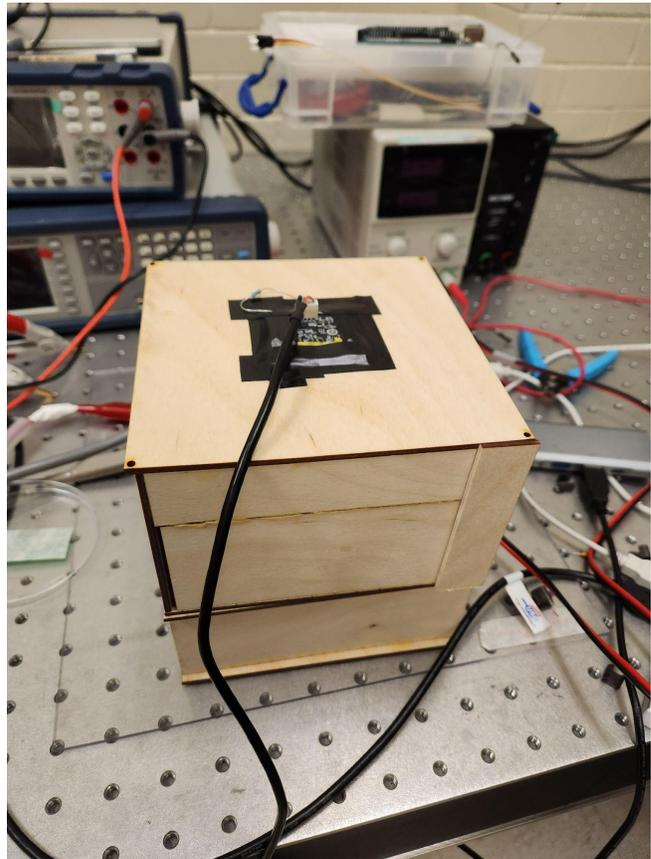
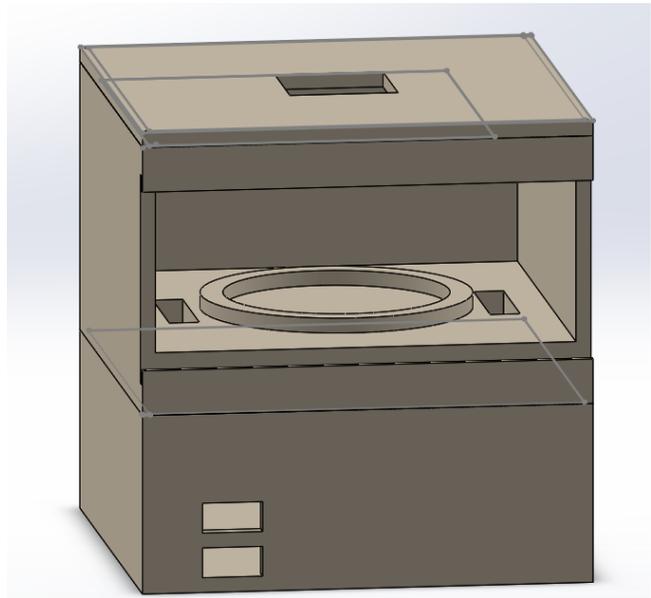
All versions of the physical system were developed in SolidWorks.

The first iteration of the physical system was designed with a 12 x 12 cm footprint, contained a single hinged door for DEA sample changes, included two small holes for running leads between the electronics housing and the DEA sample housing, and was designed to be constructed fully from 3 mm laser cut wood pieces, with M3 bolts being used to assemble all components of the design.

Prototyping of the first iteration of the physical system revealed several design deficiencies that would be rectified in the next iteration. Most

significantly, the low reflectivity and surface pattern of the wood interfered with the accuracy of the CV system, causing the system to incorrectly read and measure the area of the DEA sample. The 12 x 12 cm footprint of the design was too small for the electronic system and presented significant challenges in replacing DEA samples, while the single door could not be assembled via hinges, forcing the use of a temporary bolting solution to seal the DEA sample housing. Additionally, the small holes for leads presented large ergonomic challenges in connecting the DEA sample to the electronic system, while the electronics housing and the DEA sample housing proved difficult to assemble and disassemble due to the use of double-sided M3 studs that could not be easily turned when placed into the wood, and which left a gap between the electronic housing and the DEA sample housing when assembled. Consequently, the three components of the first physical system iteration prototype were connected with tape for ease of assembly and disassembly.

Beyond the listed issues, the prototype of the first iteration of the physical system indicated that other dimensions of the system were correctly defined, with the camera fitting into its frame and the DEA sample fitting into its housing, and that the LED system was able to provide adequate lighting for the CV system to operate consistently across trials, validating the engineering methodology and design decisions used in the design of the physical system.



**Fig. 12:** CAD and physical model of first iteration of physical system

The second iteration of the physical system was designed with a 15 x 15 cm footprint, contained two sliding door for DEA sample changes, included two larger holes for running leads

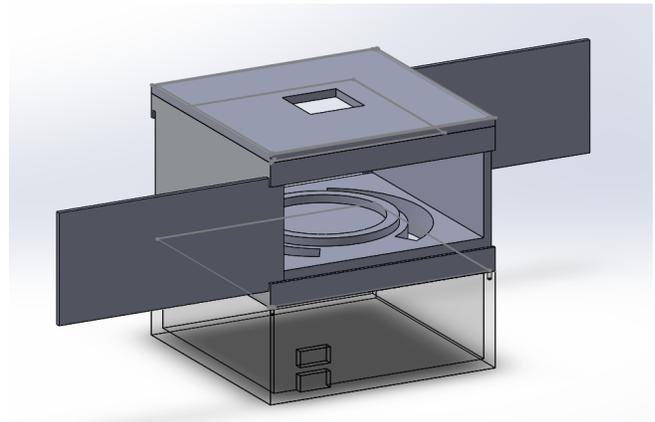
between the electronics housing and the DEA sample housing, with a cover being included to cover the holes during operation of the system to minimize undesired noise in the CV system. The DEA sample housing was designed to be 3D printed from gray PLA for dimensional accuracy and to better diffuse the light in the housing to improve the contrast between the DEA sample and the DEA sample housing, while the single color and rough surface finish of the gray PLA was selected to ensure minimal glare and visual artifacts were captured by the USB camera, in order to minimize the effect on the CV system's accuracy. The camera was designed to be mounted into the camera frame with M2 bolts, while the camera frame was designed to be bolted onto the top of the DEA sample housing via M2 bolts to securely control the position of the camera relative to the DEA sample across different trials for consistency and experimental repeatability. Four 2 x 2 mm square pegs were added to each corner of the bottom of the DEA sample housing to enable it to be mounted on top of the electronics housing, while also allowing for easy access to the electronics housing in the case of required maintenance or modifications. A 10 cm ruler is optionally included inside the DEA sample housing for calibration of the strain measurement system, being automatically cropped out of the input image by the CV system and therefore having no negative effect on the accuracy of the CV system.

The electronics housing in the second iteration of the physical system was designed to be 2 cm taller than in the first iteration of the physical system to fully accommodate the electronic system without bending of any component leads, and was additionally designed to be fully assembled from laser cut material of 3 mm thickness to save material cost.

Low-cost and easily-accessible materials and production processes (laser cutting and 3D printing) were selected as the manufacturing method for the physical system to minimize the cost of replicating the system, enabling greater scalability of the testing framework.

All other parameters, including dimensions, materials, and integration with the hardware components remained unchanged from the first iteration of the physical system, as these parameters were validated to function correctly during testing of the prototype of the first iteration of the physical system.

Prototyping and testing of the second iteration of the physical system, as well as integration of the physical system with the hardware and software system, are discussed in Section 3.4.



**Fig. 13:** CAD model of second iteration of physical system

### **3.2 Development and Iteration of Software and CV System**

The software system underwent several phases of design iteration to improve its performance, with the CV system being optimized continuously to increase accuracy (measured by percent error of strain measurement vs actual strain), robustness (number of different DEA sample contours the system is compatible with and resistance to noise in the image input), and frame processing time (measured in seconds per frame). The

development of the software system saw optimizations to improve data processing speed and the synchronization of data streams, as well as to improve the clarity of the user interface.

The CV system development consisted of three iterations of the system, each iteration offering accuracy and runtime optimizations over the previous iteration, as well as employing better code management and efficiency improvements to improve its integration with the other components of the software system. All iterations of the CV system were developed in Python using the OpenCV library for image transformations.

The first iteration of the CV system consisted of two separate CV systems, each intended to handle a different DEA contour type. DEA contouring is a manual pre-processing step where a lab technician draws a contour around the edge of a DEA sample in red or blue ink to increase the contrast of the edge and reduce the computing runtime required for edge detection - however, contouring increases the processing time of a DEA sample test by a few minutes, and therefore offers an opportunity for runtime optimization if eliminated. Given that some labs use contouring as a standard process, however, the CV system is required to be compatible with both contoured and non-contoured DEA samples. The systems were built by stacking successive image transformations and cropping to highlight the edge of the DEA and remove undesired inputs such as glare, the leads, the electrodes, the DEA sample ring, and noise in the image.

Both CV systems of the first iteration were only able to operate on sequences of image files instead of live video, this deficiency being rectified in future iterations to enable live image processing.

All testing of CV system accuracy and runtime was performed on three videos of DEA sample

tests, one with a non-contoured DEA sample and two with DEA samples contoured in red and blue ink, respectively.

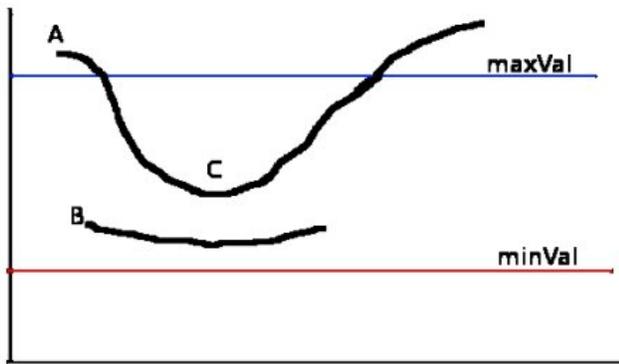
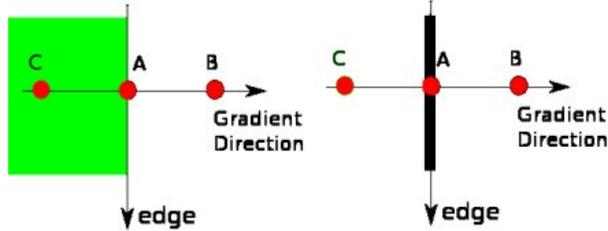
The order of transformations of the CV system for non-contoured DEA samples began with cropping the input image to remove the DEA sample ring from the image. The input image is fed through a Canny edge detection filter with thresholds of 16 and 18, which applies a Gaussian blur with a 5 x 5 pixel kernel to denoise the image, and then applies an edge gradient detection algorithm to mark detected edges with white pixels, all other pixels being set to black. While this detected the edge of the DEA sample, it also detected the electrodes, the leads, and the glare created by the light source, which impacted the processing speed of the CV system and would need to be rectified in future iterations. The system then applied the HCT with the following constant parameters:

1. Minimum distance between circle centers: 1000 pixels
2. Accumulator (grid) to image size ratio: 1:1
3. Edge detector threshold: 20 pixels
4. Accumulator threshold for circle centers: 5
5. Maximum circle radius: 250 pixels
6. Minimum circle radius: 100 pixels

The system uses the result of the HCT (circle center x-position, circle center y-position, circle radius) to draw the circle of the DEA area on the image, with the circle radius being converted into a DEA area in pixels<sup>2</sup>, strain being calculated from the ratio in the area of the initial frame to the area of the currently-processed frame. The parameters of the HCT transformation were obtained from the expected number of circles in the image (1) and the expected radius range of the DEA sample (100-250 pixels), and other parameters being selected via tuning of the system, making the system poorly adaptable in the case of different DEA sample sizes.

$$\text{Edge\_Gradient } (G) = \sqrt{G_x^2 + G_y^2}$$

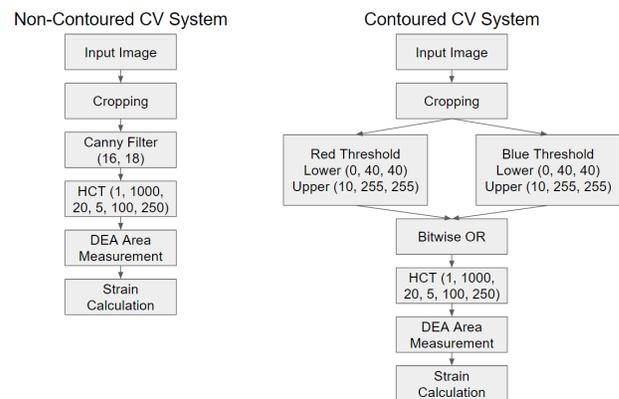
$$\text{Angle } (\theta) = \tan^{-1} \left( \frac{G_y}{G_x} \right)$$



**Fig. 14:** Canny edge detection algorithm edge gradient and angle equations, edge detector (Edge gradient of Point A is referenced with point B and C to see if it forms a local maximum along the gradient direction. If so, it is passed through the thresholding function, if not, it is set to zero), and hysteresis thresholding function [13]

The order of transformations of the CV system for contoured DEA samples was identical to the CV system for non-contoured DEA samples with the exception of the initial edge detection algorithm. The CV system for contoured DEA samples used two parallel simple thresholding transformations, which function by setting all pixels outside of a range of HSV (hue, saturation, value) values in the input image to black and setting all pixels within the range of HSV values to white, enabling the isolation of pixels of a particular color range. This was used to create two parallel image masks for the red pixels and the blue pixels present in the DEA sample edge of the input image, with the resultant image masks being combined in a

pixel-by-pixel OR operation to form the black-and-white image of the detected DEA sample edge. This transformation had much lower runtime than the Canny edge detection filter as it involved much fewer and simpler image transformations, and had the added benefit of significantly reducing the detection of the electrodes and sample edge, enabling faster HCT runtime and better measured strain precision than the CV system for non-contoured DEA samples.



**Fig. 15:** Architecture of first iteration of CV strain measurement system

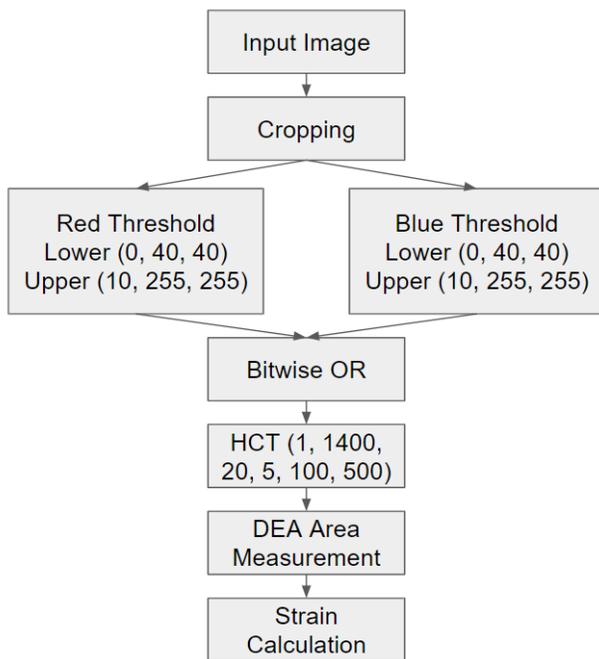
In testing on the sample videos, both systems had high runtimes (5.211 s/frame for contoured, 8.487 s/frame for non-contoured) due to a high number of transformations and poorly-optimized HCT parameters, and the disconnected architecture of the CV system meant that two instances would have to run concurrently to robustly function for strain measurement of both non-contoured and contoured DEA sample, further straining system resources and increasing runtime.

The second iteration of the CV system consisted of a single integrated system capable of operating on both non-contoured and contoured DEA samples, offering improved detection accuracy and strain measurement runtime. The changes from the first iteration include the use of simple edge thresholding for both contoured and

non-contoured edge detection, and tuning of the HCT parameters to the following values:

1. Minimum distance between circle centers: 1400 pixels
2. Accumulator (grid) to image size ratio: 1:1
3. Edge detector threshold: 20 pixels
4. Accumulator threshold for circle centers: 5
5. Maximum circle radius: 500 pixels
6. Minimum circle radius: 100 pixels

These changes to the HCT parameters reduced the grid size and reduced the number of circle drawing attempts, and increased the range of acceptable circle radii, significantly decreasing runtime of HCT while increasing its robustness to different sizes of DEA samples.



**Fig. 16:** Architecture of second iteration of CV strain measurement system

In testing on the sample videos, the second iteration demonstrated improved runtime versus the first iteration, with an average runtime of 4.978 seconds per frame. Further improvements in runtime were achievable by reduction of the number of convolutions in the HCT, as well as by

reduction of the resolution of the input image to reduce the grid size of HCT. Finally, the CV system needs to be modified to function with inputs of a live camera feed to be fully integrated with the physical system and hardware.

The third iteration of the CV system consists of the same architecture as the third system, with tuning applied to restrict the range of red and blue HSV pixel thresholds to eliminate the detection of leads and electrodes as edges, further optimization of HST parameters to reduce runtime, and structuring of the CV analysis code as a function that can be called on a given frame to allow for the CV analysis to be integrated into existing code, such as the data collection and synchronization code, with low runtime impact.

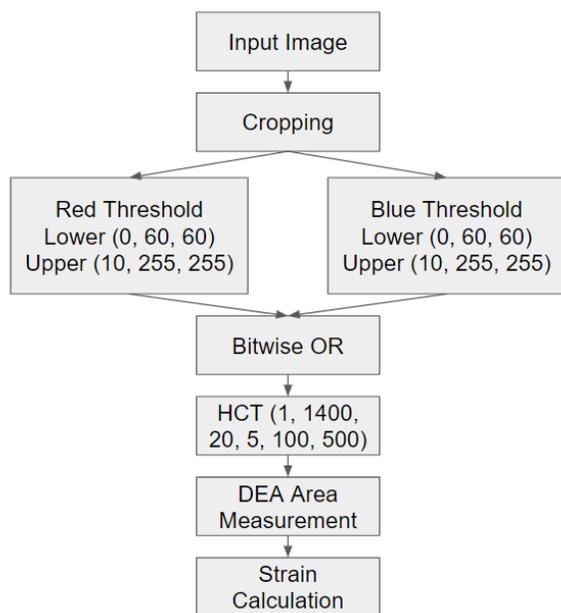
The third iteration of the CV strain measurement system demonstrated improved runtime versus the second iteration, with an average runtime of 1.232 seconds per frame and a maximum runtime of 2.632 seconds per frame, representing a decrease of 75.25% of average runtime and 3.746 seconds per frame versus the second iteration. While this runtime is insufficient to perform CV strain measurement at 120 Hz (the framerate captured by the camera), the system can parse video footage at 40 Hz, making the average runtime of the CV system sufficiently low to capture accurate strain results.

**Table 1:** Results of First, Second, and Third CV System Testing

	Runtime (s/frame)	Percent Runtime Reduction
First Iteration	6.849	0.00%
Second Iteration	4.978	-27.32%
Third Iteration	1.232	-82.01%

The system additionally contains a webcam capture parser to convert MJPEG video files (the video capture format of the USB camera) to 3-dimensional arrays containing pixel HSV data for the captured frame, enabling the system to process live video frames as they are captured by the USB camera.

The third iteration of the CV strain measurement system inputs a given video frame developed by the parser, and outputs a 3-dimensional array of the containing pixel HSV data of the detected circle superimposed on the filtered input frame, along with the area of the detected circle in pixels<sup>2</sup>, with an average runtime of just under 3 seconds per frame.



**Fig. 17:** Architecture of third iteration of CV strain measurement system

The data processing and synchronization backend of the software system existed in two iterations, with the first iteration providing a baseline for functionality of the system, and the second iteration containing optimizations to improve data processing speed and the synchronization of data streams, as well as to improve the clarity of the user interface.

The first iteration of the data processing and synchronization backend was developed around the Python time module, which implements a global clock with millisecond resolution, and the Python CSV module, which allows for reading and writing of CSV files. The global timer begins counting at the start of the code, and continues operating until the code stops executing due to user input. The backend is built around a while loop that continues as long as the webcam is recording, and can be broken by the user pressing the “Q” key. The webcam capture runs separately to the while loop, with the most recently captured frame being used when the strain measurement function is called. The LabView data collection software was modified to dump voltage and impedance data to a csv file every second, which the Python backend parses every second, offset by 0.2s after the LabView data dump, inserting the data into the experimental data array. The strain measurement is collected by running the CV strain measurement function on the most recently captured frame at the time that the strain measurement function is called, with the timestamp of the frame being recorded. Once the strain measurement calculations are complete, the circle area and strain measurement are inserted into the experimental data matrix, with the circle area and strain columns being backfilled until the timestamp of the last recorded circle area and strain data point. Once the user stops the webcam and data recording loop by pressing “Q”, the backend system writes the data array to a CSV file and saves it to the user’s selected file path (which can be modified in the code parameters).

The first iteration of the backend system was a simple and effective method of data collection and synchronizing the different data streams, but could still be optimized to reduce runtime and to streamline data processing and analysis work by graphing relevant relationships and displaying

live data readings for experimental parameter monitoring.

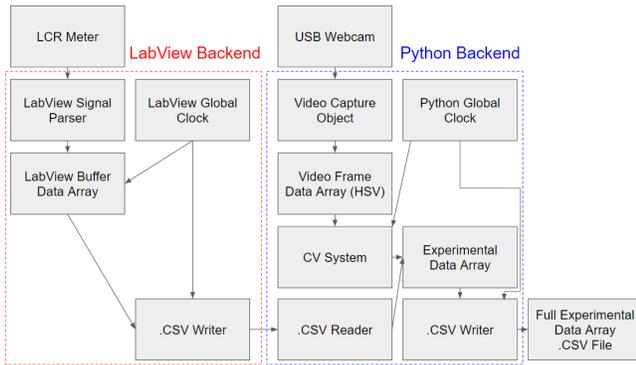


Fig. 18: Block diagram of first iteration of backend system

	A	B	C
1	70685.83	0	00:00:00
2	70685.83	500	00:16.7
3	71631.45	1000	00:33.3
4	71631.45	1500	00:00:50
5	71631.45	2000	01:06.7
6	71631.45	2500	01:23.3
7	71631.45	3000	00:01:40
8	71631.45	3500	01:56.7
9	71631.45	4000	02:13.3
10	71631.45	4500	00:02:30
11	71631.45	5000	02:46.7
12	71631.45	5500	03:03.3
13	71631.45	6000	00:03:20
14	71631.45	6500	03:36.7
15	70685.83	7000	03:53.3
16	70685.83	7500	00:04:10
17	72583.36	8000	04:26.7
18	72583.36	8500	04:43.3
19	71631.45	9000	00:05:00
20	71631.45	9500	05:16.7
21	72583.36	10000	05:33.3
22	72583.36	10500	00:05:50
23	73541.54	11000	06:06.7
24	73541.54	11500	06:23.3
25	73541.54	12000	00:06:40
26	73541.54	12500	06:56.7
27	73541.54	13000	07:13.3

Fig. 19: Full experimental data CSV file

The second iteration of the backend system was built on the framework of the first iteration of the backend system, with three key differences: The second iteration uses an optimized array structure, storing data points as float64 data types, improving data processing time and reducing runtime, creates a TCP/IP connection between Python and LabView to enable live transmission of voltage and LCR data, significantly improving runtime, and houses all user functionality in a lightweight UI that allows the user to start and stop the data collection, displays voltage, LCR, and strain, plots voltage-resistance and voltage-strain data, and saves experimental data and a video of the frames processed by the CV

strain measurement system over the course of the experiment, enabling the resultant data and frames to be parsed into a dataset of DEA testing results for training of future ML-based models for prediction of DEA behavior.

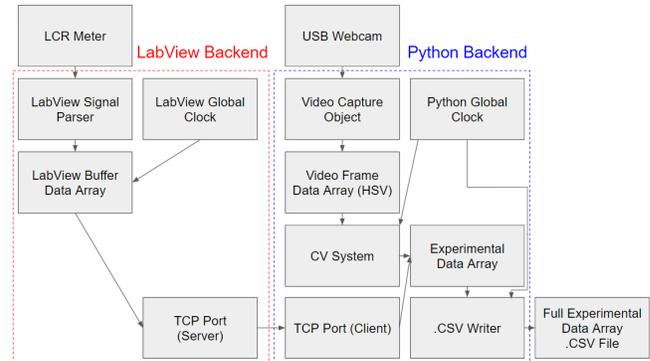


Fig. 20: Block diagram of second iteration of backend system

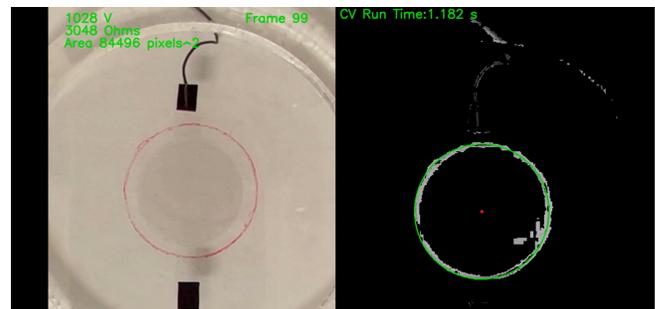


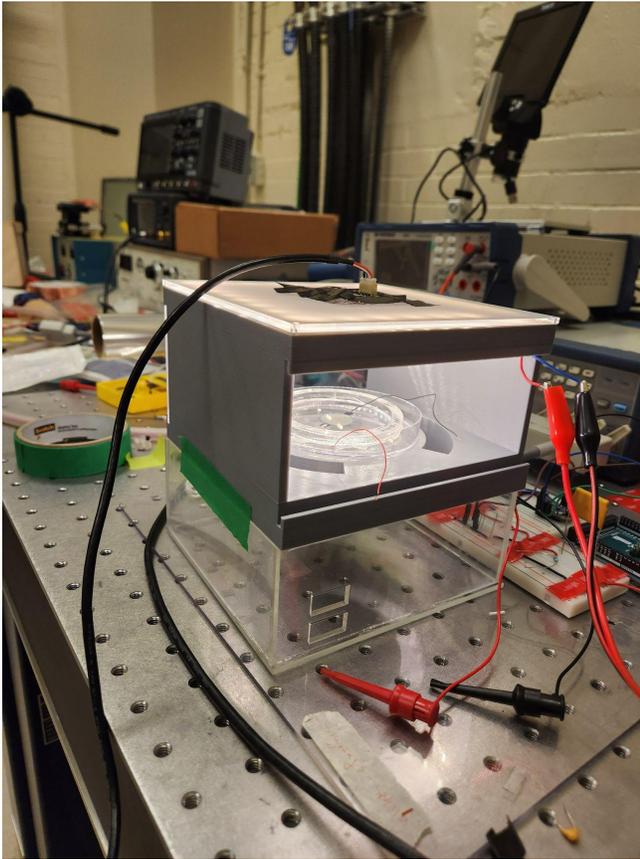
Fig. 21: UI of second iteration of backend system

### 3.3 Prototyping and Integration of Framework

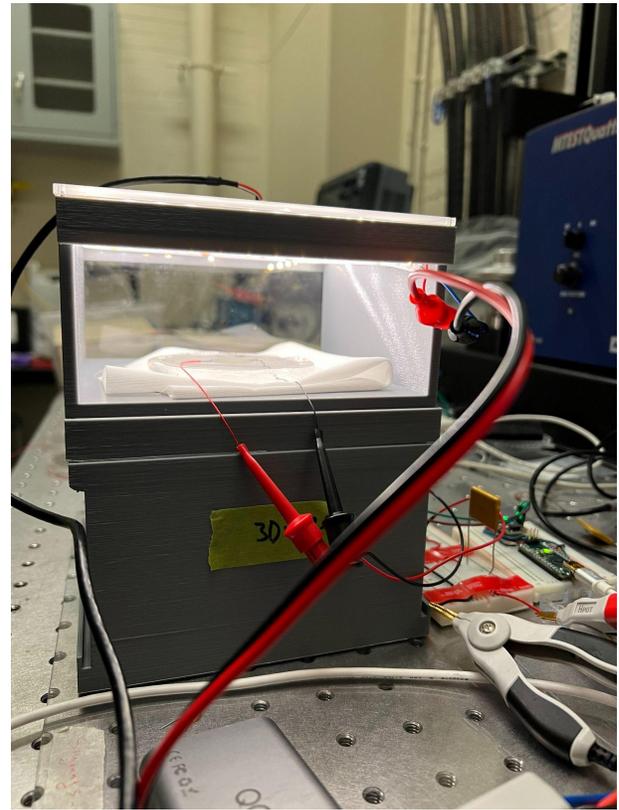
The prototype of the framework was designed to replicate the system's physical, hardware, and software design as closely as possible, to allow for testing of the framework's functionality and integration, as well as to allow for testing of the framework with a DEA sample to assess the performance of the system in an experimental context.

The physical prototype was assembled to the specifications and dimensions of the prototype model, with the camera frame and the DEA sample housing being 3D printed from gray PLA, and the electronics housing being cut from

transparent 3 mm acrylic sheets and assembled using glue. The USB camera was secured to the camera frame by means of tape to allow for easy removal, while wires were used instead of M2 bolts to connect the camera frame and the DEA sample housing to allow for easy disassembly. The LED lighting system was mounted to the inside of the DEA sample housing with transparent tape, with the power wire being run through one of the door openings of the DEA sample housing. Two sets of doors, consisting of two opaque and two transparent doors, were cut from 3 mm acrylic sheets to allow for testing of the effect of external lighting on the performance of the CV strain measurement system.

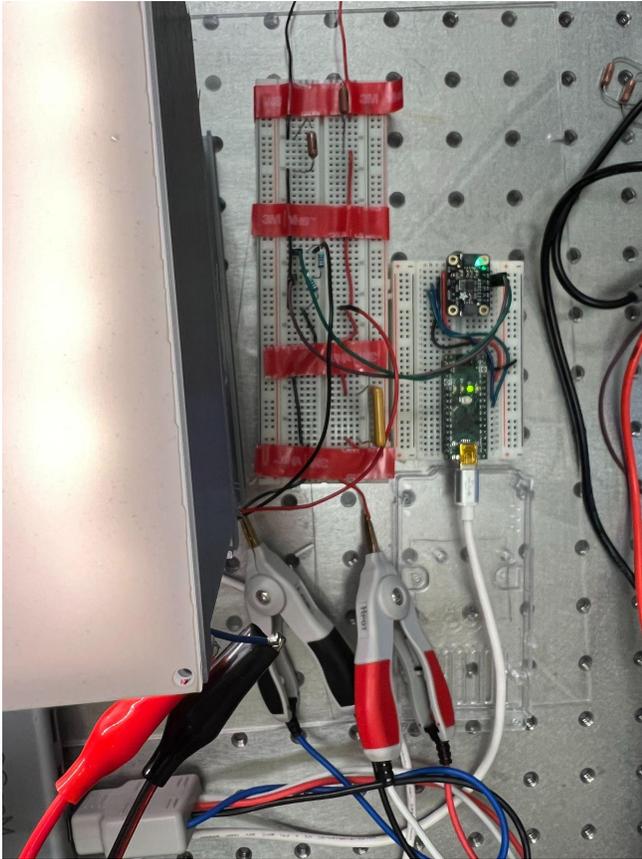


**Fig. 22:** Image of physical prototype and hardware system

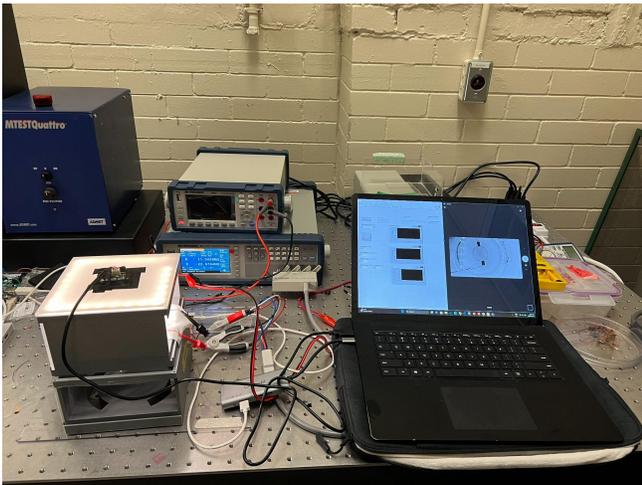


**Fig. 23:** Image of physical prototype and hardware system, showing DEA sample in DEA sample housing and white paper covering for lead holes

The hardware system, consisting of the power supply, LCR meter, electronics, and the USB camera, was configured identically to its intended configuration in the integrated design, with the exception of the assembly of the electronics system, which was assembled on a breadboard instead of a PCB for ease of assembly and placed outside of the electronics housing during testing, with leads being run through the DEA sample housing doors. The electronic system contained identical components to the components specified in the integrated design and was connected and configured according to the specifications in the design, and therefore operated identically to the final framework design.



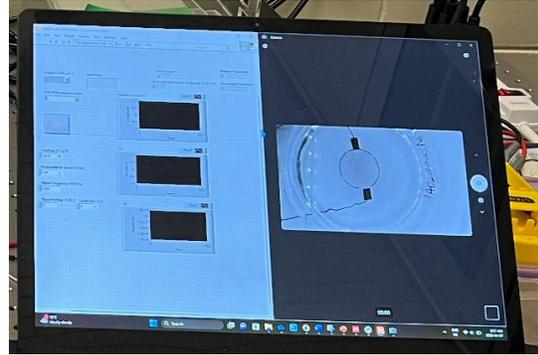
**Fig. 24:** Image of electronic system, used to isolate LCR meter from high voltage signal supplied to DEA sample



**Fig. 25:** Image of full testing framework, consisting of physical prototype, hardware (including electronic system), and software (running on computer)

The software system consisted of the same Python code and LabView code and configuration as in the final framework design, with the only difference being that the Python code was

compiled as a .ipynb file to allow for selective code execution and easier collection and analysis of relevant performance statistics, and that the first iteration of the Python and LabView backend was operated instead of the second iteration to allow for easier modification of data collection and data synchronization parameters.



**Fig. 26:** Image of testing framework software Left: LabView backend, Right: Webcam output and Python backend

The Python and LabView code were run on a laptop running Windows 11 with 8 GB of RAM, with no other applications running beyond Python and LabView. This computer and these testing conditions were held constant throughout testing to assess the relative performance of the framework to the previous testing methodology.

#### 4. Testing of System and Results

Testing of the integrated framework prototype was conducted to assess the time required to perform a DEA sample test with the framework and to compare its timing and accuracy performance to the previous testing framework, as well as to validate the integration and individual performance of the physical, hardware, and software systems.

## 4.1 Integrated Framework vs. Existing Framework Performance Test

### 4.1.1 Test Procedure

A standard test for comparing the performance of the integrated framework to the previous testing framework was devised and consisted of the following components and procedure:

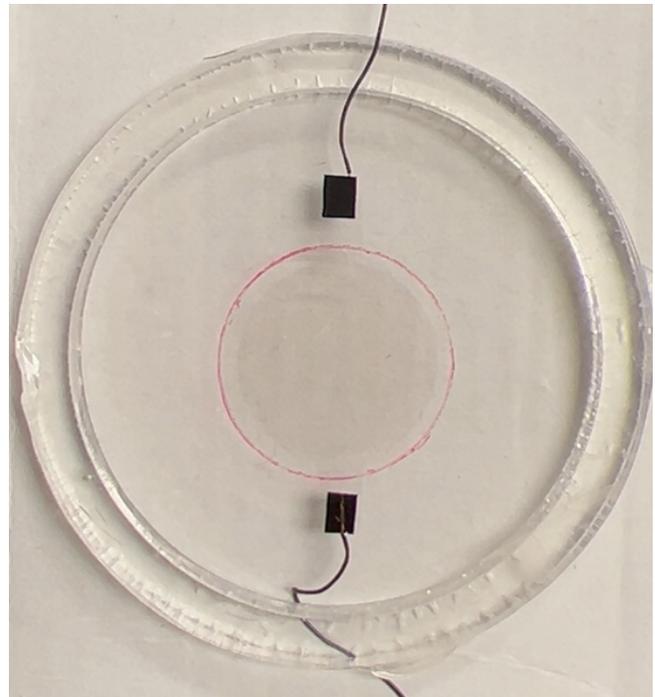
1. A previously-unseen 3-minute video file of a contoured DEA sample undergoing testing to failure is loaded onto the standard testing computer (Due to time and resource constraints, a recorded video of a DEA sample test was used instead of live video from the USB camera, this was considered appropriate as the preparation of the sample would take the same amount of time across both the integrated and existing testing frameworks. The performance of the integrated framework live footage of a DEA sample was assessed in a separate test, described below. A contoured DEA sample was used to allow for human measurement of strain in Step 3 of this test, as a non-contoured DEA sample increases the difficulty of and time required for manual strain measurement)
2. The Python and LabView applications are started and process the input video, with 50 equally-spaced frames being processed for strain measurement, and plots of voltage-resistance and voltage-strain being generated, the time between the start and end of the application operation being recorded (50 frames were selected to enable comparison to manual strain measurement)
3. A lab technician performs strain measurement by manual processing of individual images on the same 50 input frames and generates plots of

voltage-resistance and voltage-strain using Excel or any other data manipulation tool, the time required for the completion of these tasks being recorded

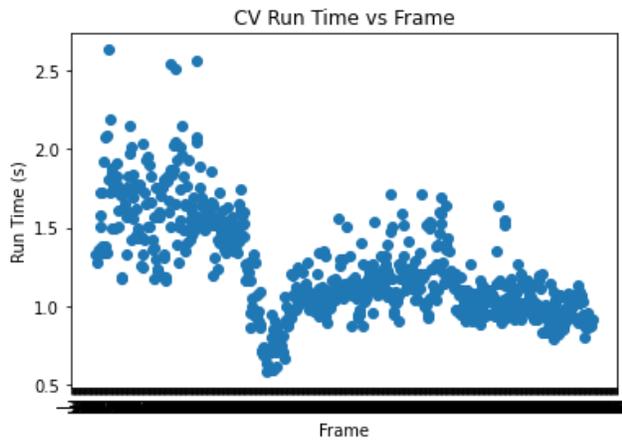
4. The total data collection and processing time for the integrated framework and existing framework are compared

Given that this test is performed in a single trial, by a single lab technician on a single given sample video, the results are not intended to be an exact measurement of the performance of the integrated testing framework, but rather a rough comparison of the performance of the integrated testing framework and the previously existing framework, to demonstrate the large-scale gains in performance achieved by the integrated testing framework.

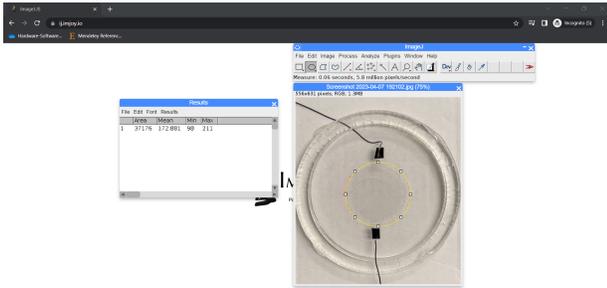
This test was performed once due to time constraints, with the results of the test being tabulated in Table 2 below.



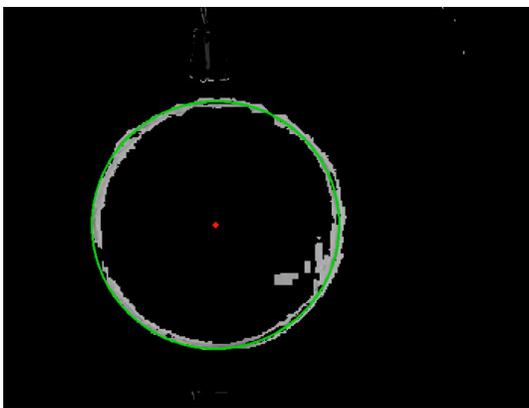
**Fig. 27:** Frame of previously-unseen DEA sample testing video used in Test 4.1



**Fig. 28:** Image of CV system runtime for test of integrated framework (only the 50 equally-spaced frames were counted in the strain measurement time of the integrated framework runtime)



**Fig. 29:** Image of manual image measurement performed in ImageJ



**Fig. 30:** Image of fully processed frame with circle detected by HCT drawn in green, with center of circle marked in red

**Table 2:** Results of Integrated Framework vs. Existing Framework Performance Test

	Integrated Framework	Existing Framework
Data Collection (s)	180.000	180.000
Strain Measurement (s)	61.600	1484.998
Plotting (s)	0.523	97.801
<b>Total Runtime (s)</b>	<b>242.123</b>	<b>1762.799</b>

#### 4.1.2 Results and Discussion

The results in Table 2 demonstrate that the integrated framework takes 242.123 seconds for collecting, processing, and plotting of relevant experimental data (voltage, impedance, strain) for a single DEA sample test, compared to 1762.799 seconds for the existing framework, representing a 86.26% decrease in total runtime with the integrated framework. If two or more integrated testing frameworks are operated in parallel (as discussed in Section 5), these time savings increase, with the overall runtime for a single DEA sample test being inversely proportional to the number of setups being run in parallel. These results indicate that this framework can reliably collect and process DEA testing data faster and with more consistency than the existing testing framework, and demonstrates that collection of sufficient data to form a functional dataset of DEA performance is feasible with a relatively small number of sample testing units (consisting of the physical system, the camera, and the electronics system) in a parallel testing setup.

## 4.2 Integrated System Live DEA Sample Test

### 4.2.1 Test Procedure

A test for verifying the functionality of the integrated framework and validating the ability of the system to operate correctly with live video was devised and consisted of the following components and procedure:

1. Load contoured DEA sample (or similarly colored ring) into assembled testing setup and attach electrodes and leads, but do not turn on the power supply (Testing of the expansion of a DEA was not tested due to time and resource constraints, though a test of the integrated framework on a previously-unseen DEA sample testing video that demonstrates considerable testing and processing runtime reductions, as performed in Section 4.1, provides an sufficient indication of the functionality of the CV system.)
2. Close doors of DEA sample housing, using opaque doors
3. Start Python and LabView applications, leave system running for 2 minutes and record CV strain measurement system processing time
4. Repeat Step 3, replacing opaque doors of DEA sample housing with transparent doors

The results of this test are tabulated in Table 3 below.

**Table 3:** Results of Live DEA Sample Test for Opaque Doors and Non-Opaque Doors (Processing time in seconds/frame)

	Contoured DEA
Opaque Doors	1.289
Non-Opaque Doors	1.340

### 4.2.2 Results and Discussion

The results in Table 3 demonstrate that the integrated framework is able to function with live video and data inputs, and that the system is able to sufficiently minimize the impact of outside lighting on the CV strain measurement system.

## 5. Discussion and Future Work

The success of the presented integrated DEA testing framework in accurately measuring DEA strain and reducing DEA sample test runtime, as well as the the low cost and resources required for replication of the framework, enables this framework to be recreated and used for DEA sample testing in any soft robotics lab with access to a 3D printer. The consistency of the testing setup between trials and between different instances of the same setup ensures that DEA performance data collected using this framework is usable in characterizing the performance of the tested DEA sample independently of the computing and measurement equipment of the lab that collected the data or the skill of the lab technician operating the tests.

The framework is also highly scalable and can be easily parallelized to further increase time savings, with the only components that need to be replicated across a parallel setup being the physical system, the USB camera, the voltage isolation electronics, and the voltage and LCR measurement device (although a LCR meter that can use multiple leads allows a single LCR meter to measure data across multiple testing setups),

with the only limitation on the number of setups that can be operated in parallel being the computing resources and bench space available to a given lab.

The ability of this framework to allow the operation of upwards of 10 parallel testing setups by a single lab technician with relatively modest computing resources and a minimal power supply setup enables a significant acceleration of DEA sample testing and data collection.

Consequently, the presented framework can be feasibly used by a single lab to compile a dataset of DEA performance, with the quality and value of this dataset growing exponentially with larger parallel testing setups and replication of the testing setup in other labs, offering the opportunity for the operation of a collaborative and low-cost research initiative to investigate the properties and performance of a wide range of DEA compositions and configurations, greatly increasing the depth of knowledge of DEA characterization and performance in the soft robotics field.

Significantly, the dataset can be used in training ML models to predict DEA performance given inputs of voltage, composition, and configuration, potentially enabling the development of better control systems for DEAs, increasing the feasibility of DEAs as artificial muscles in soft robotics and prosthetic applications.

Future work with this framework should use the system to characterize the performance of a wide range of DEA compositions and configurations to compile sufficient data for a training dataset, and should then build and test a ML model for predicting the performance of DEAs, with the most likely model candidates for an effective system being LSTMs or Transformers - however, more research is needed to investigate the

feasibility of using such models for reliable prediction of DEA performance.

Additionally, further research could benefit from application of this framework with modifications to the testing of other soft robotic actuator classes that are similar in geometry and intended applications to DEAs, with the most likely candidates being Ionic Polymeric Artificial Muscles, Soft Magnetic Artificial Muscles, and Piezoelectric Artificial Muscles.

This thesis aimed to develop and demonstrate the feasibility of an integrated testing framework to accelerate the characterization of DEA samples, with emphasis on reducing test runtime, increasing repeatability and reproducibility of tests, and system scalability. The requirements of the system were compiled and used to guide the design of the physical, hardware, and software systems of the framework, with the challenges encountered during the design process and the implemented solutions being explored. Testing of the integrated framework on a sample DEA video and a live video input indicated that the system accurately captures voltage, impedance, and strain data for a DEA sample test, while significantly reducing the test runtime, demonstrating the feasibility of the integrated framework for conducting wide-scale tests of DEA performance.

The results of this thesis demonstrate the high feasibility of intensive exploration of DEA performance and control compared to other classes of artificial muscles, increasing the potential of this class of soft robotic actuators for use as artificial muscles in advanced robotic, industrial, or prosthetic applications.

## 6. References

1. M. Shi and E. M. Yeatman, "A comparative review of artificial muscles for microsystem applications," *Microsystems & Nanoengineering*, vol. 7, no. 1, 2021.
2. M. Duduta, E. Hajiesmaili, H. Zhao, R. J. Wood, and D. R. Clarke, "Realizing the potential of dielectric elastomer artificial muscles," *Proceedings of the National Academy of Sciences*, vol. 116, no. 7, pp. 2476–2481, 2019.
3. J. Chipka, M. A. Meller, A. Volkov, M. Bryant, and E. Garcia, "Linear dynamometer testing of hydraulic artificial muscles with variable recruitment," *Journal of Intelligent Material Systems and Structures*, vol. 28, no. 15, pp. 2051–2063, 2017.
4. J. Garbulinski, S. C. Balasankula, and N. M. Wereley, "Characterization and analysis of extensile fluidic artificial muscles," *Actuators*, vol. 10, no. 2, p. 26, 2021.
5. T. W. Henderson, Y. Zhi, A. Liu, and M. C. Yip, "Data-driven actuator selection for artificial muscle-powered robots," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021.
6. A. Bailly, C. Blanc, É. Francis, T. Guillotin, F. Jamal, B. Wakim, and P. Roy, "Effects of dataset size and interactions on the prediction performance of logistic regression and Deep Learning Models," *Computer Methods and Programs in Biomedicine*, vol. 213, p. 106504, Nov. 2021.
7. G. Li, Q. Tan, Q. Sun, and Y. Hou, "Normal strain measurement by Machine Vision," *Measurement*, vol. 50, pp. 106–114, 2014.
8. Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A Review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.
9. B. Li, "Application of machine vision technology in geometric dimension measurement of small parts," *EURASIP Journal on Image and Video Processing*, vol. 2018, no. 1, 2018.
10. H. K. Yuen, J. Princen, J. Illingworth, and J. Kittler, "Comparative study of hough transform methods for circle finding," *Image and Vision Computing*, vol. 8, no. 1, pp. 71–77, 1990.
11. D. Martin, "A practical guide to machine vision lighting," [advancedillumination.com](https://www.advancedillumination.com). [Online]. Available: <https://www.advancedillumination.com/wp-content/uploads/2018/10/A-Practical-Guide-to-Machine-Vision-Lighting-v.-4-Generic.pdf>. [Accessed: 07-Mar-2023].
12. S. Maji and M. P. Bruchez, "Inferring biological structures from super-resolution single molecule images using Generative Models," *PLoS ONE*, vol. 7, no. 5, 2012.
13. "Canny edge detection," [opencv.org](https://docs.opencv.org/3.4/da/d22/tutorial_py_canny.html). [Online]. Available: [https://docs.opencv.org/3.4/da/d22/tutorial\\_py\\_canny.html](https://docs.opencv.org/3.4/da/d22/tutorial_py_canny.html). [Accessed: 07-Mar-2023].