Development of a data-driven kinematic model for a soft pneumatic actuator

Ibrahima Sow, Gilles Decroly

***Abstract*—Minimally invasive surgery can benefit greatly from the safety and navigation capabilities offered by soft robotics. In this report, we review the development steps of a kinematic model of a one-degree-of-freedom soft actuator. We designed a pneumatic actuator with an integrated strain sensor that deforms through a strain-limiting layer. We manufactured it using specially designed molds. In addition, we placed a pressure sensor at the inlet of the pneumatic actuator. The radius was calculated using an iterative algorithm acting on images provided by a camera placed in front of the actuator. The system was operated with an existing pressure station, to which a PI controller was added. The three specified modalities - strain, pressure, radius - provide a representation of the actuator state and a massive amount of data has been collected with different pressure input scenarios. Future work includes training an LSTM array using the data collected by the experimental apparatus.**

***Index Terms*—soft robotics, surgery, perception, design, control.**

I. Introduction

Recent advances in soft robotics have opened up many possibilities for minimally invasive surgery and endoscopy. In particular, actuated soft bodies constitute a safer and more dexterous tool than current steerable catheters. Their inherent deformability frees them from the traditional constraints of rigid bodies and allows them to perform complex and compliant movements in confined spaces. This physical adaptability ensures the robustness of the device and the safety of surgical operations [1]. In addition to their improved navigation capabilities, soft robots hold the promise of future autonomous actuators relieving professional practitioners of critical tasks.

These challenging tasks require a sufficient understanding of the underlying kinematics and dynamics of the robotic system in order to derive a reliable control strategy. However, the intrinsic non-linear properties of the soft material and the infinite number of degrees of freedom have made it difficult to formulate a suitable mathematical model [2]. However, recent works [3] [4] with data-driven techniques have shown promising results which allows to bypass many of the difficulties encountered in modeling the soft robot dynamics.

Perception precedes action, which means that any controlled system relies on an internal representation of its own state to perform the required actions. The quality of this perception often affects the performance of the control system, especially in critical applications such as minimally invasive endoscopy. State estimation can be achieved through multi-modal sensing, combining proprioceptive and exteroceptive, active and passive, and/or embedded and external sensors. The combination provides information on the robot itself as well as on the environments in which it operates.

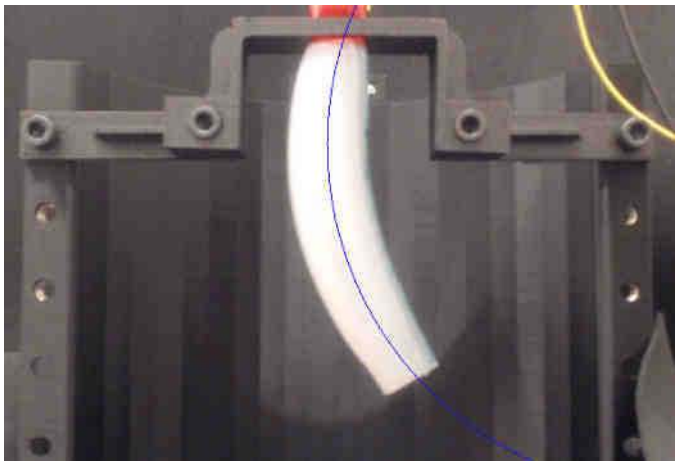

Fig. 1: Soft actuator

The present work aims at building an internal representation of the actuator to be used in a feedback loop. Specifically, this paper describes the development of an experimental setup needed to develop such a real-time sensory system for a soft endoscopic device based on extensive data collection. The system collects pressure and deformation data from the flexible actuator and uses an external camera to calculate the radius of curvature from the images, while keeping track of the timing between samples. The report concludes with the plan and future prospects for the continuation of the project, namely training the model. All codes and designs are available upon request from the authors.

II. Materials and Methods

Our approach relied on a custom designed soft planar pneumatic actuator with an embedded commercial strain sensor composed of a single pressure channel. Experiments were carried based on three feedback components: the pressure in the chamber p, the strain ϵ measured in terms of voltage by the strain sensor, and a computed radius of curvature r extracted from images via an algorithm. Together, they allow for a representation of the internal state of the actuator.

The low-level pneumatic actuation was handled by a modified pressure board inspired by the Soft robotics Toolkit [5]. A micro-controller board acted as intermediate between the computer and pressure board. It recorded pressure and strain sensor readings from the actuator at a sample rate of 1 kHz and sent back the current sample when requested to the computer while computing the new signal for the pressure station board

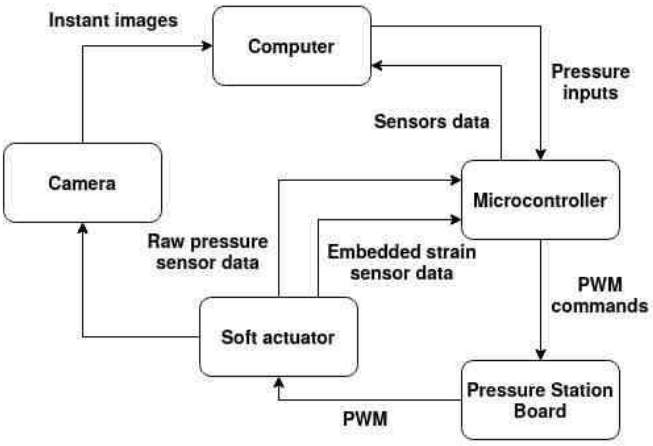

Fig. 2: General overview of the system.

based on its PI controller. In parallel, instant images were processed by the computer to extract the instantaneous radius of curvature. Simultaneous data collection was achieved by issuing a request to send a sample of data from the computer to the microcontroller. A general overview of the experimental set-up is shown in Figure 10. The sample rate was set at 10 Hz for the entire dataset.

A. Actuator fabrication and sensor embedding

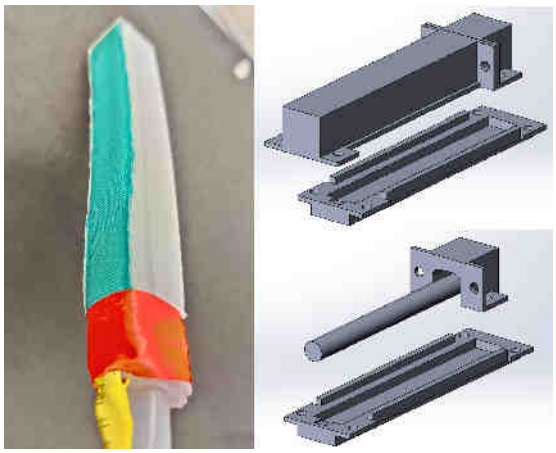

Fig. 3: Physical soft actuator (left) - CAD molds model (right)

A conventional fabrication process has been used by designing 3D printed PLA mould whose dimensions were chosen arbitrarily to accommodate resources availability and overall set-up design. The actuator was fabricated out of a common silicon material (Ecoflex-50, Smooth-on Inc.) based on two distinct layers dictated by the moulding process: a semi-cylindrical top layer containing a cylindrical actuation chamber of 9 mm diameter and a rectangular plate-shaped bottom layer. A strain limiting layer in form of a sheet of paper has been added below the latter in order to constrain the bending motion in one direction and prevent any extension of the actuator. A 4.5" commercial Flex sensor was placed on top of the lower layer, as indicated by the small cut-out extruded onto the mould during the curing process. This position far from the neutral axis of the actuator guarantees a large range of sensor readings. A small amount of silicone was poured between the two layers to take advantage of the inherent adhesion of silicone to bind them together. A view of the actuator along the molds is shown in Figure 3. Silicone tubing was inserted into the pressure chambers to ensure appropriate sealing. Special attention was paid to the sealing and removal of the actuator from the mould.

The embedded Flex sensor was a simple resistive strain sensor. The variable resistor was soldered on a protoboard to a voltage divider with a $20k\Omega$ resistor along a single-sided operational amplifier whose output voltage is used. The sensor electrical schematics is represented on Figure 4.

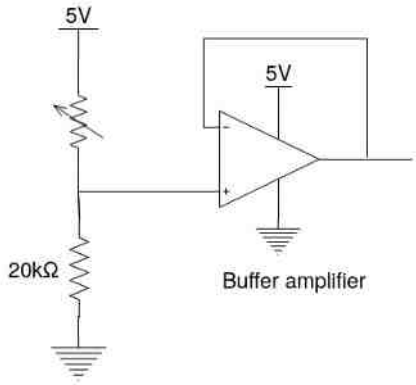

Fig. 4: Strain sensor circuit schematics.

B. Actuation and internal controller

The current work used an already existing pressure station present in the lab shown in Fig. 5 using a dedicated custom board driven by an ATmega328P chip. Air is injected into the chambers using a 12V diaphragm pump and 3/2-way valves whose frequency were fixed at 30Hz due to mechanical constraints.

An analog differential pressure transducer coupled with an instrumentation amplifier has been calibrated using a reference Bourdon tube pressure gauge as shown in Figure 6. Pressure is measured at a sample rate of 1kHz at the inlet of the actuator chambers. We assume pressure uniformity with respect to the actuator. The air flow was controlled via the PWM duty cycle of the valves which ranges from 0 to 70%.

A sample step response for a 35% duty cycle is shown on figure 7. This initial underdamped system showed a initial rising time of about 15 sec. To track the reference pressure at any given time and reduce the significant rise time, a proportional integral (PI) controller based on the pressure sensor signal and running at 50Hz has been tuned and implemented on an ATmega2560. The board communicates over two serial channels set-up to 115200 bit per seconds: one to receive

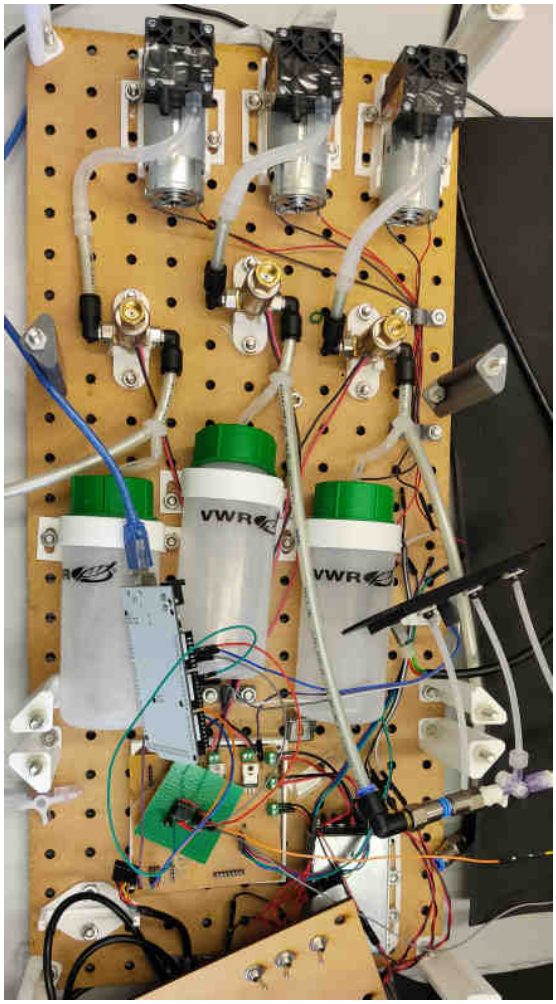

Fig. 5: Pressure station

reference pressure command from the computer and the other to send the adjusted PWM command to the pressure station.

The rise time have been reduced by 67% to 3 seconds thanks to the PI controller. The usual derivative term of the PID controller was left off due to mechanical constraints. The high frequency component of the signal caused the valves to respond abruptly due to the derivative action which could damage the system components. Despite the improvement, Figure 8 still shows a high frequency part of the signal coming from the valve oscillations, which is not negligible compared to the required accuracy. On the inspection of the power spectrum generated by the discrete Fourier transform of the signal as shown in Figure 9, we designed a first-order low-pass filter with a selected cut-off frequency ω_c at 100Hz whose transfer function is the following:

$$\begin{aligned} H(s) &= \frac{Y(s)}{X(s)} \\ &= \frac{\omega_0}{s+\omega_0} \end{aligned} \tag{1}$$

with s a complex variable.

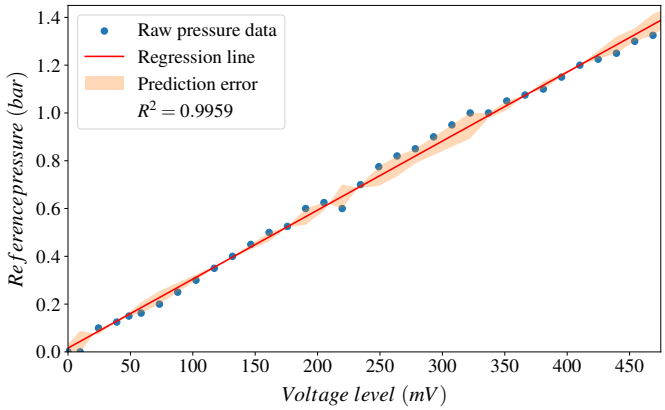

Fig. 6: Differential sensor calibration

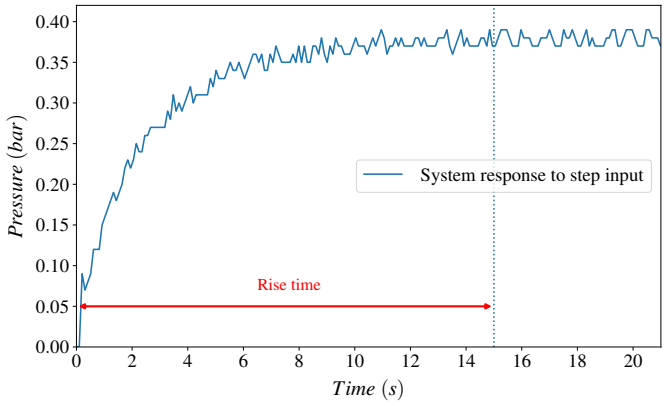

Fig. 7: System step response

For the hardware implementation, the discrete transfer function is computed using Tustin's approximation [6] which allows from a continuous to a discrete representation of the filter by injecting

$$s = \frac{2}{t}\frac{z-1}{z+1} \tag{2}$$

into the transfer function (1) where t is the sample time and z a complex variable:

$$\begin{aligned} H(z) &= \frac{\omega_0}{\frac{2}{t}\frac{z-1}{z+1} + \omega_0} \\ &= \frac{\omega_0 t(z+1)}{2(z-1) + \omega_0 t(z+1)} \\ &= \frac{\omega_0 t(z+1)}{(2+\omega_0 t)z + \omega_0 t - 2} \end{aligned} \tag{3}$$

The transfer function is made causal and its terms can be further reorganised in the following form:

$$H(z) = \frac{\frac{\omega_0 t}{2+\omega_0 t} + \frac{\omega_0 t}{2+\omega_0 t} z^{-1}}{1 + \frac{\omega_0 t - 2}{2+\omega_0 t} z^{-1}}$$

A linear difference equation for the digital filter can be obtained by applying the Z-transform on the transfer function:

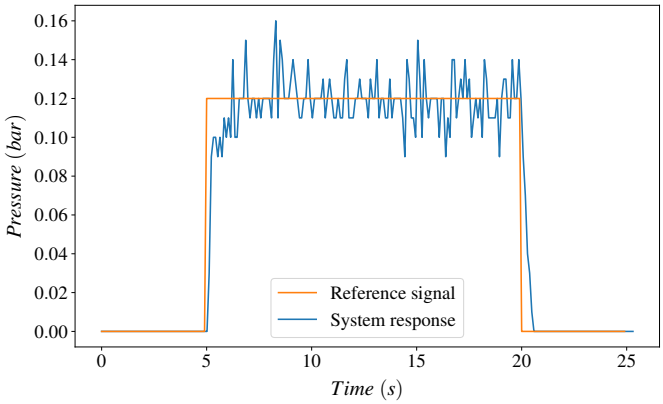

Fig. 8: PI controlled signal

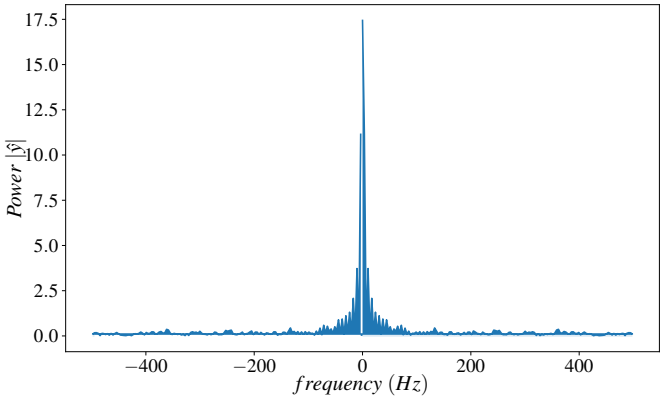

Fig. 9: Power spectrum

$$(1+\frac{\omega_0 t-2}{2+\omega_0 t})Y[z]=(\frac{\omega_0 t}{2+\omega_0 t}+\frac{\omega_0 t}{2+\omega_0 t})X[z]$$
$$Y[n]+\frac{\omega_0 t-2}{2+\omega_0 t}Y[n-1]=\frac{\omega_0 t}{2+\omega_0 t}X[n]+\frac{\omega_0 t}{2+\omega_0 t}X[n-1]$$
$$Y[n]=\frac{\omega_0 t}{2+\omega_0 t}X[n]+\frac{\omega_0 t}{2+\omega_0 t}X[n-1]-\frac{\omega_0 t-2}{2+\omega_0 t}Y[n-1] \quad (4)$$

The recursive equation is applied to the raw measurements before entering the internal controller. Figure 10 shows an example of the resulting system response.

C. Radius extraction algorithm

Compared to other approaches found in the literature we decided to track the radius of curvature itself instead of a reference point. In order to reconstruct the kinematic body of the actuator, we needed to take into account the longitudinal expansion resulting from the bending which directly affects the radius of curvature. We considered a constant curvature of the actuator, e.g. the radius is "constant in space but variable in time" [7].

We mounted the actuator with the tip facing backwards against gravity. The soft body was in front of a black background to facilitate the processing of the images. We used

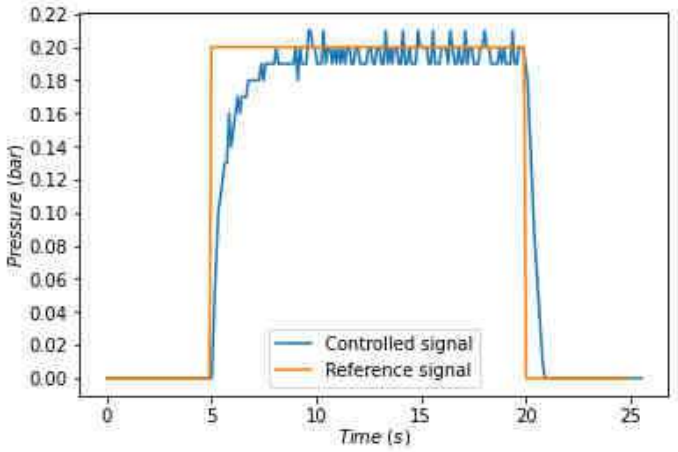

Fig. 10: Controlled system with first-order low-pass filter.

a C920 Pro HD 720p at 60 FPS for the recordings which was calibrated to our applications and placed in front of the actuator. The radius extraction algorithm as well as the overall data collection system was written in C++.

An image was captured from the video feed and undistorted according to the calibration. To reduce noise and unnecessary details, the image was blurred with a 7x7 kernel matrix and greyed out. We applied a binary threshold to the resulting frame followed by an erosion. The latter is a morphological operation which allows to shrink the shape contained in the image. The skeletonization of the shape was achieved by applying the Zhang-Suen thinning algorithm [8]. However, the processing reduced the video feed down to an average of 4 FPS, well below the 10 Hz requirements. We solved this issue by using a greater kernel with a matrix size of 31x31 in the erosion process, which reduced extensively the pixel number to be processed by the thinning algorithm. Overall, we achieved achieved a steady average of 18 FPS.

We retrieved the m 2D non zero points (x_i, y_i) for $i = 1, ..., m$ from the resulting binary frame that represents the skeleton of the actuator. Under the constant curvature assumption, this skeleton is the arc of a circle of radius R, whose equation is parameterized as follows:

$$R^2=(x-c_x)^2+(y-c_y)^2$$
$$R=\sqrt{(x-c_x)^2+(y-c_y)^2} \quad (5)$$

where $c=(c_x, c_y)$ is the center of the circle.

To fit the circle passing through the set of points, we minimize the squares of the distance between the data points and the fitting circle centered at c. We formulate our objective function as a unconstrained non-linear least square optimization problem:

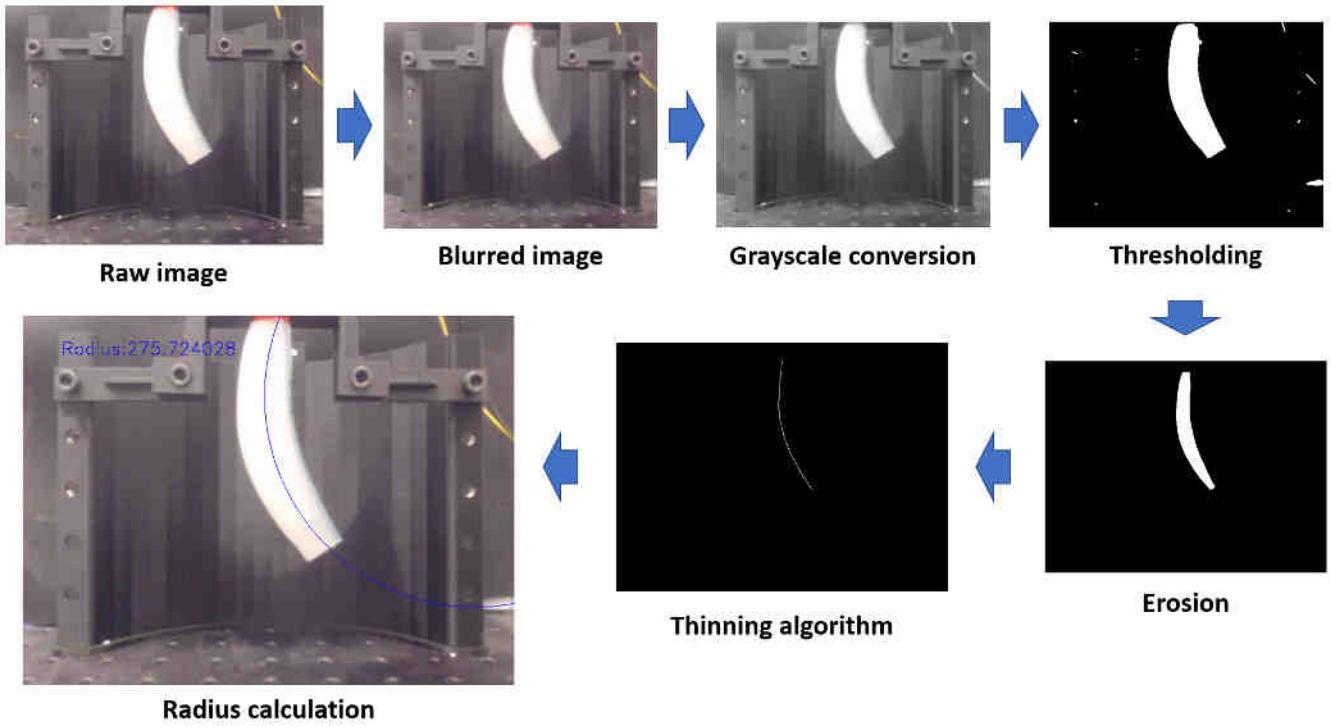

Fig. 11: Radius extraction process from raw images.

$$R_i = \sqrt{(x_i - c_x)^2 + (y_i - c_y)^2} \quad for \quad i = i, ..., m$$
$$minimize \quad \sum_{i=1}^{m} \left\| R_i - \hat{R} \right\|^2 \quad with \quad \hat{R} = \frac{1}{m} \sum_{i=1}^{m} R_i \tag{6}$$

We used Levenberg–Marquardt algorithm [9] for each frame to find the center $c = (c_x, c_y)$. The radius of the circle is computed by taking the mean of the radii of each data sample computed with Equation (5) and the found center. The initial guesses used were always the mean of the initial binary coordinates.

A depiction of the entire radius extraction procedure is shown in Figure 11. The process implementation revealed a frame rate lying in the range [11-15], which is the limiting time of the experimental set-up.

III. Results

In order to test the system and its capabilities, a pseudo-random sequence of pressure inputs was introduced into the system. This allows for the real-time retrieval of a massive amount of data, a sample of which is shown in figure 13. Each data sample is timestamped accordingly.

As shown in Figure 14, each sensor contains its own set of outliers that must be removed during data cleaning. We also notice the appearance of hysteresis effects. This shows the importance of multi-modality in such applications. Indeed, outliers are unavoidable due to environmental or unpredictable factors but hysteresis effects completely change the determination of the actuator state. Therefore, the perceptual model must account for these exceptions, hence the preference for a data-driven method.

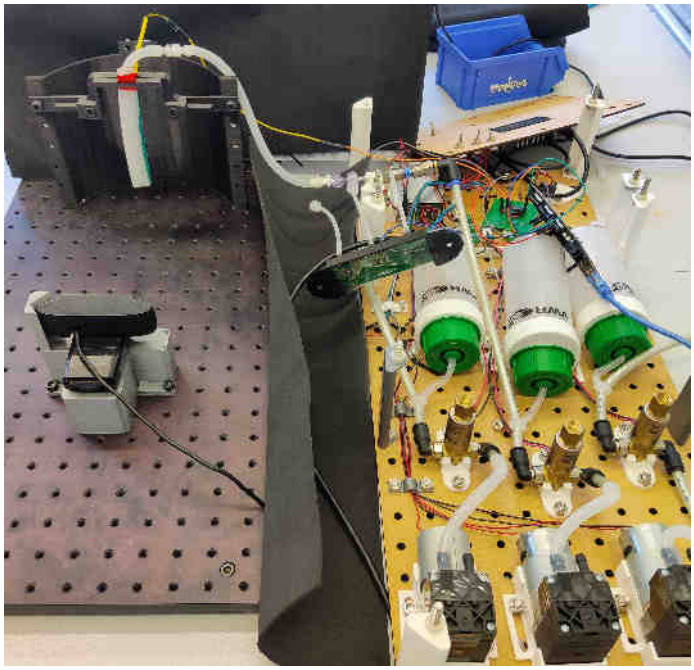

Fig. 12: Experimental set-up

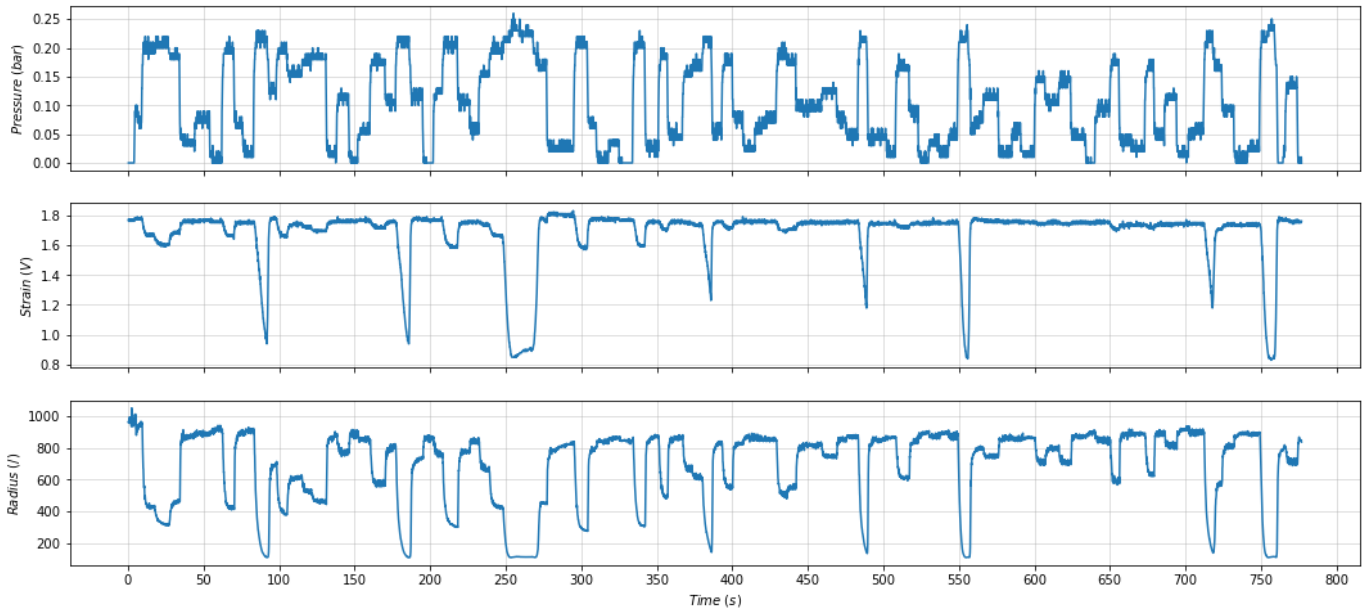

Fig. 13: Sample actuator pressure, strain and radius data over time.

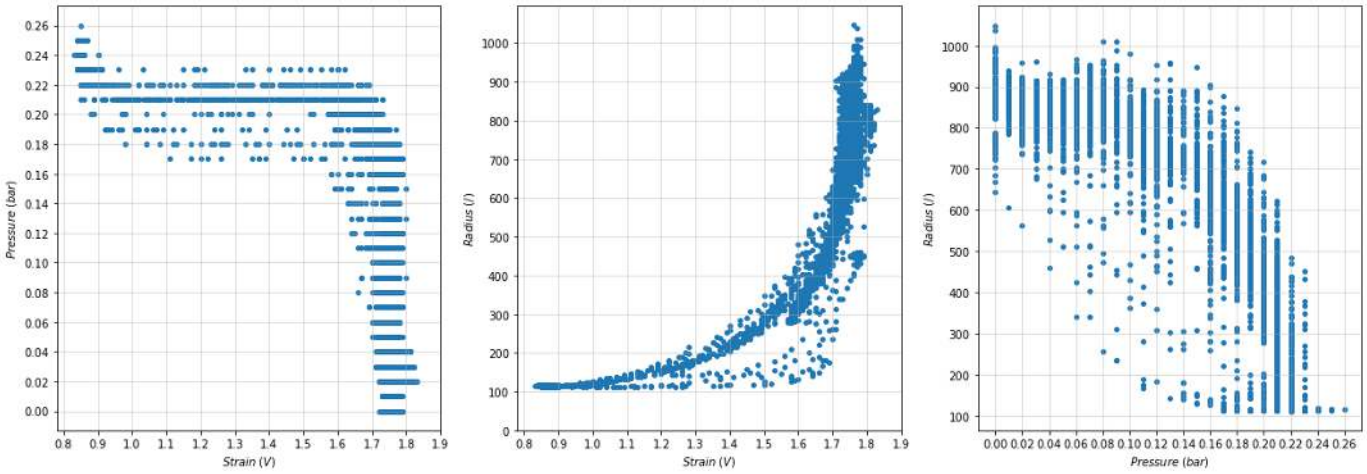

Fig. 14: Direct comparisons of modality samples.

It can be noticed that the different phenomena have non-linear relationships between them, as expected from the complexity of the phenomena. However, it should be noted that the data collected and the results obtained reflect only the soft actuator designed in this project. The realization of another actuator, even with the same characteristics, will generate new non-linear relations with the same data-driven method.

IV. Conclusion

Unlike their mechanical counterparts, soft robots offer the possibility to exploit the advantages of data-driven techniques instead of the usual theoretical modeling to model their inherently complex dynamics. The current setup demonstrated real-time data recording and processing capabilities. However, some limitations may have been introduced that, if removed, could have resulted in a more accurate system which includes, but not limited to:

- On one hand, the system would have performed better accuracy with a commercial motion capture system such as OptiTrack instead of handcrafting a recognition algorithm. On the other hand, such system is expensive and deriving useful experiments from a camera permits better replication among the scientific community.
- The pressure sensor resolution was limited to 0.1 bar, which was fairly high when considering to the perceived change of pressure.
- The Levenberg–Marquardt algorithm is an iterative algorithm converging only towards local minimum, which may or may not be a global minimum.

The developed system opens many possibilities for the next

phase of the work. One of them is the training of a predictor by an LSTM network, a type of artificial recurrent neural networks. Indeed, the temporal relationships between samples cannot be expressed by an ordinary neural network and will require a ”memory” since the state of the actuator at a given time depends on previous states. Such a predictor could be used as the basis for the estimation of the system state and constitute the internal model of the actuator.

REFERENCES

[1] G. Decroly, B. Mertens, P. Lambert, and A. Delchambre, “Design, characterization and optimization of a soft fluidic actuator for minimally invasive surgery,” *International Journal of Computer Assisted Radiology and Surgery*, vol. 15, no. 2, pp. 333–340, Feb. 2020. [Online]. Available: https://doi.org/10.1007/s11548-019-02081-2

[2] D. Trivedi, A. Lotfi, and C. D. Rahn, “Geometrically exact models for soft robotic manipulators,” *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 773–780, Aug. 2008.

[3] T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, “Soft robot perception using embedded soft sensors and recurrent neural networks,” *Science Robotics*, vol. 4, 2019.

[4] K. Elgeneidy, N. Lohse, and M. R. Jackson, “Data-driven bending angle prediction of soft pneumatic actuators with embedded flex sensors,” *IFAC-PapersOnLine*, vol. 49, pp. 513–520, 2016.

[5] D. P. Holland, C. Abah, M. Velasco Enriquez, M. Herman, G. J. Bennett, E. A. Vela, and C. J. Walsh, “The soft robotics toolkit: Strategies for overcoming obstacles to the wide dissemination of soft-robotic hardware,” *IEEE Robotics and Automation Magazine, Special Issue on Open Source and Widely Disseminated Robot Hardware*, vol. 24, no. 1, pp. 57–64, 2017 2017. [Online]. Available: http://dx.doi.org/10.1109/MRA.2016.2639067

[6] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 6th ed. Pearson, 2010.

[7] R. K. Katzschmann, C. D. Santina, Y. Toshimitsu, A. Bicchi, and D. Rus, “Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model,” pp. 454–461, 2019.

[8] T. Y. Zhang and C. Y. Suen, “A fast parallel algorithm for thinning digital patterns,” *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, Mar. 1984. [Online]. Available: https://doi.org/10.1145/357994.358023

[9] D. W. Marquardt, “An Algorithm for Least-Squares Estimation of Nonlinear Parameters,” *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, Jun. 1963. [Online]. Available: http://epubs.siam.org/doi/10.1137/0111030