Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 59-18 (2025) 145–150

Visual Servoing Control of a Tendon-driven Soft Robotic Neck [★]

Bastian Deutschmann [*] **Milan Akim** [*] **Cosimo Della Santina** [*,**]

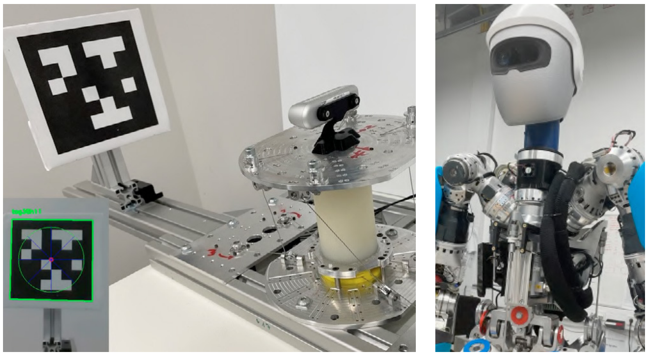
[*] *Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Wessling* `Bastian.Deutschmann@dlr.de`
[**] *Cognitive Robotics department, Delft University of Technology, Mekelweg 2, 2628 CD Delft, Netherlands* `C.DellaSantina @tudelft.nl`

Abstract: Generating precise motions with continuum soft robots calls for ways of closing the loop through nonconventional sensors - like cameras. This paper considers, for the first time, model-based visual servoing control of tendon-driven continuum soft robots. We take into account both regulation and trajectory tracking. We especially focus on a system inspired by the human neck, which employs an eye-in-hand configuration. The considered control architecture maps visual feedback to motor commands using the overall system's Jacobian, aiming for accurate end-effector (i.e., robot's head) positioning. This is made possible by blending classic results in visual serving with reduced-order models for continuum soft robots. In this work, we place significant focus on evaluating the controller's effectiveness on the physical prototype, for which we develop a tailored testing setup, which is a novel contribution of this work. Central to this setup is a tendon-driven soft robotic neck. We extensively characterize the control algorithm's performance for several control gains and different operational scenarios. We show that incorporating feedforward velocity estimation into the controller consistently improves performance in trajectory tracking tasks.

1. INTRODUCTION

Continuum soft robots (Hauser et al., 2023; Della Santina et al., 2021) are actuated mechanical systems comprised of inherently soft bodies that move by generating continuous deformations through distributed actuation - e.g., tendons pulled by electric motors and linked with kinematically designed connection nodes (Bai et al., 2024). Generating precise, stable, and effective movements with closed-loop algorithms in soft robots has been the focus of active research in the past years (Chen et al., 2024; Franco et al., 2023; Armanini et al., 2023; Della Santina et al., 2023; Grazioso et al., 2022; Alora et al., 2023; Ribeiro et al., 2025). Fundamental challenges in the modeling and controlling these systems are the unideal behavior of deformable materials, the theoretically infinite-dimensional nature, and the difficulties in directly measuring the systems' state.

With this work, we are interested in exploring the application of visual servoing (Chaumette et al., 2016) to controlling soft robots. This class of techniques, aiming to regulate states defined in camera images, has achieved significant success in controlling rigid robots (D'Avella et al., 2023; Cong et al., 2023). Visual serving is a promising avenue in soft robotics control as it leverages rich sensory feedback, potentially compensating for lacks in the model while not requiring collocated sensors. The large majority of works, however, look at the strategy in a model-free fashion, learning the model from data (Wang et al., 2016; Xu et al., 2019; Fang et al., 2019) or directly the end-to-end controller (Gao et al., 2023). The model-based implementation has received much less attention. In (Zhang et al., 2017), authors implement and experimentally validate a model-based visual servoing algorithm based on a linearized FEM model. An interesting variation is (Xu et al., 2021). Here, the camera points toward the robot instead of being on board. Apart from the limited exploration of model-based strategies, none of these works targeted trajectory tracking in the visual servoing context. Finally, experimental validations have so far been limited, and mostly focused on easier-to-control small scale systems.



(a) Setup

(b) Robot integration

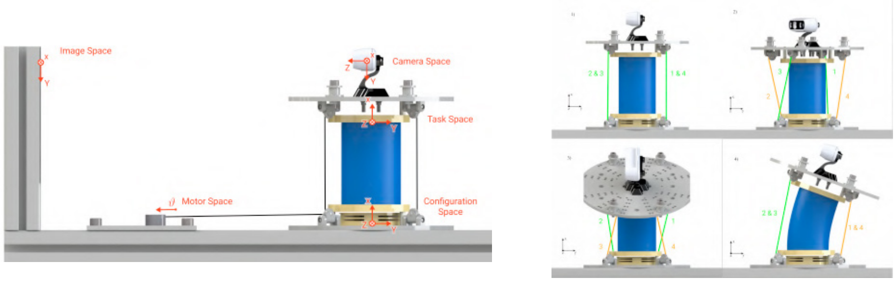
Fig. 1. This work addresses closed-loop control from image to motors, applied to a soft robotic neck. Panel (a) shows the experimental setup: the tendon-driven neck with a top-mounted RGB camera, facing a fixed AprilTag target whose position is varied to test regulation and tracking. The camera view appears at the bottom left. Panel (b) shows the intended application, with a blue version of the neck integrated into the humanoid torso David.

This work aims to tackle these open issues by proposing and extensively validating a fully nonlinear model-based visual servoing controller capable of regulation and tracking. We are motivated to pursue this direction by an application in which visual serving naturally appears as a natural solution: controlling a tendon-driven soft robotic neck (Fig. 1). The nature of this paper is markedly experimental - as it looks not just at developing a control algorithm but also at introducing

---

[★] This work has received funding from the European Research Council (ERC) under the European Union's Horizon Europe research and innovation programme (Grant Agreement No. 101165078, project RIPLEy), and from the German Research Foundation (DFG) within the Priority Programme "Soft Material Robotics System" (Grant No. 405032572).

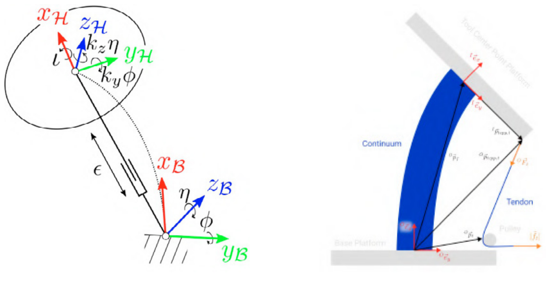
10.1016/j.ifacol.2025.10.211



(a) Lateral view

(b) Possible motions

Fig. 2. Panel (a) shows a side view of the soft-neck system with coordinate frames. Panel (b) illustrates the independent motions from pulling one set of tendons at a time: (1) x-translation via all tendons, (2) x-rotation, (3) y-rotation, (4) z-rotation. The robot can both pend and twist, making it a good setup to test advanced control algorithms.



(a)

(b)

Fig. 3. Panel (a) shows the kinematic model used for this experimental investigation, which reduces the kinematics to three rotational and one translational DoF (Deutschmann, 2020). Panel (b) illustrates how tendon routing determines the robot configuration under full actuation hypotheses.

Table 1. Symbolic Abbreviation and Meaning of each Subspace used in the derivation of the Visual Servoing Controller

Symbol	Description
$\mathcal{S}$	**Image Space**: coordinates of a set of keypoints in camera system, serving as the feedback for visual servoing.
$\mathcal{C}$	**Camera Space**: coordinate system associated with the camera, where image processing and feature detection occur.
$\mathcal{I}$	**Task Space**: coordinate system of the robot end effect, described in terms of positions and orientations.
$\mathcal{O}$	**Configuration Space**: represents the principal steady-state deformation modes of the soft robot.
$\mathcal{M}$	**Motor Space**: configuration of the actuators driving the robot via tendons.

a novel experimental platform and performing an extensive characterization of performance. For this reason, a good portion of the paper is devoted to Sec. 4, where the novel experimental platform is detailed, together with extensive validation of the control architecture in regulation and trajectory tracking.

2. MOTORS-TO-IMAGE KINEMATIC MODEL

We introduce a kinematic model for a tendon-driven soft robot with a camera, merging quasi-static modeling of soft robots and classic visual servoing theory from rigid robots. Our focus is on the robot in Fig. 1, used as our reference. Fig. 2(a) illustrates two additional coordinate systems for the soft neck's model. The derivations apply broadly to robots of this class, with Table 1 listing symbolic abbreviations for each subspace or coordinate frame.

2.1 Coordinates

Kinematics of the continuum soft robot Several kinematic models for soft robots have been introduced in the literature in the past decade (Della Santina et al., 2023). The goal is always to parameterized the overall shape of the theoretically infinite soft robots through a reduced and finite set of coordinates $\boldsymbol{q} \in \mathbb{R}^n$. In this work, we consider the the inverted pendulum model introduced by (Deutschmann, 2020). The idea here is not to focus on the soft material deformations but on the overall end-to-end geometric behavior. This model consists of four independent kinematic variables

$$\boldsymbol{q} = (\,\epsilon\;\iota\;\phi\;\eta\,) \in \mathbb{R}^4. \tag{1}$$

The kinematic variables are depicted in Fig. 3(a) which illustrates that the three variables ι, ϕ, η describe a rotation and ϵ describes a translation.

As concisely described byFig. 3(a), the pose of the terminal point of the soft robot expressed w.r.t to the inertial frame ${}^{O}\boldsymbol{T}_{\mathcal{I}}$ can be defined by subsequent composition of homogenous transformations

$${}^{O}\boldsymbol{T}_{\mathcal{I}} = {}^{O}\boldsymbol{T}_{\mathrm{r,y}}(\phi)\,{}^{O}\boldsymbol{T}_{\mathrm{r,z}}(\eta)\,{}^{O}\boldsymbol{T}_{\mathrm{t,x}}(\epsilon)\,{}^{O}\boldsymbol{T}_{\mathrm{r,y}}(\phi)\,{}^{O}\boldsymbol{T}(\iota) \tag{2}$$

$${}^{O}\boldsymbol{T}_{\mathcal{I}}(\epsilon,\iota,\phi,\eta) = \begin{pmatrix} {}^{O}\boldsymbol{R}_{\mathcal{I}} & {}^{O}\boldsymbol{p}_{\mathcal{I}} \\ 0 & 1 \end{pmatrix} \in \mathbb{R}^{4\times 4}, \tag{3}$$

where, $\boldsymbol{T}_{\mathrm{r,y}}$ is a rotation around the local y axis, $\boldsymbol{T}_{\mathrm{r,z}}$ is a rotation around the local z axis, and $\boldsymbol{T}_{\mathrm{t,x}}$ is a rotation around the local z axis.

Kinematics in the image space The position of point i in the image space is defined as $\tilde{\boldsymbol{s}}_i = (x_i, y_i) \in \mathbb{R}^2$. The positions of k points in the image space is defined as $\boldsymbol{s} = [\tilde{\boldsymbol{s}}_1, \ldots, \tilde{\boldsymbol{s}}_k] \in \mathbb{R}^{2k}$ with ${}^{C}\boldsymbol{R}_{O} \in \mathbb{R}^{3\times 3}$ being the rotational matrix between the inertial and the camera frame and ${}^{C}\boldsymbol{p}_{CO} \in \mathbb{R}^3$ being the position of the inertial frame in the camera frame.

2.2 Differential Kinematics

The Jacobian $\boldsymbol{J} \in \mathbb{R}^{2k\times r}$ describes the relation between the time derivative of the r tendon lengths ($\dot{\boldsymbol{\theta}}$) and the velocity of the marker points in the image space ($\dot{\boldsymbol{s}} \in \mathbb{R}^{2k}$),

$$\dot{\boldsymbol{s}} = \boldsymbol{J}_{\mathcal{SM}}\,\dot{\boldsymbol{\theta}}, \tag{4}$$

$$\boldsymbol{J}_{\mathcal{SM}} = {}^{C}\boldsymbol{L}_{\mathcal{SC}}\,{}^{C}\boldsymbol{Ad}_{O}\,{}^{O}\boldsymbol{J}_{\mathcal{IM}} \tag{5}$$

with the interaction matrix ${}^{C}\boldsymbol{L}_{\mathcal{SC}} \in \mathbb{R}^{2k\times 6}$, the tendon coupling matrix ${}^{O}\boldsymbol{J}_{\mathcal{IM}} \in \mathbb{R}^{6\times r}$ and the adjoint matrix ${}^{C}\boldsymbol{Ad}_{O} \in \mathbb{R}^{6\times 6}$ projecting motions from the inertial frame O to the camera frame C,

$${}^{C}\boldsymbol{Ad}_{O} = \begin{pmatrix} {}^{C}\boldsymbol{R}_{O} & {}^{C}\boldsymbol{p}_{CO}{}^{C}\boldsymbol{R}_{O} \\ 0 & {}^{C}\boldsymbol{R}_{O} \end{pmatrix}. \tag{6}$$

We now derive in the rest of the section the matrices necessary to completely specify (4).

Interaction Matrix Coherently with the visual servoing literature (Chaumette et al., 2016), we define the interaction matrix ${}^{C}\boldsymbol{L}_{\mathcal{SC}} \in \mathbb{R}^{2k\times 6}$ describes the relation between the time derivative

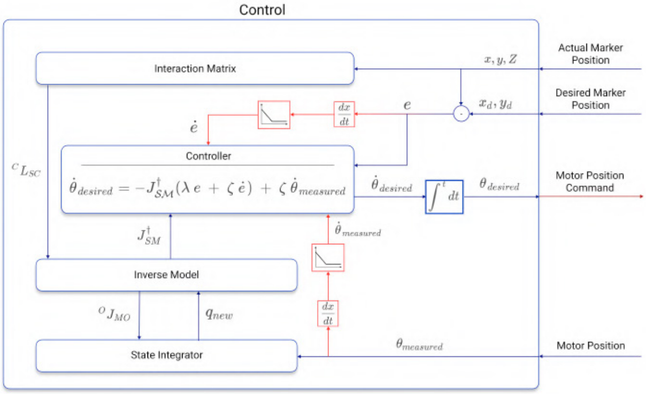


Fig. 4. Block diagram of the implemented image-based visual servoing controller. The input to the controller are the measured motor position, the desired marker- and the actual marker-position(s). The marker position is used for calculating the interacting matrix as well as the control error. The state $\boldsymbol{q}, \dot{\boldsymbol{q}}$ of the system are computed by the state estimator using the inverse model of the kinematic equations. Finally, the controller computes the desired motor velocity after integration with respect to time sent as the control command to the motors. The controller dedicated to setpoint regulation, the parameter ζ is set to 0.

of k marker points $\dot{\boldsymbol{s}} \in \mathbb{R}^{2k}$ in the image space and the generalized body velocities in the camera space ${}^{C}\boldsymbol{V} \in \mathbb{R}^6$,

$$\dot{\boldsymbol{s}} = {}^{C}\boldsymbol{L}_{SC}\,{}^{C}\boldsymbol{V}. \tag{7}$$

with ${}^{C}\boldsymbol{L}_{SC}$ being the interaction matrix for k points in the image space

$$^{C}\boldsymbol{L}_{SC} = \begin{pmatrix} {}^{C}\tilde{\boldsymbol{L}}_{SC,1} \\ {}^{C}\tilde{\boldsymbol{L}}_{SC,2} \\ \vdots \\ {}^{C}\tilde{\boldsymbol{L}}_{SC,k} \end{pmatrix} \in \mathbb{R}^{2k\times 6} \tag{8}$$

and ${}^{C}\tilde{\boldsymbol{L}}_{SC,i}$ being the interaction matrix for point i

$$^{C}\tilde{\boldsymbol{L}}_{SC,i} = \begin{pmatrix} \frac{-1}{Z_i} & 0 & \frac{x_i}{Z_i} & x_i y_i & -(1+x_i^2) & y_i \\ 0 & \frac{-1}{Z_i} & \frac{y_i}{Z_i} & 1+y_i^2 & -x_i y_i & -x_i \end{pmatrix} \tag{9}$$

with $Z_i \in \mathbb{R}$ being the current depth of the point i which has to be measured or estimated. In case the Z_i cannot be measured or estimated usually (Chaumette et al., 2016) the desired position $\boldsymbol{s}_{d,i}$ and desired depth $Z_{d,i}$ of the point i which yields that the interaction matrix becomes constant ${}^{C}\tilde{\boldsymbol{L}}_{SC,i} = {}^{C}\tilde{\boldsymbol{L}}_{SC,di}$.

Kinematic model of the tendon coupling In this paragraph, we seek to derive the Jacobian ${}^{O}\boldsymbol{J}_{\mathcal{IM}}$ that describes the relation between the generalized linear and angular body velocities given in the inertial frame ${}^{O}\boldsymbol{V} \in \mathbb{R}^6$ and the r time derivative of the tendon lengths in the motor space $\dot{\boldsymbol{\theta}} \in \mathbb{R}^r$,

$$^{O}\boldsymbol{V} = {}^{O}\boldsymbol{J}_{\mathcal{IM}}\dot{\boldsymbol{\theta}}, \tag{10}$$

with the vector of the time derivatives of each tendon length,

$$\dot{\boldsymbol{\theta}} = \begin{pmatrix} \dot{\theta}_1 & \dot{\theta}_2 & s & \dot{\theta}_r \end{pmatrix}^T. \tag{11}$$

For tendon driven systems, the relationship between the time derivative of the tendon length $\boldsymbol{l}(\boldsymbol{q})_t \in \mathbb{R}^r$, called tendon velocities, and the generalized coordinates $\boldsymbol{q}^n$ of the system is established by the tendon coupling matrix $\boldsymbol{J}_{\mathcal{M},O} \in \mathbb{R}^{r\times n}$,

$$\dot{\boldsymbol{l}}(\boldsymbol{q})_t = \dot{\boldsymbol{\theta}} = \boldsymbol{J}_{\mathcal{M},O}\,\dot{\boldsymbol{q}} \tag{12}$$

In order to compute equation (10), the representation Jacobian ${}^{O}\boldsymbol{J}_{\mathcal{I}O} \in \mathbb{R}^{6\times n}$,

$$^{\mathcal{I}}\boldsymbol{V} = {}^{O}\boldsymbol{J}_{\mathcal{I}O}\,\dot{\boldsymbol{q}}, \tag{13}$$

and the inversion of the tendon coupling matrix $\boldsymbol{J}_{\mathcal{M},O}$ is required,

$$^{O}\boldsymbol{V} = {}^{O}\boldsymbol{J}_{\mathcal{I}O}(\boldsymbol{J}_{\mathcal{M},O})^{-1}\dot{\boldsymbol{\theta}}, \tag{14}$$

which is not generally possible. (may be introduce cases.)

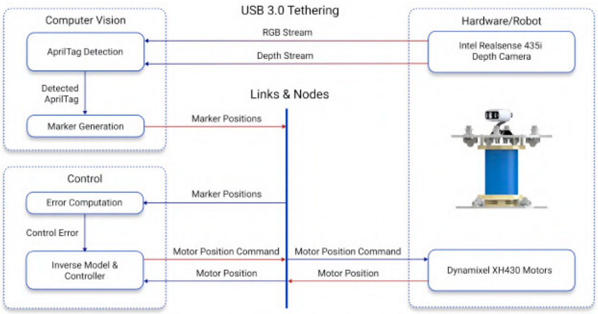


Fig. 5. Overview of the Data Flow within the System, with the relevant hardware and software modules highlighted.

3. IMAGE-BASED VISUAL SERVOING CONTROL

We introduce here the control architecture implementing the closed loop visual serving. As for the modeling part, for the sake of clarity and conciseness, the algorithm is per se general, but we discuss mostly its application to the discussed system. A block diagram of the control structure is depicted in Fig. 4. The goal is to minimize the error in the image space, $\boldsymbol{e} = \boldsymbol{s} - \boldsymbol{s}_d \in \mathbb{R}^{2k}$. A control law achieving this goal can be immediately obtained using the closed loop inverse kinematics (CLIK) framework as follows

$$\dot{\boldsymbol{\theta}}_{\mathrm{d}} = -\boldsymbol{\lambda}\,\boldsymbol{J}^{\dagger}_{S\mathcal{M}}\,\boldsymbol{e} + \boldsymbol{J}^{\dagger}_{S\mathcal{M}}\,\dot{\boldsymbol{s}}_{\mathrm{d}}, \tag{15}$$

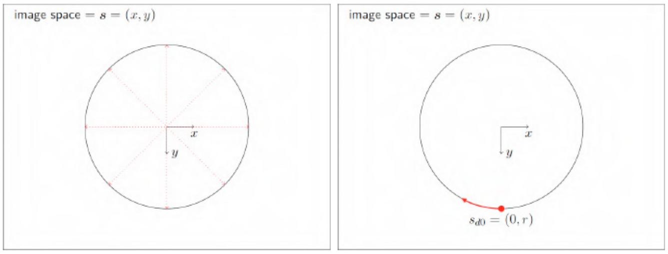
which outputs the desired motor velocities $\dot{\boldsymbol{\theta}}_{\mathrm{d}} \in \mathbb{R}^r$ to minimize the control error. The asymptotic stability towards the zero error equilibrium of the closed-loop system can immediately be proven by plugging (15) into (4), generating an exponentially stable closed-loop error dynamics, $\dot{\boldsymbol{e}} = -\boldsymbol{\lambda}\,\boldsymbol{e}$, with the diagonal, positive definite matrix of controller gains $\boldsymbol{\lambda} \in \mathbb{R}^{2k\times 2k}$. Note that $\dot{\boldsymbol{s}}_{\mathrm{d}}$ may not be available at run time and may need to be estimated. The inverse Jacobian of $\boldsymbol{J}_{\mathcal{M}O}$ from equation (14) is utilized for estimating the configuration $\boldsymbol{q}$ in real time and complements the state of the system by numerical differentiation to $\dot{\boldsymbol{q}}$. As part of that, for every iteration the measured motor position $\boldsymbol{\theta}_{\mathrm{measured}}$ is mapped into the configuration space by being multiplied with $\boldsymbol{J}^{-1}_{\mathcal{M}O}$.

4. EXPERIMENTS

This section details the development of the experimental platform, the experimental conditions, and the experimental results. It is worth stressing that all these results are from real-world experiments. We use the term *simulation* to refer to the execution of the control loop (15).

4.1 Experimental platform

We provide here more details on the hardware platform already preliminarily introduced in Sec. 1 and Fig. 1, and the associated



(a) Steps (b) Tracking

Fig. 6. Pictorial representation of the generation of the references in image space. For the step response, we generate sequential steps going from the center of the picture to the eight points shown in the left panel. For tracking, we generate a reference evolution smoothly traversing the circle counter-clock-wise.

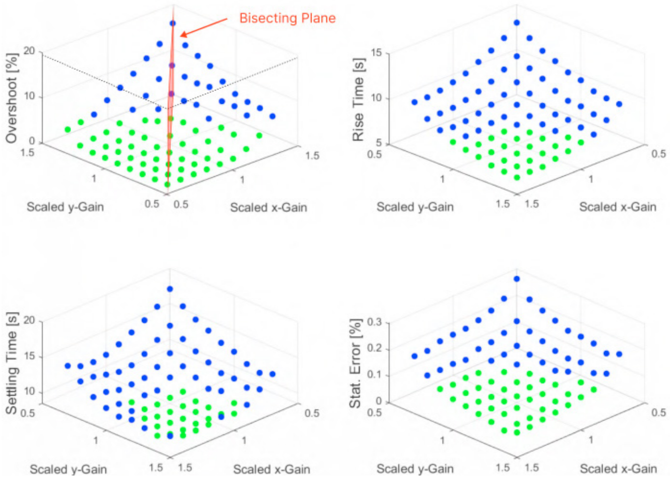


Fig. 7. The average step response performance in terms of the overshoot, rise time, settling time, and stationary error plotted over the scaled x and y gains. The green dots indicate KPIs being located in the satisfactory value range from Tab. 3.

software architecture with a particular focus on the control and the computer vision node.

The soft neck is an extension on the open source continuum soft robot platform introduced in (Deutschmann et al., 2022). The system is actuated by four tendons configured to allow for three independent rotations, see Fig. 2(b). The tendon tensions are generated by four position controlled Dynamixel XH430 motors which serve as tendon actuators. The hardware component of the experimental platform is then completed by the inclusion of an RGB camera, a camera holder, and a frame that rigidly connects the base of the neck to the target. We decided to keep the target still during experiments and move the reference position of the target in the frame instead of moving the target. From the perspective of the controller, the two are the same. However, from an experimental point of view, proceeding this way allows us to have repeatable conditions necessary to provide meaningful comparisons.

The camera mounted on the upper platform of the tendon-driven continuum soft robot is an Intel Realsense D435i. This is an RGB-D camera, but in thiswork we only acquire the RGB data stream and disregard the depth data. To decouple the control strategy evaluation from possible failures in off-the-shelf tracking algorithms, we decided to include an AprilTag as the target. Its contours are extracted from the image data using a feature detection model from `OpenCV`. An example is depicted in the bottom left of Fig. 1 where a tracked marker, highlighted in magenta, is extracted from the center of the AprilTag detection. After normalizing the image coordinates corresponding to the camera model, the computer vision node forwards the marker positions to the control node, using a POSIX-based ad hoc communication architecture `links and nodes`. The control node implemented in `Simulink` computes the desired motor velocities using the control law (15). After temporal integration of the desired motor velocities, the control node sends the desired motor positions to the actuators using links and nodes. Figure 5 shows the overall structure with its corresponding nodes and data flows. The detection model as well as the OpenCV library in general operate in pixel coordinates, while the controller does so in normalized coordinates, so we normalize the generated markers using an affine transformation such that $(0, 0)$ is the center of the frame, and ±0.5 are the two extremes.

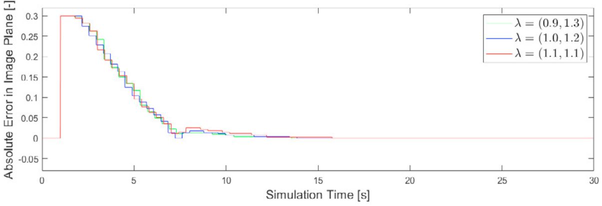


Fig. 8. Step responses from three feasible gain combinations, when performing a diagonal step from $s_d = (-0.15/\sqrt{2}, -0.15/\sqrt{2})$ to $s_d = (0.15/\sqrt{2}, 0.15/\sqrt{2})$.

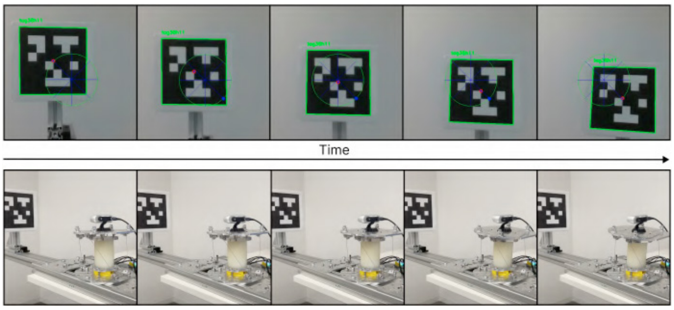
Table 2. Experimental Conditions and definition of relevant quantities. With Sim/real time ratio, we refer to the amount of times the internal control loop is executed for each interval within which the control action is changed.

Symbol	Parameter	Value	Unit
-	Tracked markers	1	[-]
-	Gains	2	[-]
Z_0	Init. marker distance	0.4	[m]
$\boldsymbol{q}_0$	Init. config.	$(0, 0, 0, 0)$	[m, °, °, °]
$\boldsymbol{s}_{d0}$	Desired marker state	$(0, 0)$	[-, -]
$\boldsymbol{s}_0$	Actual marker state	$(0, 0)$	[-, -]
A_{step}	Step amplitude	0.3	[-]
T_s	Sample time	0.001	[s]
i_{pre}	Pretension current	50	[mA]
$r_{\text{sim/real}}$	Sim/real time ratio	10	[-]

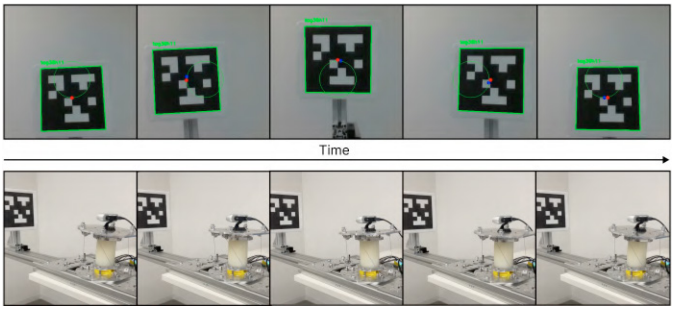
Table 3. Summary of threshold for KPIs. The mean, median, and maximum of the tracking error across the whole trajectory are chosen as performance measures for trajectory experiments. They are dimensionless measures since the image space is normalized.

Parameter	Value Range	Unit
Overshoot	< 4	[%]
Rise Time	< 0.7	[s]
Settling Time	< 0.9	[s]
Stationary Error	< 0.1	[%]
Mean Tracking Error	< 0.04	[-]
Median Tracking Error	< 0.035	[-]
Maximum Tracking Error	< 0.05	[-]

To perform reproducible experiments, the tendon-driven continuum soft robot is set to $\boldsymbol{q}_0$ and an AprilTag is placed at a distance of 40cm from the camera in such a way that its center is located at the center of the image space $\boldsymbol{s}_0 = (0, 0)$.



(a) Step response



(b) Trajectory tracking

Fig. 9. Exemplary image sequence of the soft neck controlled with the proposed architecture. Panel (a) shows the response to a step input moving the target from top left to bottom right, using gains $\boldsymbol{\lambda} = (1.1, 1.1)$. Panel (b) shows frames from a trajectory tracking experiment under harmonic excitation with the same gains.

4.2 *Initial conditions*

At the beginning of the experiment, the desired position of the target is ($\boldsymbol{s}_{\mathrm{d}} = \boldsymbol{s}_0 = (0, 0)$). This is the configuration the neck will see if it looks straight ahead, i.e., at configuration $\boldsymbol{q}_0$. In order to ensure tension in the tendons, a pretension current on each motor turned out to be sufficient. A summary of the experimental conditions can be obtained from table 2.

4.3 *Results: extensive setpoint control evaluation*

We are interested in evaluating the effect of gain tuning on the step response performance. For each value of the gain, we perform eight step-responses, as summarized in Fig. 6. Results are succinctly summarized in Fig. 7 in terms of overshoot, rise time, settling time, and static error. We represent with different colors the area that fulfill the key performance indicators (KPIs) reported in Tab. 3. Note that we oriented the x and y axis so as to have the region of feasibility on the same side. This is important to stress because - as expected - overshoot has better performance the lower are the gains, while the vice-versa is true for the others.

4.4 *Results: a closer look to setpoint regulation*

We report in Fig. 8 the evolution in time for setpoint regulation when considering gains that fulfill all the KPIs. This is the case for the following gain combinations: $\boldsymbol{\lambda}_1 = (0.9, 1.3)$, $\boldsymbol{\lambda}_2 = (1.0, 1.2)$, $\boldsymbol{\lambda}_3 = (1.1, 1.1)$. All evolutions are comparable. Fig. 9(a) also reports the evolutions corresponding to $\boldsymbol{\lambda}_3$ looked from the image frame, when going from the two extremes of the circle (top left to bottom right).

4.5 *Results: comparison transpose vs. inverse*

We consider here the following variant of (15)

$$\dot{\boldsymbol{\theta}}_{\mathrm{d}} = -\boldsymbol{\lambda}\, \boldsymbol{J}_{SM}^{\top}\, \boldsymbol{e}, \tag{16}$$

where we used the transpose instead of ($\dot{\boldsymbol{s}}_{\mathrm{d}} = 0$ in this example) . This choice will also result in asymptotic stability, but the proof has to follow Lyapunov's arguments as the closed loop dynamics is not linear this time. We omit the derivations. We repeat the complete experimental campaign discussed in the previous subsection by using this variant of the control law. The results are shown in Fig. 10. Interestingly, the two controllers perform quite similarly.

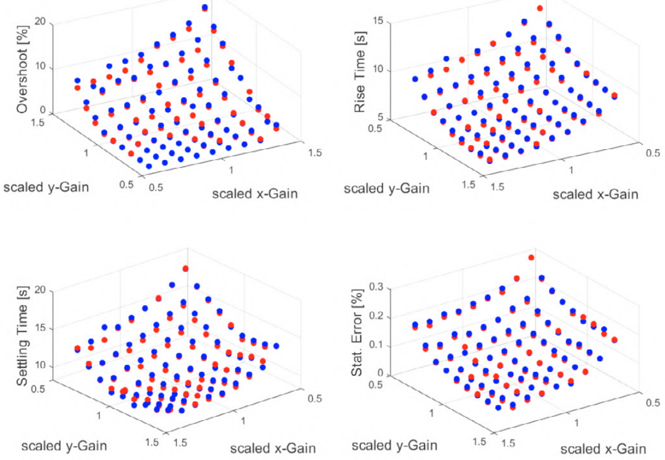


Fig. 10. Comparison between the performance of (15) and its Jacobian-transpose variant during step responses. Plots show average overshoot, rise time, settling time, and steady-state error over scaled x and y gains.

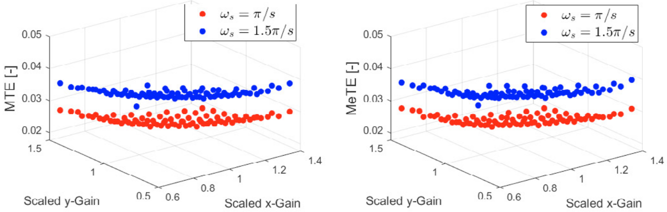


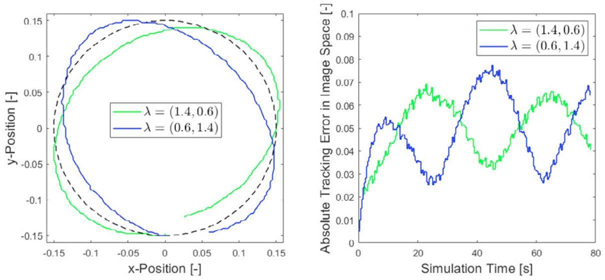
Fig. 11. The trajectory tracking experiment is conducted for different angular velocities of the desired marker and various choices of the control gains.

4.6 *Results: trajectory tracking*

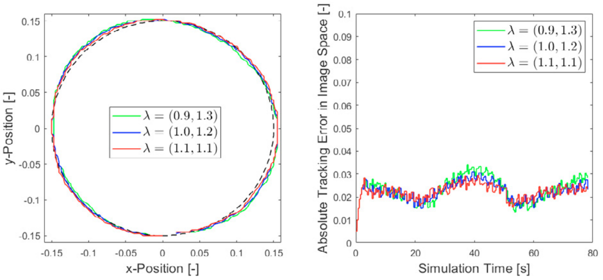
Fig. 6b reports a pictorial representation of the reference used in the trajectory tracking experiments. Two different angular velocities are considered πrad/s and $3\pi/2$rad/s. As per the step response, we perform extensive validation across several choices of gains. The results are reported in Fig. 11. Interestingly, the variation of performance due to gain is less evident in this plot compared to the state response one as the effect of chance of velocity is dominant.

4.7 *Results: a closer look to tracking*

We consider the gains: $\boldsymbol{\lambda}_1 = (0.9, 1.3)$, $\boldsymbol{\lambda}_2 = (1.0, 1.2)$, $\boldsymbol{\lambda}_3 = (1.1, 1.1)$. The performance measures obtained from those gains are shown in table 4. We report part of the evolutions as seen by the camera in Fig. 9(b). Finally, in Fig. 12, we show the performance with and without feedforward compensation of the target velocity, underscoring the importance of this term in the control law.



(a) Without feedforward compensation



(b) With feedforward compensation

Fig. 12. Experimental results in trajectory tracking in image space, for several choices of the gains, with and without feedforward compensation of the estimated velocity of the target.

Table 4. Performance Measures Resulting from Feasible Gains.

Abbreviations: Ov - Overshoot, RT - Rise Time, ST - Settling Time, SE - Steady-State Error, mte - Maximum Tracking Error, mete - Maximum Endpoint Tracking Error, mate - Maximum Absolute Tracking Error.

Gains	Ov [%]	RT [s]	ST [s]	SE [%]	mte	mete	mate
λ_1	2.34	6.88	8.58	0.074	0.023	0.023	0.034
λ_2	3.04	6.77	8.89	0.075	0.022	0.022	0.031
λ_3	3.50	6.73	8.89	0.066	0.022	0.023	0.030

5. CONCLUSIONS

With this work, we developed and tested a model-based visual servoing controller for regulation and tracking with a tendon-driven soft neck—the first application of this strategy to tendon-driven soft robots. Future work will focus on integrating the neck into the humanoid torso (Fig. 1) and designing a controller that compensates for motion-induced perturbations. The current approach relies on a simplifying full actuation assumption, thus allowing in Sec. 2 for a change of coordinates from actuation to configuration space (Pustina et al., 2024). We aim to extend it by combining visual servoing with CLIK algorithms that handle unactuation (Della Santina, 2025).

REFERENCES

Alora, J.I. et al. (2023). Robust nonlinear reduced-order model predictive control. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, 4798–4805. IEEE.

Armanini, C. et al. (2023). Soft robots modeling: A structured overview. *IEEE Transactions on Robotics*.

Bai, H. et al. (2024). Unlocking the potential of cable-driven continuum robots: A comprehensive review and future directions. In *Actuators*, volume 13, 52. MDPI.

Chaumette, F. et al. (2016). *Visual Servoing*, 841–866. Springer International Publishing, Cham.

Chen, Z. et al. (2024). A novel and accurate bilstm configuration controller for modular soft robots with module number adaptability. *arXiv preprint arXiv:2401.10997*.

Cong, V.D. et al. (2023). A review and performance comparison of visual servoing controls. *International Journal of Intelligent Robotics and Applications*, 7(1), 65–90.

Della Santina, C. (2025). Pushing the boundaries of actuators-to-task kinematic inversion: From fully actuated to underactuated (soft) robots. doi: 10.36227/techrxiv.173738377.73804320/v1. Preprint.

Della Santina, C. et al. (2021). Soft robots. *Encyclopedia of Robotics*, 489.

Della Santina, C. et al. (2023). Model-based control of soft robots: A survey of the state of the art and open challenges. *IEEE Control Systems Magazine*, 43(3), 30–65.

Deutschmann, B. (2020). *Modeling and Control for a Class of Tendon-Driven Continuum Mechanisms*. Ph.D. thesis.

Deutschmann, B. et al. (2022). Open source tendon-driven continuum mechanism: A platform for research in soft robotics. In *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, 54–61.

D'Avella, S. et al. (2023). Ros-industrial based robotic cell for industry 4.0: Eye-in-hand stereo camera and visual servoing for flexible, fast, and accurate picking and hooking in the production line. *Robotics and Computer-Integrated Manufacturing*, 80, 102453.

Fang, G. et al. (2019). Vision-based online learning kinematic control for soft robots using local gaussian process regression. *IEEE Robotics and Automation Letters*, 4(2), 1194–1201.

Franco, E. et al. (2023). Energy shaping control of a class of underactuated mechanical systems with high-order actuator dynamics. *European Journal of Control*, 72, 100828.

Gao, J. et al. (2023). Learning end-to-end visual servoing using an improved soft actor-critic approach with centralized novelty measurement. *IEEE Transactions on Instrumentation and Measurement*.

Grazioso, S. et al. (2022). Soft growing robot to enable monitoring applications in remote constrained environments. In *2022 6th International Symposium on Instrumentation Systems, Circuits and Transducers (INSCIT)*, 1–5. IEEE.

Hauser, H. et al. (2023). Leveraging morphological computation for controlling soft robots: Learning from nature to control soft robots. *IEEE Control Systems Magazine*, 43(3), 114–129.

Pustina, P., Della Santina, C., Boyer, F., De Luca, A., and Renda, F. (2024). Input decoupling of lagrangian systems via coordinate transformation: General characterization and its application to soft robotics. *IEEE transactions on robotics*.

Ribeiro, L.N., Borja, P., Della Santina, C., and Deutschmann, B. (2025). Singular-perturbation control of a tendon-driven soft robot: Theory and experiments. *IEEE Transactions on Control Systems Technology*.

Wang, H. et al. (2016). Visual servoing of soft robot manipulator in constrained environments with an adaptive controller. *IEEE/ASME Transactions on Mechatronics*, 22(1), 41–50.

Xu, F. et al. (2019). Underwater dynamic visual servoing for a soft robot arm with online distortion correction. *IEEE/ASME Transactions on Mechatronics*, 24(3), 979–989.

Xu, F. et al. (2021). Visual servoing of a cable-driven soft robot manipulator with shape feature. *IEEE Robotics and Automation Letters*, 6(3), 4281–4288.

Zhang, Z. et al. (2017). Visual servoing control of soft robots based on finite element model. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2895–2901. IEEE.