IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Globally-Stable and Robust Image-Based Visual Servoing for Positioning With Respect to a Cylinder

Alessandro Colotti, *Associate Member, IEEE*, and François Chaumette, *Fellow, IEEE*

***Abstract*—This letter proposes a new image-based visual servoing controller for positioning a camera with respect to a cylindrical object. Traditional image-based approaches often rely on estimating planar parameters from the cylinder's projected edges, making them sensitive to noise and modeling errors. In this work, we introduce a novel controller that uses pure image features while directly tied to the cylinder's 3D pose, which depends solely on the cylinder radius. Crucially, this controller offers formal global stability irrespective of the radius estimate. Simulations and real experiments with a robotic arm confirm the controller improved convergence and robustness under practical conditions.**

***Index Terms*—Visual Servoing, sensor-based control.**

I. Introduction

VISUAL servoing is a consolidated control strategy that uses visual information to control the motions of a robot [4]. This control technique is well appreciated for its robustness with respect to modeling and calibration errors, as well as its low sensitivity to image measurement noise, making it a viable strategy in real-world applications. While, classically, the visual servoing literature has been focused on the positioning of the camera with respect to a set of points or markers [4], during its long history a plethora of different objects and, more generally, geometric primitives have been proposed and used to successfully perform visual servoing tasks [2], [5], [6], [7], [17].

Among these, the positioning with respect to cylindrical objects have had great success thanks to their practical implications, ranging from submarine pipeline maintenance [1], [12] to high-speed aerial maneuvering to snatch objects in mid-air [16]. The cylinders' geometrical symmetry and the general absence of surface texture makes the full reconstruction of the camera pose impossible. Consequently, providing a robust approach to ensure the positioning with respect to these objects through image-based means is a key step toward broadening the real-world applicability of visual servoing across diverse scenarios.

Classically, visual servoing tasks concerning cylindrical targets rely on taking as visual features geometric parameters that represent the two lines forming the sides of the projection of the cylinder in the image [3], [12]. However, the computation of the corresponding interaction matrix requires the estimation of the parameters of the plane on which the corresponding 3D lines lay on. This estimation process can be sensitive to noise and modeling errors, affecting the precision and stability of the whole system [9]. Other approaches exist in the literature, notably the use of Plücker coordinates to represent the lines in the image [1] or using the coordinates of two points lying on these lines [16] for controlling 3 DOF of an UAV. Observing the cylinder's ends also allows to use image moments [14] or the intersection points between the cylinder's sides and faces [13], which, coupled with an appropriate path-planner, can guarantee global convergence. However, these strategies rely on always measuring the cylinder's ends, which is not always viable in practice (e.g., in long pipeline inspections).

The main contribution of this letter is the design of a new controller that is built on visual features that are directly measured from the image of the cylinder but that are also directly related to its 3D pose. This set of visual features was originally proposed in [15] for estimating the cylinder radius using a dedicated control scheme for this reconstruction task. Our approach avoids the need for any radius estimation and pursues a fundamentally different objective, that is, reaching a desired view of the cylinder. The proposed controller, based on a generalized inverse and not the classical pseudo-inverse, requires only a nominal value for the cylinder radius, and a crucial aspect of it is its proven global stability, regardless of the choice of the aforementioned nominal value.

As a secondary minor contribution, we propose an improvement on the classical pseudo-inverse controller that uses as visual features the polar coordinates of the two straight lines resulting from the perspective projection of a cylinder. With the same parameterization described above, it is possible to express the 3 3D parameters involved in the corresponding interaction matrix as a function of the cylinder radius, which remains once again the only parameter that has to be known or estimated. This significantly reduces the estimation process complexity and improves the controller's robustness.

The letter is structured as follows. In Section II, we provide an overview of the necessary background on the problem's modeling and the classical image-based visual servoing control scheme. In Section III, we recall the cylinder reconstruction strategy discussed above, and show how it can improve and simplify the interaction matrix estimation process needed for the classical pseudo-inverse controller. In Section IV, we introduce our new axis-plane controller, the letter's main contribution,

Received 9 September 2025; accepted 26 September 2025. Date of publication 9 October 2025; date of current version 16 October 2025. This article was recommended for publication by Associate Editor F. Morbidi and Editor P. Vasseur upon evaluation of the reviewers' comments. This work was supported by ANR SESAME under Grant ANR-18-CE33-0011. *(Corresponding author: Alessandro Colotti.)*

The authors are with Inria, Univ Rennes, CNRS, IRISA, 35042 Rennes, France (e-mail: Alessandro.Colotti@inria.fr).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3619710, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3619710

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

together with a formal proof of its global stability regardless of the radius estimate's value. Finally, in order to validate these controllers effectiveness, we conducted several simulations and real experiments on a robotic arm, which are presented in Section V.

II. IBVS Model and Control

We consider the task of positioning a camera with respect to a cylindrical object. We assume that the camera has 6 degrees of freedom and that it is controllable in velocity. We furthermore assume that the cylinder is static and arbitrarily long. The classical approach is to use as inputs of the control scheme the parameters of the 2 lines resulting from the perspective projection of the cylinder in the image [3]. For each line $\mathcal{L}_i$, these parameters are its polar coordinates, that is, the couple (ρ_i, θ_i) such that $x\cos\theta_i + y\sin\theta_i = \rho_i, \forall (x,y) \in \mathcal{L}_i$. By defining the vector of visual features $\mathbf{s} = (\rho_1, \theta_1, \rho_2, \theta_2)$ and by denoting with $\mathbf{s}^*$ their desired value, the dynamics of the features error $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ are

$$\dot{\mathbf{e}} = \mathbf{L}_\mathbf{e}\boldsymbol{\tau}, \tag{1}$$

where $\boldsymbol{\tau} = (\mathbf{v}, \boldsymbol{\omega})$ is the camera's spatial velocity, with $\mathbf{v}$ its linear velocity and $\boldsymbol{\omega}$ the angular one, and where the interaction matrix $\mathbf{L}_\mathbf{e} \in \mathbb{R}^{4\times 6}$ is given by [3]

$$\mathbf{L}_\mathbf{e} = \left[\begin{array}{c|c} \mathbf{L}_{v,1} & \mathbf{L}_{\omega,1} \\ \mathbf{L}_{v,2} & \mathbf{L}_{\omega,2} \end{array}\right], \tag{2}$$

with

$$\mathbf{L}_{v,i} = \begin{bmatrix} \lambda_{\rho,i}\cos\theta_i & \lambda_{\rho,i}\sin\theta_i & -\lambda_{\rho,i}\rho_i \\ \lambda_{\theta,i}\cos\theta_i & \lambda_{\theta,i}\sin\theta_i & -\lambda_{\theta,i}\rho_i \end{bmatrix} \tag{3}$$

$$\mathbf{L}_{\omega,i} = \begin{bmatrix} (1+\rho_i^2)\sin\theta_i & -(1+\rho_i^2)\cos\theta_i & 0 \\ -\rho_i\cos\theta_i & -\rho_i\sin\theta_i & -1 \end{bmatrix} \tag{4}$$

where $\lambda_{\rho,i} = -A\rho_i\cos\theta_i - B\rho_i\sin\theta_i - C, \lambda_{\theta,i} = B\cos\theta_i - A\sin\theta_i$ and where the parameters (A, B, C) represent the 3D plane $\mathcal{P}$ that intersects with the cylinder's sides (see Fig. 1) such that $1/Z = Ax + By + C, \forall (x,y) \in \mathcal{L}_i$.

The most basic and classical control scheme is given by [4]

$$\boldsymbol{\tau} = -\lambda \widehat{\mathbf{L}}_\mathbf{e}^+ \mathbf{e}, \tag{5}$$

where $\lambda \in \mathbb{R}_+$ and $\widehat{\mathbf{L}}_\mathbf{e}^+$ is the pseudo-inverse of an estimate or an approximation of the interaction matrix $\mathbf{L}_\mathbf{e}$. This control scheme tries to ensure a decoupled exponential decrease of the error $\mathbf{e}$ and is known to be globally asymptotically stable (GAS) if $\mathbf{L}_\mathbf{e}\widehat{\mathbf{L}}_\mathbf{e}^+ > 0$, which occurs when $\widehat{\mathbf{L}}_\mathbf{e} = \mathbf{L}_\mathbf{e}$ since $\mathbf{L}_\mathbf{e}$ is always of full rank 4. However, in our case, $\mathbf{L}_\mathbf{e}$ depends on the plane parameters (A, B, C) that are unknown and have to be estimated for each new pose of the camera since they are expressed in the (moving) camera frame. An estimated $\widehat{\mathbf{L}}_\mathbf{e}$ has thus to be considered in the control scheme, and demonstrating the GAS property is currently out of reach apart assuming (A, B, C) are perfectly estimated, which is a strong assumption.

III. Measuring the 3D Information

In this section, we analyze the method presented in [15] for measuring directly from the visual features $\mathbf{s}$ almost all 3D information regarding the cylinder pose in the camera frame. Then, we see how this method can be applied to the controller (5) in order to lessen the impact of model uncertainties on the controller's behavior.

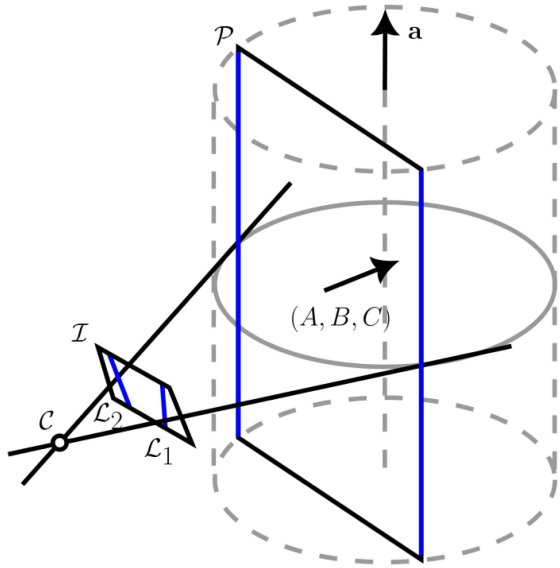

Fig. 1. Construction of the plane $\mathcal{P}$ parameterized by (A, B, C). The camera is denoted with $\mathcal{C}$ and its image plane with $\mathcal{I}$. The cylinder's sides and their projections on the image plane, denoted with $\mathcal{L}_1$ and $\mathcal{L}_2$, are highlighted in blue.

Let the vectors $\mathbf{n}_i \in \mathbb{R}^3$, $i = 1, 2$, be defined as $\mathbf{n}_i = (\cos\theta_i, \sin\theta_i, -\rho_i)$, where (ρ_i, θ_i) represent the polar coordinates of line $\mathcal{L}_i$ in the image, as described in Section II. Additionally, we denote R the radius of the cylinder, $\mathbf{a} \in S^2$ its axis' direction and $\mathbf{P}_0 \in \mathbb{R}^3$ the point on the axis that is the closest to the camera optical center.

By defining $\Delta \in \mathbb{R}^3$ as

$$\Delta := \frac{1}{2}\left(\frac{\mathbf{n}_1}{\|\mathbf{n}_1\|} + \frac{\mathbf{n}_2}{\|\mathbf{n}_2\|}\right), \tag{6}$$

we have the following expressions for $\mathbf{P}_0$ and $\mathbf{a}$ (see [15] for the details)

$$\mathbf{P}_0 = R\frac{\Delta}{\|\Delta\|^2}, \tag{7}$$

$$\mathbf{a} = \frac{[\mathbf{n}_2]_\times \mathbf{n}_1}{\left\|[\mathbf{n}_2]_\times \mathbf{n}_1\right\|} \tag{8}$$

where $[\,\cdot\,]_\times$ represents the cross product matrix.

The first contribution of this letter is to exploit (7) to significantly simplify the estimation of the interaction matrix $\widehat{\mathbf{L}}_\mathbf{e}$ involved in the controller (5). Indeed, in order to compute $\widehat{\mathbf{L}}_\mathbf{e}$, we need to estimate the parameters (A, B, C) describing the 3D plane $\mathcal{P}$ defined in Section II. It is known [3] that these parameters can be expressed as a function of $\mathbf{P}_0 = (X_0, Y_0, Z_0)$, as follows

$$A = \frac{X_0}{K},\ B = \frac{Y_0}{K},\ C = \frac{Z_0}{K}, \tag{9}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

with $K = \|\mathbf{P}_0\|^2 - R^2$. By defining $\boldsymbol{\delta} = (\delta_x, \delta_y, \delta_z)$ as

$$\boldsymbol{\delta} = \frac{\Delta}{\|\Delta\|^2} = \frac{\mathbf{P}_0}{R}, \tag{10}$$

that can be directly computed from the polar coordinates of the two lines using (6), we obtain

$$A = \frac{X_0}{\|\mathbf{P}_0\|^2 - R^2} = \frac{X_0}{R^2(\|\boldsymbol{\delta}\|^2 - 1)} = \frac{1}{R}\frac{\delta_x}{\|\boldsymbol{\delta}\|^2 - 1}, \tag{11}$$

and, analogously

$$B = \frac{1}{R}\frac{\delta_y}{\|\boldsymbol{\delta}\|^2 - 1}, \; C = \frac{1}{R}\frac{\delta_z}{\|\boldsymbol{\delta}\|^2 - 1}. \tag{12}$$

In conclusion, it is possible to compute (A, B, C) directly from $\boldsymbol{\delta}$ up to the scalar factor R, leaving only the cylinder radius as a parameter to be estimated in the computation of $\widehat{\mathbf{L}}_\mathbf{e}$ (in case it is not known *a priori*). Additionally, since (A, B, C) only appear linearly in the elements (3) related to the linear velocity, by decomposing the true interaction matrix as $\mathbf{L}_\mathbf{e} = [\mathbf{L}_v \mid \mathbf{L}_\omega]$, the estimated interaction matrix $\widehat{\mathbf{L}}_\mathbf{e}$ neatly factorizes as follows

$$\widehat{\mathbf{L}}_\mathbf{e} = \left[\tfrac{R}{\widehat{R}}\,\mathbf{L}_v \mid \mathbf{L}_\omega\right]. \tag{13}$$

implying that the estimated radius $\widehat{R}$ only acts as a partial gain, "moderating" between the linear and angular velocity.

As discussed in Section II, the pseudo-inverse controller (5) is GAS when the estimation is perfect (i.e., when $\widehat{R} = R$). Unfortunately, since the pseudo-inverse operation destroys the block-matrix structure of the interaction matrix, we have not been able to prove that the system is GAS when $\widehat{R} \neq R$, even it seems to be from the extensive simulations that we performed. This motivates us to propose in the next section a new set of visual features for positioning with respect to a cylinder, as well as a new controller that is demonstrated to be GAS in practice as long as $\widehat{R} > 0$.

IV. Axis-Plane Controller

Given the results in the previous section, we propose to consider directly $(\boldsymbol{\delta}, \mathbf{a}) \in \mathbb{R}^3 \times S^2$ as our features (where $\boldsymbol{\delta}$ is defined in (10) and $\mathbf{a}$ in (8)). Note that both $\boldsymbol{\delta}$ and $\mathbf{a}$ are computed from the polar coordinates (ρ_i, θ_i) of the lines $\mathcal{L}_i$ in the image, leading to a pure image-based visual servoing scheme. The dynamics of $(\boldsymbol{\delta}, \mathbf{a})$ are given by (see [15])

$$\begin{bmatrix}\dot{\boldsymbol{\delta}}\\ \dot{\mathbf{a}}\end{bmatrix} = \begin{bmatrix}\frac{1}{R}[\mathbf{a}]_\times^2 & [\boldsymbol{\delta}]_\times\\ \mathbf{0} & [\mathbf{a}]_\times\end{bmatrix}\boldsymbol{\tau} := \mathbf{L}_c\boldsymbol{\tau}, \tag{14}$$

where $\boldsymbol{\tau} \in \mathbb{R}^6$ represents our input. Classically, we would use the pseudo-inverse of $\mathbf{L}_c$ in the controller (and simulation results presented in the next section consider this possibility). However, in doing so it would be impossible to demonstrate that the system is GAS since $\mathbf{L}_c \in \mathbb{R}^{6\times6}$ is of rank 4 only. Furthermore, we would lose its nice block triangular structure, being the explicit form of the pseudo-inverse exceedingly complicated in our case (see, e.g., [11]). We take then a different route, by determining a generalized inverse of $\mathbf{L}_c$ with a simple analytical expression.

Let us make the (false!) assumption that $\mathbf{L}_c$ and its blocks are invertible. Then, we would have

$$\mathbf{L}_c^{-1} = \begin{bmatrix}R\left([\mathbf{a}]_\times^2\right)^{-1} & -R\left([\mathbf{a}]_\times^2\right)^{-1}[\boldsymbol{\delta}]_\times[\mathbf{a}]_\times^{-1}\\ \mathbf{0} & [\mathbf{a}]_\times^{-1}\end{bmatrix} \tag{15}$$

By substituting the inverse with the pseudo-inverse on the right-hand side, we obtain a *generalized reflexive inverse*[1] of $\mathbf{L}_c$, that we denote with $\mathbf{L}_c^\ddagger$, which can be used in the feedback loop. Given that $\|\mathbf{a}\| = 1$, we have $[\mathbf{a}]_\times^+ = -[\mathbf{a}]_\times$, and since $\boldsymbol{\delta}$ and $\mathbf{a}$ are orthogonal by definition (see Section III), we obtain that

$$\mathbf{L}_c^\ddagger = \begin{bmatrix}R[\mathbf{a}]_\times^2 & \mathbf{0}\\ \mathbf{0} & -[\mathbf{a}]_\times\end{bmatrix}. \tag{16}$$

We can then define as controller

$$\boldsymbol{\tau} = -\lambda\widehat{\mathbf{L}}_c^\ddagger\begin{bmatrix}\boldsymbol{\delta} - \boldsymbol{\delta}^*\\ \mathbf{a} - \mathbf{a}^*\end{bmatrix}, \tag{17}$$

where $\lambda \in \mathbb{R}_+$, $(\boldsymbol{\delta}^*, \mathbf{a}^*)$ are the desired values of the features, and $\widehat{\mathbf{L}}_c$ denotes that an estimate $\widehat{R}$ is being used. Once again, we can notice that the radius estimate plays the role of a partial gain, acting only on the linear velocity while leaving the angular velocity unaffected. With this controller, by combining (14) and (17), the closed loop behavior of the system is given by

$$\begin{bmatrix}\dot{\boldsymbol{\delta}}\\ \dot{\mathbf{a}}\end{bmatrix} = \lambda\begin{bmatrix}\frac{\widehat{R}}{R}[\mathbf{a}]_\times^2 & [\boldsymbol{\delta}]_\times[\mathbf{a}]_\times\\ \mathbf{0} & [\mathbf{a}]_\times^2\end{bmatrix}\begin{bmatrix}\boldsymbol{\delta} - \boldsymbol{\delta}^*\\ \mathbf{a} - \mathbf{a}^*\end{bmatrix}. \tag{18}$$

On one hand, we can note that the rotational part of our controller is the same as the line-alignment one proposed in [2], which is logical since both use the features $\mathbf{a}$. On the other hand, the translational part is different since they have different objectives: in our case, the distance to the cylinder is controlled thanks to the features $\boldsymbol{\delta}$ that encode an information about that distance. Furthermore, unlike the cascaded design in [2], our controller provides simultaneous 6-DoF regulation and it is GAS in practice, no matter the radius estimate's accuracy.

Theorem 1: The closed-loop system (18) has a single asymptotically stable equilibrium $(\boldsymbol{\delta}, \mathbf{a}) = (\boldsymbol{\delta}^*, \mathbf{a}^*)$, regardless of the choice of $\widehat{R} > 0$. The system converges to it from all initial states such that $\mathbf{a} \neq -\mathbf{a}^*$. If $\mathbf{a} = -\mathbf{a}^*$, the system converges to the unstable equilibrium $(\boldsymbol{\delta}, \mathbf{a}) = (\boldsymbol{\delta}^*, -\mathbf{a}^*)$.

Proof: We begin by noticing from (18) that the dynamics of $\mathbf{a}$ are

$$\dot{\mathbf{a}} = \lambda[\mathbf{a}]_\times^2(\mathbf{a} - \mathbf{a}^*) = -\lambda[\mathbf{a}]_\times\mathbf{e}_a, \tag{19}$$

with $\mathbf{e}_a := [\mathbf{a}]_\times\mathbf{a}^*$. These dynamics do not depend on $\boldsymbol{\delta}$ and it is well-known that such a differential equation converges to $\mathbf{a} = \mathbf{a}^*$ for almost all initial states, with the only exception of $\mathbf{a} = -\mathbf{a}^*$, which is an unstable equilibrium [2].

[1]This is a weaker form of the Moore-Penrose pseudo-inverse, since it satisfies only the first two conditions $\mathbf{L}_c\mathbf{L}_c^\ddagger\mathbf{L}_c = \mathbf{L}_c$ and $\mathbf{L}_c^\ddagger\mathbf{L}_c\mathbf{L}_c^\ddagger = \mathbf{L}_c^\ddagger$ for it, but not the two last ones: $(\mathbf{L}_c\mathbf{L}_c^\ddagger)^T \neq \mathbf{L}_c\mathbf{L}_c^\ddagger$, and $(\mathbf{L}_c^\ddagger\mathbf{L}_c)^T \neq \mathbf{L}_c^\ddagger\mathbf{L}_c$. In general, it is not unique.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

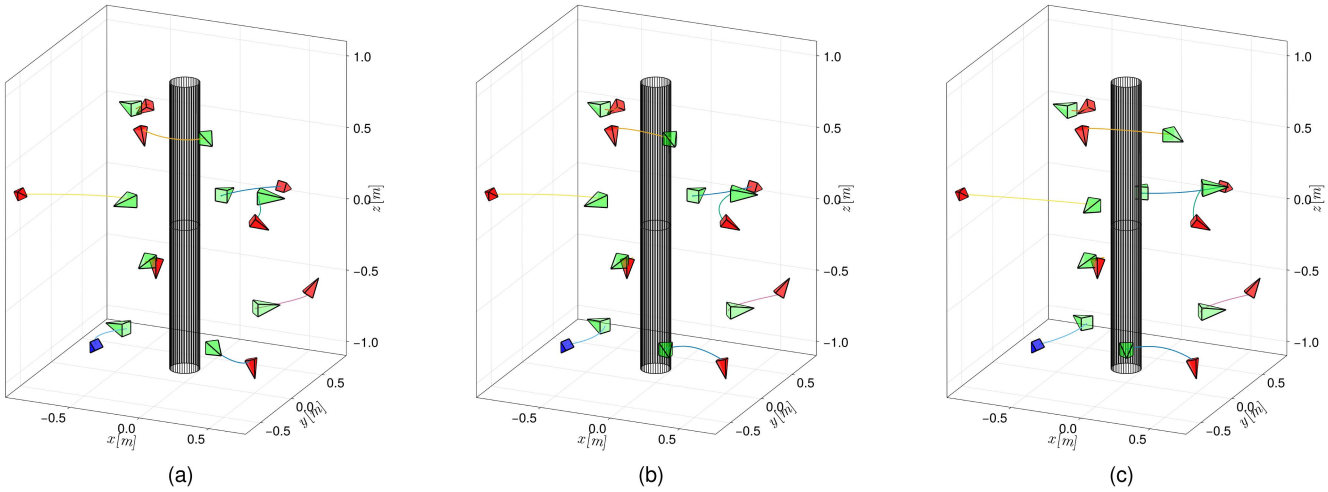

Fig. 2. Set of 9 simulations starting from random initial poses (red and blue), and converging to a final pose (green) such that the final image is the desired one (shown in green on Fig. 3(a)). (a): Improved classical controller. (b): New pseudo-inverse controller. (c): Axis-plane controller.

We can then prove that $\boldsymbol{\delta}$ globally converges to $\boldsymbol{\delta}^*$ by treating $(\mathbf{a}-\mathbf{a}^*) := \mathbf{u}$ as an input and showing that the dynamical system $\mathbf{e}_\delta := (\boldsymbol{\delta}-\boldsymbol{\delta}^*)$ is Input-to-State Stable (ISS). Let $V(\mathbf{e}_\delta) = \frac{1}{2\lambda}\|\mathbf{e}_\delta\|^2$. With a bit of work and by severely exploiting the orthogonality of $\boldsymbol{\delta}$ and $\mathbf{a}$, we obtain that, by defining $r := \widehat{R}/R > 0$,

$$\begin{aligned}\dot{V}(\mathbf{e}_\delta) &= -r\|\mathbf{e}_\delta\|^2 - \mathbf{u}^\top \boldsymbol{\delta}^* \mathbf{u}^\top (\mathbf{e}_\delta - (r-1)\boldsymbol{\delta}^*) \\ &\leq -r\|\mathbf{e}_\delta\|^2 + \|\boldsymbol{\delta}^*\|\|\mathbf{u}\|^2(\|\mathbf{e}_\delta\| + |r-1|\|\boldsymbol{\delta}^*\|), \quad (20)\end{aligned}$$

where we used the Cauchy-Schwarz inequality to obtain the second term. This is quadratic in $\|\mathbf{e}_\delta\|$, and it is easy to verify that $\dot{V}(\mathbf{e}_\delta) < 0$ if the following condition on $\mathbf{e}_\delta$ is verified

$$\|\mathbf{e}_\delta\| > \frac{1}{2r}\|\boldsymbol{\delta}^*\|\|\mathbf{u}\|(\|\mathbf{u}\| + \sqrt{\|\mathbf{u}\|^2 + 4r|r-1|}). \quad (21)$$

By [8, Th. 4.19], $\mathbf{e}_\delta$ is ISS, and, since $\mathbf{u}(t) \to 0$ for all initial states s.t. $\mathbf{a} \neq \mathbf{a}^*$, $(\boldsymbol{\delta}, \mathbf{a}) = (\boldsymbol{\delta}^*, \mathbf{a}^*)$ is asymptotically stable and the system converges to it from all initial states s.t. $\mathbf{a} \neq \mathbf{a}^*$. When $\mathbf{a} = -\mathbf{a}^*$, by orthogonality, we obtain $\dot{\mathbf{e}}_\delta = -\mathbf{e}_\delta$, meaning that the system converges to $(\boldsymbol{\delta}^*, -\mathbf{a}^*)$ that is trivially unstable due to the instability of $-\mathbf{a}^*$ for (19).

In practice, we do not expect that the cylinder's axis orientation plays a role, i.e., that taking $\mathbf{a}^*$ or $-\mathbf{a}^*$ as desired value does not impact the task's success. If this is the case, we can choose the sign of $\mathbf{a}^*$ that is most convenient for the task at hand. One possible choice is to pick the same sign as $\mathbf{a}_0^\top \mathbf{a}^*$, with $\mathbf{a}_0$ denoting the initial value of $\mathbf{a}$, which would prevent large angular displacements and enforce the controller to be GAS in practice.

V. Simulation and Experimental Results

In this section, we compare the performances of the improved classical controller (5) using (11) and (12) in the computation of $\widehat{\mathbf{L}}_\mathbf{e}$, against the two controllers based on our new choice of features, namely the pseudo-inverse controller using $\widehat{\mathbf{L}}_c^+$ in the control loop, and our axis-plane controller (17) using $\widehat{\mathbf{L}}_c^\ddagger$. In Section V-A, we analyze the overall behavior of the three controllers in simulation. Then, in Section V-B, we focus on the impact of modeling uncertainties on the controllers. Finally, in Section V-C, we validate experimentally the results found in simulation. The accompanying video illustrates these experimental results.

All simulations and experiments have been performed using the ViSP C++ library [10]. For all simulations, the gain λ involved in the controllers has been set to $\lambda = 0.2$, while a different choice of gains has been used in the experiments, as detailed in Section V-C.

A. General Behavior With Perfect Estimation

In order to compare the three controllers, we perform a set of nine simulations, each starting from random initial poses and with the same desired image shown in green on Fig. 3(a). The initial poses are picked uniformly in a $2\,\mathrm{m} \times 2\,\mathrm{m} \times 2\,\mathrm{m}$ box centered on the world-frame origin, with a random orientation so that the point $\mathbf{P}_0$ is in front of the camera. We set the cylinder radius to 10cm and consider in these simulations that it is perfectly known, i.e., $\widehat{R} = R$.

A comparison of the 3D trajectories of the camera optical center can be seen in Fig. 2. As expected, all simulations converge to the desired image, which, due to the geometry of the problem, does not require the same final pose to be reached.[2] All three controllers follow very similar paths, with most of them being close to straight lines in 3D, the exceptions being for initial poses that are close to the cylinder and having a large orientation displacement w.r.t. the desired one.

From the results presented above, it is difficult to differentiate the three controllers from the translational trajectory of the camera optical center. That is why we selected one representative

[2]As mentioned above, both $\mathbf{L}_\mathbf{e}$ and $\mathbf{L}_c$ have rank 4, and it is possible to show that they have the same kernel. Thus, there is a 2-dimensional subspace of movements that does not change the image of the cylinder, which is the same for both choices of features.

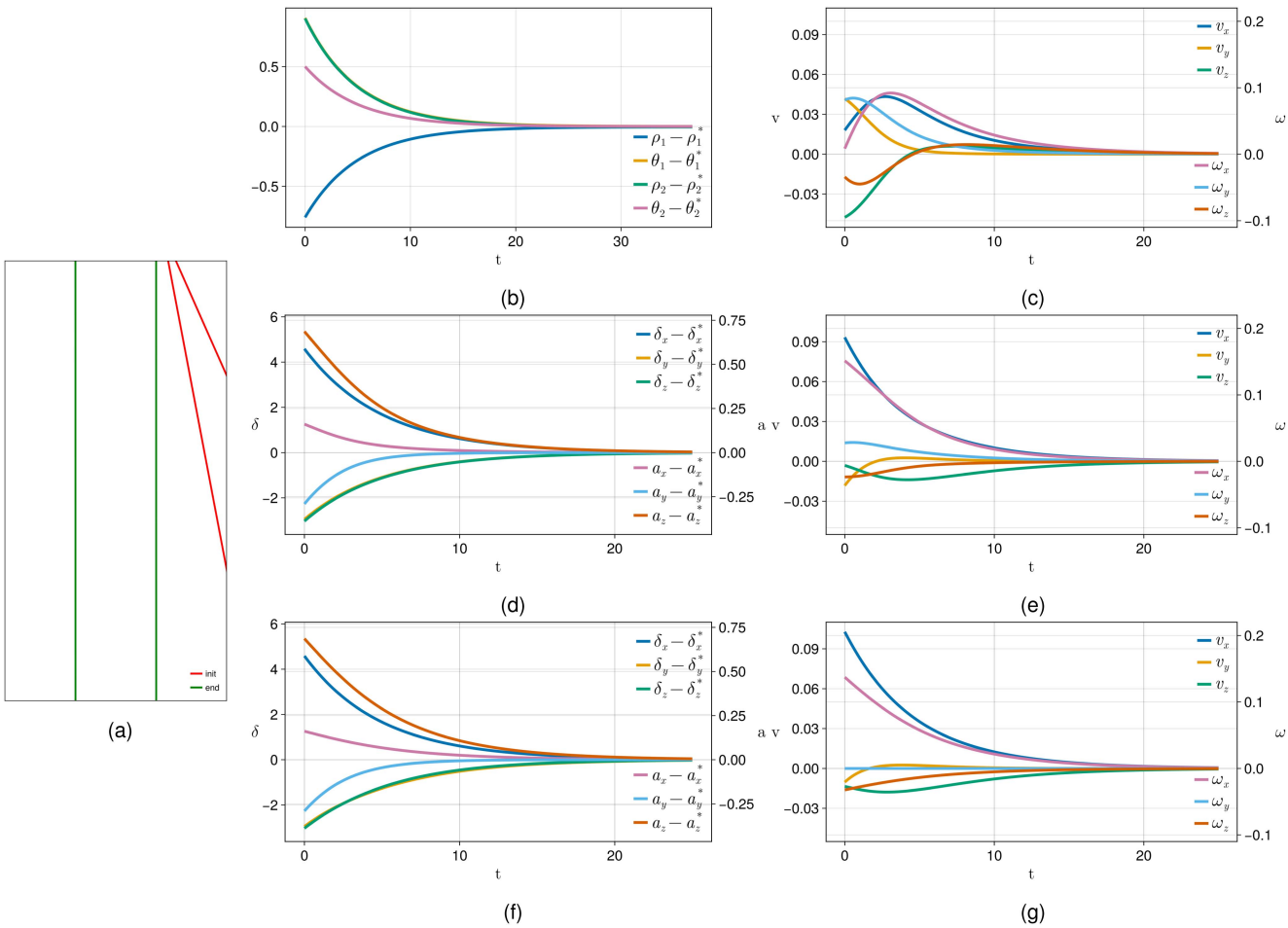

Fig. 3. Evolution of the features errors and velocities (in m/s and rad/s) for the configuration highlighted in blue in Fig. 2. On the left column, we have the initial and desired camera views. In the center and right columns we have the errors and velocities, respectively, for the improved classical (top), the new pseudo-inverse (center) and the axis-plane controller (bottom).

case among the 9 shown above to illustrate the behavior in the image and the output of the different controllers. As shown on Fig. 3(a), the initial position of the two lines in the image is far away from the desired one, meaning there is a large displacement to be achieved between the initial and desired camera poses, especially for the rotational part. As for the improved classical controller, we can first note on Fig. 3(b) the perfect decoupled exponential convergence to zero of the polar coordinates' errors $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ (note that the plots of $\theta_1 - \theta_1^*$ and $\rho_2 - \rho_2^*$ are perfectly superimposed since they have the same initial value). This was expected, since, when $\widehat{\mathbf{L}}_\mathbf{e} = \mathbf{L}_\mathbf{e}$, the features' dynamics are decoupled, i.e., $\dot{\mathbf{e}} = -\lambda \mathbf{e}$ (see Section II). However, the camera velocity components do not follow a similar nice behavior (see Fig. 3(c)), due to the strong non-linearities in $\mathbf{L}_\mathbf{e}$ and, thus, also in $\mathbf{L}_\mathbf{e}^+$. When using the new visual features $(\boldsymbol{\delta}, \mathbf{a})$, they do not follow a pure decoupled exponential decrease since $\mathbf{L}_c \mathbf{L}_c^\ddagger \neq \mathbf{I}_6$ (see Fig. 3(d) and (f)) but still have a very nice behavior. Furthermore, we can note on Fig. 3(e) and (g) that the camera velocity components are more satisfactory, especially for the axis-plane controller for which the rotational components follow a clear exponential decrease, meaning the camera follows an almost perfect geodesic. This is due to the almost linear and pure block diagonal form of $\mathbf{L}_c^\ddagger$. Remarkably, the ω_y component is strictly null along the whole trajectory. This is due to the orthogonality between $\boldsymbol{\omega}$ and $\mathbf{a}^*$, being the latter equal to (0,1,0) for the selected desired configuration. Indeed, from (17), we have $\boldsymbol{\omega} = \lambda [\mathbf{a}]_\times \mathbf{a}^*$. Again, this reinforces the observation that the camera follows an almost perfect geodesic, which is especially true for the rotational part, as it can be deduced from the proof of Theorem 1.

B. *Impact of Modeling Errors*

In order to assess the impact of errors in the radius estimation, we compared the behavior of the controllers for three choices of $\widehat{R}$, namely $\widehat{R} = \{0.1R, R, 10R\}$, selecting the same initial pose considered in the previous section. Let us note that the case $\widehat{R} = 10R$ is very severe since, for our configuration, it is as if the camera was inside the estimated cylinder.

A comparison of the 3D trajectories can be seen in Fig. 4 . Once again, all simulations converge to the desired image, and we observe an overall behavior that is globally similar when $\widehat{R} = R$ and $\widehat{R} = 0.1R$. However, when $\widehat{R} = 10R$, the trajectories of the improved classical and the new pseudo-inverse controllers

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

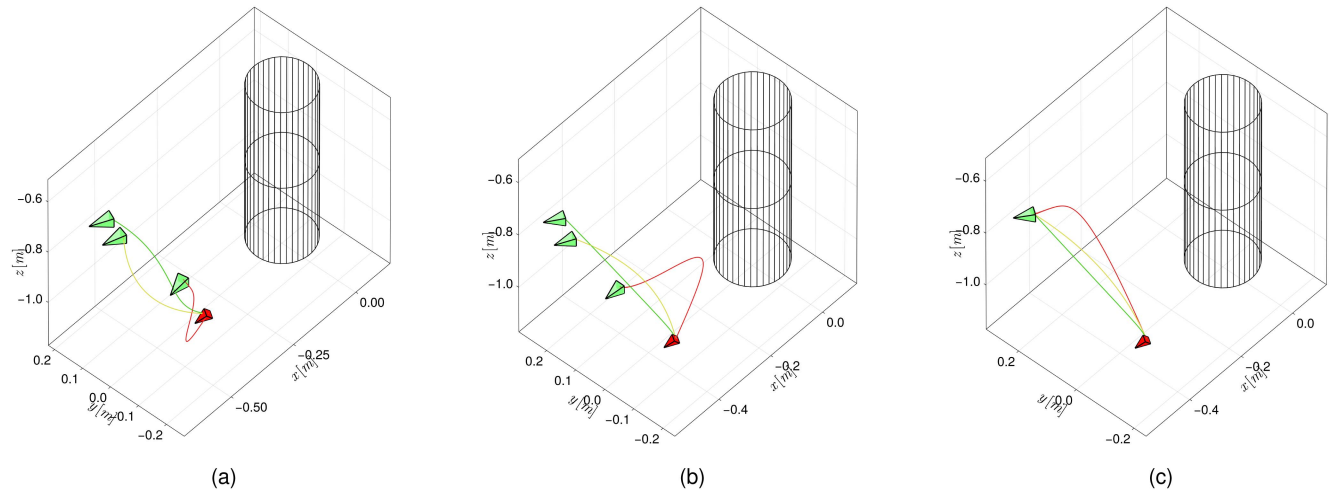

Fig. 4. Set of 3 simulations with $\widehat{R} = \{0.1R, R, 10R\}$, going from green (smallest) to red (largest). (a): Improved classical controller. (b): New pseudo-inverse controller. (c): Axis-plane controller.

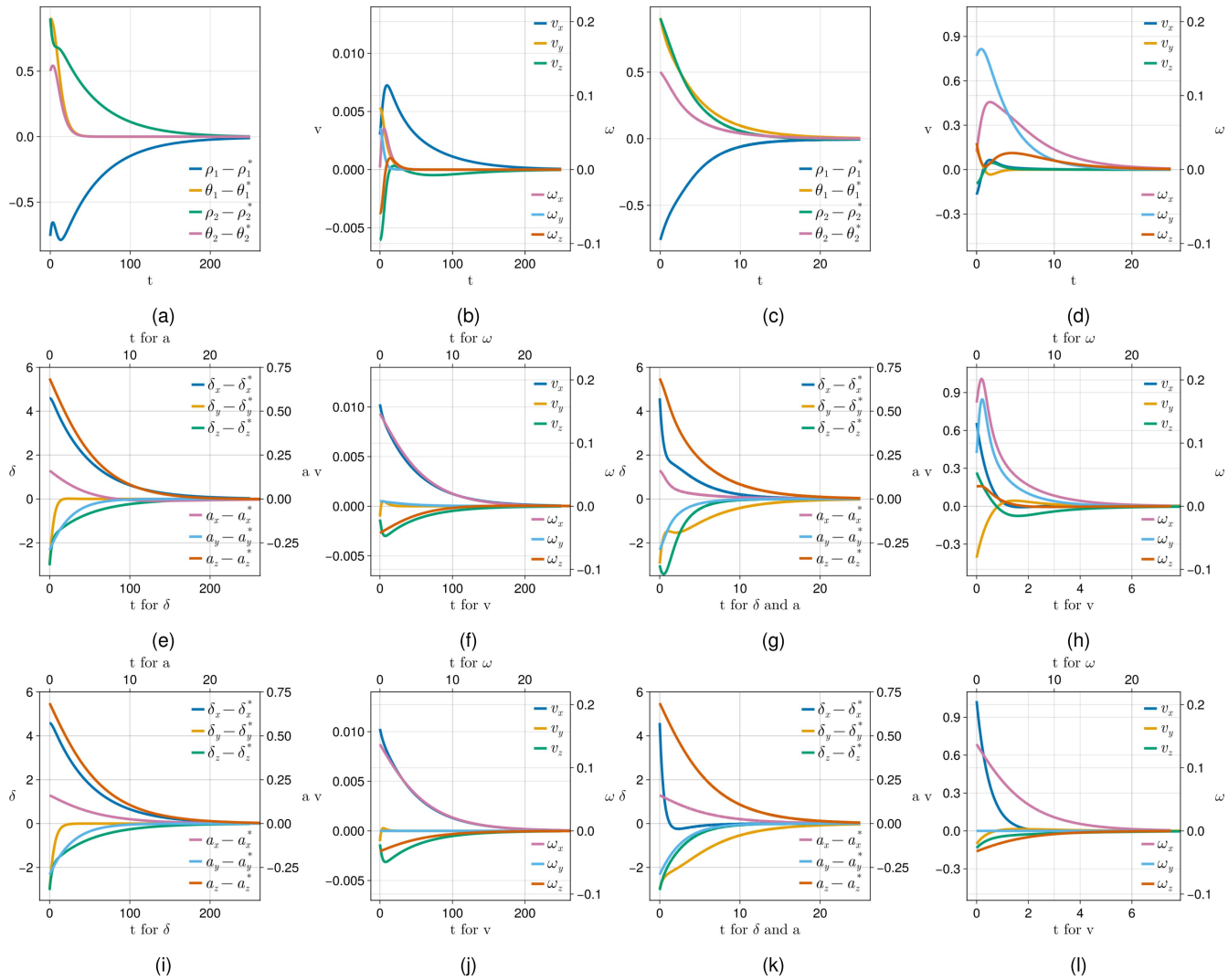

Fig. 5. Comparison of the features errors (first and third columns) and velocities (second and fourth columns), for $\widehat{R} = 0.1R$ (first two columns) and $\widehat{R} = 10R$ (last two columns). Top row: Improved classical controller. Center row: New pseudo-inverse controller. Bottom row: Axis-plane controller.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

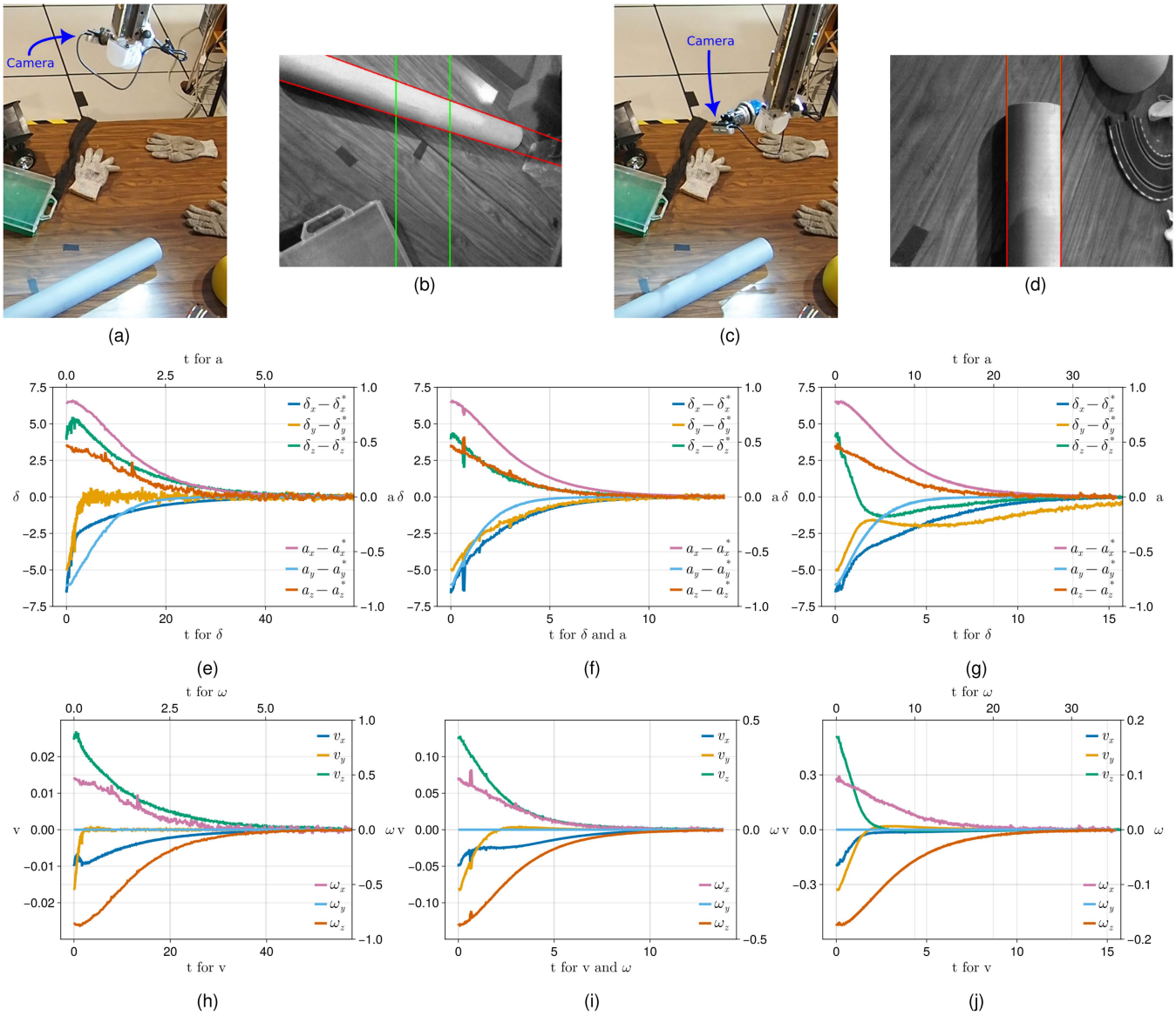

Fig. 6. Robot setup, and evolution of the features' error and velocities of the axis-plane controller for the three choices of $\widehat{R}$. Top row: external and camera view, at the initial (left) and final (right) pose. Center row: features' error evolution, with $\widehat{R}$ increasing from left to right. Bottom row: same for the velocities in m/s and rad/s.

are more impacted than the axis-plane controller. Remarkably, we also observe that, regardless of the choice of $\widehat{R}$, all trajectories of the axis-plane controller converge to the same final pose, which is a positive, albeit unexpected, characteristic.

As for the behavior in the image when using the improved classical controller, we can see on Fig. 5(a) that an underestimation of R strongly modifies the exponential decrease of the visual features and the time-to-convergence, while, from Fig. 5(c), it is not so much the case for $\widehat{R} = 10R$. By comparing Fig. 5(e) and (i) and Fig. 5(f) and (j), we can also note that the behavior of the two controllers based on the new visual features are very similar when $\widehat{R} = 0.1R$, while it is not the case for $\widehat{R} = 10R$ (see Fig. 5(g), (k) and Fig. 5(h), (l)). Finally and most importantly, using the axis-plane controller, we can check on Figs. 3(f), 5(i), 5(k) and 3(g), 5(j), 5(l) that the behavior of $\mathbf{a}$ and $\boldsymbol{\omega}$ is exactly the same whatever the value of $\widehat{R}$, while a bad estimation of R changes drastically the time to convergence of $\boldsymbol{\delta}$ and $\mathbf{v}$ (note the different time scales that have been used for plotting $\mathbf{a}$ and $\boldsymbol{\delta}$, and $\mathbf{v}$ and $\boldsymbol{\omega}$). This was expected from the dynamics established in (18) and the particular form (16) of $\mathbf{L}_c^{\ddagger}$. More in general, the time-to-convergence difference between linear and angular velocities reinforces once again the idea that the radius estimate $\widehat{R}$ plays the role of a partial gain that "moderates" between these two components, as it was observed in Sections III and IV.

C. Experimental Results

In this section, we validate experimentally the simulation results that we presented above. We employ a 6 DoF Gantry

robot with an eye-in-hand INTEL(R) REALSENSE D435 camera, and we aim to position it with respect to a pipe with a radius of 4cm that is laying on the floor. Note that only the RGB images are used for the image processing part while the depth images are not used at all.

As before, the desired configuration is so that the cylinder appears vertical and centered in the image (see Fig. 6(d)), in which case the camera optical axis is perpendicular to the cylinder axis, and so that the distance between the camera and the cylinder axis is 40cm. In the experiments, similarly to what we did in Section V-B, we selected a representative initial pose (see Fig. 6(a) and (b)) and tested the impact of the radius value set in the controllers on their performances. We selected 3 different values for $\widehat{R}$, namely $\widehat{R} = \{0.1R, R, 10R\}$, with corresponding gain choices of $\lambda = \{1, 0.5, 0.2\}$, respectively, as a result of the behaviors observed in the previous simulations. The results for the axis-plane controller with these 3 choices of estimates can be seen in Figs. 6(e) to (g). By looking at the evolution of both the features' error and the velocities, we can see that the controller's behavior on the robot closely matches the one we had in simulation. In particular, by comparing the leftmost and rightmost columns with the corresponding simulation in Fig. 5, we find once again that the radius estimate plays an important moderating role between the angular and linear velocities. That is why we used different gains for each case so that the robot is able to correctly apply the computed velocities. This explains the difference in the time-to-convergence for each case, the one using the correct radius value being logically the best one. Taking these different gain values into account, we can check once again that the behavior of the axis direction components $\mathbf{a}$ and rotational camera velocities $\boldsymbol{\omega}$ are the same whatever the value of $\widehat{R}$. Additionally, we observed that the controller converges to the same final pose, regardless of the choice of $\widehat{R}$, showing that this characteristic persists even under the small disturbances that are naturally present in real experiments.

Finally, analogous observations can be made about the practical behavior of the classical and the new pseudo-inverse controllers with respect to what was obtained in simulation, the reason why we decided to leave them out of the letter. The interested reader can find videos of all three controllers in action in the accompanying video.

VI. CONCLUSION

In this work, we proposed a new set of visual features to design a visual servoing controller for positioning a camera with respect to a cylindrical object. Based on these features, we introduced a novel controller, called the axis-plane controller, that guarantees global stability even when the estimation of the cylinder radius, which is the only unmeasured parameter, is inaccurate.

We argue that this work showcases several positive traits for our new axis-plane controller, making it an attractive choice. From a theoretical point of view, it is the only controller that guarantees a robust and global convergence to the desired image. In practice, the extensive simulations and the experiments on the robot showed that the axis-plane controller is less sensitive to changes in the radius estimate, even converging to the same final pose when starting from a given initial state, while having also better performances in the perfect estimation case. We believe that these results highlight both the theoretical and practical advantages of the axis-plane controller for this kind of visual servoing tasks.

In this letter, we deliberately refrained from any on-line estimation of the cylinder radius so as to isolate and highlight the intrinsic capabilities of the axis-plane controller and to keep the comparison with the other control schemes as transparent as possible. Looking ahead, an obvious extension, which exploits [15]'s full range of results, is to append an observer that recovers this radius.

ACKNOWLEDGMENT

Experiments presented in this paper were carried out thanks to a platform of the Robotex 2.0 French research infrastructure.

REFERENCES

[1] G. Allibert, M. -D. Hua, S. Krupiński, and T. Hamel, "Pipeline following by visual servoing for autonomous underwater vehicles," *Control Eng. Pract.*, vol. 82, pp. 151–160, 2019.
[2] N. Andreff, B. Espiau, and R. Horaud, "Visual servoing from lines," *Int. J. Robot. Res.*, vol. 21, no. 8, pp. 679–700, 2002.
[3] F. Chaumette, "Visual servoing using image features defined upon geometrical primitives," in *Proc. 33rd IEEE Conf. Decis. Control*, 1994, pp. 3782–3787.
[4] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.
[5] N. J. Cowan and D. E. Chang, "Geometric visual servoing," *IEEE Trans. Robot.*, vol. 21, no. 6, pp. 1128–1138, Dec. 2005.
[6] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Autom.*, vol. 8, no. 3, pp. 313–326, Jun. 1992.
[7] D. Fioravanti, C. Colombo, and B. Allotta, "Visual servoing with a pair of coaxial circles," in *Proc. 8th IFAC Symp. Robot Control*, 2006, pp. 37–42.
[8] H. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ, USA: Prentice Hall, 2002.
[9] E. Malis, Y. Mezouar, and P. Rives, "Robustness of image-based visual servoing with a calibrated camera in the presence of uncertainties in the three-dimensional structure," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 112–120, Feb. 2010.
[10] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: A generic software platform with a wide class of robot control skills," *IEEE Robot. Autom. Mag.*, vol. 12, no. 4, pp. 40–52, Dec. 2005.
[11] J. -M. Miao, "General expressions for the moore-penrose inverse of a 2 block matrix," *Linear Algebra Appl.*, vol. 151, pp. 1–15, Jun. 1991.
[12] P. Rives and J.-J. Borrelly, "Underwater pipe inspection task using visual servoing techniques," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot Syst.*, 1997, vol. 1, pp. 63–68.
[13] T. Shen, G. Chesi, Y. Hu, and J. Yang, "Visual servoing path-planning for approaching cylindrical objects," *J. Franklin Inst.*, vol. 360, no. 2, pp. 735–758, 2023.
[14] T. Shen, J. Yang, Y. Cai, D. Li, and G. Chesi, "Visual servoing with cylinders: Reaching the desired location following a straight line," in *Proc. 36th Chin. Control Conf*, 2017, pp. 11183–11188.
[15] R. Spica, P. R. Giordano, and F. Chaumette, "Active structure from motion: Application to point, sphere, and cylinder," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1499–1513, Dec. 2014.
[16] J. Thomas, G. Loianno, K. Sreenath, and V. Kumar, "Toward image-based visual servoing for aerial grasping and perching," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 2113–2118.
[17] H. Wang, Y. -H. Liu, and D. Zhou, "Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 843–857, Aug. 2008.