Coupling Active Depth Estimation and Visual Servoing via a Large Projection Operator

Riccardo Spica, Paolo Robuffo Giordano, François Chaumette

The International Journal of Robotics Research
XX(X):1–18

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Coupling Active Depth Estimation and Visual Servoing via a Large Projection Operator

Riccardo Spica[1], Paolo Robuffo Giordano[2], and François Chaumette[3]

Abstract

The goal of this paper is to propose a coupling between the execution of a Image-Based Visual Servoing (IBVS) task and an *active* Structure from Motion (SfM) strategy. The core idea is to modify *online* the camera trajectory in the null-space of the (main) servoing task for rendering the camera motion 'more informative' w.r.t. the estimation of the 3-D structure. Consequently, the SfM convergence rate and accuracy is maximized during the servoing transient. The improved SfM performance also benefits the servoing execution, since a higher accuracy in the 3-D parameters involved in the interaction matrix improves the IBVS convergence by significantly mitigating the negative effects (instability, loss of feature visibility) of a poor knowledge of the scene structure. Active maximization of the SfM performance results, in general, in a deformed camera trajectory w.r.t. what would be obtained with a classical IBVS: therefore, we also propose an adaptive strategy able to *automatically* activate/deactivate the SfM optimization as a function of the current level of accuracy in the estimated 3-D structure. We finally report a thorough experimental validation of the overall approach under different conditions and case studies. The reported experiments support well the theoretical analysis and clearly show the benefits of the proposed coupling between visual control and active perception.

Keywords

Visual Servoing, Motion Control, Adaptive Control

1 Introduction

In many sensor-based robot applications, the state of the robot w.r.t. the environment can only be partially retrieved from its onboard sensors. In these situations, state estimation schemes can be exploited for recovering online the 'missing information' then fed to any planner/motion controller in place of the actual unmeasurable states. When considering non-trivial cases, however, state estimation must often cope with the nonlinear sensor mappings from the observed environment to the sensor space. The perspective projection performed by cameras is a classical example in this sense (Ma et al. 2003). Because of these nonlinearities, the estimation convergence and accuracy can be strongly affected by the particular trajectory followed by the robot/sensor which, loosely speaking, must guarantee a sufficient level of *excitation* during motion (Cristofaro and Martinelli 2010; Achtelik et al. 2013).

In the context of Structure from Motion (SfM), for example, a poor choice of the camera trajectory can make the 3-D scene structure non-observable *whatever* the employed estimation strategy (Martinelli 2012; Spica and Robuffo Giordano 2013; Eudes et al. 2013; Grabe et al. 2013). Trajectories with low information content will also result, in practice, in inaccurate (or noisy) state estimation. This, in turn, can degrade the performance of any planner/controller that needs to generate actions as a function of the reconstructed states, possibly even leading to failures/instabilities (De Luca et al. 2008; Malis et al. 2010).

The dependence of the estimation performance on the robot trajectory, and of the control performance on the estimation accuracy, clearly creates a tight coupling between perception and action: perception should be optimized for the sake of improving the action execution performance, and the chosen actions should allow

[1]University of Rennes 1, Irisa and Inria, Rennes, France
[2]CNRS, Irisa and Inria, Rennes, France
[3]Inria and Irisa, Rennes, France

Corresponding author:
Paolo Robuffo Giordano, CNRS at Irisa and Inria Rennes Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes Cedex, France.
Email: prg@irisa.fr

maximization of the information gathered during motion for facilitating the estimation task (Valente et al. 2012).

In this respect, the goal of this paper is to propose an *online* coupling between action and perception in the context of robot visual control. We consider, in particular, the class of Image-Based Visual Servoing (IBVS) schemes (Chaumette and Hutchinson 2006) as a representative case study. Indeed, besides being a widespread sensor-based control technique (see e.g., Tahri and Chaumette (2005); Gans and Hutchinson (2007); Mahony and Stramigioli (2012)), IBVS is also affected by *all* the aforementioned issues. On the one hand, whatever the chosen set of visual features (e.g., points, lines, planar patches), the associated *interaction matrix* always depends on some additional 3-D parameters not directly measurable from the visual input (e.g., the depth of a feature point). These parameters must, then, be approximated or estimated online, via a SfM algorithm, with a sufficient level of accuracy for not degrading the servoing execution or even incurring in instabilities or loss of feature visibility (Malis et al. 2010). On the other hand, the SfM performance is directly affected by the particular trajectory followed by the camera during the servoing (Martinelli 2012; Spica and Robuffo Giordano 2013; Spica et al. 2014a): the IBVS controller should then be able to realize the main visual task while, *at the same time*, ensuring a sufficient level of information gain for allowing an accurate state estimation.

In this paper these objectives are met by investigating the *online* coupling between a recently developed framework for active SfM (Spica et al. 2014a) and the execution of a standard IBVS task. For this purpose, we exploit and extend the preliminary results obtained by Spica et al. (2014b): in particular, the main idea is to project any optimization of the camera motion (aimed at improving the SfM performance) within the null-space of the considered visual task in order to not affect the servoing execution. For any reasonable IBVS application, however, a simple null-space projection of a camera trajectory optimization turns out to be ineffective because of a structural lack of redundancy. Therefore, in order to gain the needed freedom, we suitably extend the redundancy framework introduced by Marey and Chaumette (2010) to the case at hand, which requires an action at the camera acceleration level. In addition, an adaptive mechanism is also introduced with the aim of activating/deactivating online the camera trajectory optimization as a function of the accuracy of the estimated 3-D structure for minimizing any 'distorting' effect on the camera motion.

The proposed (adaptive) coupling between active perception and visual control constitutes in our opinion an original contribution w.r.t. the existing literature. Other works have already studied how to fuse visual measurements and different metric cues (e.g. camera velocity/acceleration, observed target velocities and so on) to estimate the geometry of a scene and/or the camera motion (see e.g. De Luca et al. (2008); Martinelli (2012); Eudes et al. (2013); Grabe et al. (2013); Chwa et al. (2016)). Some of these works also identified and discussed the singularities of the problem, but without proposing any active control strategy to avoid them. There obviously exists a vast literature on the topic of *trajectory optimization* for improving the identification/estimation of some unknown parameters/states (see e.g., Achtelik et al. (2013); Wilson et al. (2014); Hollinger and Sukhatme (2014); Miller et al. (2016)).In the context of SfM, the so-called *Next Best View* (NBV) problem has also been addressed before, see Whaite and Ferrie (1997); Chen et al. (2011) for a classical work and a recent survey on this topic. However, many of these strategies are meant for an *offline* use (a whole trajectory is planned, executed, and then possibly re-planned based on the obtained results), and, in any case, do not take into account the *online* realization of a visual task *concurrently* to the optimization of the estimation. At the other end of the spectrum, several works have already investigated how to plug the online estimation of the 3-D structure into a visual servoing loop, see, e.g., Chesi and Hashimoto (2004); Fujita et al. (2007); De Luca et al. (2008); Malis et al. (2009); Petiteville et al. (2010); Corke (2010); Mahony and Stramigioli (2012); Mebarki et al. (2015). In all of these works, however, the SfM scheme is just fed with the camera trajectory generated by the IBVS controller which, on the other hand, has no guarantee of generating a sufficient level of excitation w.r.t. the estimation task.

With respect to this previous literature, our work provides, instead, an *online* solution to the problem of concurrently optimizing the execution of a IBVS task (visual control) and the performance of the 3-D structure estimation (active perception). We also wish to stress that the proposed machinery is not restricted to the sole class of IBVS problems presented in this paper: indeed, one can easily generalize the reported ideas to other servoing tasks (e.g., exploiting different discrete/dense/geometric visual features than those considered in this work), or apply them to Pose-Based Visual Servoing (PBVS) schemes.

The rest of the paper is organized as follows: Sect. 2 describes the theoretical setting of the paper and summarizes the active SfM framework presented in Spica and Robuffo Giordano (2013). Then, Sect. 3 details the machinery needed for coupling IBVS execution and optimization of the 3-D structure estimation. The proposed machinery is, then, validated in Sect. 4 via a number of experiments. Subsequently, Sect. 5 introduces an extension of the strategy detailed in Sect. 3 for allowing a smooth activation/deactivation of the camera trajectory optimization as a function of the current estimation accuracy. This extension is experimentally validated in

Sect. 6. Finally, Sect. 7 concludes the paper and proposes some possible future directions.

2 Problem description

2.1 Image-Based Visual Servoing

Consider a moving camera that measures a set of visual features $\boldsymbol{s} \in \mathbb{R}^m$ (e.g., the x and y coordinates of a point feature) to be regulated at a desired constant value $\boldsymbol{s}^*$. As well-known (Chaumette and Hutchinson 2006), the following relationship holds

$$\dot{\boldsymbol{s}} = \boldsymbol{L_s}(\boldsymbol{s}, \boldsymbol{\chi})\boldsymbol{u} \tag{1}$$

where $\boldsymbol{L_s} \in \mathbb{R}^{m\times 6}$ is the *interaction matrix* of the considered visual features, $\boldsymbol{\chi} \in \mathbb{R}^p$ is a vector of unmeasurable 3-D quantities associated to $\boldsymbol{s}$ (e.g., the depth Z for a point feature), and $\boldsymbol{u} = (\boldsymbol{v}, \boldsymbol{\omega}) \in \mathbb{R}^6$ is the camera linear/angular velocity expressed in the camera frame. By defining $\boldsymbol{e} = \boldsymbol{s} - \boldsymbol{s}^*$ as the visual error vector, one also has $\dot{\boldsymbol{e}} = \boldsymbol{L_s}\boldsymbol{u}$.

If the camera/robot system is *redundant* w.r.t. the visual task ($\mathrm{rank}(\boldsymbol{L_s}) < 6$), a control law that exponentially regulates $\boldsymbol{e}(t) \to \boldsymbol{0}$ can be obtained by solving the following quadratic optimization problem

$$\begin{aligned} \min_{\boldsymbol{u}} \ & \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{r}\|^2 \\ \text{s.t. } & \boldsymbol{L_s}\boldsymbol{u} = -\lambda\boldsymbol{e} \end{aligned} \tag{2}$$

where $\boldsymbol{r} \in \mathbb{R}^6$ represents, in general, the gradient of some suitable scalar cost function representative of secondary objectives. As well-known, the resolution of (2) results in the following control law

$$\boldsymbol{u} = -\lambda\boldsymbol{L_s^\dagger}\boldsymbol{e} + (\boldsymbol{I}_6 - \boldsymbol{L_s^\dagger}\boldsymbol{L_s})\boldsymbol{r} = -\lambda\boldsymbol{L_s^\dagger}\boldsymbol{e} + \boldsymbol{P}\boldsymbol{r}, \quad \lambda > 0, \tag{3}$$

where $\boldsymbol{L_s^\dagger}$ denotes the Moore-Penrose pseudoinverse of matrix $\boldsymbol{L_s}$, and $\boldsymbol{P} = (\boldsymbol{I}_6 - \boldsymbol{L_s^\dagger}\boldsymbol{L_s}) \in \mathbb{R}^{6\times 6}$ is used to project the action of $\boldsymbol{r}$ in the null-space of the main visual task so that $\|\boldsymbol{u} - \boldsymbol{r}\|$ is minimized while not perturbing the achievement of the main task (Siciliano et al. 2009).

Any implementation of (3) (or variants) must deal with the lack of a direct measurement of vector $\boldsymbol{\chi}$. A common workaround is to replace the exact interaction matrix $\boldsymbol{L_s}(\boldsymbol{s}, \boldsymbol{\chi})$ with an estimation $\hat{\boldsymbol{L}}_{\boldsymbol{s}} = \boldsymbol{L_s}(\boldsymbol{s}, \hat{\boldsymbol{\chi}})$ evaluated on some *approximation* $\hat{\boldsymbol{\chi}}$ of the unknown true vector $\boldsymbol{\chi}$. In this approximated case, assuming for simplicity $\boldsymbol{r} \equiv 0$, the closed-loop error dynamics, becomes

$$\dot{\boldsymbol{e}} = -\lambda\boldsymbol{L_s}\hat{\boldsymbol{L}}_{\boldsymbol{s}}^{\dagger}\boldsymbol{e}, \tag{4}$$

and stability is determined by the eigenvalues of the matrix $\boldsymbol{L_s}(\boldsymbol{s}, \boldsymbol{\chi})\boldsymbol{L_s}(\boldsymbol{s}, \hat{\boldsymbol{\chi}})^\dagger$ (Malis and Chaumette 2002).

Special approximations, such as $\hat{\boldsymbol{\chi}} = \boldsymbol{\chi}^* = const$, where $\boldsymbol{\chi}^*$ is the value of $\boldsymbol{\chi}$ at the desired pose, can, at best, only guarantee local stability in a neighborhood of $\boldsymbol{s}^*$ (see Chaumette and Hutchinson (2006)) and, in any case, require some prior knowledge on the scene (the value of $\boldsymbol{\chi}^*$ must be obtained independently from the execution of the servoing task). Additionally, too rough estimations of the final $\boldsymbol{\chi}^*$ (or other approximation choices for $\hat{\boldsymbol{\chi}}$) may result in a poor, or even unstable, closed-loop behavior for the servoing (see Malis et al. (2010) and the illustrative example in Sect. 4.3).

In this context, the use of an incremental estimation scheme, able to generate online a converging $\hat{\boldsymbol{\chi}}(t) \to \boldsymbol{\chi}(t)$ from (ideally) any initial approximation $\hat{\boldsymbol{\chi}}(t_0)$, can represent an effective alternative. Indeed, such a scheme can improve the servoing execution by approximating the ideal control law (3) also when *far* from the desired pose and without needing special assumptions/approximations of $\boldsymbol{\chi}$ since, as $\hat{\boldsymbol{\chi}}(t) \to \boldsymbol{\chi}(t)$, one obviously has $\hat{\boldsymbol{L}}_{\boldsymbol{s}} \to \boldsymbol{L_s}$.

Other factors (e.g., estimation gains) being equal, the convergence rate of a SfM scheme is mainly affected by the particular trajectory followed by the camera w.r.t. the observed scene, with some trajectories being more informative/exciting than other ones. Therefore, the IBVS controller should select (online) the 'most informative' camera trajectory, among all the possible ones solving the visual task, for obtaining the fastest possible SfM convergence during the servoing transient. Section 3 will detail how to attain this goal.

2.2 Active Structure from Motion

Excluding degenerate cases (e.g., when a line projects on a single point or a circle projects on a segment, and so on.), the dynamics of any image-based visual feature vector $\boldsymbol{s}$ in (1) can always be expanded linearly w.r.t. the unknown vector $\boldsymbol{\chi}$ as follows (see Espiau et al. (1992); Chaumette (2004))

$$\dot{\boldsymbol{s}} = \boldsymbol{f}_m(\boldsymbol{s}, \boldsymbol{\omega}) + \boldsymbol{\Omega}^T(\boldsymbol{s}, \boldsymbol{v})\boldsymbol{\chi} \tag{5}$$

where vector $\boldsymbol{f}_m(\boldsymbol{s}, \boldsymbol{\omega}) \in \mathbb{R}^m$ and matrix $\boldsymbol{\Omega}(\boldsymbol{s}, \boldsymbol{v}) \in \mathbb{R}^{p\times m}$ are functions of *known* quantities. As for vector $\boldsymbol{\chi}$, since its dynamics depends on the particular geometry of the scene, no special structure is assumed apart from a generic smooth dependence on the system states and inputs, i.e.,

$$\dot{\boldsymbol{\chi}} = \boldsymbol{f}_u(\boldsymbol{s}, \boldsymbol{\chi}, \boldsymbol{u}). \tag{6}$$

Owing to the linearity of (5) w.r.t. $\boldsymbol{\chi}$, the sensitivity of the feature dynamics w.r.t. the unknown $\boldsymbol{\chi}$ is $\partial\dot{\boldsymbol{s}}/\partial\boldsymbol{\chi} = \boldsymbol{\Omega}^T(\boldsymbol{s}, \boldsymbol{v})$, that is, a function of *only known quantities* (the measured $\boldsymbol{s}$ and the 'control vector' $\boldsymbol{v}$). Therefore, it is possible to act on $\boldsymbol{v}$ in order to increase the conditioning of the 'sensitivity' $\boldsymbol{\Omega}^T(\boldsymbol{s}, \boldsymbol{v})$ during the camera motion. This insight has been exploited by Spica et al. (2014a) for proposing an active SfM scheme, built upon the

dynamics (5–6), and yielding an estimation error with an *assignable* convergence rate. The machinery of Spica et al. (2014a) is here briefly summarized.

Let $(\hat{\boldsymbol{s}}\,, \hat{\boldsymbol{\chi}}) \in \mathbb{R}^{m+p}$ be an estimation of $(\boldsymbol{s}\,, \boldsymbol{\chi})$, and define $\boldsymbol{\xi} = \boldsymbol{s} - \hat{\boldsymbol{s}}$ as the 'prediction error' and $\boldsymbol{z} = \boldsymbol{\chi} - \hat{\boldsymbol{\chi}}$ as the 3-D structure estimation error. An estimation scheme for system (5–6), meant to recover the unmeasurable $\boldsymbol{\chi}(t)$ from the measured $\boldsymbol{s}(t)$ and known $\boldsymbol{u}(t)$, can be devised as

$$\left\{ \begin{array}{rcl} \dot{\hat{\boldsymbol{s}}} & = & \boldsymbol{f}_m(\boldsymbol{s},\, \boldsymbol{\omega}) + \boldsymbol{\Omega}^T(\boldsymbol{s},\, \boldsymbol{v})\hat{\boldsymbol{\chi}} + \boldsymbol{H}\boldsymbol{\xi} \\ \dot{\hat{\boldsymbol{\chi}}} & = & \boldsymbol{f}_u(\boldsymbol{s},\, \hat{\boldsymbol{\chi}},\, \boldsymbol{u}) + \alpha\boldsymbol{\Omega}(\boldsymbol{s},\, \boldsymbol{v})\boldsymbol{\xi} \end{array} \right. \tag{7}$$

where $\boldsymbol{H} > 0$ and $\alpha > 0$ are suitable gains.

By coupling observer (7) to (5–6), one obtains the following error dynamics

$$\left\{ \begin{array}{rcl} \dot{\boldsymbol{\xi}} & = & -\boldsymbol{H}\boldsymbol{\xi} + \boldsymbol{\Omega}^T(\boldsymbol{s},\, \boldsymbol{v})\boldsymbol{z} \\ \dot{\boldsymbol{z}} & = & -\alpha\boldsymbol{\Omega}(\boldsymbol{s},\, \boldsymbol{v})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{z},\, t) \end{array} \right. \tag{8}$$

with $\boldsymbol{g}(\boldsymbol{z},\, t) = \boldsymbol{f}_u(\boldsymbol{s},\, \boldsymbol{\chi},\, \boldsymbol{u}) - \boldsymbol{f}_u(\boldsymbol{s},\, \hat{\boldsymbol{\chi}},\, \boldsymbol{u})$ being a vanishing perturbation term ($\boldsymbol{g}(\boldsymbol{z},\, t) \to \boldsymbol{0}$ as $\boldsymbol{z}(t) \to \boldsymbol{0}$). As discussed in Spica et al. (2014a), the error system (8) can be proven to be semi-globally exponentially stable provided the $p \times p$ square matrix $\boldsymbol{\Omega}\boldsymbol{\Omega}^T$ remains full rank during motion (therefore, availability of $m \geq p$ independent measurements is needed). Furthermore, the unperturbed version of (8) (i.e., with $\boldsymbol{g} = \boldsymbol{0}$) enjoys a port-Hamiltonian structure with the associated Hamiltonian (storage function)

$$\mathcal{H}(\boldsymbol{\xi},\, \boldsymbol{z}) = \frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi} + \frac{1}{2\alpha}\boldsymbol{z}^T\boldsymbol{z}. \tag{9}$$

These facts will be important for the developments of Sect. 5.

Following Spica et al. (2014a), the transient response of the SfM estimation error $\boldsymbol{z}(t) = \boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)$ can be exactly *characterized* and *affected* by acting *online* on the camera linear velocity $\boldsymbol{v}$. Indeed, the convergence rate of $\boldsymbol{z}(t)$ is determined by the norm of the square matrix $\alpha\boldsymbol{\Omega}\boldsymbol{\Omega}^T$ (in particular by its smallest eigenvalue $\alpha\sigma_1^2$) which plays the role of an *observability measure* for system (5–6). For a given choice of gain α (a free parameter), the larger σ_1^2 the faster the error convergence with, in particular, $\sigma_1^2 = 0$ if $\boldsymbol{v} = 0$ (as well-known, only a translating camera can estimate the scene structure).

Since $\boldsymbol{\Omega} = \boldsymbol{\Omega}(\boldsymbol{s},\, \boldsymbol{v})$, one also has $\sigma_1^2 = \sigma_1^2(\boldsymbol{s},\, \boldsymbol{v})$ and

$$\dot{(\sigma_1^2)} = \boldsymbol{J}_{\sigma_v}\dot{\boldsymbol{v}} + \boldsymbol{J}_{\sigma_s}\dot{\boldsymbol{s}}, \tag{10}$$

where the Jacobian matrices $\boldsymbol{J}_{\sigma_v} = \frac{\partial \sigma_1^2}{\partial \boldsymbol{v}} \in \mathbb{R}^{1\times 3}$ and $\boldsymbol{J}_{\sigma_s} = \frac{\partial \sigma_1^2}{\partial \boldsymbol{s}} \in \mathbb{R}^{1\times m}$ have a *closed form* expression function of $(\boldsymbol{s},\, \boldsymbol{v})$ (again, known quantities). It is then possible to exploit relationship (10) for affecting *online* $\sigma_1^2(t)$ during motion in order to, e.g., maximize its value and, as a consequence, increase the convergence rate of the estimation error $\boldsymbol{z}(t)$.

To conclude, we detail the above machinery for the particular case of point features considered in this paper. Let $\boldsymbol{s} = \boldsymbol{p} = (x,\, y) = (X/Z,\, Y/Z)$ be the perspective projection of a 3-D point $(X,\, Y,\, Z)$, and $\chi = 1/Z$ with, thus, $m = 2$ and $p = 1$ (note that $m > p$ as required). From Spica et al. (2014a) we have

$$\left\{ \begin{array}{l} \sigma_1^2 = \boldsymbol{\Omega}\boldsymbol{\Omega}^T = (xv_z - v_x)^2 + (yv_z - v_y)^2 \\ \boldsymbol{J}_{\sigma_v} = 2\begin{bmatrix} v_x - xv_z & v_y - yv_z & (xv_z - v_x)\,x + (yv_z - v_y)\,y \end{bmatrix}. \\ \boldsymbol{J}_{\sigma_s} = 2\begin{bmatrix} (xv_z - v_x)v_z & (yv_z - v_y)v_z \end{bmatrix} \end{array} \right. \tag{11}$$

3 Plugging active sensing in Image-Based Visual Servoing schemes

In the redundant case, the execution of a servoing task can be naturally coupled with the (concurrent) optimization of the estimation of vector $\boldsymbol{\chi}$ by exploiting vector $\boldsymbol{r}$ in (3) for projecting any action aimed at maximizing σ_1^2 in the null-space of the visual task. The expression (10) shows that the optimization of $\sigma_1^2(t)$ requires an action at the *camera acceleration level*. In particular, since

$$\nabla_{\boldsymbol{u}}\sigma_1^2 = \begin{bmatrix} \boldsymbol{J}_{\sigma_v}^T \\ \boldsymbol{0} \end{bmatrix} \tag{12}$$

local maximization of σ_1^2 can be achieved by just following its positive gradient via a camera acceleration vector

$$\dot{\boldsymbol{u}}_\sigma = \begin{bmatrix} k_\sigma \boldsymbol{J}_{\sigma_v}^T \\ \boldsymbol{0} \end{bmatrix}, \qquad k_\sigma > 0. \tag{13}$$

Being $\dot{\boldsymbol{e}} = \boldsymbol{L}_{\boldsymbol{s}}\boldsymbol{u}$ and, thus, $\ddot{\boldsymbol{e}} = \boldsymbol{L}_{\boldsymbol{s}}\dot{\boldsymbol{u}} + \dot{\boldsymbol{L}}_{\boldsymbol{s}}\boldsymbol{u}$, and by formulating an optimization problem analogous to (2) (Siciliano et al. 2009), one can show that the *second-order/acceleration level* counterpart of the classical law (3) for regulating the error vector $\boldsymbol{e}(t)$ to $\boldsymbol{0}$ is simply

$$\dot{\boldsymbol{u}} = \dot{\boldsymbol{u}}_{\boldsymbol{e}} = \boldsymbol{L}_{\boldsymbol{s}}^{\dagger}(-k_v\dot{\boldsymbol{e}} - k_p\boldsymbol{e} - \dot{\boldsymbol{L}}_{\boldsymbol{s}}\boldsymbol{u}) + \boldsymbol{P}\boldsymbol{r} \tag{14}$$

with $k_p > 0$ and $k_v > 0$. Therefore, by setting $\boldsymbol{r} = \dot{\boldsymbol{u}}_\sigma$ in (14), one would obtain the desired maximization of σ_1^2 (i.e., of the convergence rate of the 3-D estimation error) concurrently to the execution of the main visual task. This straightforward strategy, although appealing for its simplicity, is unfortunately not viable in most practical situations because of the structural lack of *redundancy* for implementing action (13) (or any equivalent one) in (14). Indeed, in most visual servoing applications, the feature set $\boldsymbol{s}$ is purposely designed to constrain all the camera DOFs (i.e., $\mathrm{rank}(\boldsymbol{L}_s) = 6$), and, as a consequence, no optimization of the camera linear velocity $\boldsymbol{v}$ can be performed via the null-space projector operator $\boldsymbol{P}$. This fundamental limitation motivates the development of the alternative strategy presented in the following section.

3.1 Second-order Visual Servoing using a Large Projection Operator

An alternative control strategy, able to circumvent the redundancy limitations discussed above, can be devised by suitably exploiting the redundancy framework originally proposed by Marey and Chaumette (2010). In this work, it is shown how regulation of the full visual error vector $\boldsymbol{e}$ (a m-dimensional task) can be replaced by the regulation of its norm $\|\boldsymbol{e}\|$ (a 1-dimensional task). This manipulation results in a null-space of (maximal) dimension $6-1=5$ available for additional optimizations. The machinery presented in Marey and Chaumette (2010) (at the first order) can be exploited as follows: letting $\nu=\|\boldsymbol{e}\|$, we have

$$\dot{\nu}=\frac{\boldsymbol{e}^T\dot{\boldsymbol{e}}}{\|\boldsymbol{e}\|}=\frac{\boldsymbol{e}^T\boldsymbol{L_s}}{\|\boldsymbol{e}\|}\boldsymbol{u}=\boldsymbol{L}_\nu\boldsymbol{u},\qquad \boldsymbol{L}_\nu\in\mathbb{R}^{1\times 6},$$

and, at second-order,

$$\ddot{\nu}=\boldsymbol{L}_\nu\dot{\boldsymbol{u}}+\dot{\boldsymbol{L}}_\nu\boldsymbol{u}.$$

Regulation of $\nu(t)\to 0$ can then be achieved by the following control law analogous to (14)

$$\dot{\boldsymbol{u}}=\dot{\boldsymbol{u}}_\nu=\boldsymbol{L}_\nu^\dagger(-k_v\dot{\nu}-k_p\nu-\dot{\boldsymbol{L}}_\nu\boldsymbol{u})+\boldsymbol{P}_\nu\boldsymbol{r},\qquad(15)$$

with $k_p>0$, $k_v>0$, $\boldsymbol{L}_\nu^\dagger=\dfrac{\|\boldsymbol{e}\|}{\boldsymbol{e}^T\boldsymbol{L_s}\boldsymbol{L_s}^T\boldsymbol{e}}\boldsymbol{L_s}^T\boldsymbol{e}$, and $\boldsymbol{P}_\nu=\boldsymbol{I}_6-\dfrac{\boldsymbol{L_s}^T\boldsymbol{e}\boldsymbol{e}^T\boldsymbol{L_s}}{\boldsymbol{e}^T\boldsymbol{L_s}\boldsymbol{L_s}^T\boldsymbol{e}}$ being the null-space projection operator of the error norm with rank $6-1=5$ (see Marey and Chaumette (2010)).

By implementing controller (15) in place of (14) one can still obtain regulation of the whole visual task error since, obviously, $\nu(t)=\|\boldsymbol{e}(t)\|\to 0$ implies $\boldsymbol{e}(t)\to\boldsymbol{0}$. However, contrarily to (14), the new null-space projector $\boldsymbol{P}_\nu$ allows implementing a broader range of optimization actions including (13) or equivalent ones.

On the other hand, a shortcoming of (15) w.r.t. (14) is that the interaction matrix $\boldsymbol{L}_\nu$ is singular for $\|\boldsymbol{e}\|=0$ and, consequently, the projection matrix $\boldsymbol{P}_\nu$, and the pseudoinverse $\boldsymbol{L}_\nu^\dagger$, are not well-defined when the visual task is close to full convergence. As discussed in Marey and Chaumette (2010), this singularity can be avoided by switching from controller (15) to the classical law (14) when $\|\boldsymbol{e}\|$ becomes sufficiently small. Unfortunately, however, the 'first-order' switching strategy proposed by Marey and Chaumette (2010) is not directly transposable to the second-order case. Section 3.3 details, therefore, a suitable 'second-order' approach able to guarantee a proper switching from (15) to the classical law (14).

Remark 3.1. *Note that (15) also suffers from another singularity occurring when* $\boldsymbol{e}\in\ker(\boldsymbol{L_s}^T)$. *This corresponds, however, to a* local minimum *for the servoing itself, also affecting (14): if* $\boldsymbol{e}\in\ker(\boldsymbol{L_s}^T)$, *no camera motion can* instantaneously *realize the task. Therefore,* any *'local' control action would be equally affected by this issue, and no simple switching strategy could be employed in this case. Local minima escaping strategies, such as random walks or global optimizations, are, on the other hand, out of the scope of this paper.*

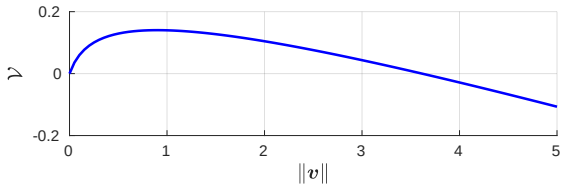


Figure 1. A representative graph of the cost function $\mathcal{V}(\boldsymbol{v})$ in (16) for $k_\sigma=1$, $k_d=0.2$, $\gamma=0.1$, $\|\boldsymbol{\omega}\|=0$, and assuming $\sigma_1^2=\|\boldsymbol{v}\|^2$. Note the presence of a finite upper bound for $\mathcal{V}(\boldsymbol{v})$ as desired.

3.2 Optimization of the 3-D Reconstruction

As discussed in Sect. 2.2, the convergence rate of the 3-D estimation error $\boldsymbol{z}(t)=\boldsymbol{\chi}(t)-\hat{\boldsymbol{\chi}}(t)$ is determined by the eigenvalue σ_1^2. To improve the estimation performance, one could attempt to maximize a cost function of the form $\mathcal{V}(\boldsymbol{u})=k_\sigma\sigma_1^2(\boldsymbol{v})$. This straightforward solution would result, however, in an unbounded growth of $\|\boldsymbol{u}\|$. Indeed, $\sigma_1^2\propto\|\boldsymbol{v}\|^2$ (see (11) for the point feature case and Spica et al. (2014a, 2015) for other examples) and, therefore, σ_1^2 can be made arbitrarily large by increasing $\|\boldsymbol{v}\|$ – the faster the camera motion, the larger value of σ_1^2.

In order to cope with this issue, it is then necessary to consider a cost function that allows for a finite upper bound w.r.t. $\|\boldsymbol{v}\|$. Among the many possible solutions meeting this requirement, we opted for the following cost function

$$\mathcal{V}(\boldsymbol{u})=k_\sigma\gamma\log\left(\frac{\gamma+\sigma_1^2(\boldsymbol{v})}{\gamma}\right)-\frac{k_d}{2}\|\boldsymbol{u}\|^2,\qquad\gamma>0,\qquad(16)$$

for which a representative graph is depicted in Fig. 1. This choice is motivated by considering that $\sigma_1^2\propto\|\boldsymbol{v}\|^2$ and $\log(x)=o(g(x))$ for any polynomial function $g(x)$. Therefore, for sufficiently large velocities ($\|\boldsymbol{v}\|\to\infty$), the damping term $\frac{k_d}{2}\|\boldsymbol{u}\|^2$ will be dominant w.r.t. the first term in (16), thereby ensuring existence of a finite upper bound w.r.t. $\|\boldsymbol{v}\|$.

Maximization of $\mathcal{V}(\boldsymbol{u})$ is, then, obtained as best as possible by plugging in vector $\boldsymbol{r}$, in (15), the following camera acceleration vector

$$\dot{\boldsymbol{u}}_\mathcal{V}=\nabla_{\boldsymbol{u}}\mathcal{V}=\frac{k_\sigma\gamma}{\gamma+\sigma_1^2}\nabla_{\boldsymbol{u}}\sigma_1^2-k_d\boldsymbol{u}.\qquad(17)$$

3.3 Second-order Switching Strategy

As explained in the previous section, the first-order switching strategy proposed by Marey and Chaumette (2010) does not simply extend to the second-order case and, therefore, we now detail a suitable second-order switching strategy meant to avoid the singularity of controller (15) when $\nu(t) = \|\boldsymbol{e}(t)\| \to 0$. We start by noting that controller $\dot{\boldsymbol{u}}_\nu$ in (15) imposes the following second-order dynamics to the error norm

$$\ddot{\nu} + k_v\dot{\nu} + k_p\nu = 0. \tag{18}$$

Define $\nu_{\|\boldsymbol{e}\|}(t)$ as the solution of (18) for a given initial condition $(\nu(t_0),\, \dot{\nu}(t_0))$.

Let now $t_1 > t_0$ be the time at which the switch from controller (15) to the classical law $\dot{\boldsymbol{u}}_{\boldsymbol{e}}$ in (14) occurs. For $t \geq t_1$, controller $\dot{\boldsymbol{u}}_{\boldsymbol{e}}$, under the assumption $\mathrm{rank}(\boldsymbol{L}_{\boldsymbol{s}}) = m$, yields

$$\ddot{\boldsymbol{e}} + k_v\dot{\boldsymbol{e}} + k_p\boldsymbol{e} = 0. \tag{19}$$

If $\mathrm{rank}(\boldsymbol{L}_{\boldsymbol{s}}) < m$, as in the case studies reported in Sect. 4, the ideal behavior (19) can, in general, only be approximately imposed.

Let $\boldsymbol{e}^*(t)$ be the solution of (19) with initial conditions $(\boldsymbol{e}(t_1),\, \dot{\boldsymbol{e}}(t_1))$, and let $\nu^*(t) = \|\boldsymbol{e}^*(t)\|$ be the corresponding behavior of the error norm. Ideally, one would like to have

$$\nu^*(t) \equiv \nu_{\|\boldsymbol{e}\|}(t), \quad \forall t \geq t_1. \tag{20}$$

In other words, the behavior of the error norm should not be affected by the control switch at time t_1, but $\nu^*(t)$ (obtained from (19)) should exactly match the 'ideal' evolution $\nu_{\|\boldsymbol{e}\|}(t)$ generated by (18) as if no switch had taken place.

While condition (20) is easily satisfied at the first-order (Marey and Chaumette 2010), this is not necessarily the case at the second-order level. Indeed, the following result holds (see appendix B)

Proposition 3.2. *For the second-order error dynamics (18–19), condition (20) holds if and only if, at the switching time t_1, vectors $\boldsymbol{e}(t_1)$ and $\dot{\boldsymbol{e}}(t_1)$ are* parallel.

It is then necessary to introduce an intermediate phase, before the switch, during which any component of $\dot{\boldsymbol{e}}$ orthogonal to $\boldsymbol{e}$ is made negligible. To this end, let

$$\boldsymbol{P}_{\boldsymbol{e}} = \left(\boldsymbol{I}_m - \frac{\boldsymbol{e}\boldsymbol{e}^T}{\boldsymbol{e}^T\boldsymbol{e}}\right) \in \mathbb{R}^{m\times m}$$

be the null-space projector spanning the $(m-1)$-dimensional space orthogonal to vector $\boldsymbol{e}$. Let also

$$\boldsymbol{\delta} = \boldsymbol{P}_{\boldsymbol{e}}\dot{\boldsymbol{e}} = \boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{L}_{\boldsymbol{s}}\boldsymbol{u}. \tag{21}$$

The scalar quantity $\boldsymbol{\delta}^T\boldsymbol{\delta} \geq 0$ provides a measure of the misalignment among the directions of vectors $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ ($\boldsymbol{\delta}^T\boldsymbol{\delta} = 0$ iff $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ are parallel, $\forall \boldsymbol{e} \neq \boldsymbol{0}$, $\dot{\boldsymbol{e}} \neq \boldsymbol{0}$). One can then minimize $\boldsymbol{\delta}^T\boldsymbol{\delta}$, compatibly with the main task (regulation of the error norm), by choosing vector $\boldsymbol{r}$ in (15) as

$$\dot{\boldsymbol{u}}_{\boldsymbol{\delta}} = -k_\delta\nabla_{\boldsymbol{u}}\left(\frac{\boldsymbol{\delta}^T\boldsymbol{\delta}}{2}\right) = -k_\delta\boldsymbol{L}_{\boldsymbol{s}}^T\boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{L}_{\boldsymbol{s}}\boldsymbol{u} = -k_\delta\boldsymbol{J}_\delta^T \tag{22}$$

where $\boldsymbol{J}_\delta = \boldsymbol{u}^T\boldsymbol{L}_{\boldsymbol{s}}^T\boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{L}_{\boldsymbol{s}}$, and the properties $\boldsymbol{P}_{\boldsymbol{e}} = \boldsymbol{P}_{\boldsymbol{e}}^T = \boldsymbol{P}_{\boldsymbol{e}}\boldsymbol{P}_{\boldsymbol{e}}$ were used.

A possible switching strategy, shown in the flowchart in Fig. 2, consists of the following three different control phases:

1. apply the norm controller $\dot{\boldsymbol{u}}_\nu$ given in (15) with the null-space vector $\boldsymbol{r}$ defined in (17) as long as $\nu(t) \geq \nu_T$, with $\nu_T > 0$ being a suitable threshold on the error norm. During this phase, the error norm will be governed by the closed-loop dynamics (18) and the convergence rate in estimating $\hat{\boldsymbol{\chi}}$ will be maximized thanks to (17);

2. when $\nu(t) = \nu_T$, keep applying controller $\dot{\boldsymbol{u}}_\nu$, but replace (17) with (22) in vector $\boldsymbol{r}$. Stay in this phase as long as some terminal condition on the minimization of $\boldsymbol{\delta}^T\boldsymbol{\delta}$ is reached. In our case, we opted for a threshold δ_T on the minimum norm of vector $\|\boldsymbol{P}_\nu\boldsymbol{J}_\delta^T\|$ as an indication of when no further minimization of $\boldsymbol{\delta}^T\boldsymbol{\delta}$ is possible in the null-space of the error norm. Note also that, during this second phase, $\nu(t)$ keeps being governed by the closed-loop dynamics (18) since $\boldsymbol{r}$ acts in the null-space of the error norm (i.e., no distorting effect is produced on the behavior of $\nu(t)$ by the change in $\boldsymbol{r}$);

3. when $\boldsymbol{\delta}^T\boldsymbol{\delta}$ has been minimized, switch to the classical controller $\dot{\boldsymbol{u}}_{\boldsymbol{e}}$ given in (14) until completion of the task. The minimization of $\boldsymbol{\delta}^T\boldsymbol{\delta}$ will ensure a smooth switch as per Prop. 3.2 (and as also demonstrated by the experimental results of Sects. 4 and 6).

Remark 3.3. *We stress again that the main benefit of the proposed switching strategy is to guarantee a* monotonic *decrease of the error norm $\nu(t)$ during all phases, in particular when switching from the norm controller (15) to the classical controller (14). Such a monotonic decrease would not be granted, in general, without a specific action (phase 2) in the flowchart). Guaranteeing a monotonic decrease of the error norm in all conditions is particularly relevant for, e.g., ensuring that the features do not leave the camera fov (since their location will keep on converging towards their desired value) and, in general, avoid erratic*

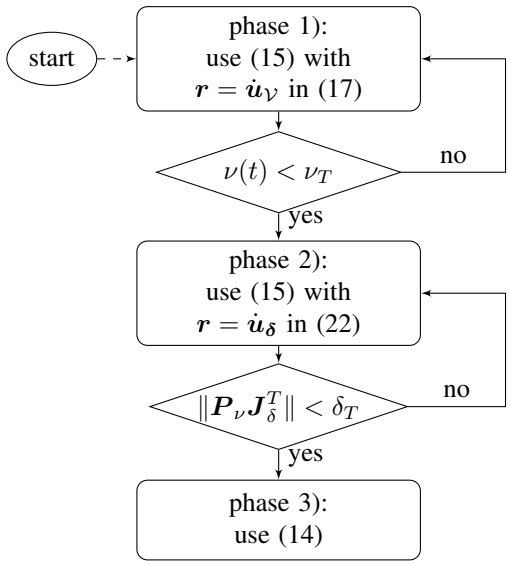


Figure 2. Flowchart representation of the switching strategy.

behaviors of the features on the image plane (that can ease the actual tracking/segmentation of the features themselves).

3.4 Final considerations

We remark that the proposed scheme (active SfM (7) coupled to the second-order visual servoing (14–15), null-space terms (17–22) and associated switching strategy of Fig. 2) only requires, as measured quantities, the visual features $\boldsymbol{s}$ and the camera linear/angular velocity $\boldsymbol{u} = (\boldsymbol{v}, \boldsymbol{\omega})$. Indeed from the estimated $\hat{\boldsymbol{\chi}}$, a (possibly approximated) evaluation of *all* the other quantities entering the various steps of the second-oder control strategy can be obtained from $(\boldsymbol{s}, \hat{\boldsymbol{\chi}})$ and $\boldsymbol{u}$ (the only 'velocity' information actually needed). We also note that the level of approximation is clearly a monotonic function of $\|\boldsymbol{\chi} - \hat{\boldsymbol{\chi}}\|$ (i.e., the uncertainty in knowing $\boldsymbol{\chi}$): thus, all quantities will asymptotically match their real values as the estimation error $\boldsymbol{z}(t) = \boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)$ converges to zero (the faster the convergence of $\boldsymbol{z}(t)$, the sooner the ideal closed-loop behaviors (18–19) will be realized).

Assuming $\|\boldsymbol{\chi} - \hat{\boldsymbol{\chi}}\|$ is small enough, one can also address the stability of the strategy in Fig. 2 in order to show that no undesired effects may arise due to the switching among the different control laws. In particular, it is easy to show that both quantities $\nu(t) = \|\boldsymbol{e}(t)\|$ and $\|\dot{\boldsymbol{e}}(t)\|$ keep bounded during motion and ultimately converge towards zero. First of all, we note that, during all phases, the error norm $\nu(t)$ is governed by the closed-loop dynamics (18) imposing an exponential convergence (with assigned poles). This is obviously the case in phases 1) and 2) (because of the norm controller (15)), and also holds when switching to phase 3) thanks to the previous optimization action of phase 2) (whose role, as explained, is to enforce condition (20) at the switching). Therefore, the error norm $\nu(t)$ will exponentially converge towards zero during all phases.

As for $\|\dot{\boldsymbol{e}}(t)\|$, the norm controller (15) used in phases 1) and 2) guarantees again exponential convergence of $\dot{\nu} = \dot{\boldsymbol{e}}^T\boldsymbol{e}/\|\boldsymbol{e}\|$, that is, of the component of $\dot{\boldsymbol{e}}$ along the direction of $\boldsymbol{e}$. The component of $\dot{\boldsymbol{e}}$ orthogonal to $\boldsymbol{e}$ remains bounded during phase 1) (because of the damping action embedded in (17)), and is afterwards driven to zero during phase 2) by the term (22) (which, indeed, is meant to minimize $\|\boldsymbol{\delta}\| = \|\boldsymbol{P}_{\boldsymbol{e}}\dot{\boldsymbol{e}}\|$). Finally, during phase 3) the closed-loop error behavior is governed by (19) which, clearly, guarantees an exponential convergence of the whole vector $\dot{\boldsymbol{e}}(t)$.

4 Experimental results

This section reports the results of several experiments meant to illustrate the approach proposed so far for coupling the execution of a visual servoing task with the concurrent optimization of the 3-D structure estimation. All experiments were run by making use of a greyscale camera attached to the end-effector of a 6-DOFs Gantry robot. The camera has a resolution of 640×480 px and a framerate of $30\,\mathrm{fps}$. The open-source ViSP library (Marchand et al. 2005) was used to implement all the image processing and feature tracking in order to obtain a measurement of the visual features $\boldsymbol{s}$ at the same frequency. To increase numerical accuracy, the SfM estimator (7) and motion controller internal states were updated with a time step of $1\,\mathrm{ms}$. A simple sample-and-hold filter was then used for $\boldsymbol{s}(t)$, which is only updated at $30\,\mathrm{Hz}$. Finally, the commands were sent to the robot at $100\,\mathrm{Hz}$.

As visual task, we considered the regulation of $N = 4$ point features $\boldsymbol{p}_i$ with, thus, $\boldsymbol{s} = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N) \in \mathbb{R}^m$, and $\boldsymbol{L}_{\boldsymbol{s}} = (\boldsymbol{L}_{\boldsymbol{s}_1}, \ldots, \boldsymbol{L}_{\boldsymbol{s}_N}) \in \mathbb{R}^{m\times 6}$, $m = 8$, with $\boldsymbol{L}_{\boldsymbol{s}_i}$ being the standard 2×6 interaction matrix for a point feature (Chaumette and Hutchinson 2006). We then have $\boldsymbol{\chi} = (\chi_1, \ldots, \chi_N) \in \mathbb{R}^p$, $p = 4$, where $\chi_i = 1/Z_i$ as explained in Sect. 2.2. The tracked points were black non-coplanar dots belonging to the surface of a white cube. A standard pose estimation algorithm was exploited to obtain the ground truth value of $\boldsymbol{\chi}(t)$ from the known object model and the measured $\boldsymbol{s}(t)$.

Because of the high contrast between black dots and white cube surface, the segmentation and tracking of the N points were easily obtained, at video-rate, via the blob tracker available in ViSP. Besides easing the image processing step, this experimental setting also allowed us to reproduce (practically) identical initial experimental conditions across the several trials illustrated in the following sections. The results reported in the next Sect. 6.2 will instead resort to a Lucas-Kanade tracker for

segmenting and tracking a generic set of points lying on a much less structured target object in order to show the viability of our method also in more realistic situations.

As for what concerns the optimization of the 3-D reconstruction, we note that each feature point is characterized by its own (independent) eigenvalue $\sigma^2_{1,i}$. Optimization of the estimation of the whole vector $\boldsymbol{\chi}$ was then obtained by considering the average of the N eigenvalues $\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}\sigma^2_{1,i}$ as quantity to be optimized. Being, obviously,

$$\nabla_{\boldsymbol{u}}\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}\begin{bmatrix}\boldsymbol{J}^T_{\sigma_{v_i}}\\ \boldsymbol{0}\end{bmatrix},$$

the acceleration command (17) was then simply replaced by

$$\dot{\boldsymbol{u}}_{\mathcal{V}} = \frac{k_\sigma\gamma}{\gamma+\sigma^2}\nabla_{\boldsymbol{u}}\sigma^2 - k_d\boldsymbol{u} \tag{23}$$

during phase 1) of all the following experiments.

We invite the reader to watch the accompanying video in Ext. 1.

4.1 First Set of Experiments

In this first set of experiments, we aim at illustrating the benefits arising from the coupling between the execution of a visual servoing task and the concurrent active optimization of the 3-D structure estimation. To this end, we consider the following four different cases, all starting from the same initial conditions:

case 1) the full strategy (three phases) illustrated in Sect. 3 and Fig. 2 is implemented. The estimator (7) is run in parallel to the servoing task for generating the estimated $\hat{\boldsymbol{\chi}}(t)$ fed to all the various control terms. The active optimization of the camera motion (23) takes place for the whole duration of phase 1;

case 2) the classical control law (14) is implemented. The estimator (7) is *still* run in parallel to the servoing task, but no optimization of the estimation error convergence is performed;

case 3) the classical control law (14) is again implemented, but the estimator (7) is *not* run. Vector $\hat{\boldsymbol{\chi}}(t)$ is, instead, taken as $\hat{\boldsymbol{\chi}}(t) = \boldsymbol{\chi}^* = const$, as customary in many visual servoing applications;

case 4) the classical control law (14) is again implemented, but by exploiting knowledge of the ground truth value $\hat{\boldsymbol{\chi}}(t) = \boldsymbol{\chi}(t)$ during the whole servoing execution. This case, then, represents the 'ideal' behavior one could obtain if $\boldsymbol{\chi}(t)$ were available from direct measurement.

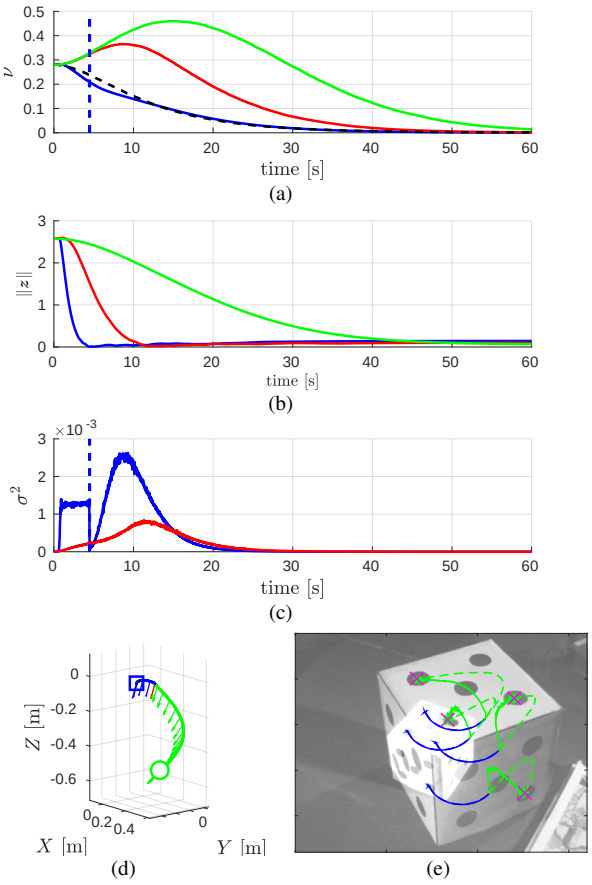


Figure 3. First set of experiments. Fig. (a): behavior of the error norm $\nu(t)$ for case 1 (blue), case 2 (red), case 3 (green) and case 4 (dashed black). Fig. (b): behavior of the norm of the approximation error $\|\boldsymbol{z}(t)\| = \|\boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)\|$ with the same color code. Fig. (c): behavior of $\sigma^2(t)$ when actively optimizing the camera motion (case 1 – blue line) or not performing any optimization (case 2 – red line). In the previous plots, the (practically coincident) vertical dashed blue lines represent the switching times between the various control phases used in case 1. Fig. (d): 3-D camera trajectory during case 1 with arrows representing the camera optical axis and square and circular markers representing the camera initial and final poses respectively. The three phases of Sect. 3.3 are denoted by the following color code: blue – phase 1, red – phase 2, green – phase 3. Fig. (e): trajectory of the four point features in the image plane during case 1 using the same color code, and with crosses indicating the desired feature positions. Superimposed, the initial and final camera images. Finally, solid lines represent the result of implementing phase 2, while dashed lines represent the effects of a direct switch from phase 1 to phase 3.

The following gains and thresholds were used in the experiments: $\alpha = 2000$ in (7), $k_p = 0.0225$ and $k_v = 0.3$ in (14–15). Moreover, only for case 1, we used $k_\sigma = 20$, $\gamma = 0.001$ and $k_d = 18$ in (23), $\nu_T = 0.21$ and $\delta_T = 0.004$ in the flowchart of Fig. 2 and finally $k_\delta = 100$ in (22). Furthermore, in cases 1 and 2, vector $\hat{\boldsymbol{\chi}}$ was initialized as $\hat{\boldsymbol{\chi}}(t_0) = \boldsymbol{\chi}^*$, that is, starting from the (assumed known) value at the desired pose $\boldsymbol{\chi}^*$ also exploited in case 3.

Let us first focus on Fig. 3(b), showing the evolution of the estimation error norm $\|\boldsymbol{z}(t)\| = \|\boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)\|$ for the four cases. From the plots one can note how the use of observer (7), in cases 1–2 (blue and red lines respectively), makes it possible for the estimation/approximation error $\|\boldsymbol{z}(t)\|$ to converge faster than in case 3 (green line), where convergence is reached only at the end of the task, when $\boldsymbol{\chi}(t) \to \hat{\boldsymbol{\chi}} = \boldsymbol{\chi}^*$ (as obvious). Furthermore, the convergence time of $\|\boldsymbol{z}(t)\|$ is almost three times shorter in case 1 (blue line) than in case 2 (red line). Indeed, $\|\boldsymbol{z}(t)\|$ becomes smaller than 5% of its initial value after about $3.5\,\mathrm{s}$ in case 1 w.r.t. $10.2\,\mathrm{s}$ in case 2. This improvement is due to the *active* optimization of the SfM occurring, during phase 1 of case 1, under the action of (23). Indeed, looking at Fig. 3(c), one can note how the value of $\sigma^2(t)$ of case 1 (blue line) is approximately 4 times larger than in case 2 (red line) during the entire phase 1.

The fast convergence of $\|\boldsymbol{z}(t)\| \to 0$ also translates into a fast accurate evaluation of the interaction matrix $\hat{\boldsymbol{L}}_{\boldsymbol{s}}$ and any related quantity. Indeed, from Fig. 3(a), one can notice that the behavior of $\nu(t)$ for case 1 (blue line) (i) quickly reaches a good match with the ideal behavior of case 4 (dashed black line), and (ii), more importantly, keeps *monotonically* decreasing during all the various phases. On the other hand, due to the larger error in estimating $\boldsymbol{\chi}(t)$ (and, hence, evaluating $\hat{\boldsymbol{L}}_s$), both cases 2 (red line) and 3 (green line) present an initial increase of the error norm $\nu(t)$. It is worth noting how this initial divergent phase has, nevertheless, a shorter duration for case 2 w.r.t. case 3 thanks, again, to the use of observer (7).

The camera trajectory, depicted in Fig. 3(d), is also helpful for better understanding the effects of the active optimization of the camera motion during phase 1 of case 1. Note, indeed, how the camera initially moves along an approximately circular path (blue line) because of the null-space term (23) that generates an 'exciting' motion for the estimation of the four point depths Z_i. It is also possible to, again, appreciate the benefits of having employed the norm controller (15) during phase 1: indeed, it is only thanks to the large redundancy granted by controller (15) that the camera is made able to follow a quite 'unusual' trajectory while, *at the same time*, ensuring a convergent behavior for the error norm $\nu(t)$. For completeness, the red line in Fig. 3(d) represents (the quite short) phase 2 of the switching strategy (i.e., the alignment among vectors $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$), while the green line represents phase 3, i.e., the use of the classical controller (14).

As a supplementary evaluation of the theoretical analysis of Sect. 3.3, we now report, for case 1 only, an additional experiment aimed at assessing the importance of having introduced phase 2 in the switching strategy of Sect. 3.3 (i.e., of having enforced the alignment of $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ before switching to the classical controller (14)). To this end,

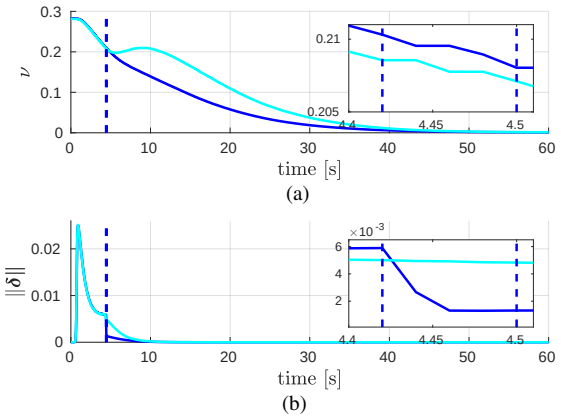


Figure 4. Regulation of 4 point features. Behavior of the error norm $\nu(t)$ (Fig. (a)) and of $\|\boldsymbol{\delta}\|$, the measure of misalignment between vectors $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ (Fig. (b)). In both plots, the blue lines represent the behavior of case 1 (full implementation of the switching strategy of Sect. 3.3), while cyan lines represent the direct switch from phase 1 to phase 3 without the action of vector $\boldsymbol{r}$ in (22). The small picture-in-picture plots provide a zoomed view of the switching phase.

Fig. 4(a) shows the behavior of the error norm $\nu(t)$ for the previous case 1 (blue line) together with the behavior of $\nu(t)$ when *not* implementing phase 2 but, instead, directly switching from phase 1 to phase 3 (cyan line). The two (almost coincident) blue vertical lines represent the switch from phase 1 to phase 2 and then phase 3 for the first experiment, and the direct switch from phase 1 to phase 3 for the second experiment. One can note how, in the second experiment, the error norm $\nu(t)$ has a large overshoot when switching to phase 3 because of the misalignment of vectors $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ at the switching time. This overshoot is, instead, clearly not present in the first experiment where $\nu(t)$ keeps converging during all phases.

A similar overshoot can be observed in Fig. 3(e), where the point feature trajectories on the image, with phase 2 *activated* (solid lines) and *deactivated* (dashed lines), are reported.

Finally, Fig. 4(b) reports the behavior of $\|\boldsymbol{\delta}\|$ from (21), i.e., the measure of misalignment among vectors $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$. One can then verify how, in the first experiment, $\|\boldsymbol{\delta}\|$ is correctly (and very quickly) minimized, during phase 2, thanks to (22).

4.2 Second Set of Experiments

We now discuss a second set of experiments that involve the same four cases 1–4 introduced in the previous section, but with the camera starting from a different initial pose and with a different desired configuration $\boldsymbol{s}^*$ w.r.t. the previous run. The results are reported in Fig. 5.

As compared to Fig. 3, it is worth noting how the sole case 1 (blue line in Fig. 5(a)) results in a successful regulation of the visual task error $\boldsymbol{e}(t)$ thanks, again, to

the fast convergence of the estimation error $\|\boldsymbol{z}(t)\|$ during the active optimization of phase 1 (blue line in Fig. 5(b)). The servoing fails, instead, in case 2 (red line in Fig. 5(a)), i.e., when coupling the classical controller (14) with observer (7) but *without* optimizing for the convergence rate of $\|\boldsymbol{z}(t)\|$. In fact, in this case, the very small value of $\sigma(t)$ during the camera motion (red line in Fig. 3(c)) makes the estimation task ill-conditioned w.r.t. noise and other unmodeled effects (including the disturbance $\boldsymbol{g}(\boldsymbol{z}, t)$ in (8)), resulting in a divergence of the estimation error $\|\boldsymbol{z}(t)\|$ at $t \approx 9\,\mathrm{s}$ (red line in Fig. 5(b)). On the other hand, the active optimization of case 1 is able to increase $\sigma(t)$ by approximately a factor of 40 w.r.t. case 2, thus ensuring a sufficiently high level of excitation for the camera motion and, consequently, a quick convergence of the estimation error $\|\boldsymbol{z}(t)\|$. Failure of the servoing is also obtained in case 3, i.e., when exploiting the *exact* final value $\hat{\boldsymbol{\chi}}(t) = \boldsymbol{\chi}^*$, because of the large initial error of the visual task that causes a loss of feature visibility (green line in Fig. 5(a)).

Finally, Figs. 5(d) and 5(e) depict the camera and feature trajectories during case 1. One can, again, appreciate, in Fig. 5(d), the initial spiralling motion of the camera that allows the increase of $\sigma(t)$ during phase 1. It is also worth noting how, in case 1, the error norm $\nu(t)$ keeps a *monotonic* decrease during the whole motion (as desired) despite the various switches among the three phases and the 'unusual' initial camera trajectory (blue line in Fig. 5(a)).

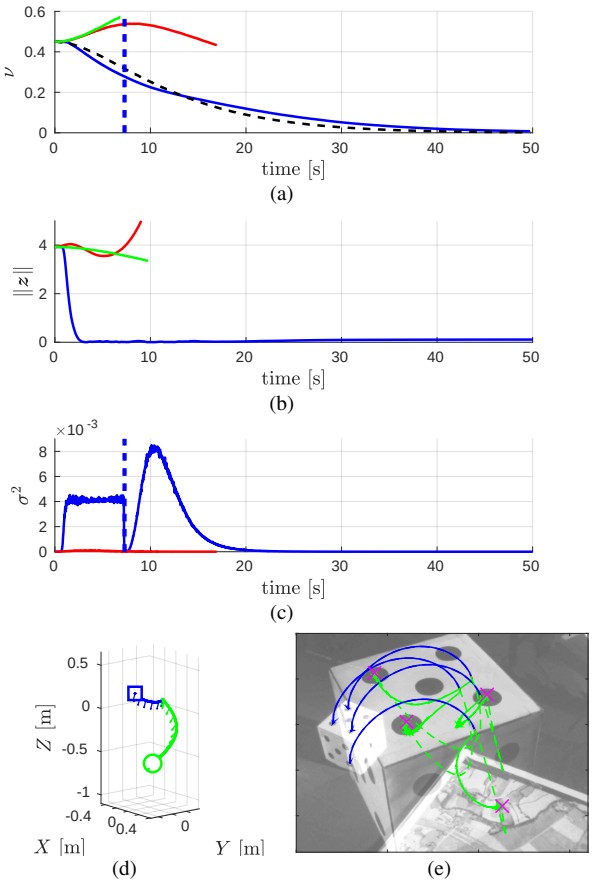


Figure 5. Second set of experiments: regulation of 4 point features starting from a different initial camera pose w.r.t. the experiments in Fig. 3. The plot pattern and color codes are the same as in Fig. 3.

4.3 Third set of Experiments

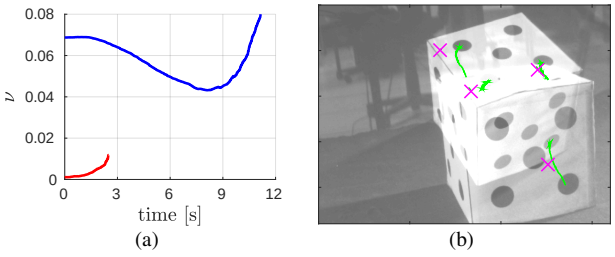
In this last section, we report the results of two experiments meant to show how even relatively small inaccuracies in determining the value $\boldsymbol{\chi}^*$ at the desired pose can cause failure of the servoing when setting $\hat{\boldsymbol{\chi}}(t) = \boldsymbol{\chi}^*$, as classically done in many visual servoing applications. The two experiments presented here involve the same problem considered in Sects. 4.1 and 4.2 (regulation of 4 point features) and differ from the starting location of the camera w.r.t. the target object: in the first experiment, the camera starts (relatively) far from the desired pose while, in the second experiment, the camera starts at almost the desired pose. In both cases, the classical second order control (14) was employed by taking $\hat{\boldsymbol{\chi}} = \boldsymbol{\chi}^*(\mathbf{1} + \boldsymbol{\epsilon})$ with $\boldsymbol{\epsilon} = (-0.0333, 0.09, 0.0424, -0.0875)$ (thus, since $|\epsilon_i| \leq 0.09$, simulating an uncertainty of up to 9% in the accuracy of $\boldsymbol{\chi}^*$).

Figure 6(a) shows the behavior of the error norm $\nu(t)$ for both cases: in the first experiment (blue line), the visual error starts converging from its initial (large) value but then, at about $t \approx 8\,\mathrm{s}$, the servoing diverges and the features leave the camera fov. An even more interesting result is obtained in the second experiment (red line): in this case, the error $\nu(t)$ starts at a very small value since the camera is already quite close to its desired pose. However, controller (14) is not able to impose a stable closed-loop behavior, and the error norm starts diverging until loss of tracking of the feature points at about $t \approx 2.5\,\mathrm{s}$.

These results then provide (for the first time, to the best of our knowledge) an experimental demonstration of the effects discussed in Sect. 2.1 and originally introduced by Malis et al. (2010): a (rather small) error in approximating $\boldsymbol{\chi}^*$ can be sufficient to move part of the eigenvalues of matrix $-\boldsymbol{L}_{\boldsymbol{s}}(\boldsymbol{s}^*, \boldsymbol{\chi}^*)\hat{\boldsymbol{L}}_{\boldsymbol{s}}(\boldsymbol{s}^*, \hat{\boldsymbol{\chi}})^\dagger$ to the right-half complex plane, thus resulting in an unstable closed-loop dynamics even when starting arbitrarily close to the desired pose. This demonstrates, once more, the importance of resorting to an online optimized estimation of $\boldsymbol{\chi}(t)$.

5 Adaptive optimization of the 3-D structure estimation

We now propose a further improvement to the strategy detailed in Sect. 3 and experimentally validated in Sect. 4. The goal is to introduce an *automatic* mechanism for adaptively *activating/deactivating* the optimization of SfM



(a) (b)

Figure 6. Third set of experiments: visual servoing of 4 point features using a constant approximation $\boldsymbol{\chi}(t) = \boldsymbol{\chi}^*$ where the value of $\boldsymbol{\chi}^*$ is corrupted by a relative error of 9%. Fig. (a): behavior of the error norm $\nu(t)$ for the first (blue line) and second (red line) experiments. Fig. (b): image plane trajectory of the 4 point features during the first experiment with crosses indicating their desired positions. The initial and final (i.e. until loss of tracking) camera images are superimposed.

as a function of the accuracy in estimating $\boldsymbol{\chi}(t)$. This modification is motivated by the following considerations w.r.t. Fig. 2 and the previous experimental results:

- the optimization of σ^2 is active during the whole phase 1, i.e., as long as the error norm is larger than some predefined threshold (i.e., $\nu(t) \geq \nu_T$). However, this is obtained at the expense of a possible distortion of the camera trajectory as clear from, e.g., Figs. 3(d) and 5(d) which depict the camera spiralling motion due to action (23) while approaching the final pose. Clearly, a more efficient strategy would implement (23) *only* when strictly needed, e.g., as long as the estimation error $\|\boldsymbol{z}(t)\| = \|\boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)\|$ is larger than some threshold;
- similarly, once in phases 2–3, the flowchart of Fig. 2 does not allow any reactivation of the optimization of σ^2. On the other hand, a reactivation could be necessary in case of unforeseen events such as, e.g., an unpredictable motion of the target that would make the estimation error $\|\boldsymbol{z}(t)\|$ to abruptly increase.

We now detail a modification of the previous strategy of Sect. 3 for addressing these issues. To this end, we first introduce a way to quantify the uncertainty level in the estimation of the unknown vector $\boldsymbol{\chi}(t)$. Since the estimation error $\boldsymbol{z}(t)$ is (obviously) not directly measurable, we consider instead the following *measurable* quantity

$$E(t) = \frac{1}{T}\int_{t-T}^{t} \boldsymbol{\xi}^T(\tau)\boldsymbol{\xi}(\tau)d\tau, \qquad T \geq \epsilon > 0, \tag{24}$$

where T represents the integration window and $\boldsymbol{\xi} = \boldsymbol{s} - \hat{\boldsymbol{s}}$ is the feedback term driving observer (7). Indeed, as discussed in appendix C, $E(t)$ plays a role comparable with the unmeasurable $\boldsymbol{z}(t)$: it provides a measure of the uncertainty of the estimated $\hat{\boldsymbol{\chi}}$ vs. the actual $\boldsymbol{\chi}$. In particular, provided the camera trajectory is sufficiently exciting (i.e., $\sigma_1^2(t) > 0$ during motion), $E(t) \equiv 0$ iff $\|\boldsymbol{z}(t)\| \equiv 0$ (i.e., the estimation has converged) and $E(t) > 0$ otherwise.

One can then leverage knowledge of $E(t)$ for, e.g., (*i*) automatically switching from phase 1 to phase 2 when the estimation error becomes smaller than a desired threshold, (*ii*) automatically switching from phase 3 back to phase 1 when the estimation error grows larger than a desired threshold, and (*iii*) adaptively weighting the first term in action (17) for smoothly activating/deactivating the optimization of σ_1^2.

Let then $0 \leq \underline{E} < \overline{E}$ be a fixed minimum/maximum threshold for $E(t)$ and define

$$k_E(E) : [\underline{E},\, \overline{E}] \mapsto [0,\, 1]. \tag{25}$$

as a monotonically increasing smooth map with $k_E(\underline{E}) = 0$, and $k_E(\overline{E}) = 1$. Function $k_E(E)$ can be exploited for suitably weighting the optimization of σ_1^2 by simply modifying the cost function (16) as

$$\mathcal{V}_E(\boldsymbol{u},\, E) = k_\sigma k_E(E)\gamma \log\left(\frac{\gamma + \sigma_1^2(\boldsymbol{v})}{\gamma}\right) - \frac{k_d}{2}\|\boldsymbol{u}\|^2, \tag{26}$$

resulting in the new optimization action

$$\dot{\boldsymbol{u}}_{\mathcal{V}_E} = \nabla_{\boldsymbol{u}}\mathcal{V}_E = \frac{k_\sigma k_E(E)\gamma}{\gamma + \sigma_1^2}\nabla_{\boldsymbol{u}}\sigma^2 - k_d\boldsymbol{u} \tag{27}$$

to be plugged in vector $\boldsymbol{r}$ in (15). This modification clearly grants a *smooth modulation* of the first term in (27) from a full activation, in case of large estimation inaccuracies ($k_E(E) = 1$ for $E \geq \overline{E}$), to a full deactivation if the estimation is sufficiently accurate ($k_E(E) = 0$ for $E \leq \underline{E}$).

Exploiting $E(t)$ and the modified optimization action given by (27), we propose the new (adaptive) switching strategy depicted in Fig. 7. This consists of the same three phases of Sect. 3.3, but it now exploits knowledge of $E(t)$ for implementing an improved switching policy.

We highlight the following features of this new adaptive strategy: first of all, the initial (possible) switch from phase 3 to phase 1 is performed only if $E(t) \geq \underline{E}$ (the estimation error is large enough for justifying an optimization of the camera motion) *and* $\nu(t) \geq \nu_T$ (the visual error norm is large enough for preventing singularities in (15)). As illustration, two scenarios will typically trigger this switch: (i) a camera starting far enough from the desired pose and with a poor enough initial estimation $\hat{\boldsymbol{\chi}}(t_0)$, or (ii) an unpredicted motion of the target object during the servoing task that causes an increase in the error norm *and* in the estimation uncertainty. The experiments of the next Sect. 6 will indeed address these two practical cases. Furthermore, while in phase 1, the optimization of the SfM will be performed only until either a good enough accuracy has been reached ($E(t) <$

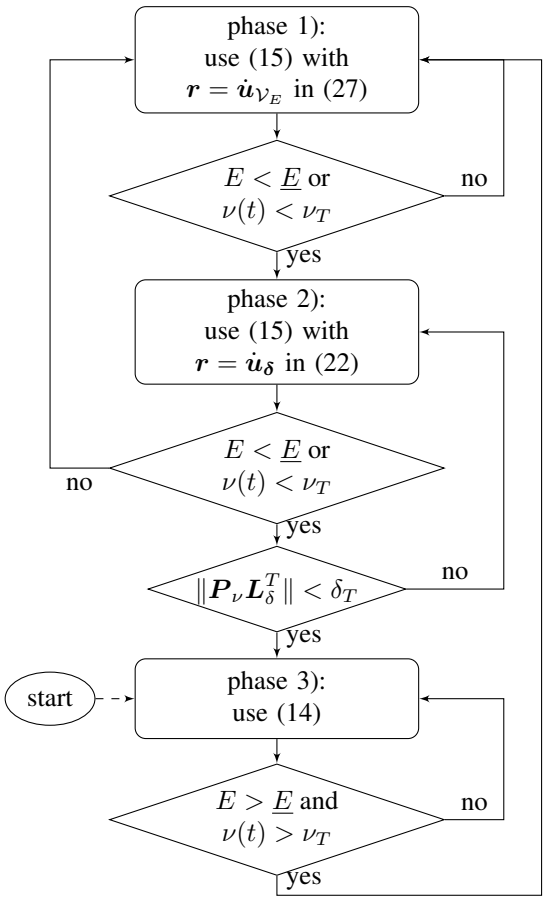


Figure 7. Flowchart representation of the switching strategy exploiting the measurable error energy for triggering changes of status.

$\underline{E}$), or controller (15) is close to become singular ($\nu(t) < \nu_T$). The new switching condition $E(t) < \underline{E}$ will then help in minimizing the distortion of the camera trajectory by allowing a quick switch to phase 2 as soon as the estimation accuracy is satisfactory (see again the experiments in Sect. 6).

As a final step, we comment about the choice of the two thresholds $\underline{E}$ and $\overline{E}$ exploited for triggering the various switches and for modulating the activation of the optimization of σ_1^2 in (27). Assume the range of possible values of $E(t)$ during the camera motion can be lower/upper bounded as $0 \leq E_{min} \leq E(t) \leq E_{max}$. It would obviously be meaningful to choose $\underline{E}$ and $\overline{E}$ such that $E_{min} \leq \underline{E} < \overline{E} \leq E_{max}$ for properly tuning the adaptive switching strategy.

Concerning the lower bound E_{min}, being $E(t) \geq 0$, a straightforward choice would be $E_{min} = 0$. However, presence of measurement noise and other non-idealities can, in practice, prevent $E(t)$ to fall below some minimum value even after convergence of the estimation error (up to some residual noise). If needed, this minimum value can be, e.g., experimentally determined by simply averaging, across a sufficient number of different camera trajectories, the (steady-state) value reached by $E(t)$ once the estimation has converged. This is indeed the solution adopted for the experiments in Sect. 6. As for E_{max}, any (arbitrarily large) positive value would in principle be a valid choice since, the larger the initial approximation error $\|\boldsymbol{z}(t_0)\| = \|\boldsymbol{\chi}(t_0) - \hat{\boldsymbol{\chi}}(t_0)\|$, the wider the possible range of $E(t)$. However, exploiting the properties of observer (7), one can prove (see appendix C) that

$$E(t) \leq \frac{\|\boldsymbol{z}(t_0)\|^2}{\alpha}. \tag{28}$$

Therefore, if an upper bound $\|\boldsymbol{z}(t_0)\| \leq z_{max}$ on the initial estimation error can be assumed (as in most practical situations), one can exploit (28) and set

$$E_{max} = \frac{z_{max}^2}{\alpha}. \tag{29}$$

For the interested reader, this result can be given an interesting energetic interpretation (Spica 2015) as a consequence of the port-Hamiltonian structure of (8).

We conclude with the following remarks: since $E(t) > 0$ as long as the estimation error has not converged, the adaptive gain $k_E(E)$ in (27) is also guaranteed to never vanish during the estimation transient (by properly placing, if needed, the minimum threshold $\underline{E}$). As a consequence, the optimization of the camera motion (i.e., of $\sigma_1^2(t)$) will always be active during phase 1. We also note that, in general, no special characterization is possible for the behavior of $E(t)$. Nevertheless, one can show that, if $\sigma_1^2(t) \approx const > 0$ during motion, then the error system (8) behaves as a second-order critically-damped linear system, with $\boldsymbol{z}(t)$ playing the role of the 'position variables' and $\boldsymbol{\xi}(t)$ that of 'velocity variables', see Spica and Robuffo Giordano (2013). In this situation, $\|\boldsymbol{\xi}(t)\|^2$ (and, thus, $E(t)$ as well) will approximate a 'bell-shaped' profile with a monotonic increase towards a maximum value followed by a monotonic decrease towards zero. Indeed, this is the profile followed by $E(t)$ during the active phases of all the experiments reported in Sect. 6, since maximization of (26) does result (as a byproduct) in $\sigma_1^2(t) \approx const$.

As for the stability during the switching strategy of Fig. 7, considerations analogous to what discussed in Sect. 3.4 hold in this case too. The main differences are the following: in an ideal condition in which $\hat{\boldsymbol{\chi}}(t_0) = \boldsymbol{\chi}(t_0)$, one would have $E(t) \equiv 0$ and, therefore, the system would start and remain in phase 3) during the whole task (by always using the full error controller (14)). If, instead, an initial (large enough) estimation error is present, the quantity $E(t)$ would start increasing, triggering a switch to phase 1). From here on, the same behavior of the previous (non-adaptive) switching strategy is implemented with, thus, a switch to phase 2) followed by phase 3) until

completion of the task. The same would also hold whenever an external 'disturbance' (as, e.g., an unmodeled target motion) occurs, making $E(t)$ to temporarily increase.

6 Experimental results of the adaptive strategy

6.1 First experiment

In this first case study, we considered the same experimental setup of Sect. 4. Vector $\hat{\boldsymbol{\chi}}(t_0)$ was taken coincident with the (assumed known) $\boldsymbol{\chi}^*$ at the final pose, resulting in a bound $\|\boldsymbol{z}(t_0)\|^2/\alpha = 5.3e{-}3$ in (28). As for the adaptive strategy thresholds, we set $\underline{E} = 10^{-5}$ and $\overline{E} = 10^{-4}$.

At the beginning of the motion (phase 3), the eigenvalue σ_1^2 is considerably small due to the low information content of the camera trajectory (Fig. 8(c)) and, analogously to case 2 in Sect. 4.2, the estimation error $\boldsymbol{z}(t)$ even starts increasing because of measurement noise, the disturbance term $\boldsymbol{g}$ in (8), and other non-idealities (Fig. 8(b)). At time $t \approx 1.1\,\mathrm{s}$, however, the quantity $E(t)$ increases over the threshold $\underline{E}$, because of the high uncertainty in the estimated $\hat{\boldsymbol{\chi}}$ (Fig. 8(d)), thus triggering the switch to phase 1 and the corresponding optimization of the camera motion. The optimization action (27) results in a fast increase of the mean eigenvalue $\sigma(t)$ (Fig. 8(c)) and, as a consequence, in a fast convergence of the estimation error $\boldsymbol{z}(t)$ (Fig. 8(b)) that practically vanishes at time $t \approx 4\,\mathrm{s}$. As a consequence, $E(t)$ decreases again below the minimum threshold $\underline{E}$ indicating that a sufficient level of accuracy has been reached. This then triggers the (very quick) switch to phase 2 and, subsequently, the switch back to phase 3 at $t \approx 4.4\,\mathrm{s}$.

Note how the adaptive gain $k_E(E)$, used in (27), correctly (and smoothly) activates and deactivates the optimization of σ^2 during phase 1 as clear from Fig. 8(e).

It is worth noting that the switch from phase 1 to phase 3 occurs when the error norm $\nu(t)$ is still well above the threshold ν_T indicating singularity of controller (15). Therefore, the distortion of the camera trajectory (depicted in Figs. 8(f) and 8(g)), needed to maximize σ^2, lasts considerably less than in the non-adaptive case where the switch would have occurred only at $\nu(t) = \nu_T$. Finally, one can also appreciate how the error norm $\nu(t)$ correctly converges monotonically towards zero once the estimation error $\boldsymbol{z}(t)$ becomes small enough, i.e., for $t \geq 4\,\mathrm{s}$, see Fig. 8(a).

At $t \approx 5.9\,\mathrm{s}$, the target object is purposely displaced causing both the servoing and the estimation error to grow with a corresponding increase of $E(t)$ above the threshold $\underline{E}$. This, in turn, triggers the switch to phase 1 at $t \approx 6.1\,\mathrm{s}$ for (re-)activating the optimization of the camera motion until convergence of the estimation error is, again, reached at $t \approx 9.1\,\mathrm{s}$. The same pattern then repeats two more times at $t \approx 10.6\,\mathrm{s}$ and $t \approx 17.2\,\mathrm{s}$ because of the two

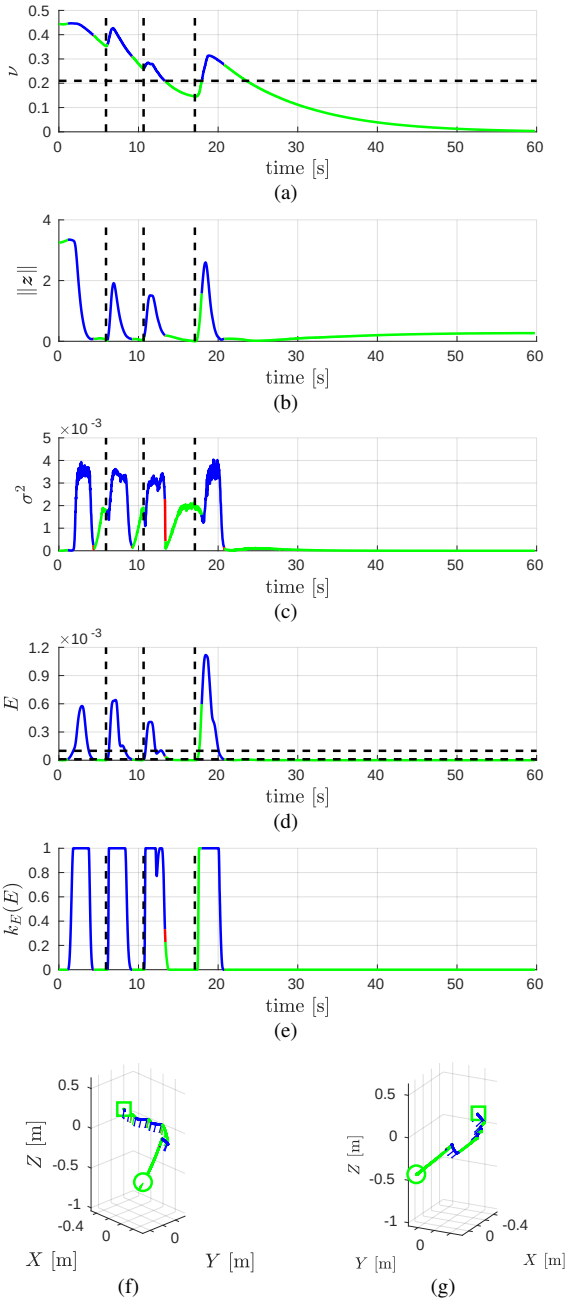


Figure 8. Regulation of 4 point features using the adaptive strategy of Sect. 5. The three phases of Fig. 7 are denoted by the following color code: blue – phase 1, red – phase 2, green – phase 3. Fig. (a): behavior of the error norm $\nu(t)$ with superimposed a horizontal dashed black line indicating the threshold ν_T. Fig. (b): behavior of the norm of the estimation error $\|\boldsymbol{z}(t)\| = \|\boldsymbol{\chi}(t) - \hat{\boldsymbol{\chi}}(t)\|$. Fig. (c): behavior of the mean eigenvalue σ^2. Fig. (d): behavior of $E(t)$ with, superimposed, two dashed horizontal lines indicating the minimum and maximum thresholds $\underline{E}$ and $\overline{E}$. Fig. (e): behavior of the adaptive gain $k_E(E)$. In all of the previous plots, vertical dashed lines represent the times at which the target object was intentionally displaced. Figs. (f) and (g): front and side views of the camera 3-D trajectory with arrows representing the camera optical axis and square and circular markers representing the camera initial and final poses, respectively.

additional displacements of the target object during the camera motion.

As explained in the previous section, the switch from phase 1 to phase 3 (and vice-versa) is also a function of the current value of the error norm $\nu(t)$ for avoiding possible singularities in (15). This is, indeed, the case of the third switch from phase 1 to phase 3 triggered, at $t \approx 13.3\,\mathrm{s}$, by the error norm falling below the threshold ν_T with $E(t)$ still above the minimum value $\underline{E}$. Similarly, the fourth switch from phase 3 to phase 1 at $t \approx 17.9\,\mathrm{s}$ is triggered only when $\nu(t) \geq \nu_T$ even though $E(t)$ has already grown over $\underline{E}$.

By looking at Fig. 8(d), it is finally worth noting how $E(t)$ always keeps below the theoretical bound $\|\boldsymbol{z}(t_0)\|^2/\alpha = 5.3e{-}3$ given in (28) despite the three intentional target displacements occurred during the servoing.

6.2 Second experiment

This last experiment is meant to illustrate the feasibility of our approach in more realistic conditions compared to the use of simple black dots on a white background as done so far. To this end, we considered regulation of 10 point features belonging to a much less structured object, that is, the shrunken piece of textured paper shown in Fig. 9(g) (and in Ext. 1). Extraction and tracking of the 10 features was achieved by exploiting the well-known Lucas-Kanade algorithm implemented in OpenCV. Finally, we made use of the threshold $\underline{E} = 0.0015$ and $\overline{E} = 0.03$, and initialized $\hat{\boldsymbol{\chi}}(t_0) = \boldsymbol{\chi}^*$ as before, with $\|\boldsymbol{z}(t_0)\|^2/\alpha = 6.3e{-}3$ for (28).

Figure 9 reports the results of the experiment: the robot starts, in phase 3, driven by the classical law (14) but, being the mean eigenvalue σ^2 rather small during this phase, the estimation error $\boldsymbol{z}(t)$ does not converge. Likewise, the error norm $\nu(t)$ slightly increases because of the too rough approximation in $\hat{\boldsymbol{\chi}}$. However, the quantity $E(t)$ starts to grow and, at $t \approx 1\,\mathrm{s}$, it exceeds the threshold $\underline{E}$ triggering the switch to phase 1 (Fig. 9(d)). During this phase (which lasts until $t \approx 5\,\mathrm{s}$) the optimization of the camera motion is then able to maximize the eigenvalue σ^2. This results in a quick convergence of the estimation error that practically vanishes at $t \approx 4.5\,\mathrm{s}$. Similarly, the quantity $E(t)$ first reaches a maximum peak value (which is anyway lower than the theoretical bound (28) as expected), and then starts decreasing back to zero thus allowing a smooth deactivation of the optimization action thanks to the adaptive gain k_E (Fig. 9(e)). Finally, at $t \approx 5\,\mathrm{s}$, the error norm $\nu(t)$ falls below the threshold ν_T inducing a quick switch to phase 2 (alignment of $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$) followed by a last switch to phase 3 until completion of the servoing task.

From these results, one can then appreciate how the behavior of the adaptive strategy is essentially equivalent to what obtained in the previous case studies, thus confirming that the proposed approach can be seamlessly applied to more complex/realistic situations.

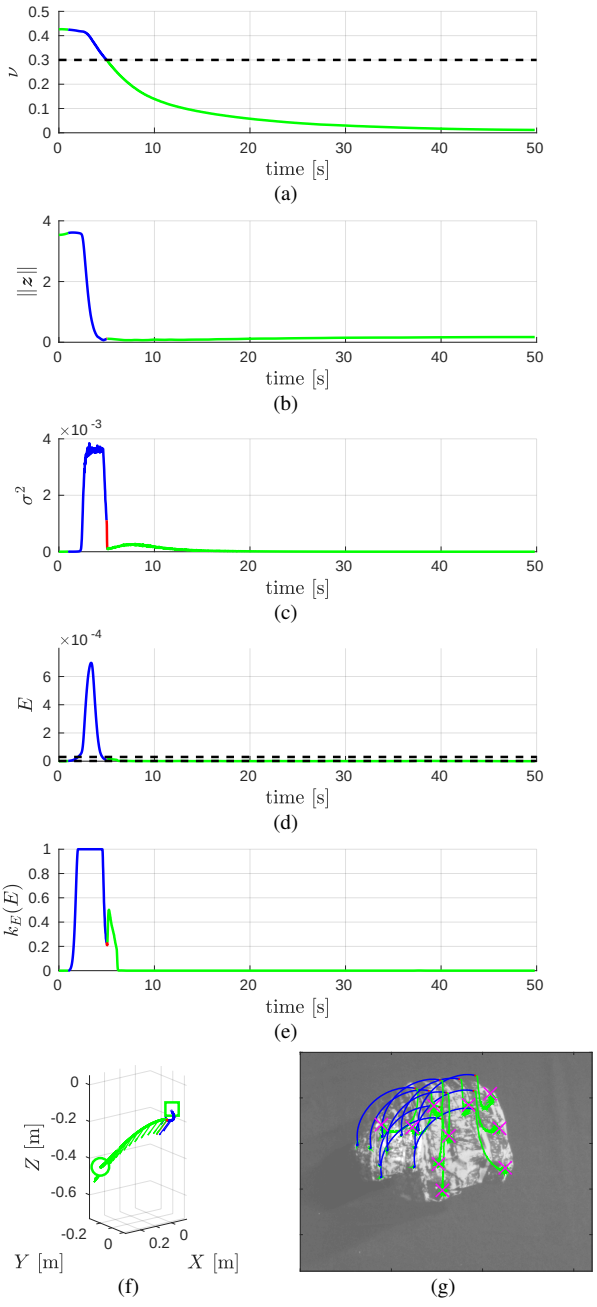


Figure 9. Regulation of 10 point features on an unstructured object using a KLT tracker and the adaptive strategy of Sect. 5. The same quantities of the previous Fig. 8 are reported here with the only exception of Fig. (g) that depicts the trajectory of the 10 point features on the image plane with crosses indicating the desired feature position and, superimposed, two (semi-transparent) camera screenshots taken at the initial and final robot configuration.

7 Conclusions

In this paper we investigated how to couple the execution of a visual servoing task with an active SfM strategy meant to optimize the reconstruction of the 3-D scene structure. This was achieved by projecting the active SfM action within the null-space of the considered IBVS task, and by suitably extending to the second order the framework originally introduced by Marey and Chaumette (2010) for granting the needed redundancy for an effective optimization of the camera motion. A (second-order) switching strategy, meant to avoid some structural singularities of such framework, was also developed and experimentally validated. As an additional contribution, we also detailed an adaptive strategy able to *automatically* activate/deactivate the optimization of the SfM as a function of the current estimation accuracy.

The reported experimental campaign clearly showed the benefits of the approach in terms of: (*i*) obtaining a faster convergence of the structure estimation error during the servoing transient w.r.t. non-active cases, (*ii*) imposing an improved closed-loop IBVS behavior by significantly mitigating the negative effects of an inaccurate knowledge of the scene structure, (*iii*) minimizing the deformation of the camera trajectory (consequence of the active SfM action) thanks to the adaptive activation/deactivation of the SfM optimization.

Despite the successful results, however, the proposed coupling between visual control and active perception has still a number of open points that deserve further developments. To start with, due to the nonlinear nature of the system dynamics, stability of each individual estimation/control block does not imply, in general, stability of their composition (the separation principle is only valid for linear time-invariant systems). While the proposed experimental results show a promising level of robustness in this sense, a more formal characterization of the convergence domain is yet to be found.

As discussed in Remark 3.3, guaranteeing a monotonic decrease of the visual error norm can help avoiding erratic behaviors of the features on the image plane. However, this may not be sufficient to ensure that the features will not leave the camera fov in all possible situations (e.g., when the desired feature location is close to the image plane borders). Similarly, other typical 'feasibility' constraints (such as joint limits or collision avoidance) were also ignored in the proposed strategy. These issues could be addressed by considering the observability maximization as an additional task in a multi-objective constrained optimization problem. This latter could then be resolved *locally* by exploiting one of the several prioritized multi-task resolution frameworks proposed in the literature (see, e.g., Escande et al. (2014); Flacco et al. (2015)). As well known, however, local optimization strategies (like the one proposed in this work and most IBVS schemes) can be prone to local minima and generate trajectories with sub-optimal observability properties. In this regard, introducing a planning phase over an extended time horizon could be beneficial also for what concerns a better handling of the visibility constraint (see, e.g., Chesi and Vicino (2004) for an example in this sense).

Finally, we also plan to apply our machinery to mobile (ground/flying) robots, equipped with onboard cameras, and possibly subject to non-holonomic constraints.

A Index to multimedia extensions

Extension	Media type	Description
1	Video	Video of the esperiments.

B Proof of Prop. 3.2

Let $\mathbf{\Phi}(t) = [\Phi_{ij}(t)] \in \mathbb{R}^{2\times 2}$ be the state-transition matrix associated to the linear time-invariant system (18). From classical system theory (Kailath 1998), we have

$$\nu_{\|\boldsymbol{e}\|}(t) = \Phi_{11}(t-t_1)\nu_1 + \Phi_{12}(t-t_1)\dot{\nu}_1, \quad \forall t \geq t_1, \tag{30}$$

where we set $\nu_1 = \nu(t_1)$ and $\dot{\nu}_1 = \dot{\nu}(t_1)$ for simplicity. We also note that (19) is governed, component-wise, by the same dynamics of (18). Therefore, the solution of (19) is

$$\boldsymbol{e}^*(t) = \Phi_{11}(t-t_1)\boldsymbol{e}_1 + \Phi_{12}(t-t_1)\dot{\boldsymbol{e}}_1, \quad \forall t \geq t_1, \tag{31}$$

where, again, $\boldsymbol{e}_1 = \boldsymbol{e}(t_1)$ and $\dot{\boldsymbol{e}}_1 = \dot{\boldsymbol{e}}(t_1)$.

If $\boldsymbol{e}_1$ and $\dot{\boldsymbol{e}}_1$ are parallel then (20) holds: assuming $\boldsymbol{e}_1$ and $\dot{\boldsymbol{e}}_1$ are parallel, vector $\dot{\boldsymbol{e}}_1$ can be expressed as

$$\dot{\boldsymbol{e}}_1 = \|\dot{\boldsymbol{e}}_1\| \frac{\boldsymbol{e}_1}{\|\boldsymbol{e}_1\|} = \|\dot{\boldsymbol{e}}_1\| \frac{\boldsymbol{e}_1}{\nu_1}. \tag{32}$$

Therefore, (31) becomes

$$\boldsymbol{e}^*(t) = \left(\Phi_{11}(t-t_1) + \Phi_{12}(t-t_1)\frac{\|\dot{\boldsymbol{e}}_1\|}{\nu_1}\right)\boldsymbol{e}_1, \quad \forall t \geq t_1, \tag{33}$$

resulting in an error norm $\|\boldsymbol{e}^*(t)\|$

$$\begin{aligned}\|\boldsymbol{e}^*(t)\| = \nu^*(t) &= \left(\Phi_{11}(t-t_1) + \Phi_{12}(t-t_1)\frac{\|\dot{\boldsymbol{e}}_1\|}{\nu_1}\right)\|\boldsymbol{e}_1\| \\ &= \left(\Phi_{11}(t-t_1) + \Phi_{12}(t-t_1)\frac{\|\dot{\boldsymbol{e}}_1\|}{\nu_1}\right)\nu_1 \\ &= \Phi_{11}(t-t_1)\nu_1 + \Phi_{12}(t-t_1)\|\dot{\boldsymbol{e}}_1\|, \quad \forall t \geq t_1.\end{aligned} \tag{34}$$

Now, being $\nu = \|\boldsymbol{e}\|$ one has

$$\dot{\nu}_1 = \frac{\boldsymbol{e}_1^T\dot{\boldsymbol{e}}_1}{\nu_1} \tag{35}$$

which, exploiting (32), yields $\dot{\nu}_1 = \|\dot{\boldsymbol{e}}_1\| \boldsymbol{e}_1^T \boldsymbol{e}_1 / \nu_1^2 = \|\dot{\boldsymbol{e}}_1\|$. Plugging $\|\dot{\boldsymbol{e}}_1\| = \dot{\nu}_1$ in (34) finally results in

$$\nu^*(t) = \Phi_{11}(t - t_1)\nu_1 + \Phi_{12}(t - t_1)\dot{\nu}_1, \quad \forall t \geq t_1,$$

thus showing that $\nu^*(t) \equiv \nu_{\|\boldsymbol{e}\|}(t)$, i.e. fulfillment of condition (20) as claimed.

If (20) holds then $\boldsymbol{e}_1$ and $\dot{\boldsymbol{e}}_1$ are parallel: from (30–31) we have (omitting the time dependency for brevity)

$$\nu_{\|\boldsymbol{e}\|}^2 = \Phi_{11}^2 \nu_1^2 + 2\Phi_{11}\Phi_{12}\nu_1\dot{\nu}_1 + \Phi_{12}^2 \dot{\nu}_1^2 \tag{36}$$

and

$$\begin{aligned} \|\boldsymbol{e}^*(t)\|^2 =& \Phi_{11}^2 \boldsymbol{e}_1^T \boldsymbol{e}_1 + 2\Phi_{11}\Phi_{12}\boldsymbol{e}_1^T \dot{\boldsymbol{e}}_1 + \Phi_{12}^2 \dot{\boldsymbol{e}}_1^T \dot{\boldsymbol{e}}_1 \\ =& \Phi_{11}^2 \nu_1^2 + 2\Phi_{11}\Phi_{12}\nu_1\dot{\nu}_1 + \Phi_{12}^2 \dot{\boldsymbol{e}}_1^T \dot{\boldsymbol{e}}_1 \end{aligned} \tag{37}$$

where (35) was used. By imposing condition (20) to (36–37) we then have

$$\nu_{\|\boldsymbol{e}\|}^2 \equiv \|\boldsymbol{e}^*(t)\|^2 \implies \Phi_{12}^2 \dot{\nu}_1^2 \equiv \Phi_{12}^2 \dot{\boldsymbol{e}}_1^T \dot{\boldsymbol{e}}_1 \implies \dot{\nu}_1 = \|\dot{\boldsymbol{e}}_1\|. \tag{38}$$

Since $\dot{\nu}_1$ is just the projection of vector $\dot{\boldsymbol{e}}_1$ along the direction of $\boldsymbol{e}_1$ (see again (35)), condition (38) necessarily requires vectors $\boldsymbol{e}_1$ and $\dot{\boldsymbol{e}}_1$ to be parallel as claimed.

C Properties of $E(t)$

Relationship between $E(t)$ and the estimation error $\boldsymbol{z}(t)$: if $\sigma_1^2(t) > 0$ during the camera motion then $E(t) \equiv 0$ iff $\|\boldsymbol{z}(t)\| \equiv 0$ (i.e., the estimation has converged) and $E(t) > 0$ otherwise (i.e., the estimation has not yet converged).

In order to prove this claim, we start by showing the following facts:

Proposition C.1. *If the camera motion is exciting (i.e., $\sigma_1^2(t) > 0$), then $\|\boldsymbol{\xi}(t)\| \equiv 0 \iff \|\boldsymbol{z}(t)\| \equiv 0$ and $\|\boldsymbol{\xi}(t)\| > 0$ a.e. $\iff \|\boldsymbol{z}(t)\| > 0$ a.e.*

Proof. Being σ_1^2 the smallest eigenvalue of matrix $\boldsymbol{\Omega}\boldsymbol{\Omega}^T$, the hypothesis $\sigma_1^2 > 0$ implies full row-rankness of the (low-rectangular) $p \times m$ matrix $\boldsymbol{\Omega}$. Considering now the error dynamics (8), the following holds

- $\|\boldsymbol{\xi}(t)\| \equiv 0 \implies \|\boldsymbol{z}(t)\| \equiv 0$: if $\|\boldsymbol{\xi}(t)\| \equiv 0$ then $\boldsymbol{\xi}(t) \equiv \boldsymbol{0}$ and $\dot{\boldsymbol{\xi}}(t) \equiv \boldsymbol{0}$. The first row of (8) then reduces to $\boldsymbol{\Omega}^T \boldsymbol{z} \equiv \boldsymbol{0}$ which implies $\|\boldsymbol{z}(t)\| \equiv 0$ since matrix $\boldsymbol{\Omega}$ is full row-rank by hypothesis;
- $\|\boldsymbol{z}(t)\| \equiv 0 \implies \|\boldsymbol{\xi}(t)\| \equiv 0$: if $\|\boldsymbol{z}(t)\| \equiv 0$, the first row of (8) reduces to $\dot{\boldsymbol{\xi}} = -\boldsymbol{H}\boldsymbol{\xi}$. Being the matrix gain $\boldsymbol{H}$ positive definite, it follows that, at steady-state, the only possible solution is $\boldsymbol{\xi}(t) \equiv \boldsymbol{0}$.

These two implications then prove the first item of the Proposition, that is, $\|\boldsymbol{\xi}(t)\| \equiv 0 \iff \|\boldsymbol{z}(t)\| \equiv 0$. The proof is concluded by noting that the remaining two (reverse) implications $\|\boldsymbol{z}(t)\| > 0$ a.e. $\implies \|\boldsymbol{\xi}(t)\| > 0$ a.e. and $\|\boldsymbol{\xi}(t)\| > 0$ a.e. $\implies \|\boldsymbol{z}(t)\| > 0$ a.e. (needed for proving the second item of the Proposition) are just the logical negations of the two ones listed above.

Prop. C.1 can now be exploited for proving the initial main claim. Indeed, since $E(t)$ is defined as the moving average of signal $\|\boldsymbol{\xi}(t)\|^2$ (see (24)), it follows that $E(t) = 0$ if $\|\boldsymbol{z}(t)\| \equiv 0$ over (at least) the integration window T. Therefore, convergence of the estimation error $\boldsymbol{z}(t)$ will necessarily make the quantity $E(t)$ vanish as desired. On the other hand, if $\|\boldsymbol{z}(t)\| > 0$ a.e. $\implies \|\boldsymbol{\xi}(t)\| > 0$ a.e., the moving average (24) over any non-infinitesimal integration window $T \geq \epsilon > 0$ will necessarily stay positive, thus implying that $E(t) > 0$ q.e.d..

Proof of bound (28): this bound can be easily proven by exploiting the port-Hamiltonian interpretation of the error dynamics (8) briefly introduced in Sect. 2.2. With reference to Spica and Robuffo Giordano (2013) (where a full analysis can be found), it is indeed possible to show that the Hamiltonian function (9) decreases over time towards its global minimum at $(\boldsymbol{\xi}, \boldsymbol{z}) = (\boldsymbol{0}, \boldsymbol{0})$, provided the usual hypothesis of an exciting camera motion ($\sigma_1^2(t) > 0$) is satisfied. Therefore, along the trajectories of (8) it is

$$0 \leq \mathcal{H}(\boldsymbol{\xi}(t), \boldsymbol{z}(t)) \leq \mathcal{H}(\boldsymbol{\xi}(t_0), \boldsymbol{z}(t_0)), \qquad \forall t \geq t_0. \tag{39}$$

We now note that, being the feature vector $\boldsymbol{s}$ a measurable quantity, one can *always* initialize $\hat{\boldsymbol{s}}(t_0) = \boldsymbol{s}(t_0)$ resulting in $\boldsymbol{\xi}(t_0) = \boldsymbol{0}$. By employing this initialization (adopted in all the reported case studies), and exploiting (9–39), the following bound easily follows

$$\frac{1}{2}\|\boldsymbol{\xi}(t)\|^2 \leq \mathcal{H}(\boldsymbol{\xi}(t), \boldsymbol{z}(t)) \leq \mathcal{H}(\boldsymbol{\xi}(t_0), \boldsymbol{z}(t_0)) = \frac{1}{2\alpha}\|\boldsymbol{z}(t_0)\|^2. \tag{40}$$

The proof is completed by noting that, from standard calculus,

$$E(t) = \frac{1}{T}\int_{t-T}^{t} \boldsymbol{\xi}^T(\tau)\boldsymbol{\xi}(\tau)\mathrm{d}\tau \leq \max_{\tau \in [t-T,\, t]} \left[\boldsymbol{\xi}^T(\tau)\boldsymbol{\xi}(\tau)\right] \leq \frac{\|\boldsymbol{z}(t_0)\|^2}{\alpha}. \tag{41}$$

We conclude by noting that (9) (and, consequently, (40–41)) is no longer valid in presence of (unmodeled) perturbations such as the several target displacements discussed in Sect. 6.1. In this case, an external amount of energy could (in general) be injected into system (8) with a consequent increase of the total energy $\mathcal{H}(t)$ and a possible violation of bound (39).

References

Achtelik MW, Weiss S, Chli M and Siegwart R (2013) Path planning for motion dependent state estimation on micro aerial vehicles. In: *2013 IEEE Int. Conf. on Robotics and Automation*. Karlsruhe, Germany, pp. 3926–3932.

Chaumette F (2004) Image moments: a general and useful set of features for visual servoing. *IEEE Trans. on Robotics* 20(4): 713–723.

Chaumette F and Hutchinson SA (2006) Visual servo control, part I: Basic approaches. *IEEE Robotics & Automation Mag.* 13(4): 82–90.

Chen S, Li Y and Kwok NM (2011) Active vision in robotic systems: A survey of recent developments. *Int. J. of Robotics Research* 30(11): 1343–1377.

Chesi G and Hashimoto K (2004) A simple technique for improving camera displacement estimation in eye-in-hand visual servoing. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 26(9): 1239–1242.

Chesi G and Vicino A (2004) Visual servoing for large camera displacements. *IEEE Trans. on Robotics* 20(4): 724–735.

Chwa D, Dani AP and Dixon WE (2016) Range and motion estimation of a monocular camera using static and moving objects. *IEEE Trans. on Control Systems Technology* 24(4): 1174–1183.

Corke P (2010) Spherical image-based visual servo and structure estimation. In: *2010 IEEE Int. Conf. on Robotics and Automation*. pp. 5550–5555.

Cristofaro A and Martinelli A (2010) Optimal trajectories for multi robot localization. In: *49th IEEE Conf. on Decision and Control*. Atlanta, GA, pp. 6358–6364.

De Luca A, Oriolo G and Robuffo Giordano P (2008) Feature depth observation for image-based visual servoing: Theory and experiments. *Int. J. of Robotics Research* 27(10): 1093–1116.

Escande A, Mansard N and Wieber PB (2014) Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *Int. J. of Robotics Research* 33(7): 1006–1028.

Espiau B, Chaumette F and Rives P (1992) A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation* 8(3): 313–326.

Eudes A, Morin P, Mahony R and Hamel T (2013) Visuo-inertial fusion for homography-based filtering and estimation. In: *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. pp. 5186–5192.

Flacco F, De Luca A and Khatib O (2015) Control of redundant robots under hard joint constraints: Saturation in the null space. *IEEE Trans. on Robotics* 31(3): 637–654.

Fujita M, Kawai H and Spong MW (2007) Passivity-based dynamic visual feedback control for three-dimensional target tracking: Stability and L_2-gain performance analysis. *IEEE Trans. on Control Systems Technology* 15(1): 40–52.

Gans N and Hutchinson SA (2007) Stable visual servoing through hybrid switched-system control. *IEEE Trans. on Robotics* 3(23): 530–540.

Grabe V, Bülthoff HH and Robuffo Giordano P (2013) A comparison of scale estimation schemes for a quadrotor UAV based on optical flow and IMU measurements. In: *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Tokyo, Japan, pp. 5193–5200.

Hollinger GA and Sukhatme GS (2014) Sampling-based robotic information gathering algorithms. *Int. J. of Robotics Research* 33(9): 1271–1287.

Kailath T (1998) *Linear Systems*. Prentice Hall International. ISBN 9789814024785.

Ma Y, Soatto S, Kosecka J and Sastry S (2003) *An invitation to 3D vision*. Springer. ISBN 0-387-00893-4.

Mahony R and Stramigioli S (2012) A port-Hamiltonian approach to image-based visual servo control for dynamic systems. *Int. J. of Robotics Research* 31(11): 1303–1319.

Malis E and Chaumette F (2002) Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods. *IEEE Trans. on Robotics* 18(2): 176–186.

Malis E, Hamel T, Mahony R and Morin P (2009) Dynamic estimation of homography transformations on the special linear group for visual servo control. In: *2009 IEEE Int. Conf. on Robotics and Automation*. pp. 1498–1503.

Malis E, Mezouar Y and Rives P (2010) Robustness of image-based visual servoing with a calibrated camera in the presence of uncertainties in the three-dimensional structure. *IEEE Trans. on Robotics* 26(1): 112–120.

Marchand E, Spindler F and Chaumette F (2005) ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics & Automation Mag.* 12(4): 40–52.

Marey M and Chaumette F (2010) A new large projection operator for the redundancy framework. In: *2010 IEEE Int. Conf. on Robotics and Automation*. Anchorage, AK, pp. 3727–3732.

Martinelli A (2012) Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. *IEEE Trans. on Robotics* 1(28): 44–60.

Mebarki R, Lippiello V and Siciliano B (2015) Nonlinear visual control of unmanned aerial vehicles in GPS-denied environments. *IEEE Trans. on Robotics* 31(4): 1004–1017.

Miller LM, Silverman Y, MacIver MA and Murphey TD (2016) Ergodic exploration of distributed information. *IEEE Trans. on Robotics* 32(1): 36–52.

Petiteville A, Courdesses M, Cadenat V and Baillion P (2010) On-line estimation of the reference visual features application to a vision based long range navigation task. In: *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. pp. 3925–3930.

Siciliano B, Sciavicco L, Villani L and Oriolo G (2009) *Robotics: modelling, planning and control*. Springer.

Spica R (2015) *Contributions to Active Visual Estimation and Control of Robotic Systems*. PhD Thesis, Université de Rennes 1.

Spica R and Robuffo Giordano P (2013) A framework for active estimation: Application to structure from motion. In: *52nd IEEE Conf. on Decision and Control*. Florence, Italy, pp. 7647–7653.

Spica R, Robuffo Giordano P and Chaumette F (2014a) Active structure from motion: Application to point, sphere and cylinder. *IEEE Trans. on Robotics* 30(6): 1499–1513.

Spica R, Robuffo Giordano P and Chaumette F (2014b) Coupling visual servoing with active structure from motion. In: *2014 IEEE Int. Conf. on Robotics and Automation*. Hong Kong, China, pp. 3090–3095.

Spica R, Robuffo Giordano P and Chaumette F (2015) Plane estimation by active vision from point features and image moments. In: *2015 IEEE Int. Conf. on Robotics and Automation*. Seattle, WA, pp. 6003–6010.

Tahri O and Chaumette F (2005) Point-based and region-based image moments for visual servoing of planar objects. *IEEE Trans. on Robotics* 21(6): 1116–1127.

Valente L, Tsai RYH and Soatto S (2012) Information gathering control via exploratory path planning. In: *2012 46th IEEE Annual Conf. on Information Sciences and Systems*. Princeton, NJ.

Whaite P and Ferrie FP (1997) Autonomous exploration: Driven by uncertainty. *IEEE Trans. on Pattern Analysis & Machine Intelligence* 19(3): 193–205.

Wilson AD, Schultz JA and Murphey TD (2014) Trajectory synthesis for Fisher information maximization. *IEEE Trans. on Robotics* 30(6): 1358–1370.