2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Transformation-Domain Gaussian Smoothing for Translational Direct Visual Servoing

Amneh Nasir[1,2], Djemaa Kachi[2], Antoine N. André[1], Guillaume Caron[1,2]

***Abstract*—Direct visual servoing (DVS) uses raw pixel intensities to control robot motion, yielding high accuracy at convergence. However, the associated photometric cost function is highly nonconvex, which leads to a narrow domain of convergence due to local minima. This work addresses that issue by adapting a Gaussian homotopy framework for cost function smoothing from cross-correlation to the sum of squared differences (SSD) objective used in DVS. The result is a spatially varying, transformation-domain kernel that depends on the motion model, producing smoother cost landscapes and enlarging the convergence basin. We first apply the smoothing to an SSD cost, derive its corresponding transformation kernel for the motion model in the camera domain, and then incorporate it into a DVS control law. The method is compared against uniform image domain blurring via Photometric Gaussian Mixtures. Experiments with an eye-in-hand robotic arm setup over three degrees of freedom translation and with different initial poses show that cost smoothing significantly increases the convergence domain while preserving the accuracy of DVS.**

***Index Terms*—Direct visual servoing, cost function smoothing, robotic vision, Gaussian homotopy.**

I. Introduction

Visual Servoing (VS) is a control methodology for robot motion using visual feedback, classically divided into pose-based and image-based approaches [1]. In image-based visual servoing (IBVS), the control law minimizes an error defined either from visual features (*e.g.*, points or lines) or, as in direct visual servoing (DVS), from raw pixel intensities over the entire image. DVS does not require feature extraction and matching, achieving accurate final positioning without intermediate geometric processing [2]. DVS formulates visual servoing as a direct photometric optimization problem defined over raw pixel intensities [3]. However, this comes with a highly nonconvex photometric cost landscape, typically the sum of squared differences (SSD) between the current and desired images, which results in a narrow convergence domain [4].

To mitigate this limitation, several approaches aim to reshape the optimization landscape, either by modifying the visual representation or by altering the structure of the cost function. Global image representations based on discrete orthogonal moments have been shown to reduce local ambiguities and enlarge the convergence domain of DVS under large displacements [5]. Recent theoretical analyses confirm that the extent of the convergence domain is directly linked to the presence of blur, which effectively enlarges the region of attraction [6]; this behavior is also demonstrated in direct visual servoing using nonlinear scale-space smoothing [7]. In practice, DVS often converges only when the camera is already close to the desired pose; larger pose errors create complex, multimodal cost surfaces with many local minima that can trap the optimization. The purpose of this paper is therefore to develop a smoothed SSD cost function to enlarge the convergence basin of DVS.

A classical approach in image alignment is the coarse-to-fine image pyramid introduced by Lucas and Kanade [8], where images are successively downsampled and low-pass filtered to create a series of simpler optimization problems as the resolution increases. This coarse-to-fine continuation method works well for planar translation models, but its effectiveness is limited for complex 3D motions and large viewpoint changes as it relies on a first-order Taylor expansion, linearizing nonlinear transformations, ideally around the optimum.

Methods based on function smoothing belong to the family of continuation approaches for nonconvex optimization, where the objective is initially smoothed to produce a simplified cost landscape with fewer local extrema and then progressively refined until the objective recovers the original cost [9]. From a control perspective, this suppresses narrow, geometry-induced local minima that can trap the optimization when the camera is far from the desired pose. As the smoothing is gradually reduced, the original cost landscape is recovered, enabling accurate and reliable convergence.

A smoothing method based on graduated nonconvexity was analytically formulated by Mobahi et al. [10]. In this approach, a smoother version of the photometric cross-correlation (CC) cost landscape is constructed by convolving it with an isotropic Gaussian kernel, leading to transformation-specific kernels that smooth the alignment cost for a given parametric motion model. This idea has been further formalized by showing that Gaussian smoothing corresponds to the best affine approximation to the convex envelope of a nonconvex function, thus providing an optimal convexification strategy [11]. The concept of cost function smoothing has also been explored in visual servoing via defocus-based DVS [12], which demonstrates that optical image blurring can enlarge the convergence domain without additional image processing.

Other influential methods have improved DVS robustness using Gaussian mixture or radial-basis-function representa-

[1]Joint Robotics Laboratory (JRL), CNRS–AIST, Tsukuba, Japan.
[2]MIS Laboratory, Université de Picardie Jules Verne, Amiens, France.

tions to smooth the cost function. In the Photometric Gaussian Mixtures (PGM) framework introduced by Crombez et al. [13], each pixel's contribution is modeled as a 2D Gaussian basis function rather than a point intensity, effectively blurring the desired and current images. The amount of blur is controlled by an adjustable Gaussian spread that can be tuned during servoing. By dynamically adjusting this spread, PGM VS achieves convergence from significantly larger initial pose errors than classical direct visual servoing with pointwise pixel intensities.

In parallel, kernel-based visual servoing methods have been proposed, where spatial sampling kernels unify tracking and control within a Lyapunov framework, providing formal stability guarantees for basic motions such as translation and roll [14]. Unlike our work, which applies Gaussian kernels to smooth the cost function itself, these approaches use kernels as measurement functions to design stable controllers.

The contributions of this work are as follows.

- Adaptation of the Gaussian cost smoothing framework [10] from CC to the SSD objective used in DVS.
- Closed-form derivation of the transformation-domain Gaussian kernel and interaction matrix for 3-DoF translational DVS for a motion-adaptive control law.
- Experimental validation on a 3-DoF UR5, showing wider basins and larger convergence ranges than PGM VS under matching settings.

The remainder of the paper is organized as follows. Section II reviews related approaches for enlarging the convergence domain. Section III presents the theoretical foundation of the proposed method. Section IV describes the experimental setup for the robotic visual servoing tasks. Section V reports results and discusses improvements in convergence. Finally, Section VI concludes with perspectives for future work.

II. Background

A. Image-Domain Smoothing

Image smoothing is a common strategy to attenuate local minima in image alignment and visual servoing (VS). Traditional direct alignment methods often employ multiscale Gaussian blurring to enlarge the convergence basin by suppressing high-frequency details that can induce spurious local minima. This approach is motion-model agnostic, as the same uniform blur is applied regardless of whether the camera motion is a pure translation, a rotation, or a more complex transform. In other words, it is a one-size-fits-all blur. Because the smoothing is applied in the image domain, it cannot account for the specific geometry of different motion models, which can leave residual local minima along certain directions.

B. GRBF Image Representation

One uniform smoothing strategy represents the image intensity function with a Gaussian radial basis function (GRBF). Let $\mathcal{X} \subset \mathbb{R}^n$ with $n = 2$ be the image domain, $\{\boldsymbol{x}_i\}_{i=1}^N \subset \mathcal{X}$ the normalized coordinates of the Gaussian centers, N the number of pixels, $a_i \in \mathbb{R}$ their intensities, and $\delta \in \mathbb{R}^+$ the Gaussian spread. For any normalized image coordinate $\boldsymbol{x} \in \mathcal{X}$:

$$\begin{aligned} \tilde{f}(\boldsymbol{x}, \delta) &= \sum_{i=1}^{N} a_i \, k(\boldsymbol{x} - \boldsymbol{x}_i), \\ k(\boldsymbol{x}) &= \frac{1}{2\pi\,\delta^2} \exp\Big(-\tfrac{\|\boldsymbol{x}\|^2}{2\delta^2}\Big). \end{aligned} \tag{1}$$

This representation yields a uniformly blurred image $\tilde{f}$ whose effective smoothing is isotropic. The GRBF representation is equivalent to convolving the image with a Gaussian kernel of standard deviation δ in the image domain:

$$\tilde{f}(\boldsymbol{\tau}(\boldsymbol{x}; \boldsymbol{r}), \delta) \;=\; \big(f(\boldsymbol{\tau}(\cdot; \boldsymbol{r})) * k(\cdot)\big)(\boldsymbol{x}), \tag{2}$$

where

- f denotes the original image; $\tilde{f}$ is the smoothed image.
- $\boldsymbol{\tau}(\boldsymbol{x}; \boldsymbol{r})$ warps $\boldsymbol{x}$ according to parameters $\boldsymbol{r} \in \mathbb{R}^m$, where m is the number of DoF.
- $*$ denotes convolution in the image domain.

Although this isotropic smoothing can widen the convergence basin by removing high-frequency content, it applies a fixed blur globally without considering the specific transformation, motivating kernels that depend explicitly on the motion model.

C. PGM in DVS

Photometric Gaussian Mixtures (PGM) were introduced by Crombez et al. [13] as a dense feature of images for direct visual servoing. PGM builds upon the GRBF image model (1) by representing each pixel's intensity as a Gaussian bump, but crucially allows the blur magnitude to be variable over time.

In the PGM framework, both the desired image and the current image are modeled as mixtures of Gaussian kernels centered at each pixel, influencing their neighborhood with a spread δ. By dynamically tuning δ during servoing, PGM can start with a large value (heavy image blur smooths the cost function when the robot is far from the goal) and gradually reduce it toward zero (sharpening details for accuracy). This adaptive strategy enlarges the convergence domain, with PGM demonstrating convergence from significantly larger initial pose errors than classical DVS.

Similarly to GRBF, PGM models each image as a sum of pixel-centered Gaussians. For an image f with pixel domain $\mathcal{X}$ and intensity $f(\boldsymbol{x})$ at pixel $\boldsymbol{x} \in \mathcal{X}$, the Gaussian mixture feature G at sampling location $\boldsymbol{x}$ with spread δ is defined in (1). Stacking over all $\boldsymbol{x}$ yields the feature vector:

$$\boldsymbol{g}(f, \delta) \;=\; \big[\, G(f, \boldsymbol{x}, \delta) \,\big]_{\boldsymbol{x} \in \mathcal{X}} \tag{3}$$

The PGM control law replaces the direct intensity error with a Gaussian mixture error. The error is:

$$\boldsymbol{\epsilon}(\boldsymbol{r}, \delta) \;=\; \boldsymbol{g}\big(f_1(\boldsymbol{\tau}(\boldsymbol{x}; \boldsymbol{r})), \delta\big) \;-\; \boldsymbol{g}(f_2(\boldsymbol{x}), \delta), \tag{4}$$

where f_1 is the current camera image (warped by $\boldsymbol{\tau}(\boldsymbol{x}; \boldsymbol{r})$ according to the transformation DoF $\boldsymbol{r}$) and f_2 is the desired

reference image. The time derivative of the error is related to the camera velocity $\boldsymbol{v}$ by

$$\dot{\boldsymbol{\epsilon}} \;=\; \mathbf{L}_{gm}\,\boldsymbol{v}, \tag{5}$$

where $\mathbf{L}_{gm}$ is the interaction matrix that maps the camera velocity to the time variation of the Gaussian mixture features. This matrix can be expressed in closed form with Gaussian derivatives [13].

Using a standard visual-servoing control scheme [1], the PGM-based control law is

$$\boldsymbol{v} \;=\; -\lambda\,\mathbf{L}_{gm}^{+}\,\boldsymbol{\epsilon}(\boldsymbol{r},\delta), \tag{6}$$

where $\lambda > 0$ is a scalar gain and $\mathbf{L}_{gm}^{+}$ denotes the Moore–Penrose pseudoinverse. Because δ is adjusted during servoing, one can initialize with a large δ (robust, heavily blurred features) and progressively decrease it (sharper features, higher accuracy at convergence), which substantially enlarges the convergence basin compared to raw DVS.

D. Transformation-Domain Cost Smoothing

The smoothing strategies above operate directly in the image domain, applying isotropic Gaussian blur. In contrast, Mobahi et al. [10] proposed a principled alternative: smooth the objective function itself in the space of transformation parameters. This approach, rooted in the idea of graduated nonconvexity, constructs a continuation path by progressively blurring the cost landscape.

In [10], the alignment objective is based on CC. Given a current image f_1 and a reference image f_2, with a parametric warp $\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r})$ defined by the transformation parameter vector $\boldsymbol{r}\in\mathbb{R}^m$, the CC cost is:

$$C_{\mathrm{CC}}(\boldsymbol{r}) \;=\; \int_{\mathcal{X}} f_1\big(\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r})\big)\; f_2(\boldsymbol{x})\,d\boldsymbol{x}. \tag{7}$$

Then, smoothing this objective is done by convolving it with an m-dimensional Gaussian kernel in the parameter domain of $\boldsymbol{r}$, $\mathbb{R}^m$:

$$z_\sigma(\boldsymbol{r}) \;=\; \big(C_{\mathrm{CC}} * k\big)(\boldsymbol{r}) \;=\; \int_{\mathbb{R}^m} C_{\mathrm{CC}}(\boldsymbol{r}^\dagger)\; k(\boldsymbol{r}-\boldsymbol{r}^\dagger;\sigma^2)\,d\boldsymbol{r}^\dagger, \tag{8}$$

where $z_\sigma(\boldsymbol{r})$ denotes the smoothed cost as a function of the transformation parameters $\boldsymbol{r}$ associated with the smoothing spread σ, and $k(\boldsymbol{r};\sigma^2) = (2\pi\sigma^2)^{-m/2}\exp\big(-\|\boldsymbol{r}\|^2/(2\sigma^2)\big)$ is the isotropic Gaussian kernel of variance σ^2.

Starting from the smoothed objective in (8), expanding the parameter space convolution gives:

$$z_\sigma(\boldsymbol{r}) \;=\; \int_{\mathcal{X}} f_2(\boldsymbol{x}) \underbrace{\Big([\,f_1(\boldsymbol{\tau}(\boldsymbol{x},\cdot)) * k](\boldsymbol{r})\Big)}_{\displaystyle\int_{\mathbb{R}^m} f_1\big(\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r}^\dagger)\big)\,k(\boldsymbol{r}-\boldsymbol{r}^\dagger;\sigma^2)\,d\boldsymbol{r}^\dagger} d\boldsymbol{x}. \tag{9}$$

To express the smoothing in the n-dimensional image domain rather than the m-dimensional transformation-domain, f_1 is expanded via its inverse Fourier representation. Fubini's theorem justifies exchanging the order of integration, and Parseval's theorem is then used to rewrite the correlation in the Fourier domain. The smoothed correlation is then expressed in terms of a transformation kernel $u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})$ as:

$$z_\sigma(\boldsymbol{r}) \;=\; \int_{\mathcal{X}} f_2(\boldsymbol{x})\Big(\int_{\mathcal{X}} f_1(\boldsymbol{y})\; u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})\,d\boldsymbol{y}\Big)d\boldsymbol{x}, \tag{10}$$

with:

$$u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y}) = \frac{1}{(2\pi)^n}\int_{\Omega}\int_{\mathbb{R}^m} e^{\,i\,\boldsymbol{\omega}^\top(\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r}^\dagger)-\boldsymbol{y})}\,k(\boldsymbol{r}-\boldsymbol{r}^\dagger;\sigma^2)\,d\boldsymbol{r}^\dagger\,d\boldsymbol{\omega}, \tag{11}$$

where $\boldsymbol{\omega}$ is the Fourier frequency ($\Omega\subset\mathbb{R}^n$). In effect, the $\mathbb{R}^m$ parameter-space convolution (e.g., $m=8$ for homography) becomes a two-dimensional image-space integral ($n=2$) via the warp-dependent kernel $u_{\boldsymbol{\tau},\sigma}$.

For pure planar translation, $\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r}^\dagger) = \boldsymbol{x}+\boldsymbol{r}^\dagger$, the inner integral over $\boldsymbol{r}^\dagger$ in (11) yields:

$$\begin{aligned} u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y}) &= \frac{1}{(2\pi)^n}\int_{\Omega} e^{\,i\,\boldsymbol{\omega}^\top\left((\boldsymbol{x}+\boldsymbol{r})-\boldsymbol{y}\right)}\,e^{-\frac{1}{2}\sigma^2\|\boldsymbol{\omega}\|^2}\,d\boldsymbol{\omega} \\ &= k\big((\boldsymbol{x}+\boldsymbol{r})-\boldsymbol{y};\,\sigma^2\big). \end{aligned} \tag{12}$$

Thus, parameter-domain smoothing coincides with Gaussian blurring of f_1 in the image domain. For affine and projective warps, evaluating the integrals over $\boldsymbol{\omega}$ and $\boldsymbol{r}^\dagger$ produces spatially varying, anisotropic kernels $u_{\boldsymbol{\tau},\sigma}$ whose local covariance depends on $\boldsymbol{x}$ and the motion model.

Gaussian convolution produces a smoothed cost function $z_\sigma(\boldsymbol{r}) = (C * k)(\boldsymbol{r})$ that progressively removes narrow local minima as σ grows, yielding a simpler landscape with broader basins of attraction. Importantly, the argmax of z_σ does not exactly coincide with that of C; the global maximum shifts slightly under smoothing. This bias vanishes as $\sigma\to 0$, since $z_\sigma\to C$. Conversely, for very large σ, z_σ approaches a nearly constant function (low contrast) and the gradient ∇z_σ becomes small, which slows optimization down and can obscure the true basin structure. Fig. 1 illustrates how increasing σ suppresses local minima while flattening the gradient of the unnormalized CC cost. The moving image was warped by a scale of 1.5 with a constant $\delta = 1$ and different σ values. The ground-truth inverse optimum along this function is $s_x^\star = 1/1.5 \approx 0.667$.

To benefit from smoothing while limiting induced bias, [10] uses a continuation strategy: start with a moderate σ_0 to suppress spurious minima and obtain a coarse solution, then gradually decrease the smoothing $\sigma_{i+1} < \sigma_i$ while re-optimizing at each stage. This tracks the solution from the wide-basin, low-detail landscape to the original cost, mitigating the minimum shift and restoring final accuracy.

III. Transformation-Domain SSD Smoothing

The transformation-domain smoothing framework of [10] was originally derived for CC-based alignment. This work adapts this principle to the SSD objective used in direct visual servoing, yielding a smoothed photometric cost that preserves motion-adaptive behavior while remaining compatible with standard DVS formulations.

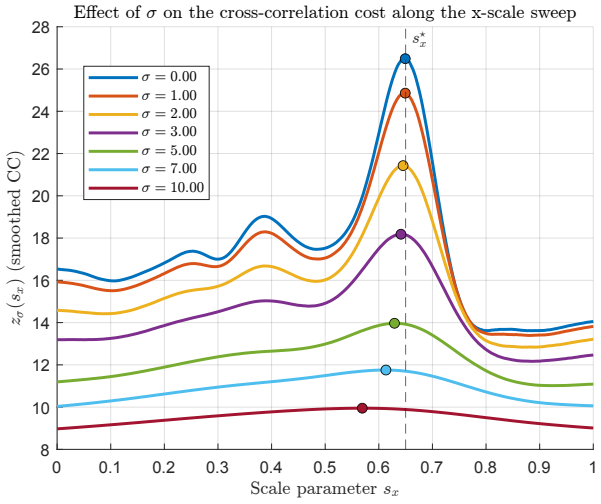


Fig. 1. CC-based alignment cost $z_\sigma(s_x)$ evaluated along a 1-D sweep of the x-scale parameter s_x (all other parameters fixed). The dashed line indicates the ground-truth inverse scale $s_x^\star$. Increasing σ suppresses narrow local extrema and widens the basin of attraction, while very large σ flattens the landscape and weakens gradients.

A. Smoothed SSD Cost (CoSmooth)

Considering a DVS problem where the objective is to minimize the SSD between the current image captured by the camera f_1 and a desired reference image f_2, with a transformation parameter vector $\boldsymbol{r} \in \mathbb{R}^m$, the corresponding photometric cost [15] is:

$$C(\boldsymbol{r}) \;=\; \frac{1}{2}\int_{\mathcal{X}} \big(f_1\big(\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\big) - f_2(\boldsymbol{x})\big)^2\, d\boldsymbol{x}, \tag{13}$$

where $\boldsymbol{\tau}(\boldsymbol{x};\boldsymbol{r})$ warps a pixel location $\boldsymbol{x}$ according to the transformation.

The base SSD (13) is convolved with an isotropic Gaussian kernel in the transformation-parameter space $\boldsymbol{r}$:

$$z(\boldsymbol{r}) \;=\; \big[C(\,\cdot\,) * k(\,\cdot\,,\sigma^2)\big](\boldsymbol{r}), \tag{14}$$

where $z(\boldsymbol{r})$ is now the smoothed SSD cost function as a function of the transformation vector and the smoothing spread σ, to be referred to as CoSmooth.

Expanding the square in (13), then applying the convolution term-wise results in:

$$\begin{aligned} z(\boldsymbol{r}) \;=& \frac{1}{2}\Big[\int_{\mathcal{X}} f_1\big(\boldsymbol{\tau}(\boldsymbol{x},\cdot)\big)^2\, d\boldsymbol{x} \;*\; k(\cdot,\sigma^2)\Big](\boldsymbol{r}) \\ &- \Big[\int_{\mathcal{X}} f_1\big(\boldsymbol{\tau}(\boldsymbol{x},\cdot)\big)\, f_2(\boldsymbol{x})\, d\boldsymbol{x} \;*\; k(\cdot,\sigma^2)\Big](\boldsymbol{r}) \\ &+ \frac{1}{2}\Big[\int_{\mathcal{X}} f_2(\boldsymbol{x})^2\, d\boldsymbol{x} \;*\; k(\cdot,\sigma^2)\Big](\boldsymbol{r}). \end{aligned} \tag{15}$$

The function $z(\boldsymbol{r})$ has three contributions: a self-term, a cross-term, and a constant term. Applying the same Fubini–Fourier construction that gives $u_{\boldsymbol{\tau},\sigma}$ in (11) to (15) where the third term is independent of $\boldsymbol{r}$ and therefore remains unchanged after convolution, we obtain:

$$\begin{aligned} z(\boldsymbol{r}) \;=& \int_{\mathcal{X}} \Big[\frac{1}{2}\int_{\mathcal{X}} f_1(\boldsymbol{y})^2 u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})\, d\boldsymbol{y} \\ &- f_2(\boldsymbol{x})\int_{\mathcal{X}} f_1(\boldsymbol{y}) u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})\, d\boldsymbol{y} \;+ \frac{1}{2} f_2(\boldsymbol{x})^2\Big] d\boldsymbol{x}, \end{aligned} \tag{16}$$

where $u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})$ is the transformation kernel identical to that in (11).

For a closed-form exponential $u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})$, substituting $\boldsymbol{\tau}$ and integrating while retaining only the diagonal GRBF self-term contributions for tractability, will result in a cost function as follows:

$$\begin{aligned} z(\boldsymbol{r}) = \int_{\mathcal{X}} \Bigg(& \frac{1}{2}\sum_{i=1}^{N} a_i^2 \left(\frac{\delta}{\sqrt{2(\frac{\delta^2}{2}+s^2(\boldsymbol{x}))}}\right)^n \exp\!\Big(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\frac{\delta^2}{2}+s^2(\boldsymbol{x}))}\Big) \\ & - f_2(\boldsymbol{x}) \sum_{i=1}^{N} a_i \left(\frac{\delta}{\sqrt{\delta^2+s^2(\boldsymbol{x})}}\right)^n \exp\!\Big(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\delta^2+s^2(\boldsymbol{x}))}\Big) \\ & + \tfrac{1}{2} f_2(\boldsymbol{x})^2 \Bigg)\, d\boldsymbol{x}, \end{aligned} \tag{17}$$

where $s^2(\boldsymbol{x})$ denotes the effective image-domain variance induced by smoothing, which depends on both σ and the local pixel location.

To obtain closed-form expressions for the integrals in (16), the current image is represented by the GRBF model in (1). This choice enables analytic evaluation of the terms in (16). Other smooth bases are possible, but GRBFs enable an easier derivation and are therefore adopted here.

B. Transformation Kernel for Translational Visual Servoing

Consider a camera undergoing pure translation with $\boldsymbol{r} = [t_X, t_Y, t_Z]^\top$. A 3D point (X, Y, Z) maps to normalized image coordinates after translation as:

$$\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r}) = \begin{bmatrix} \frac{X-t_X}{Z-t_Z} \\ \frac{Y-t_Y}{Z-t_Z} \end{bmatrix} = \frac{Z}{Z-t_Z}\begin{bmatrix} x \\ y \end{bmatrix} - \frac{1}{Z-t_Z}\begin{bmatrix} t_X \\ t_Y \end{bmatrix}, \tag{18}$$

showing how camera 3D translation induces both a dilation (mostly due to t_Z) and a shift in the image plane (mostly due to t_X and t_Y). Substituting this warp $\boldsymbol{\tau}$ into the transformation–domain smoothing integral, and using the separability of the isotropic Gaussian in $\boldsymbol{r}$ into one-dimensional factors k_X, k_Y, k_Z, the evaluation over $\boldsymbol{r} \in \mathbb{R}^3$ in (11) yields the integrals:

$$\begin{aligned} \frac{1}{(2\pi)^n}\int_{\Omega}\int_{\mathbb{R}}\int_{\mathbb{R}}\int_{\mathbb{R}} & \exp\!\left(i\,\boldsymbol{\omega}^\top\left[\frac{Z}{Z-t_Z^\dagger}\,\boldsymbol{x} - \frac{1}{Z-t_Z^\dagger}\begin{bmatrix} t_X^\dagger \\ t_Y^\dagger \end{bmatrix} - \boldsymbol{y}\right]\right) \\ & k_X(t_X - t_X^\dagger) k_Y(t_Y - t_Y^\dagger) k_Z(t_Z - t_Z^\dagger) \\ & dt_X^\dagger\, dt_Y^\dagger\, dt_Z^\dagger\, d\boldsymbol{\omega}. \end{aligned} \tag{19}$$

The result of the integration is a spatially varying kernel with variance:

$$s^2(\boldsymbol{x}) = \frac{\sigma^2}{Z^2}\left(1 + \|\boldsymbol{x}\|^2\right). \tag{20}$$

This expression shows that the effective smoothing is depth-normalized and spatially varying.

The smoothed cost $z(\boldsymbol{r})$ is differentiated with respect to the pose:

$$\begin{aligned} \dot{z}(\boldsymbol{r}) \;&=\; \frac{\partial z(\boldsymbol{r})}{\partial \boldsymbol{r}}\,\boldsymbol{v}, \qquad \boldsymbol{v}=\dot{\boldsymbol{r}}. \\ \frac{\partial z(\boldsymbol{r})}{\partial \boldsymbol{r}} \;&=\int_{\mathcal{X}}\Big[\frac{1}{2}\int_{\mathcal{X}} f_1(\boldsymbol{y})^2 \frac{\partial u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})}{\partial \boldsymbol{r}}\; d\boldsymbol{y} \\ &\quad - \; f_2(\boldsymbol{x})\int_{\mathcal{X}} f_1(\boldsymbol{y})\frac{\partial u_{\boldsymbol{\tau},\sigma}(\boldsymbol{r},\boldsymbol{x},\boldsymbol{y})}{\partial \boldsymbol{r}}\; d\boldsymbol{y}\Big] d\boldsymbol{x}. \end{aligned} \tag{21}$$

It is important to note that in (21) the convolution property allows for the derivative of the cost to be expressed entirely through the kernel derivative, without requiring spatial gradients of the image itself. This is advantageous as direct differentiation of image intensities is often noisy. By shifting differentiation to the analytic kernel $u_{\boldsymbol{\tau},\sigma}$, the formulation bypasses the need for explicit image gradients.

Differentiating the exponential in (17), the gradient of the smoothed cost $z(\boldsymbol{r})$ is as follows:

$$\begin{aligned} \frac{\partial z(\boldsymbol{r})}{\partial \boldsymbol{r}} \;=\int_{\mathcal{X}}\Bigg\{ &\frac{1}{2}\sum_{i=1}^{N} a_i^2\left(\frac{\delta}{\sqrt{2(\frac{\delta^2}{2}+s^2(\boldsymbol{x}))}}\right)^n \exp\!\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\frac{\delta^2}{2}+s^2(\boldsymbol{x}))}\right)\begin{bmatrix}\frac{x_i-\tau_x(\boldsymbol{x},\boldsymbol{r})}{\frac{\delta^2}{2}+s^2(\boldsymbol{x})}\\ \frac{y_i-\tau_y(\boldsymbol{x},\boldsymbol{r})}{\frac{\delta^2}{2}+s^2(\boldsymbol{x})}\end{bmatrix}^\top \\ &- f_2(\boldsymbol{x})\sum_{i=1}^{N} a_i\left(\frac{\delta}{\delta^2+s^2(\boldsymbol{x})}\right)^n \exp\!\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\delta^2+s^2(\boldsymbol{x}))}\right)\begin{bmatrix}\frac{x_i-\tau_x(\boldsymbol{x},\boldsymbol{r})}{\delta^2+s^2(\boldsymbol{x})}\\ \frac{y_i-\tau_y(\boldsymbol{x},\boldsymbol{r})}{\delta^2+s^2(\boldsymbol{x})}\end{bmatrix}^\top\Bigg\} \\ &\mathbf{J}_{\boldsymbol{\tau}}(\boldsymbol{x})\,d\boldsymbol{x}, \end{aligned} \tag{22}$$

where the descent direction is given by the gradient of the smoothed cost with respect to the pose parameters.

The per-pixel contribution of the smoothed cost gradient can be rearranged to expose a product structure similar to that of classical DVS. Using the commutativity and associativity of the summation, the derivative in (22) can be rearranged to obtain:

$$\begin{aligned} \mathbf{L}_\sigma(\boldsymbol{x},\boldsymbol{r}) &= \sum_{i=1}^{N} a_i\Big(\frac{\delta}{\sqrt{\delta^2+s^2(\boldsymbol{x})}}\Big)^n \exp\Big(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\delta^2+s^2(\boldsymbol{x}))}\Big) \\ &\quad \frac{1}{\delta^2+s^2(\boldsymbol{x})}(\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r}))^\top \mathbf{J}_{\boldsymbol{\tau}}(\boldsymbol{x}), \\ e_\sigma(\boldsymbol{x},\boldsymbol{r}) &= \sum_{i=1}^{N} \frac{1}{2^{\frac{n}{2}+1}} a_i\Big(\frac{\delta^2+s^2(\boldsymbol{x})}{\frac{\delta^2}{2}+s^2(\boldsymbol{x})}\Big)^{\frac{n}{2}+1} \\ &\quad \exp\Big(-\frac{\delta^2\|\boldsymbol{x}_i-\boldsymbol{\tau}(\boldsymbol{x},\boldsymbol{r})\|^2}{2(\delta^2+s^2(\boldsymbol{x}))\left(\frac{\delta^2}{2}+s^2(\boldsymbol{x})\right)}\Big) - f_2(\boldsymbol{x}). \end{aligned} \tag{23}$$

Stacking the residual terms $e_\sigma(\boldsymbol{x},\boldsymbol{r})$ over $\boldsymbol{x}\in\mathcal{X}$ forms the residual vector $\boldsymbol{e}_\sigma(\boldsymbol{r})$. The corresponding stacked interaction matrix $\mathbf{L}_\sigma(\boldsymbol{r})$ collects the per-pixel contributions $\mathbf{L}_\sigma(\boldsymbol{x},\boldsymbol{r})$ with $\mathbf{L}_\sigma(\boldsymbol{r})\in\mathbb{R}^{N\times 3}$ and $\boldsymbol{e}_\sigma(\boldsymbol{r})\in\mathbb{R}^N$.

For translational motion and constant depth Z, the local warp Jacobian takes the classical IBVS translational interaction-matrix form:

$$\mathbf{J}_{\boldsymbol{\tau}}(\boldsymbol{x}) = \begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} \\ 0 & -\frac{1}{Z} & \frac{y}{Z} \end{bmatrix}.$$

To improve the convergence and robustness of the optimization scheme, second-order minimization is often favored over first-order gradient descent by exploiting a local quadratic approximation of the objective [16]. Following this reasoning, adopting a Gauss-Newton update for the smoothed photometric objective results in:

$$\boldsymbol{v} = -\lambda\,\mathbf{L}_\sigma(\boldsymbol{r})^+\,\boldsymbol{e}_\sigma(\boldsymbol{r}), \tag{24}$$

where $\lambda>0$ is a scalar gain and $\mathbf{L}_\sigma(\boldsymbol{r})^+$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{L}_\sigma(\boldsymbol{r})$.

IV. Experimental setup

We use a 6-DoF UR5 manipulator in an eye-in-hand configuration with a FLIR Blackfly S camera rigidly mounted on the wrist as shown in Fig. 2. The images are converted to grayscale and downsampled to 125×100 pixels to ensure identical computational load across the methods. The initial pose for each trial is set from joint positions; at the beginning of every run, we impose a different relative offset between the current and desired views. During control we actuate only the three translational degrees of freedom $\boldsymbol{r}=[t_X,t_Y,t_Z]^\top$. Rotations are kept fixed. Our implementation uses $\mathbf{L}_\sigma(\boldsymbol{r})$ and $\boldsymbol{e}_\sigma(\boldsymbol{r})$ obtained from (23), and applies the control update in (24). CoSmooth is then compared to PGM VS defined in (6).

Fig. 2. Visual servoing setup: 6-DoF UR5 with wrist-mounted FLIR Blackfly S (eye-in-hand).

Both methods share the same visual pipeline and control interface; only the photometric objective differs. Gains are tuned once per method and then held fixed; all trial-specific offsets and gains are reported in the figure captions for clarity. For fairness, the PGM blur is chosen to upper–bound the effective CoSmooth spread $\delta^2_{\text{PGM}} \geq \delta^2_{\text{CoSmooth}} + \sigma^2$. The scene depth Z is manually set once and used in the interaction matrix, which avoids the need for frequent tuning of the control gain λ. Each trial uses the same start and target images for both methods, and the desired image is acquired at the goal pose prior to the experiments.

Our objective in these experiments is to assess the convergence domain rather than final alignment accuracy. We therefore keep the PGM blur fixed throughout each run, and set

CoSmooth's smoothing parameter to $\sigma = 0$ after 150 iterations to mitigate smoothing-induced optimum shift.

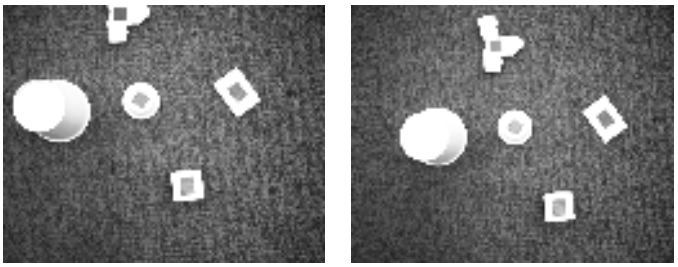
V. Results

We compare CoSmooth to a PGM VS baseline on three 3-DoF translational trials, visualized as XZ and YZ projections of the trajectories. The depth is manually set to be $Z = 1$ for all experiments.

a) Experiment A: In this experiment, the camera is shifted 3 cm in the X, Y and Z directions. The desired and initial images are shown in Fig. 3, with the parameters ($\lambda = 0.05$, $\delta = 1.5$, $\sigma = 0.1$) for CoSmooth and ($\lambda = 0.05$, $\delta = 1.6$) for PGM. Both methods converge as shown in Fig. 4. In the YZ projection, CoSmooth follows a near straight path with weaker depth–lateral coupling and terminates at the target, while the PGM path shows a depth detour. In the XZ projection, trajectories are very close to the desired path for both methods, with residual images shown in Fig. 5.

b) Experiment B: The camera is shifted 4 cm in the Z, 2 cm in the Y and 3 cm in the X as shown in Fig. 6. CoSmooth parameters are ($\lambda = 0.1$, $\delta = 1.5$, $\sigma = 0.1$) and ($\lambda = 0.05$, $\delta = 2.0$) for PGM. Fig. 7 shows that CoSmooth follows a more direct, near-straight path with weaker depth–lateral coupling and lands at the target, while the PGM path shows a depth detour with residual images in Fig. 8.

c) Experiment C: The camera is shifted 10 cm in the Z, 5 cm in the Y and 5 cm in the X as shown in Fig. 9. CoSmooth parameters are ($\lambda = 0.1$, $\delta = 2.0$, $\sigma = 0.1$), and ($\lambda = 0.1$, $\delta = 2.5$) for PGM. CoSmooth converges to the target with mild depth excursions as shown in Fig. 10. In contrast, the PGM baseline diverges from the basin of attraction. The trajectory departs from the ideal path and fails to reach the goal. We verified that this behavior is not due to under-smoothing or gain choice; even after increasing the PGM blur beyond CoSmooth's effective spread and retuning λ, trajectories still diverged. The final residual image



(a) Desired image (b) Initial image

Fig. 3. Comparison between the desired image and initial image of Experiment A.

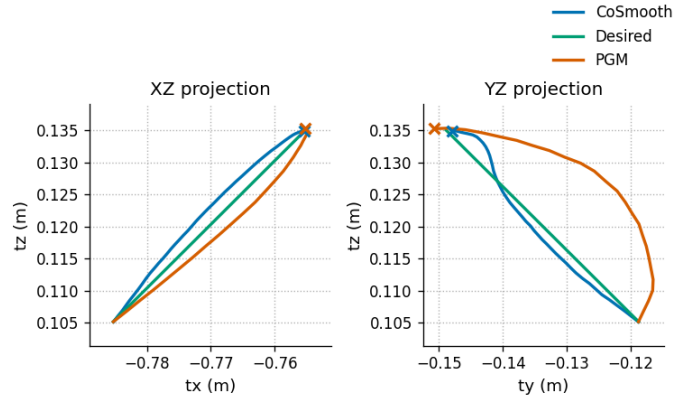
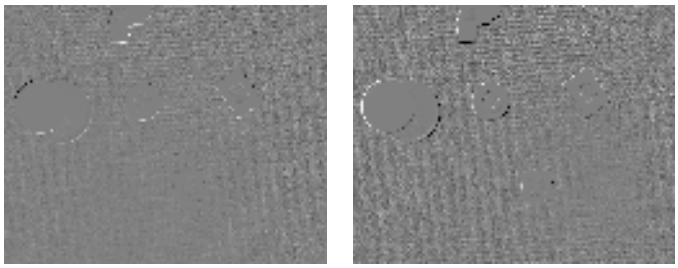
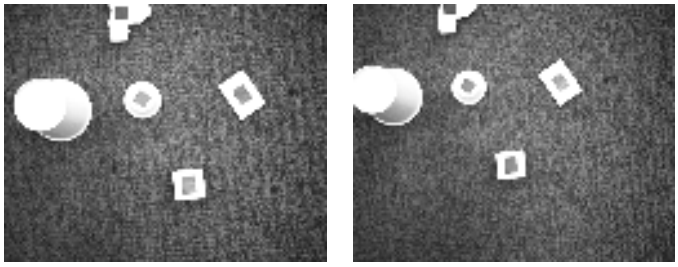


Fig. 4. **Experiment A** Trajectories projected onto XZ (left) and YZ (right). Blue: CoSmooth; Orange: PGM; Green: desired path. Crosses mark the final poses. CoSmooth adheres closely to the desired path and reaches the target; PGM shows a depth detour.



(a) CoSmooth difference image. (b) PGM difference image.

Fig. 5. Residual image at the final pose for Experiment A: pixel-wise difference between the desired and current views.



(a) Desired image (b) Initial image

Fig. 6. Comparison between the desired image and initial image of Experiment B.

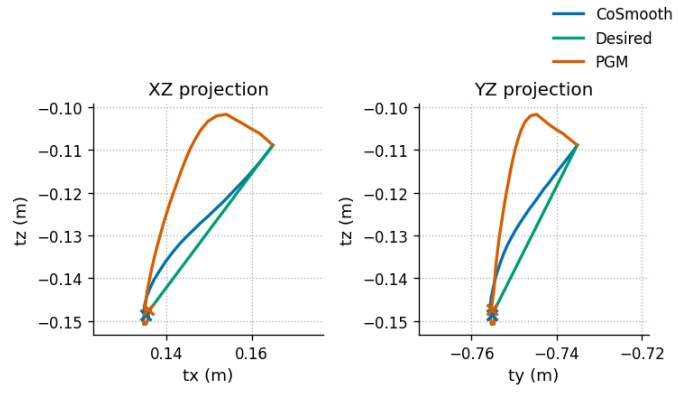


Fig. 7. **Experiment B** Trajectories projected onto XZ (left) and YZ (right). Blue: CoSmooth; Orange: PGM; Green: desired path. Crosses mark the final poses. CoSmooth adheres closely to the desired path and reaches the target; PGM shows a depth detour.

is shown in Fig. 11. The PGM run was stopped when the robot approached a joint limit.

VI. Conclusion and Future Work

We presented CoSmooth, a transformation-domain Gaussian convolution of the SSD objective for direct visual servoing. By shifting smoothing from the image to the parameter space, the method yields motion-adaptive, spatially varying kernels whose local covariance depends on the warp, suppressing geometry-induced spurious minima, while preserving the desired equilibrium as $\sigma \to 0$. On 3-DoF translational eye-in-hand experiments with a UR5, CoSmooth produced wider basins and straighter, shorter trajectories than a PGM VS baseline, converging from larger initial pose offsets where PGM VS showed depth detours or diverged. These results support the claim that transformation-domain smoothing enlarges the practical convergence domain of DVS beyond what uniform image blurs can offer.

This study is limited to translations with no continuation schedule. It does not target computational optimization. Next steps include extending the closed-form treatment to full 6-DoF and running broader quantitative studies of endpoint error, path length, success rate, robustness to occlusions, image noise, and computational optimization.

References

[1] Chaumette, F. & Hutchinson, S. Visual servo control. I. Basic approaches. *IEEE Robotics & Automation Magazine*. **13**, 82–90 (2006)

[2] Silveira, G. & Malis, E. Direct visual servoing: Vision-based estimation and control using only nonmetric information. *IEEE Transactions on Robotics*. **28**, 974–980 (2012)

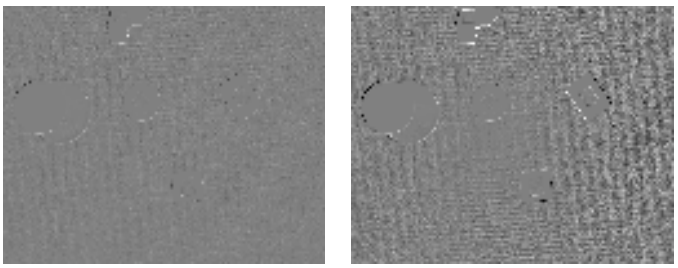
[3] Collewet, C. & Marchand, E. & Chaumette, F. Visual servoing set free from image processing. *IEEE International Conference on Robotics and Automation*. 81–86 (2008)

[4] Chaumette, F. Potential problems of stability and convergence in image-based and position-based visual servoing. *The Confluence of Vision and Control*. *Lecture Notes in Control and Information Sciences*. **237**, 66–78 (1998)

[5] Chen, Y., Meng, M. Q.-H. & Liu, L. Direct visual servoing based on discrete orthogonal moments. *IEEE Transactions on Robotics*. **40**, 1795–1812 (2024)

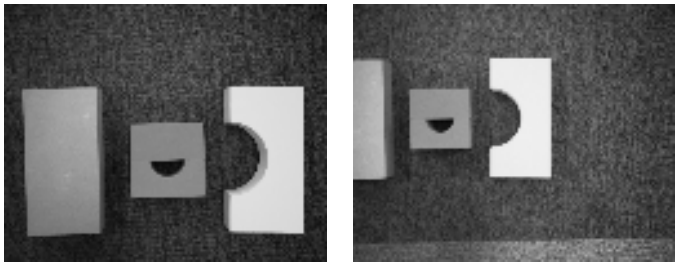
[6] Naamani, M., Caron, G., Morisawa, M. & Mouaddib, E. A mathematical characterization of the convergence domain for direct visual servoing. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 4846–4853 (2024)

[7] Caron, G. & Yoshiyasu, Y. Direct visual servoing in the non-linear scale space of camera pose. *International Conference on Pattern Recognition*. 4154–4160 (2022)



(a) CoSmooth difference image. (b) PGM difference image.

Fig. 8. Residual image at the final pose for Experiment B: pixel-wise difference between the desired and current views.



(a) Desired image (b) Initial image

Fig. 9. Comparison between the desired image and initial image of Experiment C.

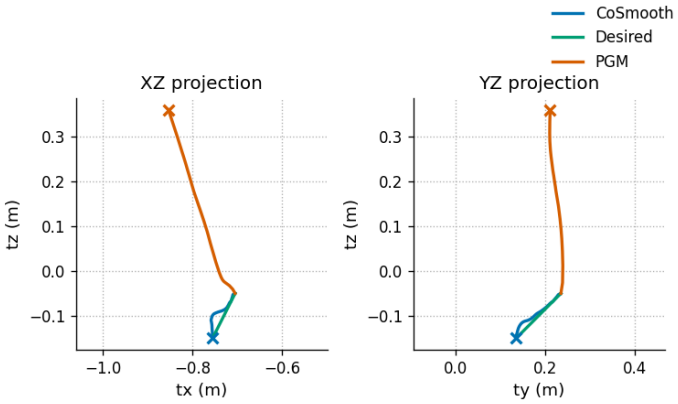
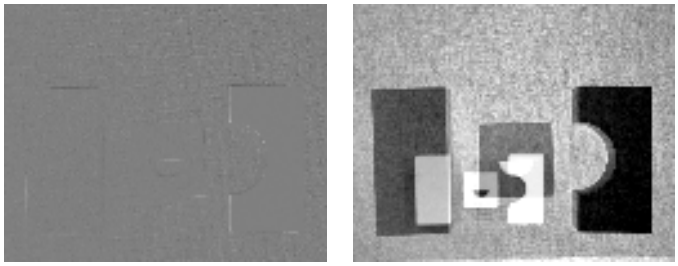


Fig. 10. **Experiment C** Trajectories projected onto XZ (left) and YZ (right). Blue: CoSmooth; Orange: PGM; Green: desired path. Crosses mark the final poses. CoSmooth follows a more direct path with mild t_Z excursions, while PGM diverges and fails to reach the goal.



(a) CoSmooth difference image. (b) PGM difference image.

Fig. 11. Residual image at the final pose for Experiment C: pixel-wise difference between the desired and current views.

[8] Lucas, B. & Kanade, T. An iterative image registration technique with an application to stereo vision. *International Joint Conference on Artificial Intelligence*. 81–88 (1981)

[9] Blake, A. & Zisserman, A. Visual reconstruction. MIT Press (1987)

[10] Mobahi, H., Zitnick, C. & Ma, Y. Seeing through the blur. *IEEE Conference on Computer Vision and Pattern Recognition*. 1736–1743 (2012)

[11] Mobahi, H. & Fisher, J. On the link between Gaussian homotopy continuation and convex envelopes. *Energy Minimization Methods in Computer Vision and Pattern Recognition*. 43–56 (2015)

[12] Caron, G. Defocus-based direct visual servoing. *IEEE Robotics and Automation Letters*. **6**, 4056–4063 (2021)

[13] Crombez, N., Caron, G. & Mouaddib, E. Photometric Gaussian mixtures based visual servoing. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5486–5491 (2015)

[14] Kallem, V., Dewan, M., Swensen, J., Hager, G. & Cowan, N. Kernel-based visual servoing. *IEEE/RSJ International Conference on Intelligent*

Robots and Systems. 1975–1980 (2007)
[15] Collewet, C. & Marchand, E. Photometric visual servoing. *IEEE Transactions on Robotics*. **27**, 828–834 (2011)
[16] Tahri, O. & Mezouar, Y. On visual servoing based on efficient second order minimization. *Robotics and Autonomous Systems*. **58**, 712–719 (2010)