Robotica: (2020) volume 38, pp. 831–844. © Cambridge University Press 2019
doi:10.1017/S0263574719001115

Position-Based Visual Servoing of a Mobile Robot with an Automatic Extrinsic Calibration Scheme

Radhe Shyam Sharma*, Santosh Shukla, Laxmidhar Behera and Venkatesh K. Subramanian

Department of Electrical Engineering, Indian Institute of Technology Kanpur, Kanpur, India.
E-mails: santosh.shukla0247@gmail.com, lbehera@iitk.ac.in, venkats@iitk.ac.in

(Accepted June 12, 2019. First published online: July 24, 2019)

SUMMARY
In this paper, we present and implement a novel approach for position-based visual servoing. The challenge of controlling the mobile robot while simultaneously estimating the camera to mobile robot transformation is solved. This is achieved using gradient descent (GD)-based estimation and the sliding-mode approach. The GD approach allows online parameter estimation for controlling the robot to achieve a desired position and orientation. The adaptive nature of the parameters demonstrates the robustness of the system. In contrast to existing work, the proposed technique achieves both estimation and control tasks in a single experiment. Simulation and experimental results are provided to validate the performance of the proposed scheme.

KEYWORDS: Visual servoing; Gradient descent; Sliding-mode control; Auto extrinsic calibration.

1. Introduction

Visual feedback is a strong approach for controlling a system. This is called visual servoing. The term visual servoing was introduced by Agin in ref. [1] to discriminate their approach from earlier "blocks world" experiments. A visual feedback control loop is used to increase the accuracy of the overall system and is given in ref. [2]. Visual servo control is the key approach that takes visual feedback to accomplish various tasks such as stabilization and trajectory tracking of a robot.[3,4] Visual servoing can be classified into three types, namely position-based visual servoing (PBVS), image-based visual servoing (IBVS) and hybrid approach. These approaches are used to drive the robot to the desired pose and orientation. A PBVS seeks to obtain positions in Cartesian space to control the vehicle, whereas IBVS directly deals with visual information on the image plane. The basic conceptual framework for visual servoing is given in ref. [5]. Authors in ref. [6] introduced the fundamental concepts and explained the basic approaches of visual servoing. Both methods of visual servoing have their own benefits and drawbacks. These methods were analyzed and compared.[7] In ref. [8], an image-based and a pose-based path reaching approaches are compared. Hybrid approach was described in ref. [9] to improve the performance of IBVS and PBVS. Light field IBVS (LF-IBVS) is proposed.[10]

Earlier work on visual servoing focused on robotic manipulators,[11,12] unmanned aerial vehicles,[13,14] humanoid robots, etc. An IBVS controller design for a class of under-actuated dynamic systems is presented.[15] There, spherical projection for the camera geometry is used. In ref. [16], it is shown that how visual servoing can be formulated as a quadratic optimization problem. Recently, the focus has increased on mobile robots. Authors in ref. [17] proposed an image sequence-based navigation method under the teaching-replay framework for robots in piecewise linear routes. The

* Corresponding author. E-mail: sharmars@iitk.ac.in

biggest challenge is that mobile robots are subject to nonholonomic constraints. Sinusoidal input-based visual control approach for nonholonomic robot is proposed.[18] The other issue is Brockett's theorem[19] which states that such a system cannot be stabilized by any smooth time invariant feedback controller. The difficulty of visual servoing is further increased by using a monocular camera which lacks depth information. This prevents reconstruction of robot pose without prior information about the target. Thus, the problem becomes challenging because of the combination of unknown depth and motion constraints. Many solutions have been provided; for example, homography-based adaptive visual servoing strategy in the presence of unknown depth is proposed by Fang et al.[20] In ref. [21], visual servoing approach for mobile robots with uncalibrated intrinsic parameters is presented.

In ref. [22], Fang et al. then proposed a switched controller to deal with unknown depth and visibility constraints. Becerra et al.[23] proposed a sliding-mode controller based on three view epipolar geometry. Zhang et al.[24] proposed a visual servoing strategy based on the homography matrix to obtain camera to robot parameters. However, the design proposes a two-stage experiment: one to estimate the rotational error between the camera frame and the robot frame, and the other to find the translation parameters between the two frames and control the robot to drive it to the desired pose. Machine learning-based approaches for mobile robot localization are presented in refs. [25–30]. Marinho et al.[26] present an approach to localize the mobile robot via a classifier with reject option. Omnidirectional images are utilized in ref. [27]. Gray-level co-occurrence matrix is used in ref. [28] as a feature extractor along with support vector machines to localize the mobile robot. In ref. [29], structural co-occurrence matrix is used as a feature extractor along with classifier. Authors in ref. [30] proposed an approach to localize and navigate a mobile robot using topological map and classifier with reject option. A convolutional neural network is adopted in ref. [30] to extract the features.

Extrinsic camera parameter measurement is a difficult task due to the invisibility of the optical axis and the origin of the camera frame. Thus, most papers generally consider them insignificant or ignore them completely. However, they are important parameters to precisely control the mobile robot, especially, in situations which are deprived of human contact like the Mars rover. This problem is addressed by Zhang et al.,[24] but the problem becomes more challenging when camera position is changed during the task. This paper proposes an approach that avoids the estimation of depth by utilizing known markers. The homography matrix[20,31] requires all features to lie in a plane. In general, good features are not always available on common planar surface, for example, a wall, unless they are modified. Hence, ArUco marker[32] is utilized to simplify the problem so that depth may be directly obtained; this obviates the estimation of one parameter. Further, the online estimation using gradient descent (GD) along with sliding-mode control has been proposed to identify all parameters and control the robot to the desired pose in a single experiment.

The body-to-camera parameters can get disturbed under external disturbances, because no mechanical coupling (screws, adhesive, etc.) is ever perfect. Thus, strong bumps and knocks can cause the mechanical coupling to lose alignment or fail. Any such incident will cause loss of calibration. This is what makes online adaptation necessary for automatic recalibration. Robustness with respect to accidental camera displacement during the task is one of the key features of the proposed approach. Another is an online estimation of parameters: while other methods require a separate trial to estimate camera parameters, our method estimates the parameters on the fly. This also means that if any parameter changes unexpectedly during execution, the convergence will continue. This has been demonstrated in simulation though experimental validation is difficult due to the impossibility of obtaining the ground truth of the disturbance in camera parameters without interrupting the flow of the experiment. Our method also works well in the presence of noise and various levels of bias up to a limited point. Derivations, mathematical proof of convergence, and integration of a newly designed estimator along with sliding-mode controller are some of the more significant contributions of this paper. A comparison Table I is included to show the advantages of the proposed approach over the existing ones. More specifically, the key contributions of this paper are the following:

- We propose PBVS of a nonholonomic mobile robot with an automatic extrinsic calibration using GD along with sliding-mode control.
- Novel adaptive laws are derived for online parameter estimation, and the convergence is proved mathematically in the sense of Lyapunov.
- The scheme achieves both estimation and robot control in a single loop.

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

Table I. Comparison with existing techniques.

Proposed method	Li et al.[33]	Zhang et al.[24]
a, b and θ_0 are estimated in one experiment	Does not estimate the extrinsic camera parameters	A separate experiment is needed to estimate θ_0
ArUco marker is a simple way to get depth on a continuous basis, hence no depth estimation	Depth estimation at waypoint but intermediate depth accuracy is not known	It requires one more parameter
For the same sampling rate, it produces more smooth trajectory as compared to Li et al.[33] It can be seen in Fig. 11(a)	More diversion	Not compared
Less control efforts than Li et al.[33] as shown in Fig. 11(b)	More control efforts	Not compared
Robustness with respect to accidental camera displacement which can be seen in Fig. 5(a)	Not applicable	Not shown

- All three unknown extrinsic parameters are estimated in a single trial in contrast to the existing work Zhang et al.,[24] which saves time.
- The proposed approach robustly adapts to changes in the camera's relative position with respect to robot frame.
- The performance of the proposed method is analyzed in the presence of noise and different levels of bias.
- Real-time experiments are performed to address implementation issues of the proposed scheme.

In short, novel adaptive laws, estimation of all three camera parameters in a single experiment, robustness with respect to accidental camera displacement, performance in the presence of noise and inaccuracies, mathematical proof of convergence and simulation as well as experimental validation are the highlights of this paper.

The remainder of this paper is organized as follows. Section 2 introduces the mathematical model of the proposed system. Novel adaptive laws are derived in Section 3. Section 4 presents the controller design methodology along with proof of convergence. The simulation and experimental results are included in Section 5. Section 6 concludes the paper.

2. Problem Formulation and System Modeling

In Fig. 1, F^r and F^c represent the robot frame and camera frame, respectively. The camera frame has a translation of a units and b units on x and y axis, respectively, with respect to the mobile robot fixed frame. The rotation error between the camera frame and the mobile robot frame is given as θ_0 (yaw) and the other two angular errors are assumed to be zero. a, b and θ_0 are the extrinsic parameters between the frames, F^r and F^c. Consider a static Cartesian point P. Let $P_r = [x_r, y_r, z_r]^T$, $P_c = [x_c, y_c, z_c]^T$ be the 3D coordinates of P in F^r and F^c, respectively. The dynamics of P_r is given as

$$\dot{P}_r = -v - \Omega \times P_r \tag{1}$$

where $v \in \Re^3$ and $\Omega \in \Re^3$ are the linear and angular velocity vectors, respectively. If we consider planar motion and nonholonomic constraints, then $v = [v, 0, 0]^T$ and $\Omega = [0, 0, \omega]^T$. Hence Eq. (1) can be expressed in the following form:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} -1 & y_r \\ 0 & -x_r \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2}$$

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

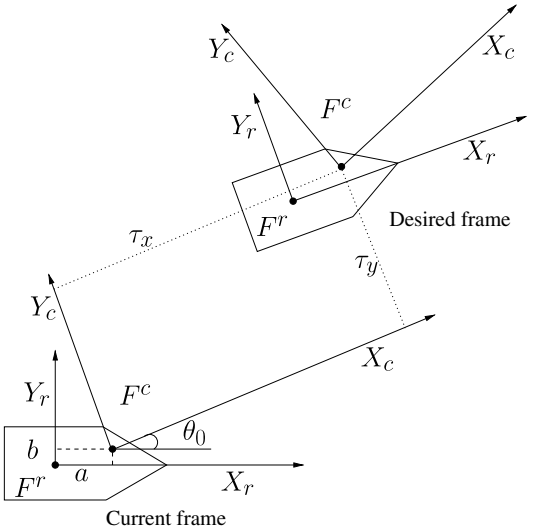


Fig. 1. Correspondence between current and desired frames.

where $v \in \Re$ and $\omega \in \Re$ are the linear and angular velocities of the mobile robot, respectively. Consider a camera placed on the robot. Then, the dynamics of the point P when its reference frame is transformed to the camera frame is represented as follows:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix} = \omega \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \omega \begin{bmatrix} \cos\theta_0 & \sin\theta_0 \\ -\sin\theta_0 & \cos\theta_0 \end{bmatrix} \begin{bmatrix} b \\ -a \end{bmatrix} - \begin{bmatrix} \cos\theta_0 & \sin\theta_0 \\ -\sin\theta_0 & \cos\theta_0 \end{bmatrix} \begin{bmatrix} v \\ 0 \end{bmatrix} \tag{3}$$

The dynamics of Eq. (3) as in ref. [24] is utilized to accomplish the task of visual servoing. The aim of visual servoing is to enable the mobile robot to translate to a desired position and orientation using image feedback. Let the current position be denoted as τ_x and τ_y. Since the feedback is obtained only through the camera, the dynamics of the point (τ_x, τ_y) using Eq. (3) can be written as follows:

$$\begin{bmatrix} \dot{\tau}_x \\ \dot{\tau}_y \end{bmatrix} = \omega \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix} + \omega \begin{bmatrix} \cos\theta_0 & \sin\theta_0 \\ -\sin\theta_0 & \cos\theta_0 \end{bmatrix} \begin{bmatrix} b \\ -a \end{bmatrix} - \begin{bmatrix} \cos\theta_0 & \sin\theta_0 \\ -\sin\theta_0 & \cos\theta_0 \end{bmatrix} \begin{bmatrix} v \\ 0 \end{bmatrix} \tag{4}$$

A third state is utilized to embed orientation information and is given as

$$\dot{\theta} = \omega \tag{5}$$

where θ indicates the orientation of the mobile robot. Equations (4) and (5) together describe the system dynamics that needs to be controlled. The quantities τ_x and τ_y are measurable by utilizing markers, for example, ArUco marker.[32] The position of the marker is obtained with respect to the camera frame. Here, the assumption is made that the robot, camera and the marker lie in the same plane. The estimation of unknown parameters a, b and θ_0 is explained in the next section.

3. Estimator Design

Parameters are estimated using GD. Using the Euler method, the system dynamics, that is, Eq. (4), is discretized. The discretized system is given as

$$\begin{bmatrix} \tau_{x_{k+1}} \\ \tau_{y_{k+1}} \end{bmatrix} = \begin{bmatrix} \tau_{x_k} \\ \tau_{y_k} \end{bmatrix} + \Delta t \left(\omega_k \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \tau_{x_k} \\ \tau_{y_k} \end{bmatrix} + \omega_k \begin{bmatrix} \cos\theta_{0_k} & \sin\theta_{0_k} \\ -\sin\theta_{0_k} & \cos\theta_{0_k} \end{bmatrix} \begin{bmatrix} b_k \\ -a_k \end{bmatrix} - \begin{bmatrix} \cos\theta_{0_k} & \sin\theta_{0_k} \\ -\sin\theta_{0_k} & \cos\theta_{0_k} \end{bmatrix} \begin{bmatrix} v_k \\ 0 \end{bmatrix} \right) \tag{6}$$

Here, $\tau_{x_{k+1}}$,$\tau_{y_{k+1}}$ are the next state variables; τ_{x_k},τ_{y_k} are the current state variables; Δt is the time step between current state and the next state; and k is the kth time step. The error function to be minimized is defined as

$$E = \frac{1}{2}(\tilde{\tau}_{x_{k+1}}^2 + \tilde{\tau}_{y_{k+1}}^2); \quad \text{here, } \tilde{\tau}_{x_k} = \tau_{x_k} - \hat{\tau}_{x_k}; \quad \tilde{\tau}_{y_k} = \tau_{y_k} - \hat{\tau}_{y_k} \tag{7}$$

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

where τ_{x_k} and τ_{y_k} are the actual measurements obtained from the camera, while, $\hat{\tau}_x$ and $\hat{\tau}_y$ are estimated from the estimates of the unknown parameters. The estimates of the parameters are represented as $\hat{a}$, $\hat{b}$ and $\hat{\theta}_0$. The GD-based update laws are given as follows:

$$\hat{a}_{k+1} = \hat{a}_k - \eta_1 \frac{\partial E}{\partial \hat{a}_k} \tag{8}$$

$$\hat{b}_{k+1} = \hat{b}_k - \eta_2 \frac{\partial E}{\partial \hat{b}_k} \tag{9}$$

$$\hat{\theta}_{0_{k+1}} = \hat{\theta}_{0_k} - \eta_3 \frac{\partial E}{\partial \hat{\theta}_{0_k}} \tag{10}$$

where η_1, η_2 and η_3 represent the learning rates. The update equation for a_k is derived as follows:

$$\begin{aligned}
\frac{\partial E}{\partial \hat{a}_k} &= \frac{\partial}{\partial \hat{a}_k}\Big(\frac{1}{2}(\tilde{\tau}_{x_{k+1}}^2 + \tilde{\tau}_{y_{k+1}}^2)\Big) \\
&= \tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{a}_k} \tilde{\tau}_{x_{k+1}} + \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{a}_k} \tilde{\tau}_{y_{k+1}} \\
&= \tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{a}_k} (\tau_{x_{k+1}} - \hat{\tau}_{x_{k+1}}) + \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{a}_k} (\tau_{y_{k+1}} - \hat{\tau}_{y_{k+1}}) \\
&= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{a}_k} \hat{\tau}_{x_{k+1}} - \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{a}_k} \hat{\tau}_{y_{k+1}} \\
&= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{a}_k} \big(\hat{\tau}_{x_k} + \Delta t(-v_k \cos \hat{\theta}_{0_k} + \omega_k \hat{\tau}_{y_k} + \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} - \hat{a}_k \omega_k \sin \hat{\theta}_{0_k})\big) \\
&\quad - \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{a}_k} \big(\hat{\tau}_{y_k} + \Delta t(v_k \sin \hat{\theta}_{0_k} - \omega_k \hat{\tau}_{x_k} - \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{a}_k \omega_k \cos \hat{\theta}_{0_k})\big) \\
&= \omega_k \Delta t(\tilde{\tau}_{x_{k+1}} \sin \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \cos \hat{\theta}_{0_k})
\end{aligned}$$

The above equation is combined with (8) to obtain the update law as

$$\hat{a}_{k+1} = \hat{a}_k - \eta_1 \omega_k \Delta t(\tilde{\tau}_{x_{k+1}} \sin \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \cos \hat{\theta}_{0_k}) \tag{11}$$

Differentiating the error function with respect to $\hat{b}_k$

$$\begin{aligned}
\frac{\partial E}{\partial \hat{b}_k} &= \tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{b}_k} \tilde{\tau}_{x_{k+1}} + \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{b}_k} \tilde{\tau}_{y_{k+1}} \\
&= \tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{b}_k} (\tau_{x_{k+1}} - \hat{\tau}_{x_{k+1}}) + \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{b}_k} (\tau_{y_{k+1}} - \hat{\tau}_{y_{k+1}}) \\
&= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{b}_k} \hat{\tau}_{x_{k+1}} - \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{b}_k} \hat{\tau}_{y_{k+1}} \\
&= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{b}_k} \big(\hat{\tau}_{x_k} + \Delta t(-v_k \cos \hat{\theta}_{0_k} + \omega_k \hat{\tau}_{y_k} + \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} - \hat{a}_k \omega_k \sin \hat{\theta}_{0_k})\big) \\
&\quad - \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{b}_k} \big(\hat{\tau}_{y_k} + \Delta t(v_k \sin \hat{\theta}_{0_k} - \omega_k \hat{\tau}_{x_k} - \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{a}_k \omega_k \cos \hat{\theta}_{0_k})\big) \\
&= \omega_k \Delta t(-\tilde{\tau}_{x_{k+1}} \cos \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \sin \hat{\theta}_{0_k})
\end{aligned}$$

The above equation is used along with (9) to obtain the update law for b_k

$$\hat{b}_{k+1} = \hat{b}_k + \eta_2 \omega_k \Delta t(\tilde{\tau}_{x_{k+1}} \cos \hat{\theta}_{0_k} - \tilde{\tau}_{y_{k+1}} \sin \hat{\theta}_{0_k}). \tag{12}$$

Differentiating the error function with respect to $\hat{\theta}_{0_k}$

$$\begin{aligned}
\frac{\partial E}{\partial \hat{\theta}_{0_k}} &= \tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} (\tau_{x_{k+1}} - \hat{\tau}_{x_{k+1}}) + \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} (\tau_{y_{k+1}} - \hat{\tau}_{y_{k+1}}) \\
&= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} \hat{\tau}_{x_{k+1}} - \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} \hat{\tau}_{y_{k+1}}
\end{aligned}$$

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

$$= -\tilde{\tau}_{x_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} \left(\hat{\tau}_{x_k} + \Delta t(-v_k \cos \hat{\theta}_{0_k} + \omega_k \hat{\tau}_{y_k} + \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} - \hat{a}_k \omega_k \sin \hat{\theta}_{0_k}) \right)$$
$$- \tilde{\tau}_{y_{k+1}} \frac{\partial}{\partial \hat{\theta}_{0_k}} \left(\hat{\tau}_{y_k} + \Delta t(v_k \sin \hat{\theta}_{0_k} - \omega_k \hat{\tau}_{x_k} - \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{a}_k \omega_k \cos \hat{\theta}_{0_k}) \right)$$
$$= \tilde{\tau}_{x_{k+1}} \Delta t(\omega_k(\hat{a}_k \cos \hat{\theta}_{0_k} + \hat{b}_k \sin \hat{\theta}_{0_k}) - v_k \sin \hat{\theta}_{0_k}) - \tilde{\tau}_{y_{k+1}} \Delta t(\omega_k(\hat{a}_k \sin \hat{\theta}_{0_k}$$
$$- \hat{b}_k \cos \hat{\theta}_{0_k}) + v_k \cos \hat{\theta}_{0_k})$$

Equation (10) is combined with the above equation to get the following update law:

$$\hat{\theta}_{0_{k+1}} = \hat{\theta}_{0_k} - \eta_3 \tilde{\tau}_{x_{k+1}} \Delta t(\omega_k(\hat{a}_k \cos \hat{\theta}_{0_k} + \hat{b}_k \sin \hat{\theta}_{0_k}) - v_k \sin \hat{\theta}_{0_k}) \\ + \eta_3 \tilde{\tau}_{y_{k+1}} \Delta t(\omega_k(\hat{a}_k \sin \hat{\theta}_{0_k} - \hat{b}_k \cos \hat{\theta}_{0_k}) + v_k \cos \hat{\theta}_{0_k}) \tag{13}$$

4. Controller Design

A sliding-mode controller is designed to control the system. The sliding-mode controller is chosen for its robustness, simple structure and its efficiency. Equations (4) and (5) are combined as follows, to describe the complete system dynamics:

$$\begin{bmatrix} \dot{\tau}_x \\ \dot{\tau}_y \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\cos \theta_0 & \tau_y + b \cos \theta_0 - a \sin \theta_0 \\ \sin \theta_0 & -\tau_x - b \sin \theta_0 - a \cos \theta_0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{14}$$

The above model can be represented in the following form:

$$\dot{\mathbf{x}} = \mathbf{G}(\mathbf{x})\mathbf{u} \tag{15}$$

where $\mathbf{u} = [v, \omega]^T$, $\mathbf{x} = [\tau_x, \tau_y, \theta]^T$ are the input and state vector, respectively. The sliding surface is taken as follows:

$$\mathbf{s} = \mathbf{x} - \mathbf{x}^d \tag{16}$$

where, $\mathbf{x}^d = [\tau_x^d, \tau_y^d, \theta^d]^T$ is the desired state vector. The control law that stabilizes the system can be designed as

$$\mathbf{u} = \mathbf{G}(\mathbf{x})^+(\dot{\mathbf{x}}^d - \mathbf{K} \tanh(\mathbf{s})) \tag{17}$$

where $\mathbf{G}(\mathbf{x})^+$ is a pseudo-inverse of $\mathbf{G}(\mathbf{x})$; $\mathbf{K} = \text{diag}\{k_1, k_2, k_3\}$; and k_1, k_2 and k_3 are positive gain constants. In Eq. (17), the conventionally used signum function has been replaced by tanh to avoid chattering.

Theorem 1. *For the given system with the kinematics as defined in (14), with sliding surface defined in (16), the control law (17) can make the overall system stable if arbitrary initial values of $\hat{a}$, $\hat{b}$ and $\hat{\theta}_0$ are updated by Eqs. (11)–(13), respectively.*

Proof. Let $\tilde{\mathbf{Y}}_1 = \tilde{\mathbf{Y}}_2 = \tilde{\mathbf{Y}}_3 = [\tilde{\tau}_{x_{k+1}}, \tilde{\tau}_{y_{k+1}}]^T$. The Lyapunov candidate is chosen as follows:

$$V = \frac{1}{2}\tilde{\mathbf{Y}}_1^T\tilde{\mathbf{Y}}_1 + \frac{1}{2}\tilde{\mathbf{Y}}_2^T\tilde{\mathbf{Y}}_2 + \frac{1}{2}\tilde{\mathbf{Y}}_3^T\tilde{\mathbf{Y}}_3 + \frac{1}{2}\mathbf{s}^T\mathbf{s} \tag{18}$$

The derivative of the function is found as

$$\dot{V} = \tilde{\mathbf{Y}}_1^T\dot{\tilde{\mathbf{Y}}}_1 + \tilde{\mathbf{Y}}_2^T\dot{\tilde{\mathbf{Y}}}_2 + \tilde{\mathbf{Y}}_3^T\dot{\tilde{\mathbf{Y}}}_3 + \mathbf{s}^T\dot{\mathbf{s}}$$
$$= \tilde{\mathbf{Y}}_1^T \frac{\partial \tilde{\mathbf{Y}}_1}{\partial \hat{a}_k}\dot{\hat{a}}_k + \tilde{\mathbf{Y}}_2^T \frac{\partial \tilde{\mathbf{Y}}_2}{\partial \hat{b}_k}\dot{\hat{b}}_k + \tilde{\mathbf{Y}}_3^T \frac{\partial \tilde{\mathbf{Y}}_3}{\partial \hat{\theta}_{0_k}}\dot{\hat{\theta}}_{0_k} + \mathbf{s}^T\dot{\mathbf{s}}$$

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

$$= \tilde{\mathbf{Y}}_1^T \begin{bmatrix} \frac{\partial \tilde{\tau}_{x_{k+1}}}{\partial \hat{a}_k} \\ \frac{\partial \tilde{\tau}_{y_{k+1}}}{\partial \hat{a}_k} \end{bmatrix} \dot{\hat{a}}_k + \tilde{\mathbf{Y}}_2^T \begin{bmatrix} \frac{\partial \tilde{\tau}_{x_{k+1}}}{\partial \hat{b}_k} \\ \frac{\partial \tilde{\tau}_{y_{k+1}}}{\partial \hat{b}_k} \end{bmatrix} \dot{\hat{b}}_k + \tilde{\mathbf{Y}}_3^T \begin{bmatrix} \frac{\partial \tilde{\tau}_{x_{k+1}}}{\partial \hat{\theta}_{0_k}} \\ \frac{\partial \tilde{\tau}_{y_{k+1}}}{\partial \hat{\theta}_{0_k}} \end{bmatrix} \dot{\hat{\theta}}_{0_k} + \mathbf{s}^T \dot{\mathbf{s}}$$

$$= \tilde{\mathbf{Y}}_1^T \begin{bmatrix} \Delta t \omega_k \sin \hat{\theta}_{0_k} \\ \Delta t \omega_k \cos \hat{\theta}_{0_k} \end{bmatrix} \dot{\hat{a}}_k + \tilde{\mathbf{Y}}_2^T \begin{bmatrix} -\Delta t \omega_k \cos \hat{\theta}_{0_k} \\ \Delta t \omega_k \sin \hat{\theta}_{0_k} \end{bmatrix} \dot{\hat{b}}_k + \tilde{\mathbf{Y}}_3^T \begin{bmatrix} \Delta t(\omega_k \hat{a}_k \cos \hat{\theta}_{0_k} + \omega_k \hat{b}_k \sin \hat{\theta}_{0_k} - v_k \sin \hat{\theta}_{0_k}) \\ -\Delta t(\omega_k \hat{a}_k \sin \hat{\theta}_{0_k} - \omega_k \hat{b}_k \cos \hat{\theta}_{0_k} + v_k \cos \hat{\theta}_{0_k}) \end{bmatrix} \dot{\hat{\theta}}_{0_k} + \mathbf{s}^T \dot{\mathbf{s}}$$

With the help of Eqs. (11)–(13), we get

$$\begin{aligned}
&= \tilde{\mathbf{Y}}_1^T \begin{bmatrix} \Delta t \omega_k \sin \hat{\theta}_{0_k} \\ \Delta t \omega_k \cos \hat{\theta}_{0_k} \end{bmatrix} (-\eta_1 \omega_k (\tilde{\tau}_{x_{k+1}} \sin \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \cos \hat{\theta}_{0_k})) + \tilde{\mathbf{Y}}_2^T \begin{bmatrix} -\Delta t \omega_k \cos \hat{\theta}_{0_k} \\ \Delta t \omega_k \sin \hat{\theta}_{0_k} \end{bmatrix} \\
&\quad (\eta_2 \omega_k (\tilde{\tau}_{x_{k+1}} \cos \hat{\theta}_{0_k} - \tilde{\tau}_{y_{k+1}} \sin \hat{\theta}_{0_k})) + \tilde{\mathbf{Y}}_3^T \begin{bmatrix} \Delta t(\omega_k \hat{a}_k \cos \hat{\theta}_{0_k} + \omega_k \hat{b}_k \sin \hat{\theta}_{0_k} - v_k \sin \hat{\theta}_{0_k}) \\ -\Delta t(\omega_k \hat{a}_k \sin \hat{\theta}_{0_k} - \omega_k \hat{b}_k \cos \hat{\theta}_{0_k} + v_k \cos \hat{\theta}_{0_k}) \end{bmatrix} \\
&\quad (-\eta_3 (\tilde{\tau}_{x_{k+1}} (\hat{a}_k \omega_k \cos \hat{\theta}_{0_k} + \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - v_k \sin \hat{\theta}_{0_k}) - \tilde{\tau}_{y_{k+1}} (\hat{a}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} \\
&\quad + v_k \cos \hat{\theta}_{0_k}))) + \mathbf{s}^T (\mathbf{G}(\mathbf{x})\mathbf{u} - \dot{\mathbf{x}}^d) \\
&= -\eta_1 \Delta t \omega_k^2 (\tilde{\tau}_{x_{k+1}} \sin \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \cos \hat{\theta}_{0_k})^2 - \eta_2 \Delta t \omega_k^2 (\tilde{\tau}_{x_{k+1}} \cos \hat{\theta}_{0_k} - \tilde{\tau}_{y_{k+1}} \sin \hat{\theta}_{0_k})^2 \\
&\quad - \eta_3 \Delta t (\tilde{\tau}_{x_{k+1}} (\hat{a}_k \omega_k \cos \hat{\theta}_{0_k} + \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - v_k \sin \hat{\theta}_{0_k}) - \tilde{\tau}_{y_{k+1}} (\hat{a}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} \\
&\quad + v_k \cos \hat{\theta}_{0_k}))^2 + \mathbf{s}^T (\dot{\mathbf{x}}^d - \mathbf{K} \tanh(\mathbf{s}) - \dot{\mathbf{x}}^d) \\
&= -\eta_1 \Delta t \omega_k^2 \|P_1\|^2 - \eta_2 \Delta t \omega_k^2 \|P_2\|^2 - \eta_3 \Delta t \|P_3\|^2 + \mathbf{s}^T (-\mathbf{K} \tanh(\mathbf{s})) \\
&\leq -\Delta t (\omega_k^2 (\eta_1 \|P_1\|^2 + \eta_2 \|P_2\|^2) + \eta_3 \|P_3\|^2) - \lambda_{\min}(\mathbf{K}) \, \|\mathbf{s}\|
\end{aligned}$$

where

$$\begin{aligned}
P_1 &= \tilde{\tau}_{x_{k+1}} \sin \hat{\theta}_{0_k} + \tilde{\tau}_{y_{k+1}} \cos \hat{\theta}_{0_k} \\
P_2 &= \tilde{\tau}_{x_{k+1}} \cos \hat{\theta}_{0_k} - \tilde{\tau}_{y_{k+1}} \sin \hat{\theta}_{0_k} \\
P_3 &= \tilde{\tau}_{x_{k+1}} (\hat{a}_k \omega_k \cos \hat{\theta}_{0_k} + \hat{b}_k \omega_k \sin \hat{\theta}_{0_k} - v_k \sin \hat{\theta}_{0_k}) - \tilde{\tau}_{y_{k+1}} (\hat{a}_k \omega_k \sin \hat{\theta}_{0_k} - \hat{b}_k \omega_k \cos \hat{\theta}_{0_k} + v_k \cos \hat{\theta}_{0_k})
\end{aligned}$$

$\because \eta_1, \ \eta_2, \ \eta_3, \ \lambda_{\min}(\mathbf{K}), \ \Delta t > 0 \implies \dot{V} \leq 0.$

A newly designed estimator is integrated along with sliding-mode controller as shown in Fig. 2 to solve two different problems in a single loop. These two problems are (1) to estimate all three body-to-camera parameters simultaneously and (2) to make a nonholonomic mobile robot to achieve its desired position and orientation. The proposed method achieves both these objectives in a single loop. Further, this scheme can detect and estimate unintended changes in body-to-camera parameters during operation. These are essential features make our approach unique and attractive in relation to other existing works.

5. Results and Discussions

The block diagram of the proposed scheme is shown in Fig. 2, which shows that the adaptive algorithm is run on the information obtained from the translation parameters ($\tau_x, \ \tau_y$); specifically, $\tau_x - \hat{\tau}_x$, $\tau_y - \hat{\tau}_y$ are the main inputs for the adaptation. Hence, if we do not estimate τ_x and τ_y, then adaptation could not proceed. The performance of the proposed approach is verified in a simulation environment first and then by real-time experiments to validate the robustness of the proposed scheme. The detailed analysis for the same is available in Sections 5.1 and 5.2. Comparison of the proposed approach with respect to other existing works is incorporated in Section 5.3.

5.1. Simulation

Initially, the algorithm for parameter estimation is tested. The initial estimates of a, b and θ_0 are selected at random. They are updated using the equations derived in Section 3. The model was provided with a series of different linear and angular velocities and the corresponding value of τ_x and τ_y

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

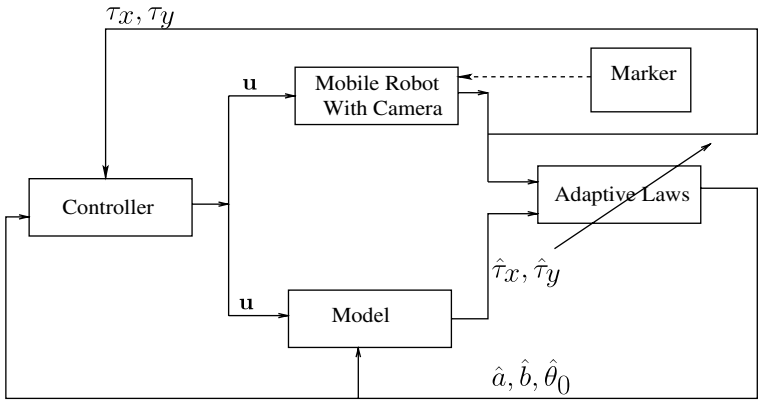


Fig. 2. Block diagram of the proposed scheme.

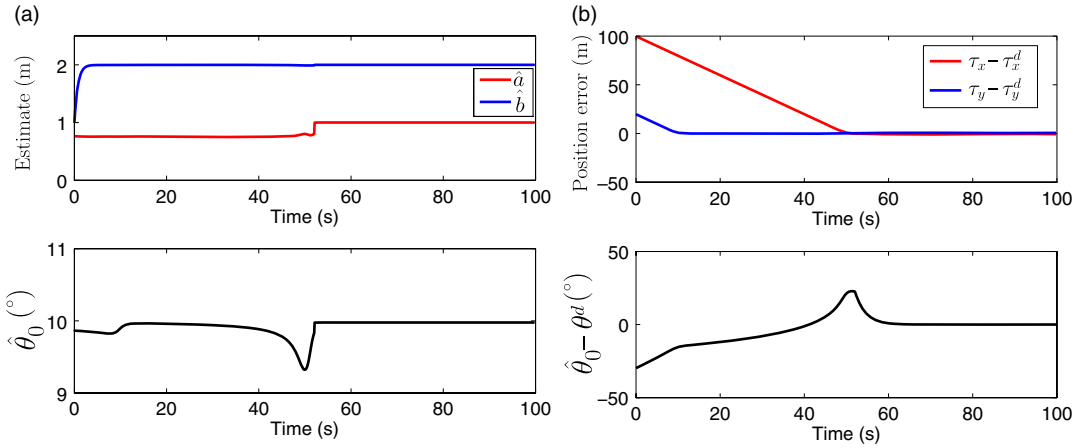


Fig. 3. Simulation: (a) estimation of a, b and θ_0 and (b) position and orientation error.

from the camera are utilized to update the parameters. The second part of the simulation consists of controlling the system to approach the desired position and orientation using the controller design, in Section 4. The desired position and orientation are taken as $[0\ \text{m}, 0\ \text{m}, 30°]^T$.

The final step of simulation consists of combining both the parts to complete the system. A true model was used to provide feedback of τ_x and τ_y. The values of a, b and θ_0 are set at 1 m, 2 m and 10°. The results for parameter estimation in the combined system can be seen in Fig. 3(a). The final estimated values are $\hat{a} = 1.0007$ m, $\hat{b} = 1.9996$ m and $\hat{\theta}_0 = 9.9773°$. Figure 4(a) shows how the desired point is reached from the reference frame of the mobile robot. The start position shows the location of the target point with respect to the mobile robot at $t = 0$ s, where t is the time dimension. The position and orientation error can be seen in Fig. 3(b). The errors approach zero, which is suggestive of the fact that the desired position and orientation are achieved. It has been observed that the convergence of parameter estimates occurs at a different rate from the convergence of the robot to its final position; even among the parameters themselves, a, b, θ_0 estimates converge at different rates due to geometrical constraints. Profiles of linear and angular velocities can be seen in Fig. 4(b).

In a different simulation, adaptive nature of the parameters is shown. Here, the desired values of a, b and θ_0 are changed in between, from $\theta_0 = 10°$, $b = 2$ m, $a = 1.4$ m to $\theta_0 = 5°$, $b = -1$ m, $a = 1.2$ m. The values of θ_0, b and a are altered at 15 s, 25 s and 50 s, respectively, from $t = 0$ s. With the help of Fig. 5(a), it can be seen that how the parameters adapt the new values fairly quickly. Therefore, it is suggestive of the fact that the camera to robot transformation can be obtained even when there is a change in the orientation of the camera or the camera slides to a different position when the robot is performing a task. The discontinuity occurs because of the abrupt change in the body-to-camera parameters, that is, a, b and θ_0. Further, we have studied the system for different levels of noise and bias. We added Gaussian white noise with zero mean and 0.2 variance, with $\hat{\tau}_x$ and $\hat{\tau}_y$. Estimates in

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

Table II. Final estimates of a, b and θ_0.

	Estimated value		
In the presence of	$\hat{a}$ **(m)**	$\hat{b}$ **(m)**	$\hat{\theta}_0$**(°)**
Noise	0.9745	2.1178	9.9380
Colored noise	1.0346	2.0240	9.9057
Bias = 0.01	0.9652	2.0244	9.9665
Bias = 0.05	0.7990	2.1329	10.5708
Bias = 0.08	0.6838	2.2086	10.7408

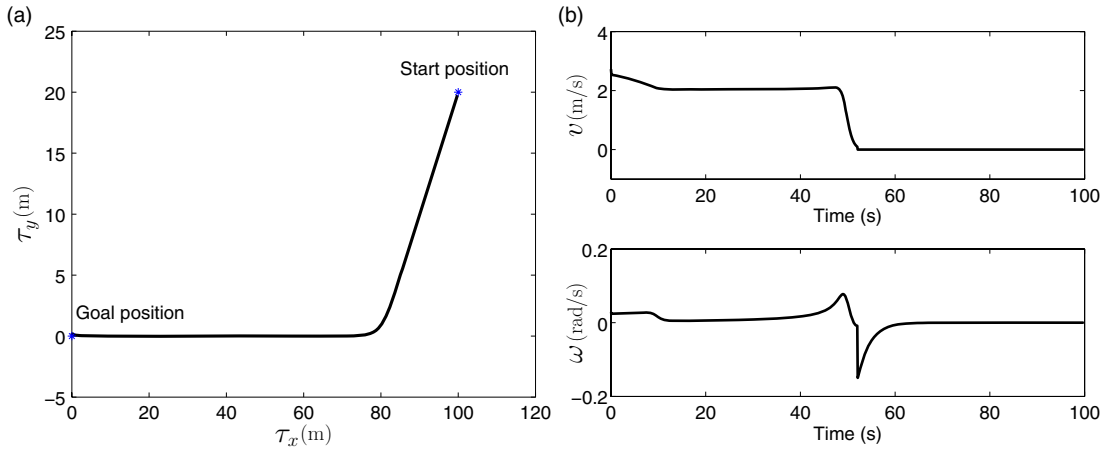


Fig. 4. Simulation: (a) motion of target point with respect to camera frame and (b) control inputs.

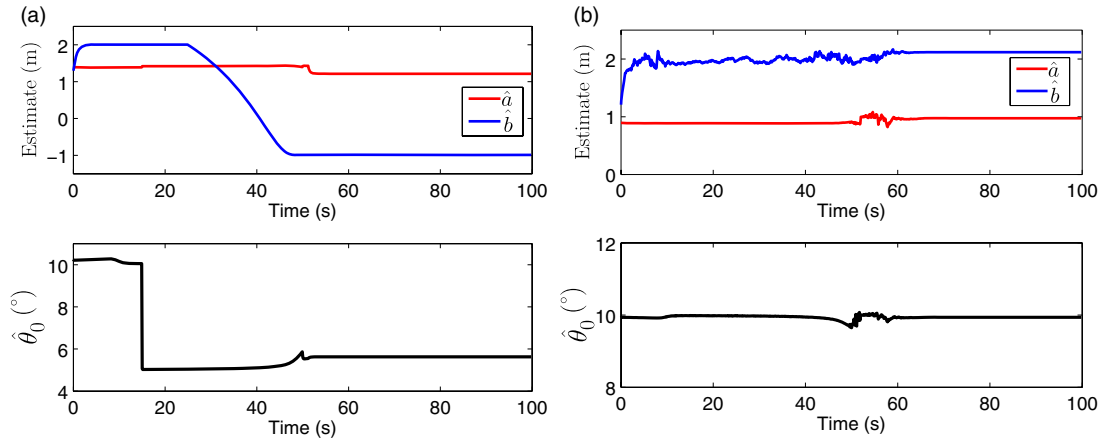


Fig. 5. Simulation: (a) adaptive nature of a, b and θ_0 and (b) estimation of a, b and θ_0 in the presence of noise.

the presence of noise are shown in Fig. 5(b). To analyze the effect of colored noise, a sequence of white Gaussian noise with zero mean and 0.82 variance is generated first and then this generated sequence is passed through a second-order Butterworth filter with a cutoff frequency of 0.3 Hz; after that, the output of low-pass filter is added with $\hat{\tau}_x$ and $\hat{\tau}_y$. The performance of the proposed scheme in the presence of colored noise can be seen in Fig. 6(a). To observe the effect of inaccuracy, we add the constant bias as 0.01 with $\hat{\tau}_x$ as well as with $\hat{\tau}_y$. In addition to this, we increased the value of bias and noted the obtained estimates. The final estimates of a, b and θ_0 in the presence of noise and bias are given in Table II. It can be inferred that as the value of bias increases, the error between final estimates and their respective true values ($a = 1$ m, $b = 2$ m, $\theta_0 = 10°$) increases. Estimates of a, b

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

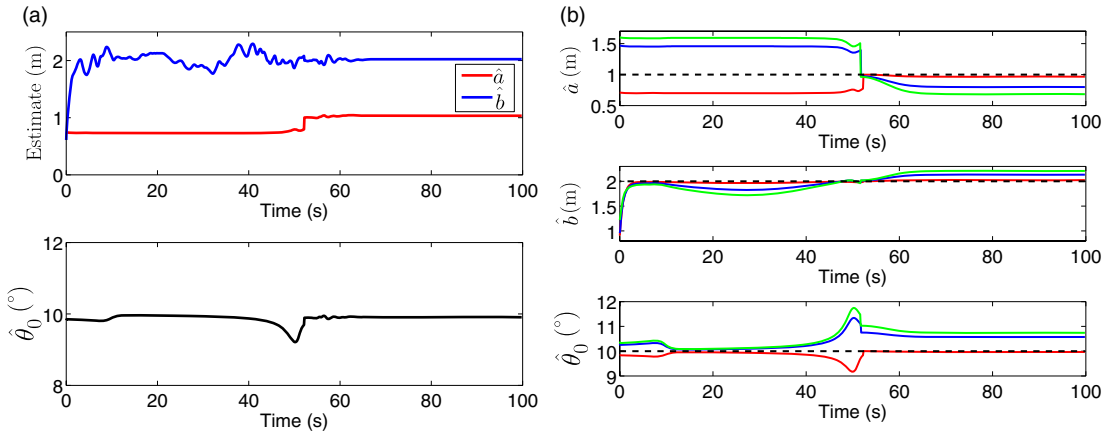
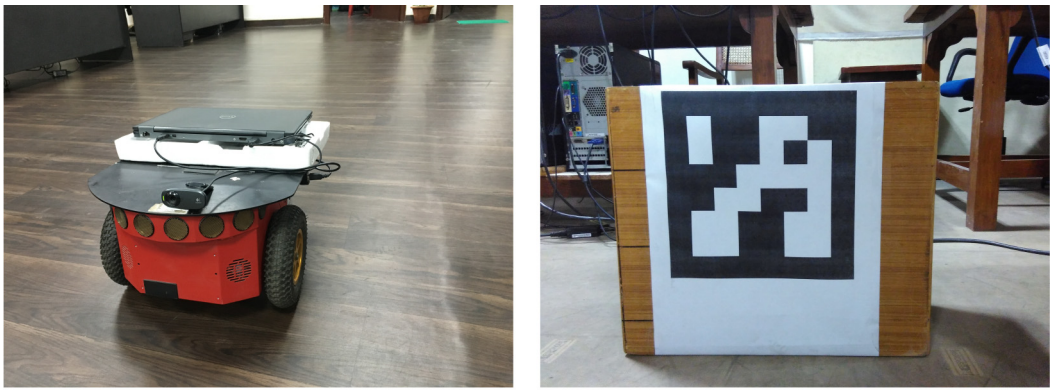


Fig. 6. Simulation in the presence of (a) colored noise: estimation of a, b and θ_0 and (b) bias: estimation of a, b and θ_0 (red line: using bias = 0.01; blue line: using bias = 0.05; green line: using bias = 0.08; dashed line: true value).



Robot with onboard camera and laptop Marker

Fig. 7. Experimental setup.

and θ_0 for the different levels of bias are shown in Fig. 6(b). It is noticed that the proposed method works well in the presence of noise and bias up to a limited point.

5.2. *Experiment*

The experimental setup consists of a Pioneer P3-DX mobile robot, an ArUco marker and a Logitech webcam. The robot was controlled using the Robot Operating System. The experimental setup is shown in Fig. 7. Let the point $(z = \tau_x + 2\,\text{m},\ \tau_y)$ define the position of the camera with respect to the ArUco marker. The initial and goal positions are set at $(\tau_x = 3.029\,\text{m},\ \tau_y = 1.452\,\text{m})$ and $(\tau_x = 0\,\text{m},\ \tau_y = 0\,\text{m})$, respectively. We begin by measuring τ_x and τ_y. This is achieved using ArUco marker.[32] An ArUco marker from a specific dictionary id (dictionary id = 9) is utilized while performing the experiments. The OpenCV library enables us to detect the position of the marker with respect to the camera frame. The readings obtained for τ_x and τ_y are very noisy, and may lead to improper parameter estimation. Therefore, the pose data from the camera are first filtered with a second-order Butterworth low-pass filter; 0.3 Hz cutoff frequency is used. The evolution of parameter estimates in this case can be seen in Fig. 8(a). The final values of the parameter estimates are $\hat{a} = 0.2154$ m and $\hat{b} = -0.0771$ m and $\hat{\theta}_0 = -8.0963°$. Since the odometry data are available at 10 Hz, we used sampling time $\Delta t = 0.1$s. The motion of the target point with respect to the camera frame can be seen in Fig. 8(b); there are perturbations that can be observed in the evolution of the point position. This is due to noise contents while estimating the pose using ArUco marker. For the controller, the error in x, y co-ordinate and robot orientation can be seen in Fig. 9(a); it can be seen that the error in robot orientation is as low as 2°. The final positioning error is $[0.024\,\text{m}, 0.096\,\text{m}]^T$. The linear velocity and

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

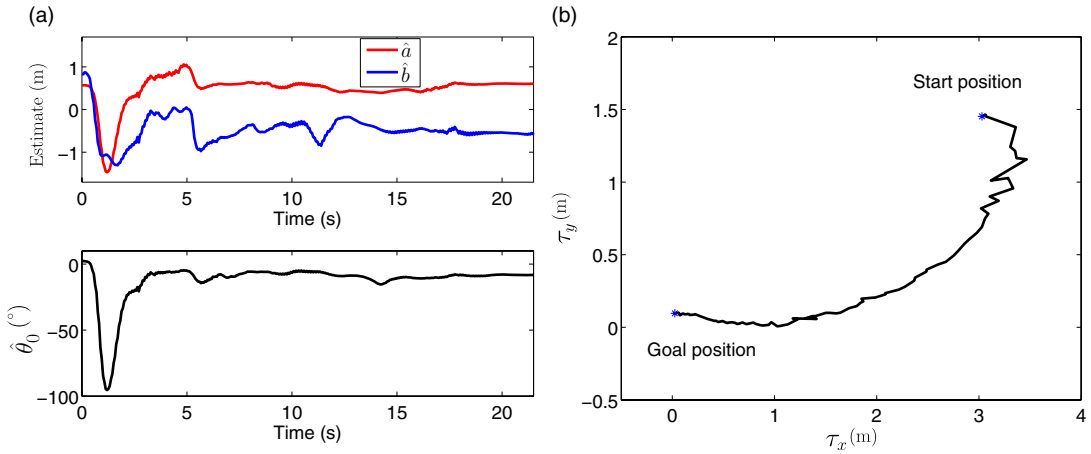


Fig. 8. Experiment: (a) estimation of a, b and θ_0 and (b) trajectory.

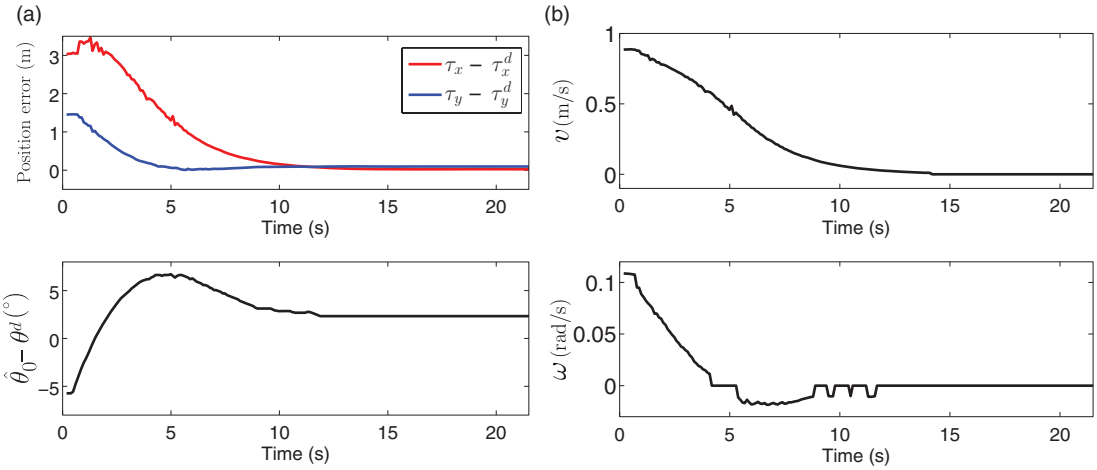


Fig. 9. Experiment: (a) position and orientation error and (b) control inputs.

angular velocity motion profile can be seen in Fig. 9(b). We have chosen the values of parameters (η_1, η_2, η_3, k_1, k_2 and k_3) heuristically and validated them using some trials. The estimation of θ_0 is performed using η_3 which differs from η_1 and η_2; all three learning parameters are assumed to be independent. The **YouTube** link for the video recorded during the experiment is given in ref. [34]. The taped video images are shown in Fig. 10.

5.3. Comparison

Let t_a, t_b and t_{θ_0} denote the convergence times for a, b and θ_0, respectively. In our case, the overall convergence time to estimate all three body-to-camera parameters is taken as $\max(t_a, t_b, t_{\theta_0})$, because all three parameters are estimated in a single trial. This is in contrast to Zhang et al.,[24] in which the overall convergence time to estimate all three body-to-camera parameters is $(t_{\theta_0} + \max(t_a, t_b))$, because, there, θ_0 needs to be estimated in a trial before the estimation of a and b. Hence, the proposed scheme is liable to be faster than Zhang et al.[24] Figure 11(a) suggests that the proposed method follows the shortest path as compared to Li et al.[33] Learning-based approaches are presented to localize and navigate a mobile robot in refs. [25–30]. In contrast to these work, this paper provides a unique solution to achieve both estimation (all three body-to-camera parameters) and robot control in a single loop. Eleven different trials have been executed to exhibit the effectiveness of the proposed method. Obtained trajectories can be seen in Fig. 11(a). Figure 12 indicates the confidence intervals for different trajectories and initial poses. Table I is included to give a point by point comparison of the proposed approach in the relation to Zhang et al.[24] and Li et al.[33]

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

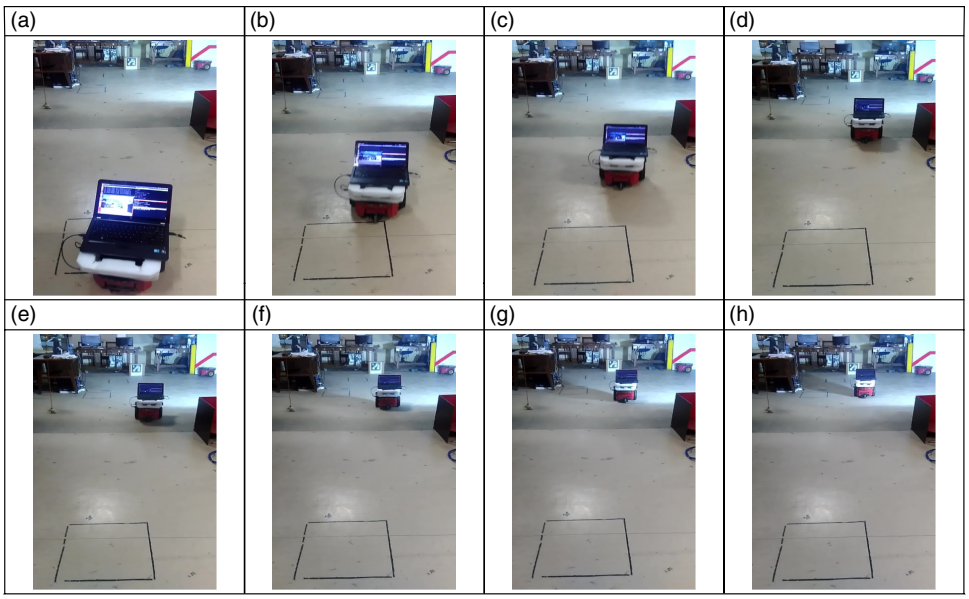


Fig. 10. Taped video image sequence.

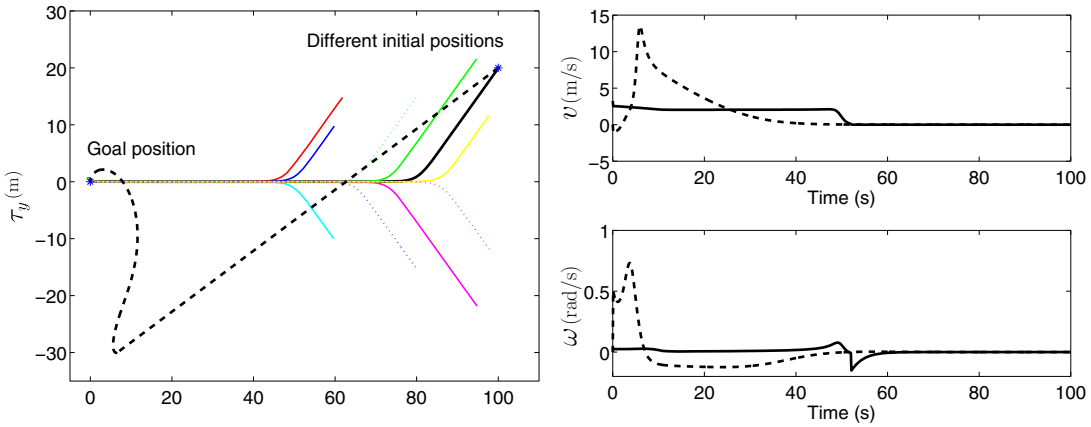


Fig. 11. Comparison: (a) trajectories (solid and dotted lines: by the proposed method; dashed black line: by Li et al.[33]) and (b) control inputs (solid line: by the proposed method; dashed line: by Li et al.[33]).

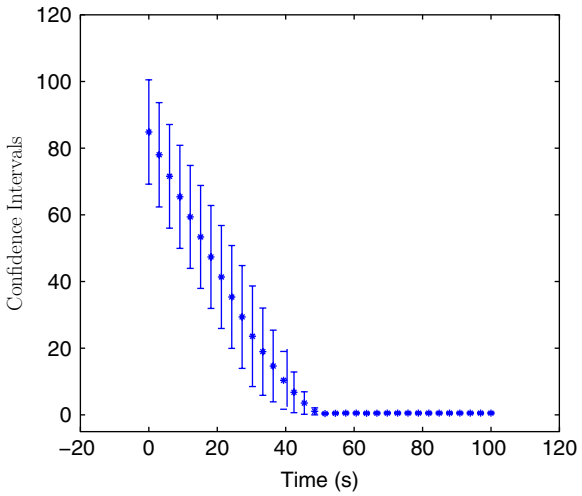


Fig. 12. Confidence intervals for different trajectories and initial poses.

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

6. Conclusion

We have proposed a PBVS scheme with the capacity to estimate body-to-camera parameters online. We showed that the robot continuously updates its parameters as it approaches the desired position. The controller successfully drives the robot to the desired point. The convergence has been proved mathematically. Some of the interesting features of the proposed scheme are that (i) this is a unique approach that achieves both control and estimation tasks in a single loop; (ii) it demands less time to estimate all three body-to-camera parameters as compared to the existing work by virtue of estimating all three parameters simultaneously; (iii) it handles effortlessly the possibility of accidental camera displacement with respect to the robot frame; and (iv) it works well in the presence of noise, various levels of bias and disturbance to the camera position. This greatly simplifies the task of visual servoing, as it allows us to arbitrarily locate the camera on the robot. We are still able to correctly predict both the translation and rotational components of the camera to robot frame, while controlling the system. Both simulation and experimental results prove that the method is attractive. It was further shown that, in case of a system disturbance which changes the camera to robot transformation, the changed parameters are effortlessly obtained. This paper is concerned with kinematic control and extrinsic calibration. It is assumed that the internal dynamic controller is able to follow the desired trajectory. Incorporation of vehicle dynamics is an interesting suggestion and has significant practical and theoretical value. Incorporation of vehicle dynamics would be taken up in the next work.

References

1. G. J. Agin, *Real Time Control of a Robot with a Mobile Camera* (SRI International, Menlo Park, California, 1979).
2. Y. Shirai and H. Inoue, "Guiding a robot by visual feedback in assembling tasks," *Pattern Recognit.* **5**(2), 99–108 (1973).
3. H. Wang, Y. H. Liu, W. Chen and Z. Wang, "A new approach to dynamic eye-in-hand visual tracking using nonlinear observers," *IEEE/ASME Trans. Mechatron.* **16**(2), 387–394 (2011).
4. M. Keshmiri, W. F. Xie and A. Mohebbi, "Augmented image-based visual servoing of a manipulator using acceleration command," *IEEE Trans. Ind. Electron.* **61**(10), 5444–5452 (2014).
5. S. Hutchinson, G. D. Hager and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. Rob. Autom.* **12**(5), 651–670 (1996).
6. F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Rob. Autom. Mag.* **13**(4), 82–90 (2006).
7. F. Janabi-Sharifi, L. Deng and W. J. Wilson, "Comparison of basic visual servoing methods," *IEEE/ASME Trans. Mechatron.* **16**(5), 967–983 (2011).
8. A. Cherubini, F. Chaumette and G. Oriolo, "Visual servoing for path reaching with nonholonomic robots," *Robotica* **29**(7), 1037–1048 (2012).
9. F. Chaumette and S. Hutchinson, "Visual servo control. II. Advanced approaches [tutorial]," *IEEE Rob. Autom. Mag.* **14**(1), 109–118 (2007).
10. D. Tsai, D. G. Dansereau, T. Peynot and P. Corke, "Image-based visual servoing with light field cameras," *IEEE Rob. Autom. Lett.* **2**(2), 912–919 (2017).
11. M. Hao and Z. Sun, "A universal state-space approach to uncalibrated model-free visual servoing," *IEEE/ASME Trans. Mechatron.* **17**(5), 833–846 (2012).
12. J. Pomares, I. Perea and F. Torres, "Dynamic visual servoing with chaos control for redundant robots," *IEEE/ASME Trans. Mechatron.* **19**(2), 423–431 (2014).
13. N. Guenard, T. Hamel and R. Mahony, "A practical visual servo control for an unmanned aerial vehicle," *IEEE Trans. Rob.* **24**(2), 331–340 (2008).
14. H. Romero, S. Salazar and R. Lozano, "Visual servoing applied to real-time stabilization of a multi-rotor UAV," *Robotica* **30**(7), 1203–1212 (2012).
15. T. Hamel and R. Mahony, "Visual servoing of an under-actuated dynamic rigid-body system: an image-based approach," *IEEE Trans. Rob. Autom.* **18**(2), 187–198 (2002).
16. D. J. Agravante, G. Claudio, F. Spindler and F. Chaumette, "Visual servoing in an optimization framework for the whole-body control of humanoid robots," *IEEE Rob. Autom. Lett.* **2**(2), 608–615 (2017).
17. Y. Fu, T. R. Hsiang and S. L. Chung, "Multi-waypoint visual homing in piecewise linear trajectory," *Robotica* **31**(3), 479–491 (2013).
18. M. Aranda, G. López-Nicolás and C. Sagüés, "Sinusoidal input-based visual control for nonholonomic vehicles," *Robotica* **31**(5), 811–823 (2013).
19. R. W. Brockett, "Asymptotic Stability and Feedback Stabilization," **In**: *Differential Geometric Control Theory*, vol. 27, no. 1 (Birkhauser, Boston, 1983) pp. 181–191.
20. Y. Fang, W. E. Dixon, D. M. Dawson and P. Chawda, "Homography-based visual servo regulation of mobile robots," *IEEE Trans Syst. Man Cybern. Part B (Cybernetics)* **35**(5), 1041–1050 (2005).
21. B. Li, Y. Fang and X. Zhang, "Visual servo regulation of wheeled mobile robots with an uncalibrated onboard camera," *IEEE/ASME Trans. Mechatron.* **21**(5), 2330–2342 (2016).

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press

22. Y. Fang, X. Liu and X. Zhang, "Adaptive active visual servoing of nonholonomic mobile robots," *IEEE Trans. Ind. Electron.* **59**(1), 486–497 (2012).
23. H. M. Becerra, G. López-Nicolás and C. Sagüés, "A sliding-mode-control law for mobile robots based on epipolar visual servoing from three views," *IEEE Trans. Rob.* **27**(1), 175–183 (2011).
24. X. Zhang, Y. Fang, B. Li and J. Wang, "Visual servoing of nonholonomic mobile robots with uncalibrated camera-to-robot parameters," *IEEE Trans. Ind. Electron.* **64**(1), 390–400 (2017).
25. J. S. Almeida, L. B. Marinho and J. W. M. Souza, "Localization system for autonomous mobile robots using machine learning methods and omnidirectional sonar," *IEEE Lat. Am. Trans.* **16**(2), 368-374 (2018).
26. L. B. Marinho, P. P. Rebouas Filho, J. S. Almeida, J. W. M. Souza, A. H. S. Junior and V. H. C. de Albuquerque, "A novel mobile robot localization approach based on classification with rejection option using computer vision," *Comput. Electr. Eng.* **68**, 26–43 (2018).
27. L. B. Marinho, J. S. Almeida, J. W. M. Souza, V. H. C. Albuquerque and P. P. Rebouas Filho, "A novel mobile robot localization approach based on topological maps using classification with reject option in omnidirectional images," *Expert Syst. Appl.* **72**, 1–17 (2017).
28. F. Sampaio, E. T. Silva, L. C. da Silva and P. P. Rebouas Filho, "An Embedded Classifier for Mobile Robot Localization Using Support Vector Machines and Gray-Level Co-occurrence Matrix," *International Conference on Artificial Intelligence and Soft Computing*, Zakopane, Poland (2018) pp. 201–213.
29. S. P. P. da Silva, L. B. Marinho, J. S. Almeida and P. P. Rebouas Filho, "A Novel Approach for Mobile Robot Localization in Topological Maps Using Classification with Reject Option from Structural Co-occurrence Matrix," *International Conference on Computer Analysis of Images and Patterns*, Ystad, Sweden (2017) pp. 3–15.
30. S. P. P. da Silva, R. V. M. da Nbrega, A. G. Medeiros, L. B. Marinho, J. S. Almeida and P. P. Reboucas Filho, " Localization of Mobile Robots with Topological Maps and Classification with Reject Option using Convolutional Neural Networks in Omnidirectional Images," *International Joint Conference on Neural Networks*, Rio, Brazil (2018) pp. 1–8.
31. X. Zhang, Y. Fang and X. Liu, "Motion-estimation-based visual servoing of nonholonomic mobile robots," *IEEE Trans. Rob.* **27**(6), 1167–1175 (2011).
32. S. Garrido-Jurado, R. Muoz-Salinas, F. J. Madrid-Cuevas and M. J. Marn-Jimnez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognit.* **47**(6), 2280–2292 (2014).
33. B. Li, X. Zhang, Y. Fang and W. Shi, "Visual servo regulation of wheeled mobile robots with simultaneous depth identification," *IEEE Trans. Ind. Electron.* **65**(1), 460–469 (2018).
34. "Position based visual servoing with online parameter estimation." https://youtu.be/IlIY5lynRgU.

https://doi.org/10.1017/S0263574719001115 Published online by Cambridge University Press