Neural Optimal Control for Constrained Visual Servoing via Learning From Demonstration

Prakash, Ravi; Behera, Laxmidhar

DOI
10.1109/TASE.2023.3272739

Publication date
2024

Document Version
Final published version

Published in
IEEE Transactions on Automation Science and Engineering

Neural Optimal Control for Constrained Visual Servoing via Learning From Demonstration

Ravi Prakash, *Member, IEEE*, and Laxmidhar Behera, *Senior Member, IEEE*

***Abstract*— This paper proposes a novel optimal control scheme for constrained image based visual servoing of a robot manipulator. For a robot manipulator with an eye-on-hand configuration, visibility constraint is an essential requirement to avoid servo failure, while robot's actuator limits must also be satisfied. To ensure this, the constraints are modelled implicitly via learning the task and defining safe regions using expert human demonstrations via mixture of Dynamic Movement Primitives (DMPs). The visual servoing problem is then formulated as a closed-loop optimal control problem using these constraint model where a desired target (possibly time-varying) is obtained by acting upon the feedback from the real-time visual sensors. The visual servo control loop consists of a single network adaptive critic optimal tracking control scheme whose weights are tuned using Lyapunov stability criteria. The stability and the performance of the proposed control scheme is shown theoretically via Lyapunov approach and also verified experimentally using a seven degree of freedom (DOF) Franka Emika and six DOF Universal Robot (UR) 10 manipulator. The approach is also demonstrated on a use case scenarios in mock-up convenience store and warehouse setup.**

***Note to Practitioners*—The applications of robots in busy warehouses, healthcare sectors and convenience store, has a big societal impact. The scenarios in these real-world problems often consist of a dynamic environment. Therefore, sensor-based localization and planning, along with satisfying robot and task constraints are needed. These naturally adds up to the programming costs in addition to the robot platform and actuation. However, modern robots need intuitive and easy programming for more pervasive in a practical societal application. In our proposed method, this sensor-based planning with environmental constraints is modelled implicitly using the Programming by Demonstration framework. The user needs to catch the robot arm by his hand and teach the task at hand, and the framework captures the task and robot constraints while generalizing to new goals. This modelling, along with cost optimal controller, generates real-time constraint aware robot manipulation trajectories for the demonstrated task in dynamic environments.**

Manuscript received 1 September 2022; revised 22 January 2023; accepted 21 April 2023. Date of publication 23 June 2023; date of current version 8 August 2024. This article was recommended for publication by Associate Editor H. Choi and Editor L. Zhang upon evaluation of the reviewers' comments. This work was supported in part by the Indian Institute of Technology Kanpur-Tata Consultancy Services (IITK-TCS) Collaborative Project under Reference TCS-EE-2019156, in part by the Indo-German Science and Technology Centre (IGSTC) Project under Reference IGSTC-EE-2019200, and in part by the European Research Council Starting Grant TERI under Reference 804907. *(Corresponding author: Ravi Prakash.)*

Ravi Prakash is with the Department of Cognitive Robotics, TU Delft, 2628 CD Delft, The Netherlands (e-mail: r.prakash-1@tudelft.nl).

Laxmidhar Behera is with IIT Mandi, Mandi 175005, India (e-mail: director@iitmandi.ac.in).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TASE.2023.3272739.

Digital Object Identifier 10.1109/TASE.2023.3272739

***Index Terms*— Neural optimal control, visual servoing, learning from demonstration, robot manipulator.**

I. Introduction

IN THE real world robotics applications, the robot task space is dynamic and unstructured. Therefore visual servo control plays an inevitable and a significant role in the real time planning and manipulation of the robot to achieve the desired task. Decades of research in this field has resulted in established methods of visual servo control. They can be categorized into position based visual servo (PBVS) [1], image based visual servo (IBVS) [2] and a combination of both [3], [4], [5]. A nice survey of visual servo control schemes can be found in [6] and [7]. Robustness to visual sensors noise and modelling errors make IBVS a popular choice [8], [9].

IBVS consists of finding control input to a robot manipulator such that a set of visual features as 2D parameters in image plane converges to the desired value of the visual features that may be a constant in case of a fixed target object or varying in case of moving target. The ease of implementing classical IBVS comes with its associated disadvantages of handling visual and kinematic constraints simultaneously causing undesirable behaviours. Therefore, the problem of visual servo control via IBVS has been investigated in the literature and can be categorized among different approaches.

One category of approach aims to design visual features for optimal visual servo control performances. A comprehensive analysis of the performance of the IBVS scheme for the choice of different visual features is shown in [10]. Some of the notable works in this category include hybrid $2\frac{1}{2}$ features [11], coordinate transformation [12], image moments [13] and fast interest points detector [14]. Although under controlled setting they resulted in improved robot motions, they do not guarantee constraint satisfaction.

Another category of approach consists of path planning and trajectory tracking for image based control while ensuring field of view (FOV) constraints [15], [16], [17], [18], [19], [20], [21]. On success, it generates optimized trajectories of the camera in the robot task space with satisfied visual constraints. However, there may not be any feasible robot tracking solution leading to failure cases. In another category, IBVS is formulated as a nonlinear optimization problem in the image plane with different kind of optimization solutions mainly based on numerical and convex solvers. The notable works include Linear Matrix Inequality (LMI) based convex optimization [22], [23], Neural Network (NN) convex optimization [24], safe numerical optimization using Barrier Function [25] and Model Predictive Control (MPC) [26], [27], [28].

Fig. 1. Figure shows a) modelling constraint-aware visual servo task using expert demonstration and b) execution for novel situations using optimal control framework.

The numerical solutions are generally not so reliable and has stability and efficiency issue. Optimum step size needs to be ensured to avoid failure due to large step divergence. The convex optimization formulation allows integration of visual and kinematic constraints which are modeled using rigorous mathematical formulations. The solutions however assumes local convexity assumptions and may require computationally expensive matrix inversion to actuate the robot manipulator. To simplify the difficulty of complex mathematical modelling visual and kinematic constraints are modeled via Learning from Demonstration (LfD) approach [23], [29] in a Lagrangian based convex optimization setup.

In this paper, we formulate the IBVS as non-linear optimization where we propose a complementary approach by combining constraints modelling via LfD and Hamilton-Jacobi-Bellman (HJB) based optimal control solution which is a necessary and sufficient condition for global optimality of a control solution with respect to a cost function. Earlier approaches of modelling constraints using LfD employs statistical model from human demonstrations. Although it is an interesting way to autonomously extract the important features of the task from multiple demonstrations, requesting multiple demonstrations of a single task would annoy the user. Moreover, this formulation is not robust to perturbations (e.g. dynamic environment including time varying target object). Instead, LfD approaches based on dynamical system known as Dynamic Movement Primitives (DMPs) offer a particularly interesting solution to model human demonstrations while being robust to perturbations or dynamical changes in the environment (e.g. time varying target object) due to its stable attractor dynamics. This allows them to be suitable visual servo control for both fixed and time varying image features. They are so designed that from a single demonstration they can model the constraint and replay the task trajectories for novel goals. They support incremental learning approach unlike statistical approaches and therefore the models can be further improved on the fly. In the proposed scheme, we model the visual servo constraints via DMPs by leveraging the user's sensory capabilities and the optimization using these models is then solved using HJB formulation. The stability of the overall closed loop system is ensured using Lyapunov stability. The modules of modelling the task and optimal execution of visual servo task is depicted in Fig. (1).

The novel contributions of this work is summarized next,

1) Visual and Kinematic constraint modelling using dynamic LfD approach which requires one shot learning form single demonstration and generalises to static as well as moving targets.
2) HJB based optimal control solution to constrained visual servo control with guarenteed stability.
3) Hardware validations of the proposed scheme for both fixed and time varying target object along with comparative experimental study of the efficacy of the proposed control w.r.t similar control schemes..

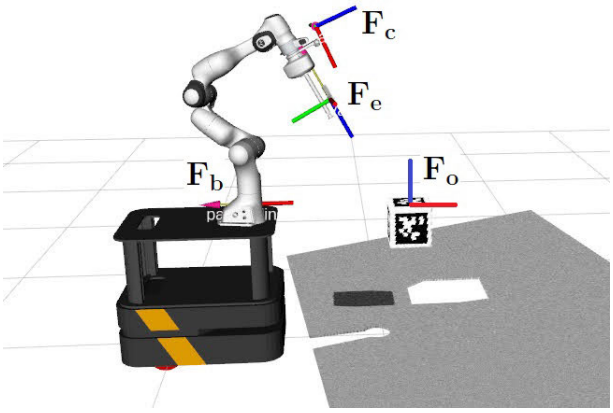


Fig. 2. The figure shows the frame assignment F_b, F_e, F_c and F_o to the robot base, the robot end-effector, the camera and the target object respectively.

The remainder of this paper is organized as follows. In Section II, details of the system models are presented. In Section III, visual servo control problem is formulated. In Section IV, novel NN optimal control design is presented along with its stability analysis. In Section V, experimental results are presented to validate the effectiveness of the proposed method and its superiority in performance compared with the existing methods. The paper is concluded in Section VI.

II. System Modeling

Let us assign the following coordinate frames for system modelling. Figure (2) shows the frames F_b, F_e, F_c and F_o is attached to the robot base, the robot end effector, the camera and the target object respectively. ${}^{2}\mathbf{T}_{1}$ represents the homogenous transformation which expresses the coordinates of the frame F_1 in the coordinates of frame F_2.

A. Object Model

An object model consists of a number of $3D$ feature points whoose coordinates $({}^{o}X_i, {}^{o}Y_i, {}^{o}Z_i)$ are defined in F_o. Object model for a target with n features is ${}^{o}\mathbf{P}_i = [{}^{o}X_i \;\; {}^{o}Y_i \;\; {}^{o}Z_i \;\; 1]^{\intercal}$ $i \in [1 \;\; n]$.

B. Robot Model

Typically, in a visual servo control applications, a kinematic robot model suffices due to the slower visual outer loop with respect to the faster inner control loop of the robot which stabilizes the dynamics. Therefore, in this work the kinematic model of an N independent joint actuated robot manipulator is used which is given by ${}^{b}\mathbf{T}_{e}(\boldsymbol{\theta}) = \mathbf{f}(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the joint angle vector of the robot manipulator. The derivative of

kinematic model of the robot manipulator with respect to its joint angle vector is the robot manipulator Jacobian (${}^b\mathbf{J}_r(\boldsymbol{\theta})$) i.e. ${}^b\mathbf{J}_r(\boldsymbol{\theta}) = \begin{bmatrix} {}^b\mathbf{J}_v(\boldsymbol{\theta}) \\ {}^b\mathbf{J}_\omega(\boldsymbol{\theta}) \end{bmatrix} = \frac{d\mathbf{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}$, where, ${}^b\mathbf{J}_v(\boldsymbol{\theta})$ and ${}^b\mathbf{J}_\omega(\boldsymbol{\theta})$ are linear and rotational components of the robot manipulator Jacobian. The control input to the robot manipulator is the joint velocity vector i.e.

$$\mathbf{u}(t) = \dot{\boldsymbol{\theta}}(t) \tag{1}$$

C. Image Feature Model

For visual servo application, a camera is mounted on the robot end-effector in eye-in-hand configuration. The camera projected image coordinates (u_i, v_i) corresponding to the object feature points as a function of $\boldsymbol{\theta}$ can be modelled as

$$\mathbf{s}_i(\boldsymbol{\theta}) = \begin{pmatrix} p_{ui} \\ p_{vi} \\ 1 \end{pmatrix} = \mathbf{C}[\mathbf{I}_{3\times3}|\mathbf{0}_{3\times1}]{}^c\mathbf{T}_e{}^e\mathbf{T}_b{}^b\mathbf{T}_o{}^o\mathbf{P}_i, \tag{2}$$

where, $\mathbf{C} = \begin{bmatrix} \kappa_u & \kappa_{uv} & u_0 \\ 0 & \kappa_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$ is the 3×3 camera matrix [30].

Finally, the image coordinate vector $\mathbf{s}$ is used for visual servo control is given by

$$\mathbf{s} = [(\mathbf{s}_1)^\intercal \quad (\mathbf{s}_2)^\intercal \quad \cdots \quad (\mathbf{s}_n)^\intercal]^\intercal. \tag{3}$$

D. Image Feature Velocity Model

The image coordinates vector ($\mathbf{s}$) of the feature points have a Jacobian relationship with $\boldsymbol{\theta}$ at the velocity level given by

$$\dot{\mathbf{s}}(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}(t) = \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}(t). \tag{4}$$

The Jacobian $\mathbf{J}_s(\mathbf{s}, \boldsymbol{\theta})$ has the following form

$$\begin{aligned} \mathbf{J}_s(\boldsymbol{\theta}) &= \begin{bmatrix} \kappa_u & \kappa_{uv} \\ 0 & \kappa_v \end{bmatrix}\mathbf{L}(\mathbf{x}, \mathbf{y}, \mathbf{Z})\begin{bmatrix} {}^c\mathbf{J}_v(\boldsymbol{\theta}) \\ {}^c\mathbf{J}_\omega(\boldsymbol{\theta}) \end{bmatrix} \text{ or,} \\ \mathbf{J}_s(\boldsymbol{\theta}) &= \begin{bmatrix} \kappa_u & \kappa_{uv} \\ 0 & \kappa_v \end{bmatrix}\mathbf{L}(\mathbf{x}, \mathbf{y}, \mathbf{Z})\begin{bmatrix} {}^c\mathbf{R}_b(\boldsymbol{\theta}) & \mathbf{0} \\ \mathbf{0} & {}^c\mathbf{R}_b(\boldsymbol{\theta}) \end{bmatrix}\begin{bmatrix} {}^b\mathbf{J}_v(\boldsymbol{\theta}) \\ {}^b\mathbf{J}_\omega(\boldsymbol{\theta}) \end{bmatrix}, \end{aligned} \tag{5}$$

${}^c\mathbf{R}_b(\boldsymbol{\theta})$ is the rotation matrix expressing coordinates of F_b in the coordinates of F_c and $\mathbf{L}(\mathbf{x}, \mathbf{y}, \mathbf{Z})$ is the image interaction matrix [6]. The sub-matrix $\mathbf{L}_i(x_i, y_i, Z_i)$ corresponding to each feature point (x_i, y_i) at a depth of Z_i is given by

$$\mathbf{L}_i = \begin{bmatrix} -\frac{1}{Z_i} & 0 & \frac{x_i}{Z_i} & x_i y_i & -(1 + {y_i}^2) & y_i \\ 0 & -\frac{1}{Z_i} & \frac{y_i}{Z_i} & (1 + {x_i}^2) & -x_i y_i & -x_i \end{bmatrix},$$

where, the feature point coordinates $[x_i \quad y_i \quad 1]^\intercal$ is related to its image counterparts $[p_{u_i} \quad p_{v_i} \quad 1]^\intercal$ by the camera matrix $\mathbf{C}$.

E. Task and Constraints Model

In visual servo applications, robot visual and kinematic constraints are essential requirements. Kinematic constraints are modelled using lower and upper limits on the robot joint positions $[\theta^{min} \quad \theta^{max}]$ and joint velocities $[\dot{\theta}^{min} \quad \dot{\theta}^{max}]$. Visual constraints are modelled using camera's field of view constraints for a target object with n features, where $[p_u^{min} \quad p_u^{max}]$ and $[p_v^{min} \quad p_v^{max}]$ are limits in horizontal and vertical directions.

In this work, the constraints are modelled implicitly via learning the task and defining safe regions using expert human demonstrations or LfD approach. The expert demonstrator's knowledge and sensing capabilities are leveraged to model these constraints in the visual servo control problem. An expert demonstrator guides the robot via kinesthetic teaching towards a general target object while ensuring kinematic and visual constraints. Image space and joint space trajectories are extracted from the demonstration. These trajectories are used to train DMPs. The task as well as the constraints are captured using the learnt shape parameters in the trained DMPs. In this framework, each degree of freedom $(n + N)$ in the demonstrated movement is modelled using the modified motion model [31].

For the generalization of the generated image and joint trajectories in novel situations, a single demonstrated trajectory is not sufficient because of the changing profiles across varying goals. Therefore, a library of many demonstrated trajectories along with a decision making skill is needed. Given the start and goal configuration of the particular signal in new condition represented by the vector δ, the formulation for optimizing the policy for trajectory generation is given by

$$\bar{\pi} = \frac{\sum_{i=1}^{N_p} \mu_i(\delta)\pi_i(\delta)}{\sum_{i=1}^{N_p} \mu(\delta)}, \tag{6}$$

where, π_i is the generated trajectories from DMP corresponding to i^{th} of the total of N_p expert demonstration for the new configuration δ, and μ_i is the design parameter. The trajectories generated by following the policy (6) serve as desired image ($\mathbf{s}_d(\boldsymbol{\theta})$) and joint trajectories ($\boldsymbol{\theta}_d(t), \dot{\boldsymbol{\theta}}_d(t)$).

III. Visual Servo Control Problem Formulation

In this section, the optimal visual servo control of a robot manipulator is formulated as a non-linear optimal control problem, and is addressed such that the image feature tracking error given by

$$\mathbf{e}(\boldsymbol{\theta}) = \mathbf{s}(\boldsymbol{\theta}) - \mathbf{s}_d(\boldsymbol{\theta}) \tag{7}$$

reduces to zero with time by following an optimal trajectory obeying visual and kinematic constraints. Here, $\mathbf{s}_d(t) \in \mathbf{R}^{n\times1}$ is the desired image feature points trajectory. For tracking purpose, we assume that the desired trajectory possesses the dynamics given by

$$\dot{\mathbf{s}}_d(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}_d(t) = \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_d(t). \tag{8}$$

The time derivative of the image feature tracking error is given by

$$\begin{aligned}\dot{\mathbf{e}}(\boldsymbol{\theta}) &= \dot{\mathbf{s}}(\boldsymbol{\theta}) - \dot{\mathbf{s}}_d(\boldsymbol{\theta}) \\ &= \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_e(t),\end{aligned} \tag{9}$$

where $\mathbf{u}_e(t) = \mathbf{u}(t) - \mathbf{u}_d(t)$.

Considering the above system error dynamics (9) with $\boldsymbol{\theta}(t)$ as the system states, the optimal control law is derived next. The optimal control problem for the system (9) is to find an optimal control $\mathbf{u}_e(t)$ that minimizes a cost-to-go function of state and control variables.

Then assume that there exists a continuous cost-to-go value function $V(\boldsymbol{\theta}(t), \mathbf{u}_e(t)$ for the constrained visual servo control problem defined as

$$V(\boldsymbol{\theta}(t), \mathbf{u}_e(t) = \int_t^{\infty} (\mathbf{e}(\boldsymbol{\theta})^T \mathbf{Q}\mathbf{e}(\boldsymbol{\theta}) + \mathbf{u}_e(t)^{\intercal}\mathbf{R}\mathbf{u}_e(t))d\tau \tag{10}$$

where $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{N\times N}$ are positive definite matrices, $\mathbf{e}(\boldsymbol{\theta})^T\mathbf{Q}\mathbf{e}(\boldsymbol{\theta})$ is a positive semi-definite function and stands for the state cost, $\mathbf{u}_e(t)^{\intercal}\mathbf{R}\mathbf{u}_e(t)$ is a positive semi-definite function and stands for the control cost. The control policy $\mathbf{u}_e(t)$ is defined to be admissible with respect to a given cost function (10), such that it stabilizes the system (9) and the cost function (10) is finite. Let $V^*(\mathbf{e}(\boldsymbol{\theta}))$ be the optimal cost function defined by

$$V^*(\boldsymbol{\theta}(t)) = \min_{\mathbf{u}_e} \left\{V(\boldsymbol{\theta}(t), \mathbf{u}_\mathbf{e}^*(t))\right\}. \tag{11}$$

Let $H(\boldsymbol{\theta}, \mathbf{u}_e, V_{\boldsymbol{\theta}})$ be the Hamiltonian function corresponding to the cost function (10) and the admissible control input $\mathbf{u}_e(t)$ defined by

$$\begin{aligned}H(\boldsymbol{\theta}, \mathbf{u}_e, V_{\boldsymbol{\theta}}) &= \mathbf{e}(\boldsymbol{\theta})^T\mathbf{Q}\mathbf{e}(\boldsymbol{\theta}) + \mathbf{u}_e(t)^{\intercal}\mathbf{R}\mathbf{u}_e(t) + V_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \\ H(\boldsymbol{\theta}, \mathbf{u}_e, V_{\boldsymbol{\theta}}) &= \mathbf{e}(\boldsymbol{\theta})^T\mathbf{Q}\mathbf{e}(\boldsymbol{\theta}) + \mathbf{u}_e(t)^{\intercal}\mathbf{R}\mathbf{u}_e(t) \\ &\quad + V_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_e(t)\end{aligned} \tag{12}$$

where $V_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \frac{\partial V(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$. The optimal control input for the given problem is obtained by employing the stationarity condition [32] on the Hamiltonian function and is given by $\frac{\partial H(\boldsymbol{\theta}, \mathbf{u}_e, V_{\boldsymbol{\theta}}^*)}{\partial \mathbf{u}_e} = 0$. The optimal control policy then equals

$$\mathbf{u}_e^*(t) = -\frac{1}{2}\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})V_{\boldsymbol{\theta}}^{*\intercal}(\boldsymbol{\theta}) \tag{13}$$

And correspondingly, the optimal control input to the robot manipulator is given by

$$\mathbf{u}^*(t) = -\frac{1}{2}\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})V_{\boldsymbol{\theta}}^{*\intercal}(\boldsymbol{\theta}) + \mathbf{u}_d(t) \tag{14}$$

The visual servo control of the robot manipulator with visual and kinematic constraints is solved by providing the optimal control (14) to the robot manipulator. However, the term $V^*(\boldsymbol{\theta})$ or $V_{\boldsymbol{\theta}}^*(\boldsymbol{\theta})$ is generally unavailable and thus we propose a NN based estimate to find the corresponding control input.

In the next section, the NN based optimal critic design is proposed to realize the control input (13).

IV. NN Optimal Visual Servo Controller

A. NN Critic Design

Owing to the universal approximation property of NN, the cost function (11) is estimated using a NN as follows,

$$V^*(\boldsymbol{\theta}) = \mathbf{W}^{\intercal}\boldsymbol{\phi}(\boldsymbol{\theta}) + \varepsilon(\boldsymbol{\theta}) \tag{15}$$

where, $\mathbf{W} \in \mathbb{R}^L$ is the ideal neural network weight vector, $\boldsymbol{\phi}(\cdot) : \mathbb{R}^n \to \mathbb{R}^L$ is a linearly independent basis vector which satisfies $\boldsymbol{\phi}(0) = 0$, and $\varepsilon(\boldsymbol{\theta})$ is the neural network reconstruction errors are assumed to be upper bounded according to $\| \mathbf{W} \| \leq \mathbf{W}_M$ and $\| \varepsilon(\boldsymbol{\theta}) \| \leq \varepsilon_M$, respectively [32]. In addition, it will be assumed that the gradient of the neural network reconstruction error with respect to $\boldsymbol{\theta}$ is upper bounded according to $\| \frac{\partial \varepsilon(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \| = \| \varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \| \leq \varepsilon'_M$. Moving on, the gradient of the NN cost function (15) is

$$\frac{\partial V^*(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = V_{\boldsymbol{\theta}}^*(\boldsymbol{\theta}) = \boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{W} + \varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \tag{16}$$

where, $\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \frac{\partial \boldsymbol{\phi}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$. Now. using (16), the control (13) is rewritten as

$$\mathbf{u}_e^*(\boldsymbol{\theta}) = -\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{W}/2 - \mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})/2 \tag{17}$$

A NN critic is designed next to approximate the optimal cost function:

$$\widehat{V}^*(\boldsymbol{\theta}) = \widehat{\mathbf{W}}^{\intercal}\boldsymbol{\phi}(\boldsymbol{\theta}) \tag{18}$$

where $\widehat{\mathbf{W}}$ is the estimate of ideal weight $\mathbf{W}$. Therefore, the NN estimate for the optimal control (17) is given by

$$\widehat{\mathbf{u}}_e^*(\boldsymbol{\theta}) = -\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\widehat{\mathbf{W}}/2 \tag{19}$$

Modified system error dynamics may be obtained by substituting (19) into the nonlinear system (9) reveals

$$\dot{\mathbf{e}}(\boldsymbol{\theta}) = -\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\widehat{\mathbf{W}}/2 \tag{20}$$

Noting the cost function and control law in terms of NN weight vector, the Hamiltonian is expressed as a function of $\boldsymbol{\theta}$ and $\mathbf{W}$ as,

$$H(\boldsymbol{\theta}, \mathbf{W}) = \mathbf{e}(\boldsymbol{\theta})^{\intercal}\mathbf{Q}\mathbf{e}(\boldsymbol{\theta}) - \frac{1}{4}\mathbf{W}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} + e_\varepsilon \tag{21}$$

where $\mathbf{A}(\boldsymbol{\theta}) = \boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})$ and the last term representing residual error of neural network expression is given by

$$\begin{aligned}e_\varepsilon = &-\frac{1}{2}\varepsilon_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{W} \\ &- \frac{1}{4}\varepsilon_{\boldsymbol{\theta}}^{\intercal}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})\end{aligned} \tag{22}$$

Corresponding to the estimated NN weight vector, the estimated hamiltonian may be obtained as

$$\widehat{H}(\boldsymbol{\theta}, \widehat{\mathbf{W}}) = \mathbf{e}(\boldsymbol{\theta})^{\intercal}\mathbf{Q}\mathbf{e}(\boldsymbol{\theta}) - \frac{1}{4}\widehat{\mathbf{W}}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\widehat{\mathbf{W}} \tag{23}$$

Let us define Hamiltonian error using (21) and (23) as

$$\begin{aligned}e_H &= \widehat{H}(\boldsymbol{\theta}, \widehat{\mathbf{W}}) - H(\boldsymbol{\theta}, \mathbf{W}) \\ &= -\frac{1}{4}\widetilde{\mathbf{W}}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} + \frac{1}{2}\widetilde{\mathbf{W}}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} - e_\varepsilon\end{aligned} \tag{24}$$

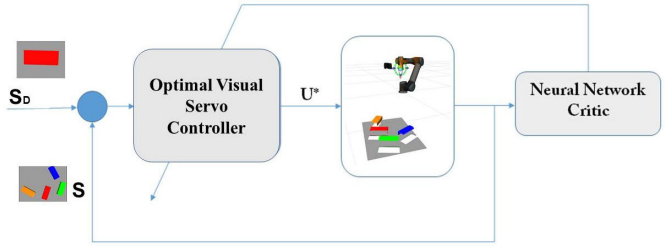


Fig. 3. Optimal visual servo control scheme.

where, $\widetilde{\mathbf{W}} = \mathbf{W} - \widehat{\mathbf{W}}$ is the error between the target and estimated weight vectors of the neural network.

Next, we train the critic neural network such that the hamiltonian error function $E = \frac{1}{2}e_H^2$ is minimised. The update law for tuning the NN weights is found by

$$\dot{\widehat{\mathbf{W}}} = -\alpha_c e_H \frac{\partial e_H}{\partial \widehat{\mathbf{W}}} + \frac{1}{2}\alpha_e \boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\mathbf{e} \tag{25}$$

$$= -\alpha_c\left(-\frac{1}{4}\widetilde{\mathbf{W}}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} + \frac{1}{2}\widetilde{\mathbf{W}}^{\intercal}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} - e_{\varepsilon}\right) \times \left(-\frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} + \frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}}\right) + \frac{1}{2}\alpha_e \boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^{\intercal}(\boldsymbol{\theta})\mathbf{e} \tag{26}$$

The update law consists of two components. The first component (25) minimizes the error function $E = \frac{1}{2}e_H^2$ and the second component ensures the system error to be bounded while the critic the optimal cost function. The learning parameters $\alpha_c > 0$ is the critic learning rate and $\alpha_e > 0$ is the gain of the stabilizing term.

B. Stability Analysis

We present following lemmas and assumption needed in the stability analysis of the controller.

Lemma 1: For the system dynamics represented by (9), with cost function (10), let $J(\mathbf{e}) = \frac{1}{2}\mathbf{e}^{\intercal}(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta})$ be a continuously differentiable Lyapunov function candidate satisfying

$$\dot{J}(\mathbf{e}) = \mathbf{e}^{\intercal}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) < 0, \tag{27}$$

where $\dot{\mathbf{e}}^*(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_e^*(t)$. Then, there exists a positive definite matrix, $\mathbf{M}$ of appropriate dimension ensuring,

$$\mathbf{e}^{\intercal}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) = -\mathbf{e}^{\intercal}(\boldsymbol{\theta})\mathbf{M}\mathbf{e}(\boldsymbol{\theta}) \leq -\alpha_{min}(\mathbf{M})\|\mathbf{e}(\boldsymbol{\theta})\|^2 \tag{28}$$

Proof: Please refer to Appendix A. □

Assumption 1: The Jacobian $\mathbf{J}_s(\boldsymbol{\theta})$ is bounded as $\|\mathbf{J}_s(\boldsymbol{\theta})\| \leq \alpha_{\mathbf{J}_s}$. The terms $\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta}), \boldsymbol{\varepsilon}_{\boldsymbol{\theta}}(\boldsymbol{\theta}), \dot{\mathbf{e}}^*, \mathbf{A}(\boldsymbol{\theta}), \mathbf{M}$ are bounded as $\|\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\| \leq \alpha_{\boldsymbol{\phi}}$, $\|\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\| \leq \alpha_{\varepsilon}$, $\|\dot{\mathbf{e}}^*\| \leq \alpha_{\dot{\mathbf{e}}^*}$, $\|\mathbf{A}(\boldsymbol{\theta})\| \leq \alpha_{\mathbf{A}}$, $\|\mathbf{M}\| \leq \alpha_{\mathbf{M}}$ where $\alpha_{\mathbf{J}_s}, \alpha_{\boldsymbol{\phi}}, \alpha_{\varepsilon}, \alpha_{\dot{\mathbf{e}}^*}, \alpha_{\mathbf{A}}, \alpha_{\mathbf{M}}$ are positive constants. Note: These are standard assumptions in NN based control of robot manipulators [33], [34].

Theorem 1: For the task of optimal constrained robot manipulator visual servo control using the proposed control shown in Fig. (3)) with NN control law given by (19) and NN weight update law as (26), the error state vector of the closed-loop system and the critic weight estimation error are all uniformly ultimately bounded (UUB). The approximate controller $\widehat{\mathbf{u}}_e^*(\boldsymbol{\theta})$ converges to a neighbourhood of the optimal feedback controller $\mathbf{u}_e^*(\boldsymbol{\theta})$ with a finite bound.

Proof: Please refer to Appendix B. The theoretical maximum bounds on the error state vector of the closed-loop system and the critic weight error vector can be estimated. In the current case study of proposed control of 7 DOF robot manipulator, the calculation of above mentioned bounds is also summarized in the Appendix. □

V. Results and Discussion

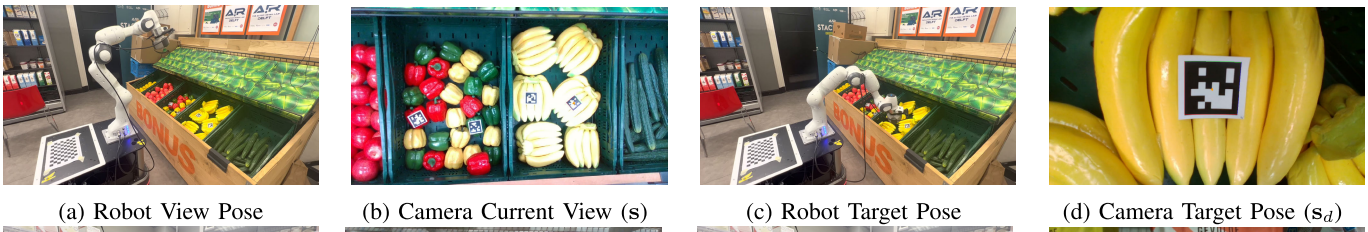
In this section, the effectiveness of the proposed visual servo control scheme is demonstrated using real time experiments on a real 7 DOF Franka Emika and a 6 DOF UR10 robot manipulator in mock-up convenience and warehouse setups. Video of the experiments are submitted as supplementary files. Also available online.[1]

A. Experimental Setup

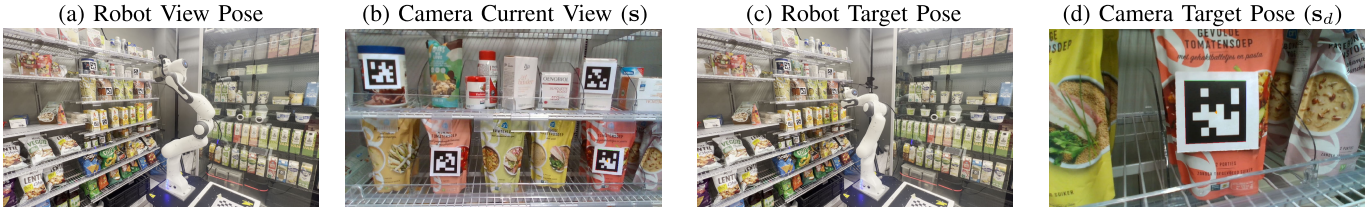
The experiments are conducted for constrained visual servoing for a robot manipulator in mock-up convenience store and warehousing applications.

1) Mock-Up Convenience Store: A mock-up convenience store is setup at the TU Delft Robotics Institute along with industry partners.[2] This store is equipped with a Franka Emika robot manipulator mounted with a Intel Realsense RGBD camera in eye-on-hand configuration. The store stocks a range of everyday items such as coffee, groceries, snack foods, confectionery, soft drinks, and replica of fruits. The fruits are stored in crates on a wooden box and other it on ems are stored in the racks. The robot is expected to fetch different items and help in delivery automation. We use this setup to test our proposed visual servo control scheme. In our experiments, the robot needs to use visual servo scheme to reach desired item for a pre-grasp configuration. Each item is attached with an unique apriltag marker ID of appropriate sizes depending on the size of the item e.g. a marker on *red pepper* is of dimension 2.5 × 2.5 *cm* while a marker on *cracker box* is of dimension 6 × 6 *cm* for correct pre-grasp configuration. The open-source ROS package *apriltag-ros*[3] is used for tracking the realtime apriltag marker for feature extraction. The input to the algorithm is the desired marker ID and its reference image (desired visual features $\mathbf{s}_d$) corresponding to an item to be grasped. The robot outputs the optimal joint velocity to reach to the target pre-grasp pose only by using real-time sensory feedback from the markers. Given a list of marker IDs and its reference image, the robot can sequentially servo to the pre-grasp configuration to each item. Fig. (4) shows the setup of the experimental validations in a mock-up convenience store setup. Fig. (4)(a-d) shows the visual servo experiment setup for reaching different fruits (attached with unique apriltag markers) placed in the crates. Fig. (4)(e-h) shows the visual servo experiment setup for reaching different items placed on the storage racks.

[1] https://youtu.be/R93W94G4cnY
[2] https://icai.ai/airlab-delft/
[3] http://wiki.ros.org/apriltag_ros

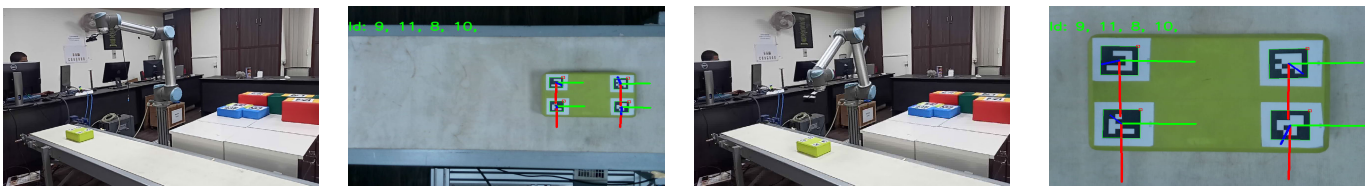


(a) Robot View Pose (b) Camera Current View ($\mathbf{s}$) (c) Robot Target Pose (d) Camera Target Pose ($\mathbf{s}_d$)



(e) Robot View Pose (f) Camera Current View ($\mathbf{s}$) (g) Robot Target Pose (h) Camera Target Pose ($\mathbf{s}_d$)

Fig. 4. Figure shows the experimental setup for validating the proposed visual servo control in a mock-up convenience store. (a,e) Robot scans the current visual features, (b,f) shows the current camera view, (c,g) robot reaches target pre-grasp pose and (d,h) shows the target camera view.



(a) Robot View Pose (b) Current Visual Features (c) Robot Tracks Target (d) Desired Visual Features

Fig. 5. Figure shows the experimental setup for validating the proposed visual servo control for tracking an item on moving conveyor belt. (a) The robot waits for the target object to come in its visual (b) the targets enter the visual field and current visual features are detected (c) robot tracks the moving item in a pre-grasp pose and (d) shows the target camera view while robot is in motion.

2) Warehousing Application: The task of item sorting in a dynamic warehouse environment, e.g., on a conveyor belt, requires time-critical movements of the robot to grasp the moving items and sort them as per their category. The grasp the moving item, a stable view (target locking) with respect to the gripper is critical. A task of item sorting on a moving conveyor setup to validate the efficacy of the proposed visual servo control for tracking time varying target visual features $\mathbf{s}_d(t)$). The setup consists of a conveyor belt transporting different items with unique markers, a 6 DOF UR 10 robot arm mounted with eye-on-hand Intel Realsense RGBD camera. Fig. (5) shows the setup of the experimental validations in for tracking an item in pre-grasp pose on a moving conveyor belt.

B. Modelling Kinematic and Visual Constraints: LfD

The kinesthetic demonstrations are given to the robot in gravity compensation mode by an expert human demonstrator for the task of visual servoing. There are a total of 6 demonstrations for different start and end configurations of the robot and the target object capturing the kinematic and visual space sufficiently. These demonstrations are used to train the DMP model for each degree of freedom corresponding to kinematic and visual trajectories. Samples from the demonstration and modeled trajectory using DMP model in [31] are shown in Fig. (6) for kinematic and visual degree of freedom. The figure also show how the individual DMPs generalize to a new traget object location while keeping the same target image. For a new/undemonstrated target object and its corresponding target image with visual features, desired visual ($\mathbf{s}_d(t)$) and kinematic ($\boldsymbol{\theta}(t), \dot{\boldsymbol{\theta}}(t)$) trajectories are generated from the mixture of DMP models using the policy in eq. (6).

C. NN Setup

The critic NN is constructed using a radial basis function network (RBFN) with single hidden layer containing 10 nodes and the output is linear in terms of NN weights ($\widehat{\mathbf{W}}$). The j^{th} radial basis activation function in the hidden layer is given by $\sigma_c(j) = 1 - \exp^{-\frac{\|x-c_j\|^2}{2\rho_c^2(j)}}$, where x, c_j and $\rho_c(j)$ are the input vector, center and width of the j^{th} activation function respectively. The NN weights ($\widehat{\mathbf{W}}$) is updated using the update rule (26) derived from the Lyapunov stability. The input to this network are the terms from the expansion of the polynomial $\sum_{\beta=0}^{2}\left(\sum_{i=1}^{2n}\mathbf{e}_i\right)^{\beta}$. In our application 4 feature points from corners of the apriltag marker sets the number of $n = 4$. The input layer is thus given as $[1, \mathbf{e}_1, \mathbf{e}_2, \cdots \mathbf{e}_8, \mathbf{e}_1^2, \mathbf{e}_2^2, \cdots \mathbf{e}_8^2, \mathbf{e}_1\mathbf{e}_2, \mathbf{e}_1\mathbf{e}_3, \cdots \mathbf{e}_7\mathbf{e}_8]^\intercal \in \mathbb{R}^{81}$. The centers $\mathbf{c}_j$ of the activation functions are chosen as $c\mathbf{1} \in \mathbb{R}^{81}$ where c is chosen uniformly from a span of $[-0.08, 0.08]$ and the width of the radial basis functions $\rho_c(j)$ were chosen uniformly from a span of $[0.05, 1.05]$ during numerous real-time responses to achieve optimal performance. The values of the critic NN weight vector $\widehat{\mathbf{W}} \in \mathbb{R}^{10}$ are intialized with each element to small random vector between -0.1 and 0.1. Critic learning rate α_c and adjustable rate of the additional stabilizing term α_e are tuning hyperparameters of the NN controller. The values of these parameters are chosen from

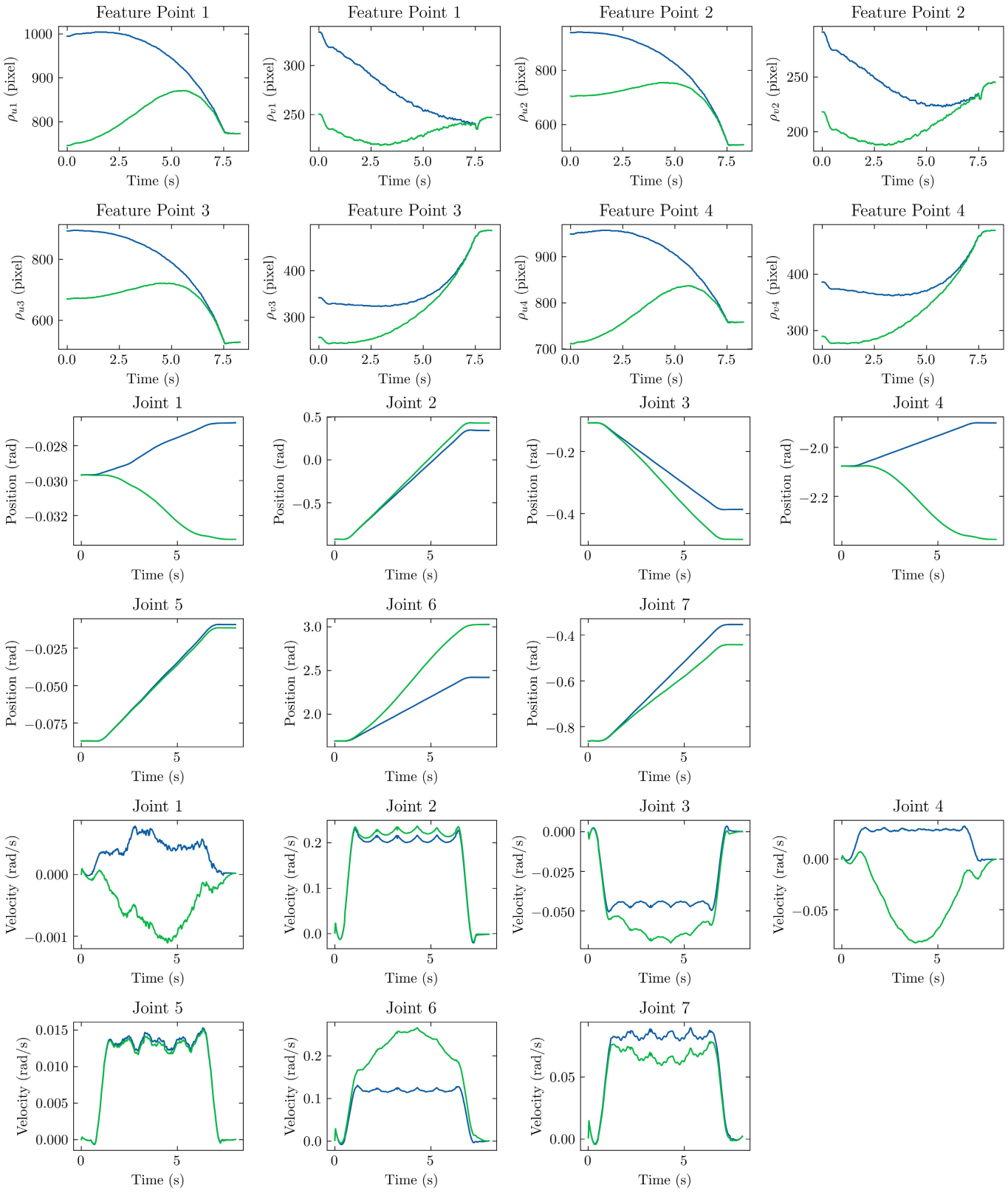


Fig. 6. Figure shows the modelling of the visual servo task to encode kinematic and visual constraints via LfD. The DMP modelling of demonstrated expert trajectory corresponding to each visual (top 2 rows) and kinematic (below 4 rows) degree of freedom is shown. — shows the human demonstrated kinematic and visual trajectory for the visual servo task which is used to train the corresponding DMP model and — represents the generated trajectories using learnt DMP models for varying goal parameters (the target object pose is changed but the reference feature image of the target object is the same). It is to be noted that the kinematic and visual trajectories retain their demonstrated shape profiles while generalizing to new goals.

a sensitivity analysis of the learning rate for the controller, also called a grid search. This can help to both highlight an order of magnitude where good learning rates may reside for satisfactory task performance.

D. Experiments in Mock-Up Convenience Store: Constant Visual Features $\mathbf{s_d}$

Real-time experiments are carried out to test the performance of the proposed optimal visual servo control of a robot manipulator. The experiments is setup as shown in Fig. 4. For a given marker ID and its reference image (visual feature $\mathbf{s_d}$), the objective is to actuate the robot manipulator using optimal joint velocities to the individual robot joints such that the current image features ($\mathbf{s}$) reaches the desired image features (s_d) in the vision space. Open-source ROS package is used to detect the four image feature points. The experimental results generated from a typical run using the proposed control is shown in the Fig (7)(a-c). Fig 7(a) shows the feature trajectory generated using the proposed scheme for the setup shown in Fig. (1))(a-d). Fig (7(b)) shows the convergence of feature error over time and the corresponding control velocity input to the robot is shown in Fig (7(c)). The controller gains $diag(\mathbf{Q})$ and $diag(\mathbf{R})$ in Eq. 19 are tuned to $diag(\mathbf{Q}) = [10^5\ 10^5\ 10^5\ 10^5\ 10^5\ 10^5\ 10^5]$ and $diag(\mathbf{R}) = [2.5\ 2.5\ 2.5\ 2.5\ 5\ 5\ 5]$ for a time domain specification of convergence time $\sim 8s$. The difference in the order of magnitude of weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ accomodates unit normalization of individual signals (error pixels and joint velocity control input). The whole essence of tuning the controller gain parameters is to achieve a motion task specifications at hand. The important time domain specifications of a motion task is its error convergence time while ensuring actuator limits. The critic learning rate $\alpha_c = 2$ and the stabilizing gain $\alpha_e = 0.1$.

Baselines: The efficacy of the proposed visual servo control in terms of its cost optimization are demonstrated via a comparative study against contemporary techniques available in the literature. The proposed controller is compared against three other baselines listed below

- SoA1 [23]: In this approach, the visual trajectories are modeled using statistical LfD approach via multiple human demonstrations to find a safe camera trajectories. This motion is back projected to the joint space using weighted Jacobian based instantaneous optimization for robot control.
- SoA2 [25]: This approach uses uses a safe transformation of visual feature dynamics into state constraints to find safe camera trajectories. The camera trajectories are then converted to robot joint speed using robot Jacobian based local optimization.
- SoA3 [24]: This approach uses a NN based convex optimization to minimize a cost function of feature error and joint velocity subjected to constraints using Karush–Kuhn–Tucker based convex solution.
- Proposed: Our approach uses dynamical LfD approach to model visual and kinematic constraints and employ a NN optimal control by solving HJB equations.

These state-of-the-art visual servo controllers are employed to run on the same setup shown in Fig. 4(a-d). The feature trajectory, error convergence and corresponding control velocity input to the robot for SoA1, SoA2 and SoA3 are shown in Fig. 7(d,e,f), Fig. 7(g,h,i) and Fig. 7(j,k,l) respectively. The controller parameters in all of them were tuned to attain similar motion task specifications (convergence, accuracy) for comparison.

From the qualitative analysis of the results in Fig. (7), it is observed that the proposed controller results in smoother feature and control trajectory profiles than its counterparts for the same motion specifications. The proposed approach and SoA1 both employs LfD modelling for constraints and optimization of a cost function to obtain velocity control input to the robot. SoA1 uses statistical method of LfD to generate desired kinematic and visual trajectories from a single learnt model, whereas in our proposed approach we use a mixture of model policy (6) which generalizes well for a new goal position. The optimization uses instantaneous local optimization and the solution needs matrix inversion which is computationally expensive. The SoA2 and SoA3 uses rigorous mathematical modeling of constraint dynamics by projecting the motion into safe region. SoA2 uses Jacobian based local inverse solution which might result in infeasible motion due to robot singularity and is computationally expensive. SoA3 uses local convex optimization solution which forces the input to a manipulator to comply with the constraints which results in non-smooth trajectory profiles in comparision to SoA1 and the proposed method. However, this has the advantage of computational efficiency since it does not require matrix inversions in comparison to SoA1.

E. Experiments in Warehousing Applications: Time Varying Visual Features $\mathbf{s_d(t)}$

The application of the proposed optimal visual servo control of a robot manipulator for time varying visual feature trajectory is very easily extended due to formulation in this paper. The performance of the proposed scheme is validated on a real experimental setup as shown in Fig. 5. The objective is to actuate the robot manipulator using optimal joint velocities to the individual robot joints such that the current image features ($\mathbf{s}$) tracks a desired image features ($s_d(t)$) in the vision space from any given start pose. The realtime experimental results generated from a typical run using the proposed control is shown in Fig (8. Fig (8(a)) shows the feature trajectory generated using the proposed scheme for the setup shown in Task (2) (refer Fig. (1)). Fig (8(b)) shows the convergence of feature error over time and the corresponding control velocity input to the robot is shown in Fig (8(c)). The controller gains were tuned as mentioned in the previous case. The extension of servo schemes in SoA1, SoA2 and SoA3 for tracking a time varying visual features are not studied in their evaluations and are not straightforward from their control solutions.

F. Quantitative Test Comparison

The performance of the IBVS using the proposed optimal control is quantified against the state-of-the-art methods SoA1

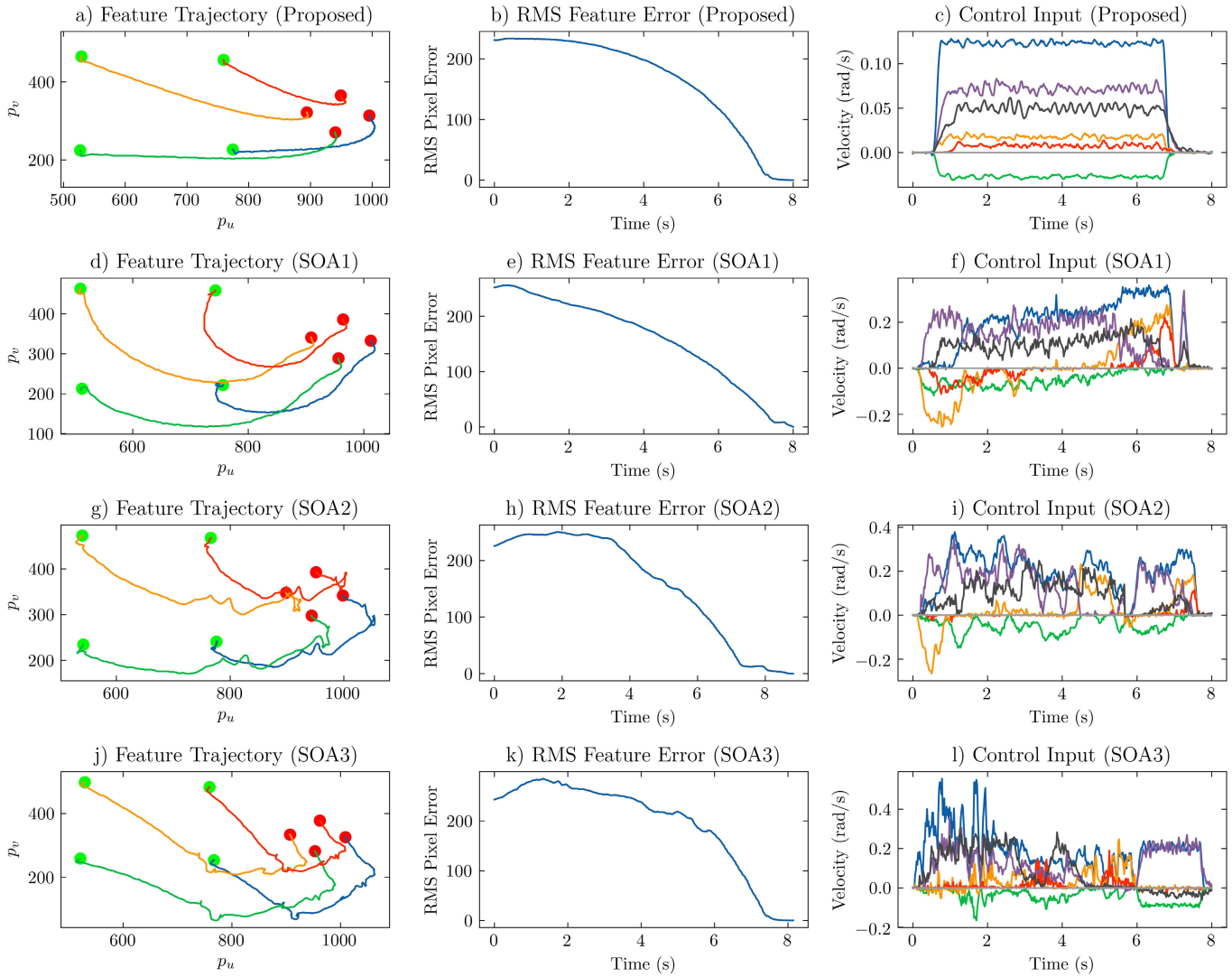


Fig. 7. Figure shows the profiles of feature trajectory, RMS feature error and optimal control velocity input to the Franka Panda robot manipulator for the task of visual servo as per experimental setup in Fig. 1(a-d). • and • demotes the start and end of the trajectories respectively. The samples from experiments are shown in the video link [35].

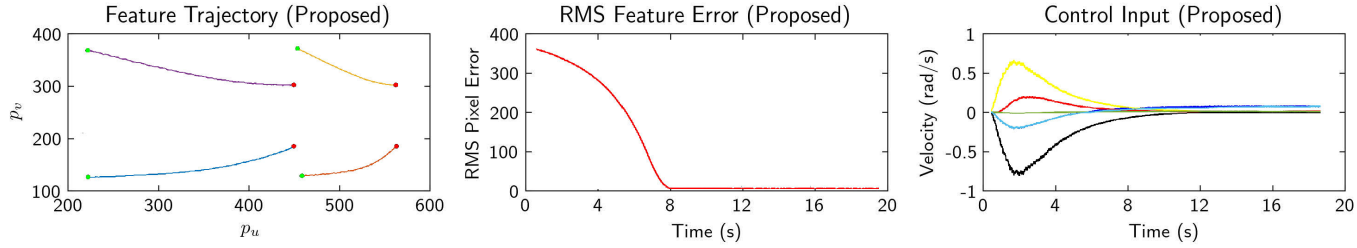


Fig. 8. Figure shows the profiles of feature trajectory, RMS feature error and optimal control velocity input to the Franka Panda robot manipulator for the task of visual servo as per experimental setup in Fig. 5. • and • demotes the start and end of the trajectories respectively. The samples from experiments are shown in the video link [35].

SoA2 and SoA3. A set of test pairs of start and target visual features in different setups of the mock-up convenience store are generated. The the test items like fruits are placed in more horizontal and inclined plane while the racks supports items in vertical plane. Moreover, different sized objects are attached with different sized markers ($2.5 \times 2.5cm - 7 \times 7cm$). These generates test sets with variance in pose, size and distance. The four control schemes for image based visual servoing are then tested and compared. The comparison is done on a performance metric Average Normalised Trajectory Cost (ANTC), Integral of Time Absolute Chattering Quantifier (ITACQ) for control input to real robot and Computational Complexity (CC).

1) ANTC: The ANTC is defined as average normalised trajectory cost over all trajectories. The normalized cost for i^{th} trajectory is given by $C_i = C/N$, where N is the

TABLE I
COMPARATIVE STUDY OF THE PROPOSED VISUAL SERVO CONTROL WITH THE STATE-OF-THE-ART CONTROLLERS

Scheme	Class	Matrix Inversion	Constraint Modelling	ANTC	ITACQ (10^3)	Algorithmic Complexity	Runtime Complexity ($10^{-4}s$)	Validations $s_d(t)$
SoA1 [23]	Numerical, Local, Jacobian Pseudoinverse	Yes	LfD (statistical)	3671±43.5	33±6.8	1248	1.49	No
SoA2 [25]	Numerical, Local, Jacobian Pseudoinverse	No	Mathematical	6707±78.35	37±8.5	997	1.19	No
SoA3 [24]	Convex, Local, KKT conditions	No	Mathematical	8027±58.25	42±9	576	0.68	No
Proposed	Optimal, Global, Approximate HJB	No	LfD (dynamical)	1573±17.7	19±3.5	3129	3.5	Yes

number of total time instances until the error converges to a threshold in case of end pose regulation and total time of execution in case of trajectory tracking. Hence the average normalized trajectory cost over all trajectory is given by $\frac{\sum_i^{N_t} V_i}{N_t}$, where N_t is total number of trajectories. Now we define the ANTC metric corresponding to trajectory cost $C = \mathbf{e}(t)^\intercal \mathbf{e}(t) + \mathbf{u}(t)^\intercal \mathbf{u}(t) + \dot{\mathbf{u}}(t)^\intercal \dot{\mathbf{u}}(t)$. This metric is a measure of accuracy and optimal effort with smooth trajectories. To ensure fairness in comparison the following methodology was adopted. First all the algorithms are run for the same problem setup using best selection of controller gain parameters until satisfactory task error performance is achieved (e.g. the RMS regulation error or tracking error converges below 10^{-3}). Then the controller gain parameter of each of the controller is tweaked to match the same time domain convergence while ensuring that the maximum control limits of the individual joints is not violated in all the cases. So, now for the same time domain task performance, both the control methods are compared on a same metric whose physical significance is that how much control effort is invested for the same accuracy of the task and also how smooth are the trajectory profiles. The smaller value of these metrics would imply smooth and optimal control for the same robot motion tasks. It would also ensure optimal actuator torque input in the secondary/low level servo control loop for the same task performance in the real world applications. For experimentation, 25 pairs of start and target images were taken for the mock-up convenience store setup. For each pair, the tests were run 10 times in order to find the mean (μ) and standard deviation (σ) of ANTC. The results of the study is reported in Table I.

2) ITACQ: It is given by $I_i = \nu \sum_{i=1}^{N_u} |\tilde{u}_i \times t|$, where $\tilde{u}$ is the amplitude of deviation from the mean of the input control, ν is chattering frequency of input control, N_u is the number of sampling instants. This quantifier is justified because we consider the magnitude and frequency along with time instant which are primarily the most basis for chattering phenomenon. Average ITACQ (AITACQ) of control input for all joints can be calculated by $\frac{\sum_i^{N_j} I_i}{N_j}$, where N_j is total number of joints. The lesser the ITACQ, the lesser is the stresses on the joint actuators and more is the longevity of the robotic hardware.

3) Computational Complexity (CC): The computational complexity of the control solutions of all the four schemes are calculated for comparison. It consists of algorithmic and runtime complexities. Algorithmic complexity is a measure of number of flops of operation needed to compute control solution per control loop, and the runtime complexity measures the actual time elasped on compute platform of the robot for the same. *The multiplication of two matrices, one with the dimensions of $l_1 \times l_2$ and the other with the dimensions of $l_2 \times l_3$, requires $l_1 l_3 (2l_2 - 1)$ flops of operations. The summation of two matrices with the dimensions of $l_1 \times l_2$ requires $l_1 l_2$ flops of operations. The inversion of a square matrix with the dimensions of $l_1 \times l_1$ requires l^3 flops of operations* [36]. For our current study of visual servo control using 7 DOF robot manipulator i.e. $N = 7$ and image features $n = 4$, the total number of flops for operation in each control solutions is tabulated in Table I. The corresponding runtime on a computer with 11th Gen Intel® Core™ i9-11900H @ 2.50GHz × 16 processor is also shown.

The comparison has been arranged as per their class of optimization and methodology of solution along with performance metrics. It also shows qualitative and quantitative performances. In essence the performance of proposed scheme based on LfD modelling and optimal control theory involving solution of HJB equation performs better than other servo schemes. The study reflects that the expert demonstration not only allows easy modelling of the constraints but also results in smoother feature trajectories and the non-inversion based methods are computationally efficient. However, owing to the usage of NN, the computational cost of the proposed method is more than those without using NN. Because in the NN based methods, we often need to have many neurons to ensure the approximation accuracy and whoose weights are computed in every iteration. The proposed optimal visual servo control accommodates easy and better constraint modelling which is robust to perturbations. And the solution is inversion free, optimal and comparably efficient for real-time implementations.

G. Effect of Environmental Conditions

The real-world environment in which a visually controlled robot is employed can have varying levels of illumination. Experiments are conducted to find the influence of such variations on the performance of the proposed visual servo control in a mockup convenience store at TU Delft robotics institute. In this study, three different illumination conditions based on luminance range measured by a lightmeter are used. The illumination conditions consists of low (20 - 75 lux), good (150-250 lux), and bright (350-500 lux) ambient lighting. The detection of the target objects of varying sizes is realized

TABLE II
ROBOT CONTROL PERFORMANCE UNDER VARYING ENVIRONMENTAL CONDITIONS

Illumination (lux)	Detection Rate (%)	ANTC	ITACQ	Min Marker Size (cm)
Low	89	1.27C	1.16Q	2.5
Good	99	C	Q	1.5
Bright	93.5	1.19C	1.38Q	1.75

through markers of appropriate sizes attached to them i.e., smaller marker on smaller object and bigger marker on bigger object. The varying illumination in the environment affects the detection rate of these markers. In addition, the detection of the markers are also dependent on its distance from the camera sensor. For a constant illumination, the detection rate of the marker decreases for an increasing camera distance. Therefore, an optimal/minimum size of a marker is determined based on consistent successful detection from a maximum distance in the manipulation workspace (i.e., 150 cm in our case). The minimum size of the marker also limits the minimum size of the objects that can be used in an environment. Once the size of the marker is fixed, the effects of the varying illumination on the robot control performance is studied. It is observed that the average detection rate drops in low and bright illumination resulting in some blind frames. In such frame stamps, a linear prediction for the value of the marker pixels are obtained based previous two previous measurements to ensure real time control execution loop. The robot is found to move with increased velocity during small intervals with slight shaky/oscillatory behaviour. This can be attributed to the sudden/temporary vision loss due to varying illuminations. The robot performance is quantified through the two parameters ANTC and ITACQ defined earlier. The experimental results from 15 runs in each illumination condition is analyzed and the outcome is tabulated in Table II. It is observed that the mean ANTC increases (27%) due to lesser detection rates in the low illumination which gives rise to non-optimal robot behaviour. In the bright illumination case, the detection is slightly better than that in the low illumination, however due to the reflection/shining of the markers the measurement is noisy resulting in increase of mean ITACQ (38%) corresponding to oscillations/non-smooth behaviour.

VI. CONCLUSION

In conclusion, this paper presents a novel method for visual servoing based on optimal control formulation, replacing the cumbersome mathematical modeling of visual and kinematic constraints with a dynamical system-based LfD approach. This approach offers several advantages, including robustness against perturbations, a single demonstration model, and the ability to improve the framework on the run. The visual servoing problem was then formulated as a closed-loop optimal control problem and solved using HJB solution, resulting in cost-optimal trajectory profiles without matrix inversion. The experimental evaluations show the efficacy of the proposed scheme compared to existing state-of-the-art schemes. However, the proposed visual servo control scheme assumes that visual features are extracted robustly, which limits its application scenario. Future work may include incorporating obstacle avoidance based constrained optimal visual servo control and robustness against measurement errors. Overall, this work focuses on developing an efficient way of modeling constraints through LfD approach and generating smoother robot trajectories using optimal control formulation.

APPENDIX A
PROOF OF LEMMA 1

Proof: The optimal control law must ensure the closed-loop system is asymptotically stable when the dynamics are known. Therefore, the closed loop system (9) can be shown to be bounded by a function of the system state after application of optimal control law (13). Thus, it can be assumed that

$$\left\|\dot{\mathbf{e}}^*(\boldsymbol{\theta})\right\| \leq \eta\|\mathbf{e}(\boldsymbol{\theta})\| \text{ with } \eta > 0, \tag{29}$$

which results in

$$\left\|\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta})\right\| \leq \eta\|\mathbf{e}(\boldsymbol{\theta})\|^2 \tag{30}$$

Therefore combining (27) with the fact that

$$\lambda_{min}(\mathbf{M})\|\mathbf{e}(\boldsymbol{\theta})\|^2 \leq \mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{M}\mathbf{e}(\boldsymbol{\theta}) \leq \lambda_{max}(\mathbf{M})\|\mathbf{e}(\boldsymbol{\theta})\|^2,$$

implies Lemma 1. □

APPENDIX B
PROOF OF THEOREM 1

Proof: Consider the Lyapunov candidate,

$$L(\mathbf{e}(\boldsymbol{\theta}), \widetilde{\mathbf{W}}) = \frac{\alpha_s}{2\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta}) + \frac{1}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\widetilde{\mathbf{W}} \tag{31}$$

where $\mathbf{e}(\boldsymbol{\theta})$ is the visual features error vector, $\widetilde{\mathbf{W}}$ is the NN critic weight estimation error vector. The time derivative of the Lyapunov candidate function is given by

$$\dot{L}(\mathbf{e}(\boldsymbol{\theta}), \widetilde{\mathbf{W}}) = \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) + \frac{1}{\alpha_c}\widetilde{\mathbf{W}}^\intercal\dot{\widetilde{\mathbf{W}}} \tag{32}$$

Observing that, $\dot{\widetilde{\mathbf{W}}} = -\dot{\hat{\mathbf{W}}}$ and using (25),

$$\begin{aligned}\dot{L} = {} & \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) + \widetilde{\mathbf{W}}^\intercal\Big(-\frac{1}{4}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} + \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\mathbf{W} \\ & - e_\varepsilon\Big)\Big(-\frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} + \frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}}\Big) \\ & + \frac{\alpha_e}{\alpha_c}\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta})\end{aligned} \tag{33}$$

Noticing (20) and expression of $\mathbf{A}(\boldsymbol{\theta})$ the following equality can be deduced

$$-\frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\mathbf{W} + \frac{1}{2}\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} = \boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \tag{34}$$

And then,

$$\begin{aligned}\dot{L} = {} & \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) - \Big(\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) - \frac{1}{4}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} + e_\varepsilon\Big) \\ & \times \widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) - \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta})\end{aligned} \tag{35}$$

By combining the control law given in (17) and (19), we obtain

$$\mathbf{u}_e^*(\boldsymbol{\theta}) - \widehat{\mathbf{u}}_e^*(\boldsymbol{\theta}) = -\frac{1}{2}\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})(\boldsymbol{\phi}_{\boldsymbol{\theta}}^\intercal(\boldsymbol{\theta})\widetilde{\mathbf{W}} + \varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})) \quad (36)$$

Observing $\dot{\mathbf{e}}^*(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_e^*(\boldsymbol{\theta})$ and $\dot{\mathbf{e}}(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\widehat{\mathbf{u}}_e^*(\boldsymbol{\theta})$ and the relation in (36), we obtain the relationship between $\dot{\mathbf{e}}^*(\boldsymbol{\theta})$ and $\dot{\mathbf{s}}$ as:

$$\begin{aligned}
\dot{\mathbf{e}}^*(\boldsymbol{\theta}) - \dot{\mathbf{e}}(\boldsymbol{\theta}) &= \mathbf{J}_s(\boldsymbol{\theta})(\mathbf{u}_e^*(\boldsymbol{\theta})) - \widehat{\mathbf{u}}_e^*(\boldsymbol{\theta}) \\
&= -\frac{1}{2}\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})(\boldsymbol{\phi}_{\boldsymbol{\theta}}^\intercal(\boldsymbol{\theta}))\widetilde{\mathbf{W}} + \varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})) \\
\dot{\mathbf{e}}(\boldsymbol{\theta}) &= \dot{\mathbf{e}}^*(\boldsymbol{\theta}) + \frac{1}{2}\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^\intercal(\boldsymbol{\theta}))\widetilde{\mathbf{W}} \\
&\quad + \frac{1}{2}\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \quad (37)
\end{aligned}$$

Using (37) in (35) results in,

$$\begin{aligned}
\dot{L} = &-\Big(\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) + \frac{1}{4}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} \\
&+ \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) + e_\varepsilon\Big) \\
&\times \Big(\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) + \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} + \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta}) \\
&\times \mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})\Big) - \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta}) \\
&\times \mathbf{e}(\boldsymbol{\theta}) + \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \\
\dot{L} = &-(\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}))^2 \\
&- \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta})\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} \\
&- \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta})\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \\
&- \frac{1}{4}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) - \frac{1}{8}(\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}})^2 \\
&- \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \\
&- \frac{1}{2}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) \\
&- \frac{1}{4}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} \\
&- \frac{1}{4}(\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}))^2 \\
&- e_\varepsilon\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) - \frac{1}{2}e_\varepsilon\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}} \\
&- \frac{1}{2}e_\varepsilon\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \\
&- \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta}) + \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \quad (38)
\end{aligned}$$

Next, completing the squares ($ab \leq 1/2(a^2 + b^2)$ or $-ab \leq 1/2(a^2 + b^2)$) with respect to $\widetilde{\mathbf{W}}^\intercal\mathbf{A}(\boldsymbol{\theta})\widetilde{\mathbf{W}}$ in terms with cubic occurrence of $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta})$, $\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ with single occurrence of $\widetilde{\mathbf{W}}$ and taking upper bounds as per Assumption 1 results in,

$$\begin{aligned}
\dot{L} < &-\alpha_{\boldsymbol{\phi}}^2\alpha_{\dot{\mathbf{e}}^*}^2\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{16}\alpha_{\mathbf{A}}^2\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_{\boldsymbol{\phi}}^2\alpha_{\dot{\mathbf{e}}^*}^2 \\
&\times \left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{2}\alpha_{\mathbf{A}}\alpha_{\dot{\mathbf{e}}^*}\alpha_\varepsilon\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{32}\alpha_{\mathbf{A}}^2\left\|\widetilde{\mathbf{W}}\right\|^4 + \frac{1}{2}\alpha_{\boldsymbol{\phi}}^2\alpha_{\dot{\mathbf{e}}^*}^2 \\
&\times \left\|\widetilde{\mathbf{W}}\right\|^2 - \frac{1}{8}\alpha_{\mathbf{A}}^2\left\|\widetilde{\mathbf{W}}\right\|^4 + \frac{1}{64}\alpha_{\mathbf{A}}^2\left\|\widetilde{\mathbf{W}}\right\|^4 + 4\alpha_{\boldsymbol{\phi}}^2\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\| \\
&\times \alpha_\varepsilon\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{2}\alpha_{\mathbf{A}}\alpha_{\dot{\mathbf{e}}^*}\alpha_\varepsilon\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{128}\alpha_{\mathbf{A}}^2\left\|\widetilde{\mathbf{W}}\right\|^4 + 2\alpha_{\boldsymbol{\phi}}^2\alpha_{\mathbf{J}_s}^4 \\
&\times \left\|\mathbf{R}^{-1}\right\|\alpha_\varepsilon\left\|\widetilde{\mathbf{W}}\right\|^2 - \frac{1}{4}\alpha_{\boldsymbol{\phi}}^2\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\|\alpha_\varepsilon\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{2}\alpha_{\boldsymbol{\phi}}^2\alpha_{\dot{\mathbf{e}}^*}^2 \\
&\times \left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{2}\alpha_\varepsilon^2 + \frac{1}{2}\alpha_\varepsilon\alpha_{\mathbf{A}}\left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{4}\alpha_{\boldsymbol{\phi}}^2\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\|\alpha_\varepsilon \\
&\times \left\|\widetilde{\mathbf{W}}\right\|^2 + \frac{1}{4}\alpha_\varepsilon^2 - \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta}) \\
&+ \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \\
\dot{L} < &-\alpha_1\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_2\left\|\widetilde{\mathbf{W}}\right\|^2 + \alpha_3^2 \\
&- \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta}) + \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta}) \quad (39)
\end{aligned}$$

where $\alpha_1, \alpha_2, \alpha_3$ are positive constants given by,

$$\begin{aligned}
\alpha_1 &= \frac{5}{128}\alpha_{\mathbf{A}}^2 \\
\alpha_2 &= \frac{1}{2}\alpha_{\boldsymbol{\phi}}^2\alpha_{\dot{\mathbf{e}}^*}^2 + \alpha_{\mathbf{A}}\alpha_{\dot{\mathbf{e}}^*}\alpha_\varepsilon + 6\alpha_{\boldsymbol{\phi}}^2\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\|\alpha_\varepsilon + \frac{1}{2}\alpha_\varepsilon\alpha_{\mathbf{A}} \\
\alpha_3 &= \frac{\sqrt{3}}{2}\alpha_\varepsilon
\end{aligned}$$

Now using the expression (37) in (39) results in

$$\begin{aligned}
\dot{L} < &-\alpha_1\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_2\left\|\widetilde{\mathbf{W}}\right\|^2 + \alpha_3^2 \\
&- \frac{\alpha_s}{2\alpha_c}\widetilde{\mathbf{W}}^\intercal\boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\mathbf{e}(\boldsymbol{\theta}) + \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}(\boldsymbol{\theta})^* \\
&+ \frac{\alpha_s}{2\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\boldsymbol{\phi}_{\boldsymbol{\theta}}^\intercal(\boldsymbol{\theta}))\widetilde{\mathbf{W}} \\
&+ \frac{\alpha_s}{2\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \\
< &-\alpha_1\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_2\left\|\widetilde{\mathbf{W}}\right\|^2 + \alpha_3^2 \\
&+ \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) \\
&+ \frac{\alpha_s}{2\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \quad (40)
\end{aligned}$$

Now substituting $\dot{\mathbf{e}}^*(\boldsymbol{\theta}) = \mathbf{J}_s(\boldsymbol{\theta})\mathbf{u}_e^*(\boldsymbol{\theta})$ in (40)

$$\begin{aligned}
\dot{L} < &-\alpha_1\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_2\left\|\widetilde{\mathbf{W}}\right\|^2 + \alpha_3^2 \\
&+ \frac{\alpha_s}{\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\dot{\mathbf{e}}^*(\boldsymbol{\theta}) \\
&+ \frac{\alpha_s}{2\alpha_c}\mathbf{e}^\intercal(\boldsymbol{\theta})\mathbf{J}_s(\boldsymbol{\theta})\mathbf{R}^{-1}\mathbf{J}_s^\intercal(\boldsymbol{\theta})\varepsilon_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \quad (41)
\end{aligned}$$

Observing Assumption 1 and Lemma 1 we get

$$\begin{aligned}
\dot{L} < &-\alpha_1\left\|\widetilde{\mathbf{W}}\right\|^4 + \alpha_2\left\|\widetilde{\mathbf{W}}\right\|^2 + \alpha_3^2 \\
&- \frac{\alpha_s}{\alpha_c}\alpha_{min}(\mathbf{M})\|\mathbf{e}(\boldsymbol{\theta})\|^2 \\
&+ \frac{\alpha_s}{2\alpha_c}\|\mathbf{e}(\boldsymbol{\theta})\|\alpha_{\mathbf{J}_s}^2\left\|\mathbf{R}^{-1}\right\|\alpha_{\boldsymbol{\varepsilon}} \quad (42)
\end{aligned}$$

And on further arrangements in (42) results in

$$\begin{aligned}
\dot{L} < &-\alpha_1\left(\left\|\widetilde{\mathbf{W}}\right\|^2 - \frac{\alpha_2}{2\alpha_1}\right)^2 + \frac{\alpha_2^2}{4\alpha_1} + \alpha_3^2 \\
&- \frac{\alpha_s}{\alpha_c}\alpha_{min}(\mathbf{M})\left(\|\mathbf{e}(\boldsymbol{\theta})\| - \frac{\alpha_{\mathbf{J}_s}^2\left\|\mathbf{R}^{-1}\right\|\alpha_{\boldsymbol{\varepsilon}}}{4\alpha_{min}(\mathbf{M})}\right)^2
\end{aligned}$$

$$+\frac{\alpha_s\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\|^2\alpha_{\boldsymbol{\varepsilon}}^2}{16\alpha_c\alpha_{min}(\mathbf{M})}$$
$$<-\alpha_1\left(\left\|\widetilde{\mathbf{W}}\right\|^2-\frac{\alpha_2}{2\alpha_1}\right)^2+\alpha_4$$
$$-\frac{\alpha_s}{\alpha_c}\alpha_{min}(\mathbf{M})\left(\|\mathbf{e}(\boldsymbol{\theta})\|-\alpha_5\right)^2 \tag{43}$$

where

$$\alpha_4=\frac{\alpha_2^2}{4\alpha_1}+\alpha_3^2+\frac{\alpha_s\alpha_{\mathbf{J}_s}^4\left\|\mathbf{R}^{-1}\right\|^2\alpha_{\boldsymbol{\varepsilon}}^2}{16\alpha_c\alpha_{min}(\mathbf{M})}$$
$$\alpha_5=\frac{\alpha_{\mathbf{J}_s}^2\left\|\mathbf{R}^{-1}\right\|\alpha_{\boldsymbol{\varepsilon}}}{4\alpha_{min}(\mathbf{M})}$$

Therefore from (43), $\dot{L}(\mathbf{e}(\boldsymbol{\theta}),\widetilde{\mathbf{W}})<0$ can be shown, provided any one of the following inequalities

$$\|\mathbf{e}(\boldsymbol{\theta})\|>\alpha_5+\sqrt{\frac{\alpha_c\alpha_4}{\alpha_s\alpha_{min}(\mathbf{M})}}\triangleq\alpha_{J_s}, \tag{44}$$

$$\left\|\widetilde{\mathbf{W}}\right\|>\sqrt{\frac{\alpha_2}{2\alpha_1}+\sqrt{\frac{\alpha_4}{\alpha_1}}}\triangleq\alpha_{\widetilde{\mathbf{W}}} \tag{45}$$

holds, and therefore according to the standard Lyapunov extension theorem, it can be concluded that he error state vector of the closed-loop system consisting of visual feature error and the critic weight error are all UUB. According to (45), note that $\left\|\widetilde{\mathbf{W}}\right\|<\alpha_{\widetilde{\mathbf{W}}}$. Furthermore, considering (36) and *Assumption* 1, it is observed that the approximate controller $\widehat{\mathbf{u}}_e^*(\boldsymbol{\theta})$ given by (19) converges to a neighbourhood of the optimal feedback controller $\mathbf{u}_e^*(\boldsymbol{\theta})$ with a finite bound as, $\left\|\mathbf{u}_e^*(\boldsymbol{\theta})-\widehat{\mathbf{u}}_e^*(\boldsymbol{\theta})\right\|\leq\frac{1}{2}\left\|\mathbf{R}^{-1}\right\|\alpha_{\mathbf{J}_s}\left(\alpha_{\boldsymbol{\phi}}\alpha_{\widetilde{\mathbf{W}}}+\alpha_{\boldsymbol{\varepsilon}}\right)\triangleq\alpha_{\mathbf{u}}$ □

Estimation of Theoretical Bounds: The estimation of the theoretical maximum bound in Equations (44) and (45) along with others can be found using bound information on individual terms. In the current case study of NN visual servo control of 7 DOF robot manipulator, the calculation of above mentioned bounds will be summarized next. Since the NN activation function is radial basis function, the gradient of activation function is bounded by maximum value of $\alpha_{\boldsymbol{\phi}}=\frac{\exp^{-0.5}}{\rho_c}$. The minimum ρ_c used in our experiments is 0.5, therefore $\alpha_{\boldsymbol{\phi}}=1.2$. The minimum and maximum eigen value of the image Jacobian matrix $\mathbf{J}_s$ of is $\alpha_{min}(\mathbf{J}_s)=0.1$ and $\alpha_{max}(\mathbf{J}_s)=1$. The NN approximation error is assumed to be slowly varying with respect to the state and hence the bound on its gradient is chosen as $\varepsilon'_M=0.01$. The minimum and maximum eigenvalue of design matrix $\mathbf{R}$ is 2.5 and 5 respectively and hence minimum and maximum eigenvalue of $\mathbf{R^{-1}}$ is 0.2 and 0.4 respectively. The minimum and maximum eigenvalue of design matrix $\mathbf{Q}$ is 10^5 each. The controller hyperparameters values are $\alpha_c=0.1,\alpha_e=5$. Using these values and the relations derived earlier the following bounds can be computed,

$$\alpha_{\mathbf{A}}=2.5,\alpha_{\varepsilon}=10^{-3},\quad \alpha_1=0.2,\ \alpha_2=0.0013,\ \alpha_3=10^{-3},$$
$$\alpha_4=6\times10^{-5},\quad \alpha_5=0.1,\ \alpha_{\widetilde{\mathbf{W}}}=0.1,\ \alpha_{\mathbf{M}}=0.5,$$
$$\alpha_{\mathbf{e}}=0.03,\quad \alpha_{\mathbf{u}}=0.05.$$

The value $\alpha_{\mathbf{e}}=0.03$ shows that the closed loop error dynamics is always bounded and the maximum value is 0.03. The norm of critic weight error dynamics is bounded by $\alpha_{\widetilde{\mathbf{W}}}=0.1$ where the average norm of the converged NN critic weight is of the order 2. The approximate optimal controller converges to optimal feedback controller and is bounded by $\alpha_{\mathbf{u}}=0.25$ where the average norm of the approximate optimal control is of the order 3.

ACKNOWLEDGMENT

The authors would like to thank Prof. Jens Kober, 3mE, Cognitive Robotics, TU Delft for providing access to experimental setup in mock-up convenience store (AirLab) at RoboHouse, Delft, and also would like to thank the anonymous reviewers for their constructive feedback in improving the quality of the manuscript.

REFERENCES

[1] C. J. Taylor and J. P. Ostrowski, "Robust vision-based pose control," in *Proc. ICRA Millennium Conf. IEEE Int. Conf. Robot. Automat. Symposia*, Apr. 2000, pp. 2734–2740.
[2] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator control with image-based visual servo," in *Proc. IEEE Int. Conf. Robot. Autom.*, Jan. 1991, pp. 2267–2268.
[3] E. Malis, F. Chaumette, and S. Boudet, "2 1/2 D visual servoing," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 238–250, Apr. 1999.
[4] P. Y. Oh and K. Allen, "Visual servoing by partitioning degrees of freedom," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 1–17, Feb. 2001.
[5] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: A switching approach," *IEEE Trans. Robot.*, vol. 20, no. 5, pp. 908–914, Oct. 2004.
[6] F. Chaumette and S. Hutchinson, "Visual servo control. I. basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.
[7] F. Chaumette and S. Hutchinson, "Visual servo control. II. Advanced approaches [tutorial]," *IEEE Robot. Autom. Mag.*, vol. 14, no. 1, pp. 109–118, Mar. 2007.
[8] Y. Peng, D. Jivani, R. J. Radke, and J. Wen, "Comparing Position- and image-based visual servoing for robotic assembly of large structures," in *Proc. IEEE 16th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2020, pp. 1608–1613.
[9] T. Xu, Y. Guan, J. Liu, and X. Wu, "Image-based visual servoing of helical microswimmers for planar path following," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 325–333, Jan. 2020.
[10] R. Mahony, P. Corke, and F. Chaumette, "Choice of image features for depth-axis control in image based visual servo control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2002, pp. 390–395.
[11] E. Malis and F. Chaumette, "Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods," *IEEE Trans. Robot. Autom.*, vol. 18, no. 2, pp. 176–186, Apr. 2002.
[12] M. Iwatsuki and N. Okiyama, "A new formulation of visual servoing based on cylindrical coordinate system," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 266–273, Apr. 2005.
[13] F. Chaumette, "Image moments: A general and useful set of features for visual servoing," *IEEE Trans. Robot.*, vol. 20, no. 4, pp. 713–723, Aug. 2004.
[14] Z. Cao, F. Ma, J. S. Dai, and J. Zhang, "A Gabor based fast interest point detector for image-based robot visual servo control," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2014, pp. 883–888.
[15] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, no. 3. Belmont, MA, USA: Athena Scientific, 2005.
[16] Y. Mezouar and F. Chaumette, "Optimal camera trajectory with image-based control," *Int. J. Robot. Res.*, vol. 22, nos. 10–11, pp. 781–803, Oct. 2003.
[17] F. Schramm and G. Morel, "Ensuring visibility in calibration-free path planning for image-based visual servoing," *IEEE Trans. Robot.*, vol. 22, no. 4, pp. 848–854, Aug. 2006.
[18] G. Chesi and Y. S. Hung, "Global path-planning for constrained and optimal visual servoing," *IEEE Trans. Robot.*, vol. 23, no. 5, pp. 1050–1060, Oct. 2007.

[19] T. Nammoto, K. Kosuge, and K. Hashimoto, "Model-based compliant motion control scheme for assembly tasks using vision and force information," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2013, pp. 948–953.

[20] C. De Farias, M. Adjigble, B. Tamadazte, R. Stolkin, and N. Marturi, "Dual quaternion-based visual servoing for grasping moving objects," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 151–158.

[21] G. B. P. Barbosa, E. C. D. Silva, and A. C. Leite, "Robust image-based visual servoing for autonomous row crop following with wheeled mobile robots," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 1047–1053.

[22] S. Durola, P. Danes, D. F. Coutinho, and M. Courdesses, "Rational systems and matrix inequalities to the multicriteria analysis of visual servos," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 1504–1509.

[23] T. Shen, S. Radmard, A. Chan, E. A. Croft, and G. Chesi, "Optimized vision-based robot motion planning from multiple demonstrations," *Auto. Robots*, vol. 42, no. 6, pp. 1117–1132, Aug. 2018.

[24] Y. Zhang and S. Li, "A neural controller for image-based visual servoing of manipulators with physical constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5419–5429, Nov. 2018.

[25] I. Salehi, G. Rotithor, R. Saltus, and A. P. Dani, "Constrained image-based visual servoing using barrier functions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 14254–14260.

[26] S. Sajjadi, M. M. H. Fallah, M. Mehrandezh, and F. Janabi-Sharifi, "Stochastic image-based visual predictive control," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 551–556.

[27] G. Allibert, E. Courtial, and F. Chaumette, "Predictive control for constrained image-based visual servoing," *IEEE Trans. Robot.*, vol. 26, no. 5, pp. 933–939, Oct. 2010.

[28] A. Chan, S. Leonard, E. A. Croft, and J. J. Little, "Collision-free visual servoing of an eye-in-hand manipulator via constraint-aware planning and control," in *Proc. Amer. Control Conf.*, Jun. 2011, pp. 4642–4648.

[29] A. Chan, E. A. Croft, and J. J. Little, "Constrained manipulator visual servoing (CMVS): Rapid robot programming in cluttered workspaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 2825–2830.

[30] O. Faugeras and O. A. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*. Cambridge, MA, USA: MIT Press, 1993.

[31] R. Prakash, L. Behera, S. Mohan, and S. Jagannathan, "Dynamic trajectory generation and a robust controller to intercept a moving ball in a game setting," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 4, pp. 1418–1432, Jul. 2020.

[32] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*. Hoboken, NJ, USA: Wiley, 2012.

[33] F. Lewis, S. Jagannathan, and A. Yesildirak, *Neural Network Control of Robot Manipulators and Non-Linear Systems*. Boca Raton, FL, USA: CRC Press, 2020.

[34] R. Prakash, L. Behera, S. Mohan, and S. Jagannathan, "Dual-loop optimal control of a robot manipulator and its application in warehouse automation," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 1, pp. 262–279, Jan. 2022.

[35] R. Prakash and L. Behera. *Neural Optimal Control for Constrained Visual Servoing via Learning From Demonstration*. Youtube. Accessed: Jun. 20, 2023. [Online]. Available: https://youtu.be/R93W94G4cnY

[36] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Ravi Prakash (Member, IEEE) received the Ph.D. degree in control and automation from the Department of Electrical Engineering, IIT Kanpur, Kanpur, India. He is currently a Post-Doctoral Researcher with Cognitive Robotics, 3mE, TU Delft, The Netherlands. His current research interests include robotics, motion modeling, imitation learning, adaptive critic-based optimal control, and visual servoing.

Laxmidhar Behera (Senior Member, IEEE) received the Ph.D. degree from IIT Delhi, New Delhi, India, in 1996. He was a Post-Doctoral Researcher with the German National Research Center for Information Technology, GMD, Sank Augustin, Germany, in 2001. He is currently working as the Director, IIT Mandi on deputation from IIT Kanpur, having research and teaching experience of more than 25 years. He has published more than 280 papers in peer-reviewed journal articles and refereed conference proceedings and has guided 19 Ph.D.'s to completion. His current research interests include the convergence of machine learning, control theory, robotic vision, and heterogeneous robotic platforms. He is a fellow of INAE.