Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 6639963, 15 pages
https://doi.org/10.1155/2023/6639963

Research Article

A Digital Twin-Based Visual Servoing with Extreme Learning Machine and Differential Evolution

Minghao Cheng,[1] Hao Tang,[2] Asad Khan,[3] Syam Melethil Sethumadhavan,[4] Muhammad Assam,[5] Di Li,[1] Yazeed Yasin Ghadi,[6] Heba G. Mohamed,[7] and Uzair Aslam Bhatti[2]

[1]*School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, China*
[2]*School of Information and Communication Engineering, Hainan University, Haikou 570100, China*
[3]*School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China*
[4]*IoT Research Center, Shenzhen University, Shenzhen 518060, Guangdong, China*
[5]*Department of Software Engineering, University of Science and Technology Bannu, Bannu, KP, Pakistan*
[6]*Department of Computer Science, Al Ain University, Abu Dhabi, UAE*
[7]*Department of Electrical Engineering, College of Engineering, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia*

Correspondence should be addressed to Hao Tang; melineth@hainanu.edu.cn, Asad Khan; asad@gzhu.edu.cn, Heba G. Mohamed; hegmohamed@pnu.edu.sa, and Uzair Aslam Bhatti; uzairaslambhatti@hotmail.com

Received 25 March 2023; Revised 21 May 2023; Accepted 27 May 2023; Published 10 June 2023

Academic Editor: Subrata Kumar Sarker

The technology of visual servoing, with the digital twin as its driving force, holds great promise and advantages for enhancing the flexibility and efficiency of smart manufacturing assembly and dispensing applications. The effective deployment of visual servoing is contingent upon the robust and accurate estimation of the vision-motion correlation. Network-based methodologies are frequently employed in visual servoing to approximate the mapping between 2D image feature errors and 3D velocities, offering promising avenues for improving the accuracy and reliability of visual servoing systems. These developments have the potential to fully leverage the capabilities of digital twin technology in the realm of smart manufacturing. However, obtaining sufficient training data for these methods is challenging, and thus improving model generalization to reduce data requirements is imperative. To address this issue, we offer a learning-based approach for estimating Jacobian matrices of visual servoing that organically combines an extreme learning machine (ELM) and a differential evolutionary algorithm (DE). In the first stage, the pseudoinverse of the image Jacobian matrix is approximated using the ELM, which solves the problems associated with traditional visual servoing and is resistant to outside influences such as image noise and mistakes in camera calibration. In the second stage, differential evolution is utilized to select input weights and hidden layer bias and to determine ELM's output weights. Experimental results conducted on a digital twin operating platform for 4-DOF robot with an eye-in-hand configuration demonstrate better performance than classical visual servoing and traditional ELM-based visual servoing in various cases.

1. Introduction

The digital twin (DT) technology endeavors to fabricate a simulated model of a tangible device, coupled with a comprehensive analysis of its life cycle and implementation model, to enhance the safety and manufacturing efficiency of the application system. In this regard, the vision servoing (VS) technology is a fundamental enabler, wherein the DT and VS are synergistically employed to achieve optimal system performance. VS is a closed-loop control approach that integrates robot motion control with visual information to rapidly process images and minimize error towards the desired position. The dynamic responsiveness and environmental adaptability of VS's closed-loop architecture make it a versatile technique with diverse applications, including industrial robots [1, 2], automated guided vehicles [3, 4], unmanned aerial vehicles (UAVs) [5, 6], underwater robots [7, 8], and medical robots [9, 10], among others.

Closed-loop feedback control is the defining feature of VS. Based on the type of visual information utilized to provide feedback error, VS can be classified into three categories: (1) position-based visual servoing (PBVS) [11, 12], which defines the error in cartesian space; (2) image-based visual servoing (IBVS) [13, 14], which defines the error in image space; and (3) Hhybrid visual servoing (HBVS) [15, 16], which combines both of these characteristics but is computationally intensive. Among these approaches, IBVS has gained popularity as a mainstream research method due to its balance of robustness and accuracy, which is the primary focus of our work.

One of the primary challenges in IBVS is the acquisition of the image Jacobian matrix and its pseudoinverse. The image Jacobian defines the mapping relationship between the error in image features in 2D space and velocities in 3D space, which is constructed using calibration parameters, feature projection models, and depth information of the features relative to the camera frame. The analytic expressions for the image Jacobian corresponding to various types of image features have been derived in [17]. However, as the complexity of image features increases, obtaining the pseudoinverse of the image Jacobian for use in VS control law can be challenging. Furthermore, stability and convergence issues can arise in certain VS scenarios, such as camera retreat when the goal is rotated about the z-axis [18].

To address these challenges, an offline or online scheme can be employed to estimate the numerical value of the pseudoinverse. In the online scheme, the image jJacobian matrix is viewed as an optimization problem that can be solved using optimization methods. For example, a dynamic Broyden's method has been proposed for model-independent visual servo control without precise calibration of kinematic and camera models in [19]. Similarly, a novel algorithm for multicamera pose estimation using virtual visual servoing has been proposed [20]. However, these iterative approaches may face issues such as unexpected camera motion due to poor selection of initial conditions or updated speed. In contrast, the offline scheme addresses the mapping problem using neural networks, which can avoid issues related to training well models. However, the computational complexity and error-proneness of vision-based robot positioning techniques often limit the number of controllable degrees of freedom. A novel method based on global picture descriptors and neural learning has been introduced in [21] to address these issues. An adaptive neural network module has also been constructed to approach unknown dynamics, allowing for control of robots with nonlinear and structurally unknown dynamics in [22]. However, the control parameters (learning rate, learning epochs, etc.) of these methods need to be tuned manually, which can be challenging when faced with local minima, and training can be time-consuming when there is a lot of training data.

The preamble machine learning techniques are used in the digital twin to analyze physical and virtual data while enhancing the design properties and operational characteristics of key aspects of the visual servo system based on a data-driven approach to further improve the system performance. In recent times, the learning efficiency of the extreme learning machine (ELM), a new neural algorithm, has been found to surpass traditional offline methods significantly. Remarkably, the ELM algorithm necessitates solely the determination of the number of hidden nodes as its solitary parameter. Previous works have attested to the benefits of ELM in this regard [23, 24]. ELM, in combination with fuzzy logic (FL), has been proffered as a viable solution for addressing three common problems in visual servoing (VS), which entail obtaining the interaction matrix and its pseudoinverse for defined feature points, selecting an appropriate gain value for the VS controller, and ensuring that features remain within the field of view (FOV) for VS permanency [25]. In addition, a novel estimation technique based on ELM and Q-learning has been put out in [26] to address complex modeling problems and achieve effective servo gain selection.

Notwithstanding the extensive research conducted on the matter, a significant oversight in the literature pertains to the inherent challenges associated with obtaining a considerable quantity of industrial field data sets to ensure a large range convergence of VS. Moreover, the weights pertaining to the input and hidden layer Q, along with the hidden layer bias B, remain static during the training phase. This characteristic may lead to suboptimal network performance when a substantial disparity exists between the training and validation data. Thus, if we are able to devise a means to update Q and B under specific circumstances while using the same or fewer training data as traditional ELM, it may be possible to enhance the prediction accuracy of the model. The Moore–Penrose (MP) generalized inverse of ELM is used to find the output weights analytically, and the differential evolutionary algorithm is used to choose the input weights. This hybrid learning technique is detailed in [27]. However, it should be noted that, to the best of our knowledge, this approach has not yet been implemented in a visual servoing application.

This paper proposes a novel digital twin-based approach to visual servoing that integrates extreme learning machine (ELM) and evolutionary algorithm (EA). The proposed approach leverages the capabilities of ELM to approximate the pseudoinverse of the image Jacobian matrix, which effectively addresses matrix singularity and provides robustness against image noise and camera calibration errors. Compared to other classification techniques [28, 29], ELM significantly reduces the training time since it avoids slow gradient-based learning algorithms and eliminates the need for iterative parameter calculation. Furthermore, because EA is frequently used as a global optimization seeking strategy, the combination of analytical methodologies and EA is anticipated to be successful for network training. In the suggested method, EA is used to pick the best input weights and hidden biases, allowing the network parameters to be adaptively changed to raise the model's prediction accuracy. Furthermore, in the context of platform development, we

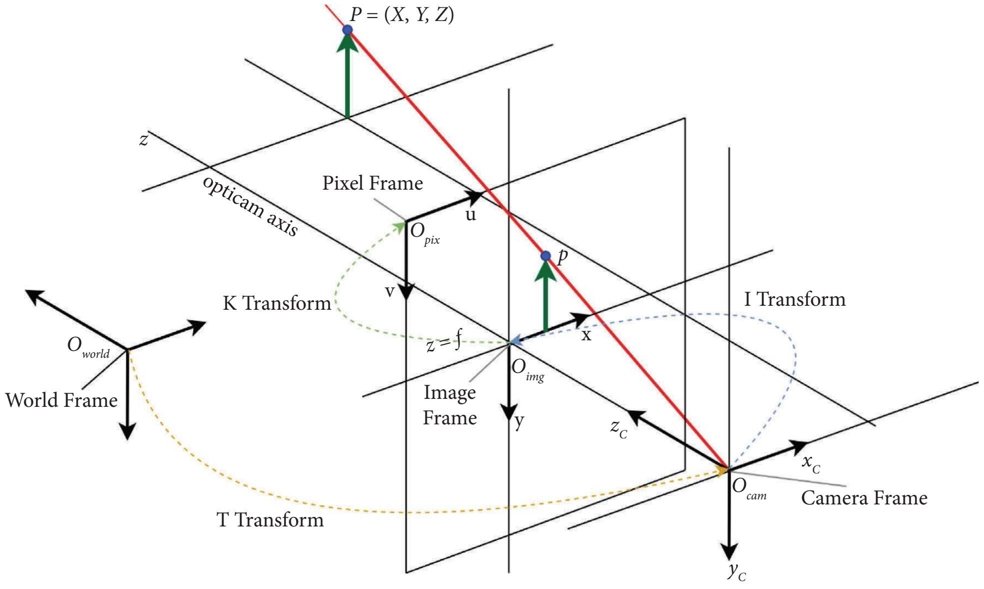


Figure 1: Pinhole model of camera.

have devised a digital twin infrastructure grounded on the EtherCAT real-time Ethernet technology, tailored to physical robotic entities, with the aim of guaranteeing the safety of scenario deployment and the efficacy of performance validation. Overall, the proposed visual servoing approach with ELM and EA is expected to provide better performance than conventional techniques in challenging robotic applications.

The present article is structured as follows: in Section 2, an exposition of the sophisticated approximation methodology for IBVS is presented. The fundamental principles of ELM and EA are introduced and subsequently employed in tandem within the visual servoing module. Section 3 presents empirical results based on a 4-axis robot digital twin operating platform, which helps to validate the effectiveness of the proposed approach. Finally, Section 4 outlines the conclusion derived from the study.

2. A Novel Visual Servoing with ELM and EA

The present section furnishes an exhaustive exposition of the proposed methodology for IBVS, with a specific focus on point features, designed for robot manipulators operating in an eye-in-hand configuration.

The present exposition commences by expounding upon the process of image plane projection. Specifically, the pinhole model of a camera, depicted in Figure 1, is introduced. It is assumed that the camera captures K pixels, denoted as $s_i = (u_i, v_i)^T, i = 1, \ldots, K$ and that the origin O_{pix} is located at the top left-hand corner of the image plane, in accordance with convention. The proposed methodology proceeds to consider n feature points, denoted as $p_i = (x_i, y_i)^T, i = 1, \ldots, n \in K$ in the image frame, and corresponding desired image coordinates p_i^*, which are typically obtained through a priori knowledge. In general, the camera projection can be expressed in the following form:

$$s = \text{Kp}, \gamma p = \text{IP}, P = T\widetilde{P}, \tag{1}$$

where

$$K = \begin{pmatrix} \dfrac{f}{\rho_w} & 0 & u_0 \\ 0 & \dfrac{f}{\rho_h} & v_0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}. \tag{2}$$

The notation $p = (x_i, y_i, 1)^T$ denotes the homogenous expression of the image point p_i, while γ represents the features' depth information relative to the camera frame. The camera parameter matrix is denoted by K, and I is the project matrix. The pixel dimensions are given by ρ_w and ρ_h for width and height, respectively, and (u_0, v_0) denotes the pixel coordinate of the principal point. Furthermore, $P = (X_i, Y_i, Z_i)^T$ represents the feature points expressed in the camera frame, with $P = (X_i, Y_i, Z_i, 1)^T$ denoting its homogenous expression. T is a 3×4 homogenous transformation matrix that relates the camera frame and the end effector frame, and $\widetilde{P} = (X_w, Y_w, Z_w, 1)^T$ denotes the feature points expressed in the world frame. Feature points expressed in different frames are obtained by (1). The goal of IBVS system is to minimize

$$e = p_i - p_i^*, \tag{3}$$

where $e = (e_1, e_2, \ldots, e_i)$ is the vector of the features error signal. To convert image space error signals to manipulator pose space, we use the following relationship:

$$\dot{e} = L_e V_c, i = 1, \ldots, n, \tag{4}$$

where $\dot{e}$ is the time derivative of features error and $V_c = (v_c, w_c) = (v_x, v_y, v_z, w_x, w_y, w_z)$ is the camera spatial velocity. The link between the variation of e and the camera velocity v_c is described by the image Jacobian matrix L_e. According to the classical kinematic theory,

$$\dot{P} = -v_c - \omega_c \times P \Leftrightarrow \begin{cases} \dot{X} = -v_x - \omega_y Z + \omega_z Y, \\ \dot{Y} = -v_y - \omega_z X + \omega_x Z, \\ \dot{Z} = -v_z - \omega_x Y + \omega_y X. \end{cases} \tag{5}$$

Take the derivative of p in (1)

$$\dot{p} = \frac{I\dot{P}}{\gamma} \Leftrightarrow \begin{cases} \dot{x} = \dfrac{(\dot{X} - x\dot{\gamma})}{\gamma}, \\ \qquad\qquad\qquad\qquad \gamma = Z \\ \dot{y} = \dfrac{(\dot{Y} - y\dot{\gamma})}{\gamma}, \end{cases} \tag{6}$$

Injecting (5) in (6), where L_e related to e is

$$L_e = \begin{bmatrix} \dfrac{-1}{Z} & 0 & \dfrac{x}{Z} & xy & -\left(1 + x^2\right) & y \\ \\ 0 & \dfrac{-1}{Z} & \dfrac{y}{Z} & 1 + y^2 & -xy & -x \end{bmatrix}. \tag{7}$$

The final general control law can be calculated by using V_c as the input to the low level controller and obtaining

$$V_c = -\lambda \widehat{L_e^+} e, \tag{8}$$

where λ is a control gain to ensure an exponential decrease of the error, $\widehat{L_e^+} \in R^{6\times 2n}$ is the estimation of Moore–Penrose pseudoinverse of the image matrix.

As shown in (8), the primary objectives of visual servoing control are to approximate the relationship between the feature error signal e and the camera spatial velocity V_c, which is primarily determined by the control gain and the estimation of the Moore–Penrose pseudoinverse of the image matrix $\widehat{L_e^+}$. However, as image attributes become increasingly intricate, computing $\widehat{L_e^+}$ becomes a laborious and complicated task. To address this challenge, ELM is employed to learn the nonlinear coupling relationship between e and V_c by randomly selecting the input weights and hidden bias. Furthermore, DE is utilized to enhance the performance of the visual servoing controller by tuning the input weights and hidden bias.

The accurate estimation of $\widehat{L_e^+}$ is essential to determine the performance of the visual servoing system, but it is a challenging task due to its dependence on camera calibration parameters and object model parameters, which are difficult to precisely determine. Considering the lack of extensive research on the estimation form of $\widehat{L_e^+}$, a novel method to approximate it, is proposed. ELM is a type of single-hidden layer feedforward neural network algorithm (SLFN), and its algorithm framework is presented in Figure 2. Distinguishing itself from conventional gradient-based learning algorithms such as Levenberg–Marquardt (LM) and backpropagation (BP), ELM employs randomly generated or artificially set weight coefficients and biases for the hidden layer, which obviates the need for gradient descent method (GDM) for error backpropagation and reduces the requirement for tuning. The objective of ELM is to determine the weights of the output layer. Based on these properties, ELM offers faster computing efficiency and stronger computing power than traditional methods.

In the context of the IBVS system, assuming a given set of N known input samples, $\mathrm{XI} = [\mathrm{XI}_1, \mathrm{XI}_2, \ldots, \mathrm{XI}_i] \in R^{m\times N}$, where m is the size of the input vector for a single sample, and $\mathrm{YO} = [\mathrm{YO}_1, \mathrm{YO}_2, \ldots, \mathrm{YO}_i] \in R^{n\times N}$, where n is the size of the output vector from a single sample. Accordingly, we can express each sample as $\mathrm{XI}_i = [e_{i1}, e_{i2}, \ldots, e_{\mathrm{im}}]^T \in R^m, i = 1, \ldots, N$ and $\mathrm{YO}_i = [V_{ci1}, V_{ci2}, \ldots, V_{cin}]^T \in R^n, i = 1, \ldots, N$.

In this paper, we employ AprilTag [30, 31] as the target and extract four corners as the image features. Subsequently, we utilize ELM to estimate the Cartesian space velocity of the camera, where the dimensions of the input and output vectors are $m = 8$ and $n = 6$, respectively. The hidden layer's output function is

$$f_L = \beta H, \tag{9}$$

where β is $n \times L$ weight matrix, which maps the hidden layer to output layer, H is $L \times N$ weight matrix, which maps the input layer to hidden layer. H is defined as

$$H = G(x) = G(\mathrm{QXI} + B). \tag{10}$$

In this regard, the function $G(x)$ is commonly known as the neuron activation function or feature mapping function. It can be defined as various functions such as the sigmoid function, RBF function, and others. In addition, Q stands for the $L \times m$ matrix, which is defined at random as the initial value for the hidden layer's input. Furthermore, $B = [B_1, B_2, \ldots, B_L]$ is a representation of the L hidden nodes' bias.

The primary objective of this process is not to learn Q and B, but rather to minimize the error between f_L and YO. Therefore, the objective function can be defined as

$$\min_{\boldsymbol{\beta}} = \| \beta H - \mathrm{YO} \| = \| \widehat{\beta} H - \mathrm{YO} \|. \tag{11}$$

The Moore–Penrose pseudoinverse matrix of H is calculated in this context, and it is multiplied by YO to produce the lowest norm least squares solution of the objective function $\widehat{\beta}$. In other words, $\widehat{\beta}$ can be expressed as $\widehat{\beta} = H^+ YO$.

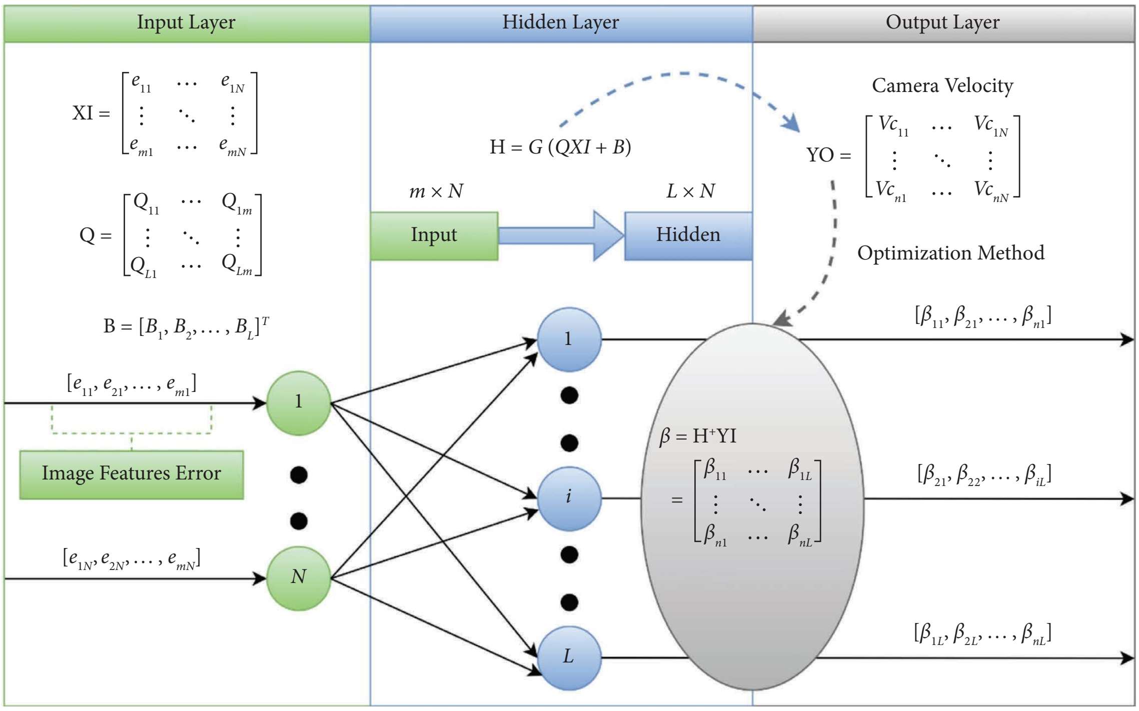

Input Layer
Hidden Layer
Output Layer
$XI = \begin{bmatrix} e_{11} & \cdots & e_{1N} \\ \vdots & \ddots & \vdots \\ e_{m1} & \cdots & e_{mN} \end{bmatrix}$
$Q = \begin{bmatrix} Q_{11} & \cdots & Q_{1m} \\ \vdots & \ddots & \vdots \\ Q_{L1} & \cdots & Q_{Lm} \end{bmatrix}$
$B = [B_1, B_2, \ldots, B_L]^T$
$[e_{11}, e_{21}, \ldots, e_{m1}]$
Image Features Error
$[e_{1N}, e_{2N}, \ldots, e_{mN}]$
1
N
H = G (QXI + B)
m × N
L × N
Input
Hidden
1
i
L
Camera Velocity
$YO = \begin{bmatrix} Vc_{11} & \cdots & Vc_{1N} \\ \vdots & \ddots & \vdots \\ Vc_{n1} & \cdots & Vc_{nN} \end{bmatrix}$
Optimization Method
$\beta = H^{+}YI = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1L} \\ \vdots & \ddots & \vdots \\ \beta_{n1} & \cdots & \beta_{nL} \end{bmatrix}$
$[\beta_{11}, \beta_{21}, \ldots, \beta_{n1}]$
$[\beta_{21}, \beta_{22}, \ldots, \beta_{iL}]$
$[\beta_{1L}, \beta_{2L}, \ldots, \beta_{nL}]$

Figure 2: Algorithm framework of ELM.

Once the matrix $\widehat{\beta}$ has been trained, it can be utilized to predict the camera space velocity V_c for a given feature error input vector using (9).

The random assignment of input weights and hidden bias can potentially impact the accuracy of the model's predictions. To mitigate this issue, differential evolution (DE) is employed. DE is a type of evolutionary algorithm (EA) first proposed by Storn and Price [32]. The fundamental concept behind utilizing DE in ELM can be described as follows:

Step 1, *Population Initialization*: here, NP represents the population size, and D is the dimension of each individual. In our particular design, each individual consists of input weights and hidden bias, with a total dimension of $D = L \times (n + 1)$. Thus, a population can be generated in the following manner:

$$\theta_{i,G}, i = 1, 2, \ldots, \mathrm{NP} \in R^{D}. \tag{12}$$

Step 2, *Mutation*: like living creatures, the individuals in the population (denoted as $\theta_{i,G}$) are subject to mutation. Consequently, mutant individuals can be described as follows:

$$\nu_{i,G+1} = \theta_{r1,G} + F \times \left(\theta_{r2,G} - \theta_{r3,G}\right). \tag{13}$$

The current investigation revolves around the selection of indices, specifically referred to as r_1, r_2, and r_3, from a given population. It is essential that these indices are chosen in a random manner and are distinct from each other. To ensure adherence to this requirement, the population size, denoted as NP, must surpass a value of four. Furthermore, the variable F, which is constrained within the interval $[0, 2]$, represents the zoom factor that governs the extent of scaling for differential variation.

Step 3, *Crossover*: to broaden the population's diversity, it is imperative to carry out crossover operations, and the D-dimensional of crossover individuals is defined as follows:

$$\mu_{i,G+1} = \left(\mu_{1i,G+1}, \mu_{2i,G+1}, \ldots, \mu_{Di,G+1}\right), \tag{14}$$

where

$$\mu_{\mathrm{ji},G+1} = \begin{cases} \nu_{ji,G+1} \text{ if } (\mathrm{rand}(j) \le \mathrm{CR} \text{ or } j = \mathrm{rnbr}(i)), \\ \theta_{\mathrm{ji},G} \quad \text{if } (\mathrm{rand}(j) > \mathrm{CR} \text{ or } j \ne \mathrm{rnbr}(i)). \end{cases} \tag{15}$$

Herein, $\mathrm{rand}(j)$, where $j = 1, 2, \ldots, D$, represents the jth value that is randomly generated and conforms to the range of $[0, 1]$. The crossover probability factor CR, which resides in the interval $[0, 1]$, is subject to determination by the user. Additionally, $\mathrm{rnbr}(i)$ denotes an index that is randomly selected from the set of integers $[1, 2, \ldots, D]$. This index guarantees that at least one parameter from $\nu_{\mathrm{ji},G+1}$ is incorporated into the vector $\mu_{i,G+1}$.

Step 4, *Selection*: regarding the selection of individuals for the ensuing generation, the mutant individual denoted by $\mu_{i,G+1}$ is subjected to comparison with $\theta_{i,G+1}$. If the former displays a higher level of performance in relation to the latter, the former shall be designated for the succeeding generation. In contrast, should $\theta_{i,G}$

```
Input
XI: the input training set, the size is $m \times N$
YO: the output training set, the size is $n \times N$
XI$'$:  the input validation set, the size is $n \times N'$
YO$'$: the output validation set, the size is $n \times N'$
Output
$V_c$: the camera cartesian space velocity
Begin
  Normalized XI and XI$'$ in $[0, 1]$;
  $Q = 2 \times \mathrm{rand}(L, m) - 1$;
  $B = 2 \times \mathrm{rand}(L, 1) - 1$;
  $\theta = \mathrm{rand}(\mathrm{NP} - D) * (b - a) + a$;
  for $i = 1{:}\ 1{:}\ \mathrm{NP}$ do
    Output value of hidden layer $H = Q \times \mathrm{XI} + B$;
    Activation function $h = 1/(1 + \exp\ (H))$;
    Output weight matrix $\beta = \mathrm{pinv}(h') \times \mathrm{YO}'$;
    Output value of hidden layer $H_1 = Q \times \mathrm{XI} + B$;
    Activation function $h_1 = ((1 + \exp(H_1))/)$;
    $V_c = (h_1{}' \times \beta)'$;
    $ob(i) = \mathrm{RMSE}(\theta_{\mathrm{ini}}, V_c, YO')$;
  end for
  for i = 1: 1: G do
    $\nu(i + 1) = \theta(r_1, i) + F \times (\theta(r_2, i) - \theta(r_3, i))$;
    $\mu(i + 1) = \mathrm{Corssover}(\nu(i + 1), \theta(i))$;
    $\mathrm{ob}(i) = \mathrm{RMSE}(\theta(i), V_c, YO')$;
  end for
  $Q, B \leftarrow$ get optimal value based on ob;
  Return $V_c$;
end
```

Algorithm 1: Framework of the proposed EA-ELM-IBVS.

outperform $\mu_{i,G+1}$, the original individual $\theta_{i,G}$, will be preserved.

Based on the above basis, a novel approach named EA-ELM-IBVS is proposed and its framework is shown in Algorithm 1.

Before testing the accuracy of the ELM network on the validation set, we know that Q is of size $L \times m$ and B is of size $1 \times L$ in (10). As a result, we create the population at random. Every member of the population possesses all Q and B parameters as a vector

$$\theta = [Q_{11}, Q_{12}, \ldots, Q_{1m}, Q_{21}, Q_{22}, \ldots, Q_{2m}, Q_{L1}, Q_{L2}, \ldots, Q_{Lm}, B_1, B_1, \ldots, B_L] \in R^D. \tag{16}$$

All values of Q_{ij} and B_i are initialized randomly within the interval $[-1, 1]$. Subsequently, we calculate the output weights β for each individual using the least squares solution in accordance with (11). The activation function adopted in this context is as follows:

$$f(t) = \frac{1}{1 + \exp(t)}. \tag{17}$$

The fitness function is described as follows:

$$\mathrm{fit} = \sqrt{\frac{\sum_{j=1}^{N} \left\| \sum_{i=1}^{L} \beta_i G\left(Q_i \times XI_j + B_i\right) - YO_j \right\|_2^2}{m \times N}}, \tag{18}$$

where fit is the root mean squared error (RMSE) to evaluate each individual.

Conventionally, the RMSE is employed as the fitness metric, derived from the complete training set. However, given that the ELM optimizes β through the least-squares solution, evaluating the fitness metric based on the entire training set may result in overfitting. Therefore, in order to achieve enhanced performance of the visual servoing controller while mitigating computational overhead, we exclusively assess the RMSE on the validation set. Subsequently, once the fitness of each individual has been ascertained, we proceed to delve into the core concept of DE, encompassing mutation, crossover, and selection operations.

The user often assigns a fixed value to the zoom factor F during mutation, which has an impact on the ability for global optimization. In the event, where F is set to a smaller value, the model will tend to escape from the local minimum, leading to a slower convergence rate. To circumvent this issue, an adaptive approach is described below for adjusting the value of F:

$$F = F_0 \times 2^{\gamma}, \tag{19}$$

where

$$\gamma = e^{1-(G+1-i/)}, i \in [1, G], \tag{20}$$

where F_0 denotes the initial value of the zoom factor, and F ranges between F_0 and $2F_0$. During the early stage of evolution, F is assigned a relatively large value to preserve the diversity of individuals and forestall the occurrence of premature convergence. Conversely, during the later stages of evolution, F is assigned a smaller value to retain valuable information, prevent the degradation of the optimal solution, and augment the likelihood of discovering the global optimum.

While performing crossover, if the zoom factor F assumes a large value, the mutated individuals may acquire values outside of the permissible range. As a result, a limit function is added that has the following definition to get around this problem:

$$\mu_{\mathrm{ji},G} = \begin{cases} a \text{ if } \mu_{\mathrm{ji},G} < a, \\ b \text{ if } \mu_{\mathrm{ji},G} > b, \end{cases} j = 1, 2, \ldots, D, \tag{21}$$

where a and b are the lower and upper bounds. Here, we define $a = -1$ and $b = 1$.

During the selection process, the mutated population is compared with the original population based on their fitness. To achieve this, the output weights of the mutated population are computed and their corresponding fitness is evaluated on the validation set. The individuals with better fitness are then chosen for the next generation. A minor validation error does not necessarily imply a small testing error, which depends mainly on the distribution of the validation data. Therefore, choosing individuals only on the basis of fitness may not be the best course of action. In order to overcome this problem, and motivated by [27], a new criterion, the norm of output weights, denoted as $\|\beta\|$, is added to the selection process. The individual who results in a smaller value of $\|\beta\|$ is chosen when the difference in fitness between several individuals is minimal. Therefore, the selection strategy for the next generation is defined as

$$\theta_{i,G+1} = \begin{cases} \mu_{i,G} \text{ if } \mathrm{fit}\left(\theta_{i,G}\right) - \mathrm{fit}\left(\mu_{i,G}\right) & > \epsilon \, \mathrm{fit}\left(\theta_{i,G}\right), \\ \mu_{i,G} \text{ if } \mathrm{fit}\left(\theta_{i,G}\right) - \mathrm{fit}\left(\mu_{i,G}\right) & > \epsilon \, \mathrm{fit}\left(\theta_{i,G}\right), \\ \left\|\beta^{\mu_i,G}\right\| & < \left\|\beta^{\theta_i,G}\right\|, \\ \theta_{i,G} & \text{else.} \end{cases} \tag{22}$$

In this formula, fit is the fitness function, and ϵ is the predetermined tolerance rate. After selecting a fresh population, determine its fitness before starting the DE process again and repeating it until the desired iteration number or range of convergence is obtained.

3. Experiment Result

3.1. Numerical Simulations. Three distinct systems' performances were compared using simulations: the suggested IBVS system (EA-ELM-IBVS), the extreme learning machine-based IBVS (ELM-IBVS), and the classical image-based visual servoing (C-IBVS). The volume and distribution of training data have a significant impact on the mapping model's capacity to learn the link between image attributes and the end effector's spatial velocity. However, acquiring training data that cover the entire workspace of the robot can be highly time-consuming. Therefore, improving the generalization ability of the mapping model with fewer training data can expand the application capability of the system. To demonstrate the advantages of the proposed method, experiments were designed and conducted using MATLAB and the Machine Vision Toolbox [18]. The camera was mounted at the end of the end effector, and the homogeneous transformation matrix between the camera and the end of the manipulator ${}_{T}^{C}H$ was assumed to be the identity. This can be expressed as

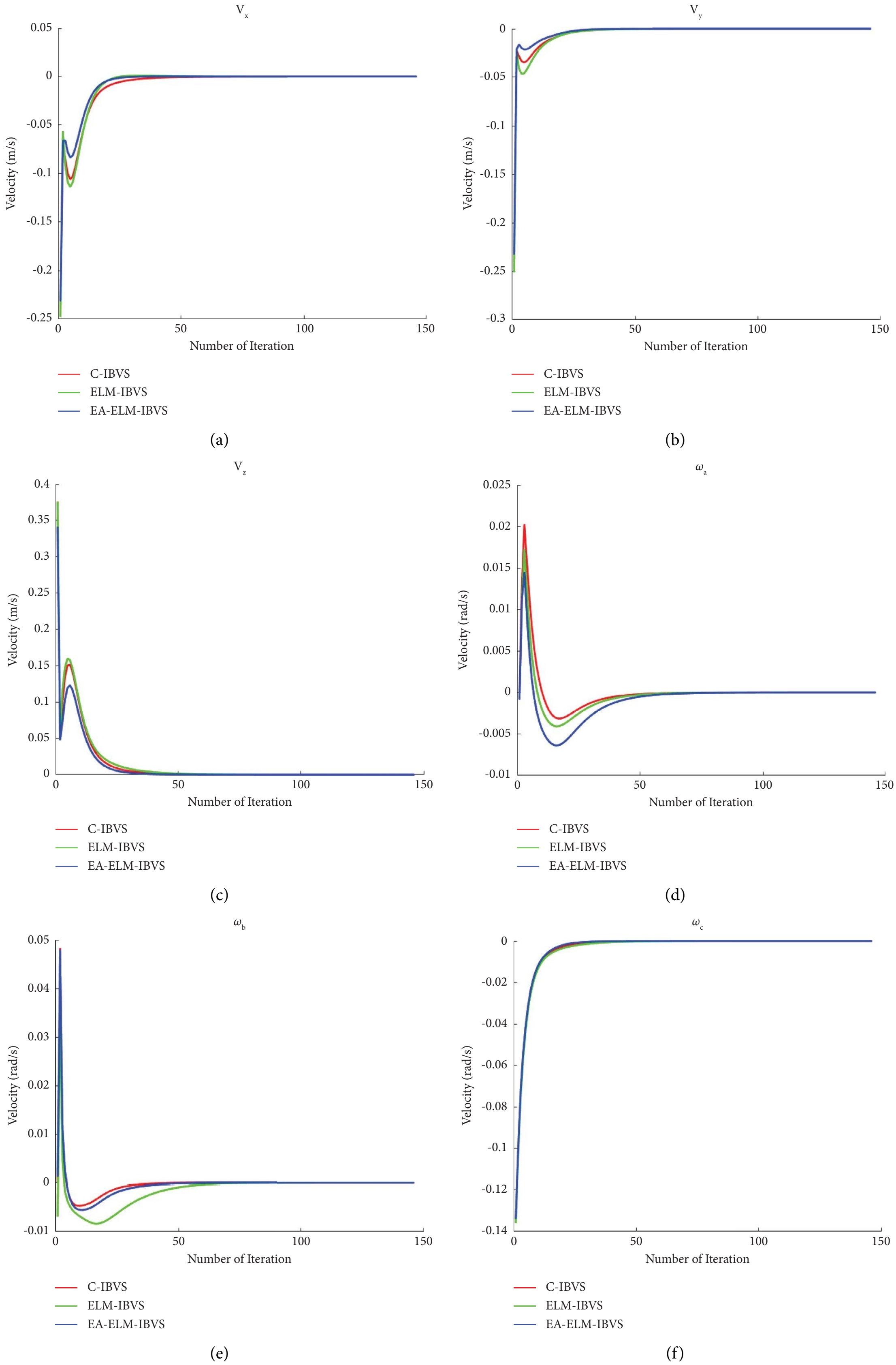
$${}_{T}^{C}H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{23}$$

The camera resolution is specified as 1024 × 1024, with the pixel coordinates of the camera's origin being $(512, 512)$. The camera frame rate is identified as $33Hz$, and the focal length is determined to be $0.008m$. Additionally, the desired feature points in both Cartesian space, denoted as P^*, and the pixel frame, denoted as s^*, are provided as

$$s^* = \begin{bmatrix} 312 & 312 & 712 & 712 \\ 312 & 712 & 712 & 312 \end{bmatrix},$$
$$P^* = \begin{bmatrix} -0.25 & -0.25 & 0.25 & 0.25 \\ -0.25 & 0.25 & 0.25 & -0.25 \\ 3 & 3 & 3 & 3 \end{bmatrix}. \tag{24}$$

The successful completion of the designated task is contingent upon the attainment of a feature error of less than σ pixels. In the current study, the number of hidden nodes, denoted by L, is set to 20, whereas the population number, denoted by NP, is established at 400. Furthermore, the algebra of population evolution, represented by G, is determined to be 200. These technical specifications are critical in guiding the implementation and optimization of the utilized algorithm, emphasizing their importance in achieving the desired outcomes.

The forthcoming subsections adopt two distinct scenarios to facilitate a comprehensive comparative evaluation of C-IBVS, ELM-IBVS, and the proposed EA-ELM-IBVS. In the first scenario, the validation data are presumed to be in close proximity to the training data, and the camera's spatial velocity is examined across the three aforementioned methods. Subsequently, in the second scenario, a greater disparity is introduced between the initial poses of the validation and training data, wherein the superiority of the EA-ELM-IBVS approach in terms of generalization performance becomes evident. This comparison analysis sheds important light on the efficacy and robustness of the various



(a)

(b)

(c)

(d)

(e)

(f)

Figure 3: Result for case 1 for C-IBVS, ELM-IBVS, and the proposed EA-ELM-IBVS.

visual servoing techniques under various circumstances, highlighting their individual advantages and disadvantages.

3.1.1. Case 1: Validation Data ≈ Training Data. In the first scenario, ${}^{w}_{e}H_1$ is designated as the initial pose of the end effector frame relative to the world frame. Subsequently, the end effector is directed to move towards the desired pose until the image feature error converges. The calculated error and spatial velocity at each iteration of the process are utilized as the inputs and outputs of the training data, respectively. The corresponding information pertaining to ${}^{w}_{e}H_2$ is designated as the validation data. The homogeneous transformation matrix of ${}^{w}_{e}H_1$ and ${}^{w}_{e}H_2$ is of critical importance in this experimental setting, which is shown as follows:

$$
\begin{aligned}
{}^{w}_{e}H_1 &= \begin{bmatrix} 0.8253 & -0.5646 & 0 & 1 \\ 0.5646 & 0.8253 & 0 & 1 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
{}^{w}_{e}H_2 &= \begin{bmatrix} 0.8253 & -0.5646 & 0 & 0.99 \\ 0.5646 & 0.8253 & 0 & 1 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
\tag{25}
$$

It is apparent that a minute discrepancy of solely $0.01m$ in the x direction exists between the initial poses of the training data and the validation data. The forecasted spatial velocity, $V = (v_x, v_y, v_z, \omega_x, \omega_y, \omega_z)$, is depicted in Figure 3, with the red curves signifying the desired output. As inferred from the experimental outcomes, the disparity in prediction between the two techniques in the first scenario is not statistically noteworthy. Since the validation and training data are proximate, the ELM technique has computed the optimal prediction via norm least squares approximation, while the differential evolution approach has had a subtle influence.

3.1.2. Case 2: Validation Data ≠ Training Data. In the second scenario, the process information of the homogenous transformation matrix ${}^{w}_{e}H_1$ is utilized as the training dataset. Nevertheless, an alternative initial pose, ${}^{w}_{e}H_3$, associated with the world frame is designated as the validation dataset. Notably, the disparity in the x-direction is augmented by a factor of ten. The value attributed to ${}^{w}_{e}H_3$ is thereby expanded as follows:

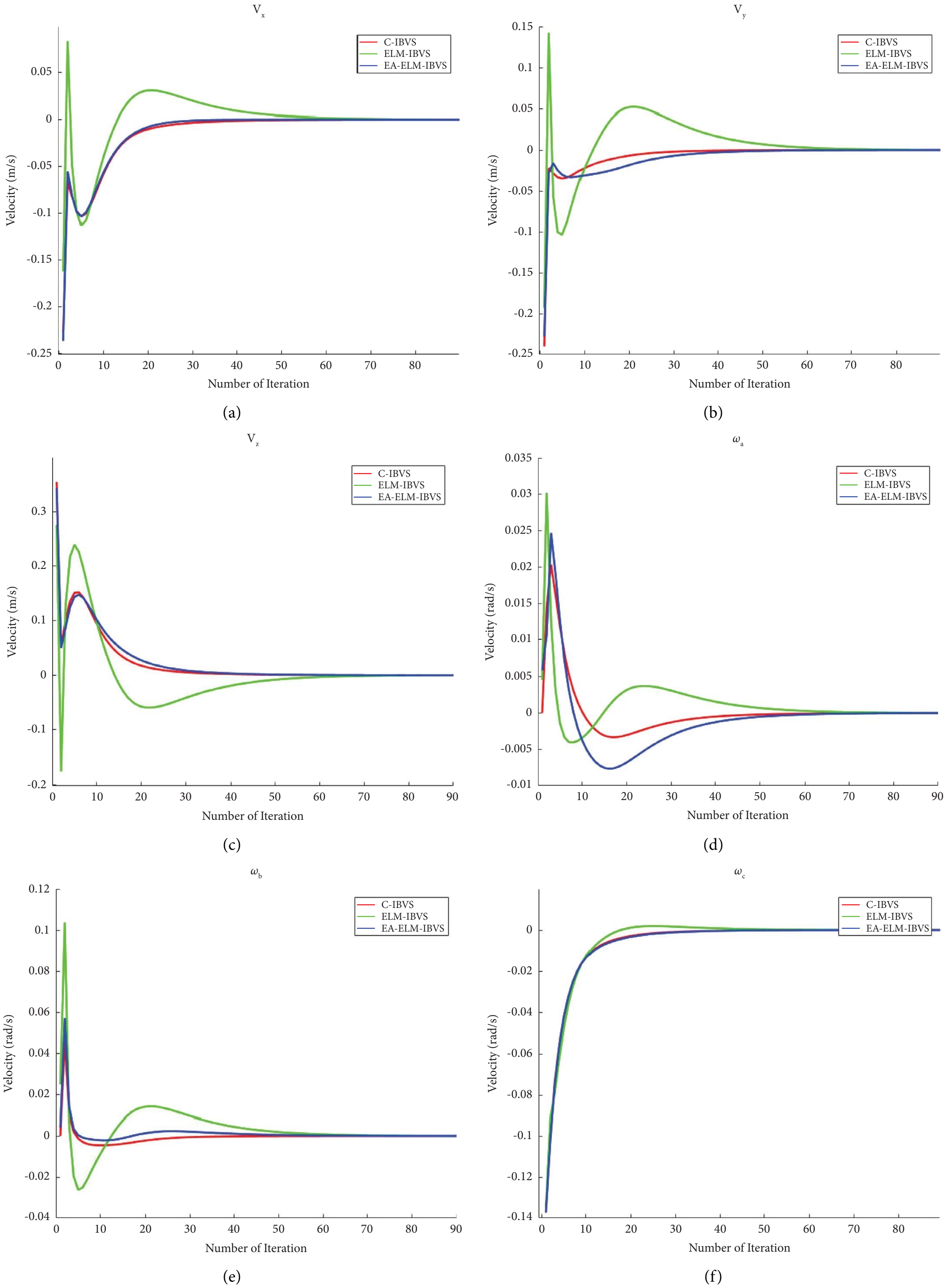
$$
{}^{w}_{e}H_2 = \begin{bmatrix} 0.8253 & -0.5646 & 0 & 0.9 \\ 0.5646 & 0.8253 & 0 & 1 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{26}
$$

Observing the results presented in Figure 4, it is evident that the discrepancy between the initial poses of the training and validation datasets is $0.1m$ in the x-direction. This value is substantially larger than the corresponding disparity in the first case. Notably, the curve denoted in green and generated by the ELM-IBVS exhibits a pronounced phenomenon of significant deviation. This trend implies that the predicted spatial velocity curve markedly increases in both slope and extreme value. Notably, if the velocity predicted by the ELM-IBVS is fed into the underlying velocity controller under identical experimental conditions, the motor will experience more significant vibrations, potentially leading to damage to the robot. In contrast, EA-ELM-IBVS manages to retain its predictive ability even when confronted with substantial differences between the testing and validation datasets.

3.2. Real Experiments. In order to ensure the safety and efficiency of physical device experimentation, we have developed a digital twin operation platform that integrates a 4-axis robot virtual model with its corresponding physical entity. This organic combination enables the concurrent evolution of both the virtual and physical components and facilitates system-level analysis and performance verification. The effectiveness of the proposed method is validated through experiments conducted on the platform, as illustrated in Figure 5, thus demonstrating its practical applicability in real-world settings. This platform encompasses a testbed composed of a high-level visual servoing controller as well as a low-level motion controller. To acquire image data, a Realsense D435 camera is affixed to the end of the SCARA robot. Subsequently, the environmental data is conveyed to the visual servoing controller, while the motion controller receives motion commands following certain processing steps.

The fundamental aspect of the construction of a digital twin framework lies in the capacity to procure accurate data from tangible apparatuses contemporaneously. In consideration of the system depicted in Figure 5, the data flow within the system is illustrated in Figure 6. To facilitate real-time activities, we opted to implement the xenomai patch on a Linux IPC (Intel i3-4150, 3.5 GHz). Subsequently, the following three programs are installed on the IPC: (1) The visual servoing controller, which processes the visual servoing algorithm, robot kinematics, and Jacobian conversion at a rate of 30–50 Hz. (2) The motion controller, which enables real-time layer control and facilitates communication of motion tasks at a frequency of 1 KHz. (3) The data collector, which generates a data buffer designed to receive pulse values for each axis in real time. Additionally, four Panasonic MADHT1507BA1 EtherCAT servo drives are configured to operate in cyclic synchronous (CSP) mode, functioning as slaves of the EtherCAT network.

In the subsequent part, we proceed to carry out experiments on actual SCARA robots in relation to cases 1 and 2, respectively. We present a scenario wherein a SCARA robot is tasked with conducting an IBVS eye-in-hand static tracking experiment. The outcomes of the experiment are showcased in Figures 7 and 8. In the context of case 1, it is evident that ELM-IBVS exhibits a consistent bias along with larger fluctuations in amplitude. In contrast, the proposed



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4: Result for case 2 for C-IBVS, ELM-IBVS, and the proposed EA-ELM-IBVS.

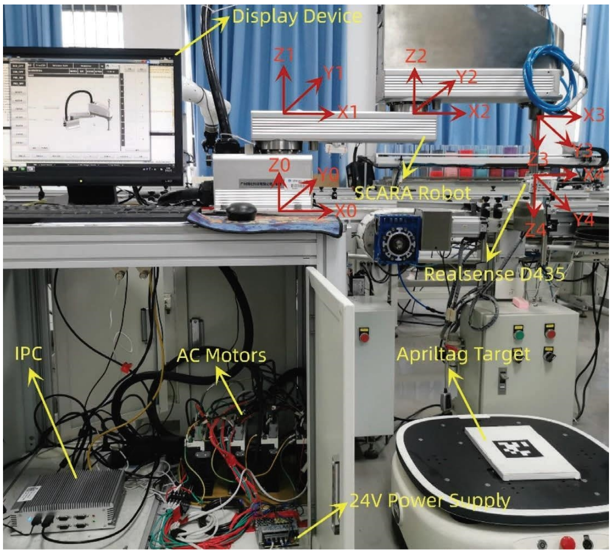


Figure 5: Structures of the testbed platform.

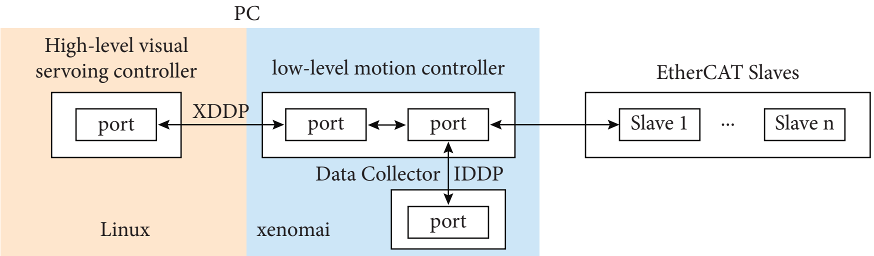


Figure 6: Data interactions in the testbed platform.

EA-ELM-IBVS successfully achieves convergence at the desired speed and maintains smaller fluctuations. Moving on to case 2, the fluctuations observed in ELM-IBVS are more pronounced. Additionally, a peculiar phenomenon arises where the predicted velocity direction contradicts the expected velocity direction, which is also observed in EA-ELM-IBVS. To address this issue, the most straightforward solution entails augmenting the training data samples. However, merely increasing the number of training samples falls short in meeting the requirements of robots operating in a wide range, thereby presenting a significant research problem that necessitates further investigation in subsequent studies.

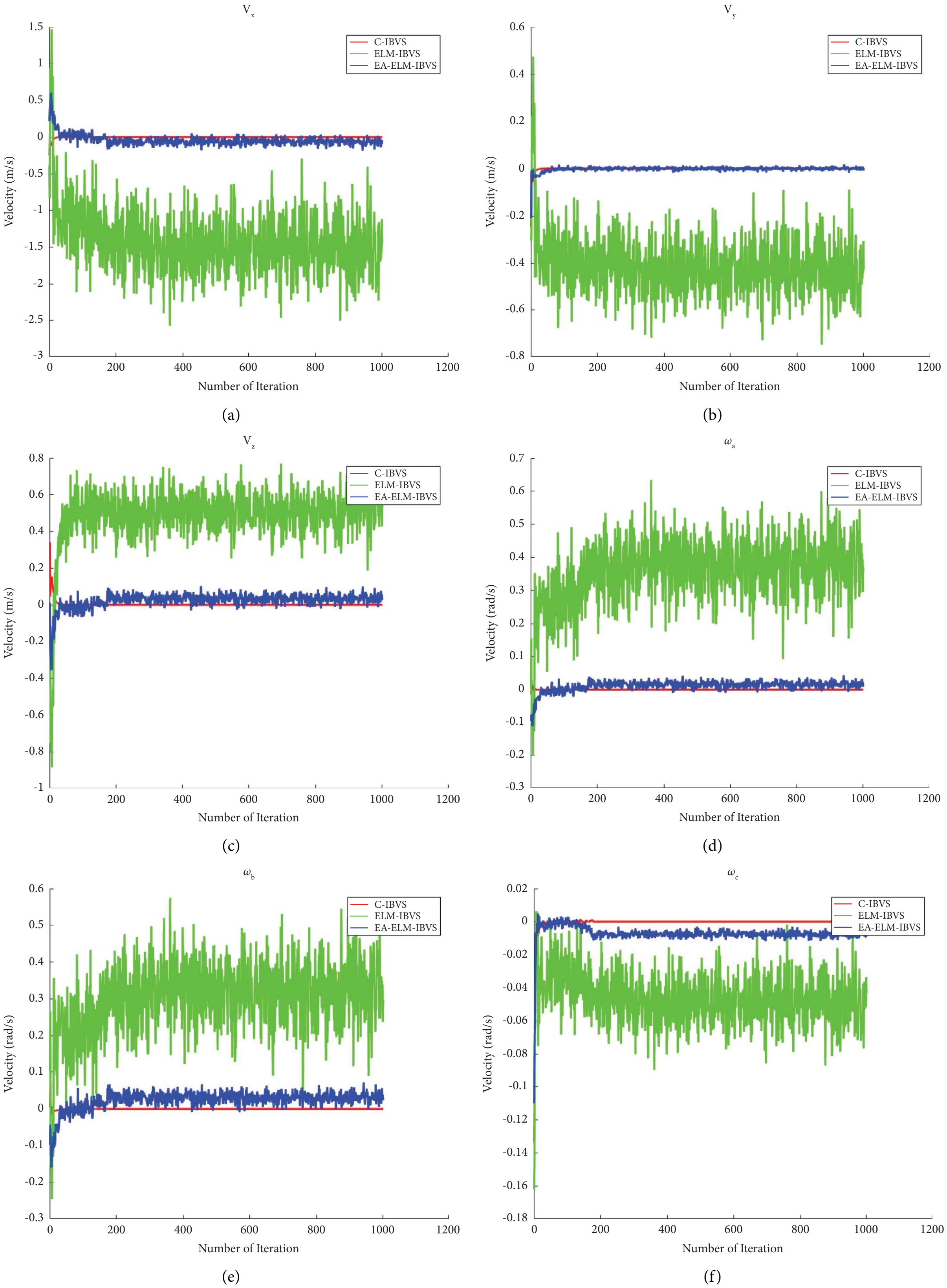
Ultimately, within the domain of error analysis, there exist two quintessential calculated indicators which hold great import, namely the sum of error squares (SSE) and the mean square error (MSE). These indicators, SSE and MSE, are commonly described as

$$\begin{aligned} \mathrm{SSE} &= \sum_{i=1}^{N}\left(\widehat{Y}_i^{\mathrm{pre}} - Y_i^{\mathrm{des}}\right)^2, \\ \mathrm{MSE} &= \frac{1}{N}\sum_{i=1}^{N}\left(\widehat{Y}_i^{\mathrm{pre}} - Y_i^{\mathrm{des}}\right)^2, \end{aligned} \tag{27}$$

where $\widehat{Y}_i^{pre}$ denotes the predicted value, and Y_i^{des} refers to the desired value. N, on the other hand, denotes the total number of samples under consideration. It is pertinent to note that $\widehat{Y}_i^{pre}$ and Y_i^{des} are four-dimensional vectors in the current experimental setup. Therefore, in order to measure the extent of improvement brought about by the experimental intervention, we propose the utilization of two indicators, namely the average of the sum of error squares (AS-SSE) and the average of the sum of mean square errors (AS-MSE), which are defined as follows:

$$\begin{aligned} \mathrm{AS-SSE} &= \frac{1}{6}\sum_{j=1}^{6}\sum_{i=1}^{N}\left(\widehat{Y}_{\mathrm{ij}}^{\mathrm{pre}} - Y_{\mathrm{ij}}^{\mathrm{des}}\right)^2, \\ \mathrm{AS-SSE} &= \frac{1}{6N}\sum_{j=1}^{6}\sum_{i=1}^{N}\left(\widehat{Y}_{\mathrm{ij}}^{\mathrm{pre}} - Y_{\mathrm{ij}}^{\mathrm{des}}\right)^2. \end{aligned} \tag{28}$$

Tables 1 and 2 present the pertinent data indicators for the aforementioned four experiments. Upon conducting an in-depth analysis of the simulation data, it was observed that the EA-ELM-IBVS method outperformed the other methods with respect to AS-SSE and AS-MSE. These findings corroborate the notion that the proposed methodology exhibits



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7: Result for case 1 on SCARA robot for C-IBVS, ELM-IBVS, and the proposed EA-ELM-IBVS.

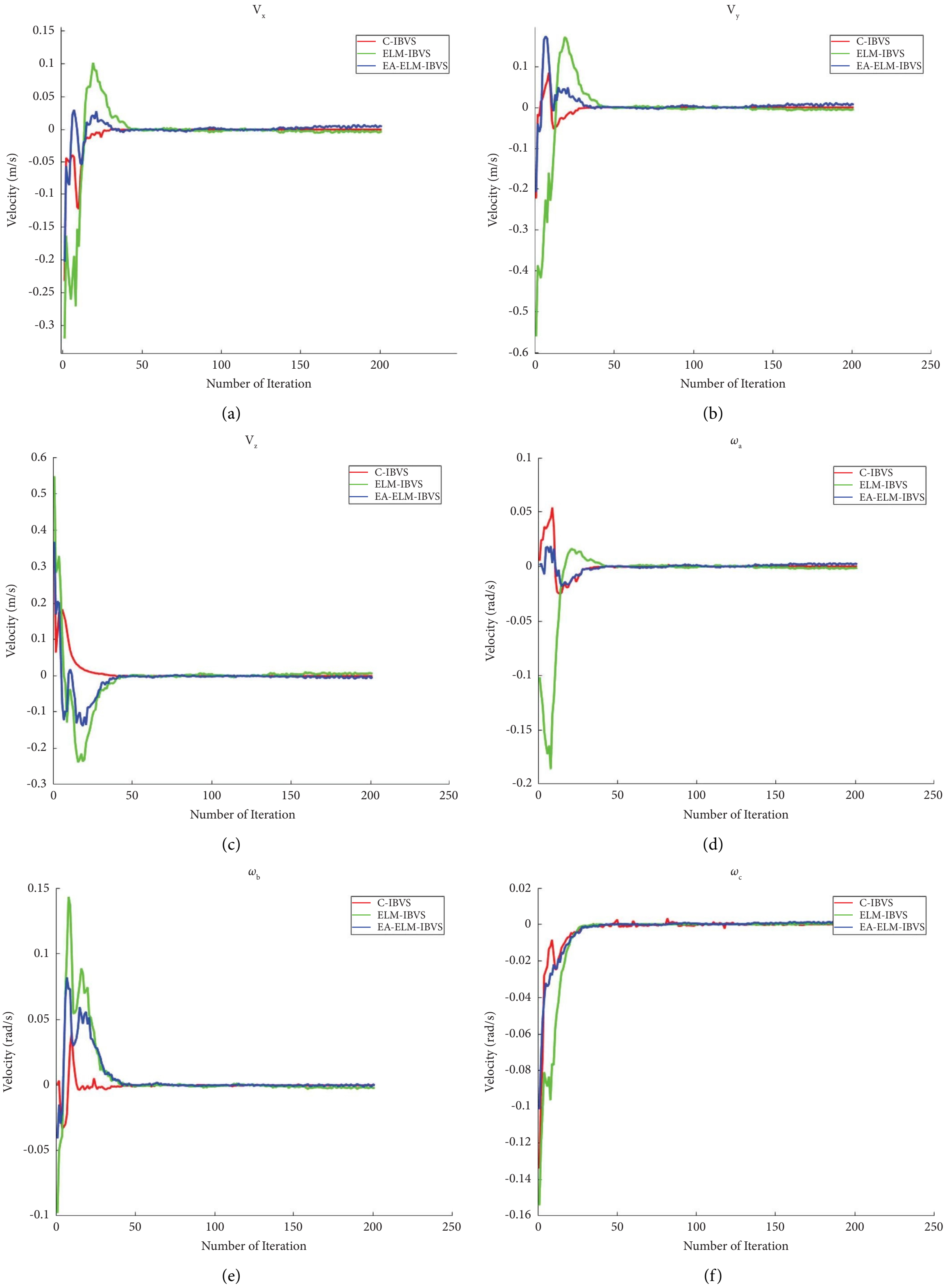


Figure 8: Result for case 2 on SCARA robot for C-IBVS, ELM-IBVS, and the proposed EA-ELM-IBVS.

Table 1: Quantitative indicators of ELM-IBVS.

Situation	AS-SSE	AS-MSE
Case 1 (simulation)	0.0614	$4.2055e - 04$
Case 2 (simulation)	0.0551	$6.1274e - 04$
Case 1 (real)	69.1485	0.3440
Case 2 (real)	0.5542	0.0028

Table 2: Quantitative indicators of EA-ELM-IBVS.

Situation	AS-SSE	AS-MSE
Case 1 (simulation)	0.0017	$1.1314e - 05$
Case 2 (simulation)	0.0011	$1.2118e - 05$
Case 1 (real)	1.1045	0.0055
Case 2 (real)	0.1296	$64462e - 04$

superior generalization ability, particularly when there is a significant divergence between the training and verification data. Moreover, an examination of the actual robot data also revealed that the EA-ELM-IBVS approach demonstrated superior performance, thereby providing evidence for its heightened robustness in the face of environmental factors such as image noise.

4. Conclusion

Given that the efficacy of learning-based methods for approximating the correlation between image feature errors and the spatial velocity of the end effector is primarily contingent on the volume of data contained within the training set, it can be a time-consuming and challenging task to amass a sufficient amount of data for the robot to operate effectively across a broad range of scenarios. To address this issue, the present study proposes an EA-ELM-IBVS approach that has been shown to exhibit superior generalization ability in comparison to the ELM-IBVS method. This signifies that the suggested method can perform better with a comparatively smaller training dataset. Meanwhile, the built digital twin operation system can better realize the organic interaction between virtual models and physical entities, which facilitates the efficiency and security of algorithm deployment and performance verification. Nevertheless, it is crucial to acknowledge that there are notable hurdles that persist in situations where the training and validation data showcase substantial dissimilarities concerning image noise. In light of this circumstance, forthcoming investigations will concentrate on undertaking algorithmic and methodological interventions and improvements that render the learning-based approach more resilient and dependable in the realm of image Jacobi estimation within the visual servo system. Furthermore, efforts will be made to incorporate additional constraints when implementing the proposed approach on tangible devices.

Data Availability

The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Minghao Cheng and Hao Tang equally contributed to the study.

Acknowledgments

This work was supported by the Princess Nourah bint Abdulrahman University Researchers Supporting Project number PNURSP2023TR140, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] S. He, Y. Xu, D. Li, and Y. Xi, "Eye-in-Hand visual servoing control of robot manipulators based on an input mapping method," *IEEE Transactions on Control Systems Technology*, vol. 31, 2022.

[2] S. Tsuchida, H. Lu, T. Kamiya, and S. Serikawa, "Characteristics based visual servo for 6DOF robot arm control," *Cognitive Robotics*, vol. 1, pp. 76–82, 2021.

[3] Y. Qiu, B. Li, W. Shi, and X. Zhang, "Visual servo tracking of wheeled mobile robots with unknown extrinsic parameters," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8600–8609, 2019.

[4] Y. Tian, G. Zhang, K. Morimoto, and S. Ma, "Automated rust removal: rust detection and visual servo control," *Automation in Construction*, vol. 134, Article ID 104043, 2022.

[5] Z. Zhang, C. Wang, Q. Zhang, Y. Li, X. Feng, and Y. Wang, "Research on autonomous grasping control of underwater manipulator based on visual servo," in *Proceedings of the 2019 Chinese Automation Congress (CAC)*, pp. 2904–2910, Hangzhou, China, November 2019.

[6] Y. Zhou, Y. Zhang, J. Gao, and X. An, "Visual servo control of underwater vehicles based on image moments," in *Proceedings of the 2021 6th IEEE International Conference on Advanced Robotics and Mechatronics (ICARM)*, pp. 899–904, Chongqing, China, July 2021.

[7] J. Lin, Y. Wang, Z. Miao, H. Zhong, and R. Fierro, "Low-complexity control for vision-based landing of quadrotor UAV on unknown moving platform," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 8, pp. 5348–5358, 2021.

[8] D. Guo, H. Wang, and K. K. Leang, "Nonlinear vision-based observer for visual servo control of an aerial robot in global positioning system denied environments," *Journal of Mechanisms and Robotics*, vol. 10, no. 6, 2018.

[9] T. Cheng, W. Li, C. S. H. Ng, P. W. Y. Chiu, and Z. Li, "Visual servo control of a novel magnetic actuated endoscope for uniportal video-assisted thoracic surgery," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3098–3105, 2019.

[10] X. Ma, C. Song, P. W. Chiu, and Z. Li, "Visual servo of a 6-DOF robotic stereo flexible endoscope based on da Vinci Research Kit (dVRK) system," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 820–827, 2020.

[11] W. J. Wilson, C. C. Williams Hulls, and G. S. Bell, "Relative end-effector control using Cartesian position based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, 1996.

[12] B. Thuilot, P. Martinet, L. Cordesses, J. Gallice, and B. Pascal, "Position based visual servoing: keeping the object in the field of vision," in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, p. 6, Washington, DC, USA, May 2002.

[13] J. T. Feddema and O. R. Mitchell, "Vision-guided servoing with feature-based trajectory generation (for robots)," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 5, pp. 691–700, Oct 1989.

[14] L. Weiss, A. Sanderson, and C. Neuman, "Dynamic sensor-based control of robots with visual feedback," *IEEE Journal of Robotics and Automation*, vol. 5, p. 14, 1987.

[15] N. R. Gans and S. A. Hutchinson, "Stable visual servoing through hybrid switched-system control," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 530–540, 2007.

[16] P. I. Corke and S. A. Hutchinson, "A new hybrid image-based visual servo control scheme," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, vol. 3, pp. 2521–2526, Sydney, NSW, Australia, December 2000.

[17] F. Chaumette, P. Rives, and B. Espiau, "Classification and realization of the different VISION-based tasks," *World Scientific Series in Robotics and Intelligent Systems*, vol. 7, pp. 199–228, 1993.

[18] P. I. Corke and O. Khatib, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Vol. 73, Springer, Berlin/Heidelberg, 2011.

[19] J. A. Piepmeier, G. V. McMurray, and H. Lipkin, "A dynamic jacobian estimation method for uncalibrated visual servoing," in *Proceedings of the 1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No. 99TH8399)*, pp. 944–949, Atlanta, GA, USA, September 1999.

[20] A. Assa and F. Janabi-Sharifi, "Virtual visual servoing for multicamera pose estimation," *IEEE*, vol. 20, no. 2, pp. 789–798, 2014.

[21] G. Wells, C. Venaille, and C. Torras, "Vision-based robot positioning using neural networks," *Image and Vision Computing*, vol. 14, no. 10, pp. 715–732, Dec 1996.

[22] F. Wang, Z. Liu, C. L. P. Chen, and Y. Zhang, "Adaptive neural network-based visual servoing control for manipulator with unknown output nonlinearities," *Information Sciences*, vol. 451–452, pp. 16–33, 2018.

[23] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2011.

[24] G.-B. Huang, "What are extreme learning machines? Filling the gap between Frank Rosenblatt's dream and John von Neumann's puzzle," *Cognitive Computation*, vol. 7, no. 3, pp. 263–278, 2015.

[25] T. Yüksel, "Intelligent visual servoing with extreme learning machine and fuzzy logic," *Expert Systems with Applications*, vol. 72, pp. 344–356, Apr 2017.

[26] M. Kang, H. Chen, and J. Dong, "Adaptive visual servoing with an uncalibrated camera using extreme learning machine and Q-leaning," *Neurocomputing*, vol. 402, pp. 384–394, Aug 2020.

[27] Q.-Y. Zhu, A. K. Qin, P. N. Suganthan, and G.-B. Huang, "Evolutionary extreme learning machine," *Pattern Recognition*, vol. 38, no. 10, pp. 1759–1763, Oct 2005.

[28] S. ten Hagen and B. Kröse, "Neural Q-learning," *Neural Computing & Applications*, vol. 12, pp. 81–88, 2003.

[29] G. B. Huang, Q. Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, vol. 2, pp. 985–990, Budapest, Hungary, July 2004.

[30] E. Olson, "AprilTag: a robust and flexible visual fiducial system," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*, pp. 3400–3407, Shanghai, China, May 2011.

[31] J. Wang and E. Olson, "AprilTag 2: efficient and robust fiducial detection," in *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4193–4198, Daejeon, South Korea, October 2016.

[32] R. Storn, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *DIFFERENTIAL EVOLUTION*, vol. 11, p. 19, 1997.