drones

MDPI

Article

Visual Servoing for Aerial Vegetation Sampling Systems

Zahra Samadikhoshkho and Michael G. Lipsett *

Mechanical Engineering Department, North Campus, University of Alberta, Edmonton, AB T6G 2G8, Canada; samadikh@ualberta.ca
* Correspondence: mlipsett@ualberta.ca

Abstract: This research describes a vision-based control strategy that employs deep learning for an aerial manipulation system developed for vegetation sampling in remote, dangerous environments. Vegetation sampling in such places presents considerable technical challenges such as equipment failures and exposure to hazardous elements. Controlling aerial manipulation in unstructured areas such as forests remains a significant challenge because of uncertainty, complex dynamics, and the possibility of collisions. To overcome these issues, we offer a new image-based visual servoing (IBVS) method that uses knowledge distillation to provide robust, accurate, and adaptive control of the aerial vegetation sampler. A convolutional neural network (CNN) from a previous study is used to detect the grasp point, giving critical feedback for the visual servoing process. The suggested method improves the precision of visual servoing for sampling by using a learning-based approach to grip point selection and camera calibration error handling. Simulation results indicate the system can track and sample tree branches with minimum error, demonstrating that it has the potential to improve the safety and efficiency of aerial vegetation sampling.

Keywords: aerial robotics; visual servoing; vegetation sampler; deep learning; knowledge distillation

1. Introduction

Advancements in aerial manipulation systems, a subset of unpiloted aerial vehicle (UAV)-based technology, have significantly enhanced the physical interaction with the environment, enabling contact tasks such as inspection, sampling, planting, and harvesting in agriculture and forestry. UAV-based technology presents a practical solution for exploring forests and sampling tree branches by overcoming limitations such as safety risks, height restrictions, low precision in collection, and the inconvenience of transport for traditional methods [1–3].

An aerial manipulation system combines the dexterity of a robotic arm with the maneuverability of an aerial base platform, making it well-suited for sampling tree branches. Studies in [4–6] investigated aerial manipulation systems for canopy sampling, from early research to interaction modeling. For example, the DeLeaves project [1,7,8] used a UAV equipped with a suspended sampling tool to collect forest canopy samples, while the methodology proposed in [2] employed a compact and portable mini-drone to collect tree samples.

Recent advances in remote sensing technology, especially LiDAR [9], have greatly improved vegetation monitoring capabilities. LiDAR data have been successfully used to estimate stand volume and above-ground biomass, resulting in rich three-dimensional insights into forest systems. Furthermore, satellite imagery, such as that from Planet [10], has been used to evaluate physiological factors in plants. However, these methods rely solely on photographs and do not include real sampling, which is required to validate assessments conducted through remote sensing, and thus provide a more detailed understanding of plant health. Sampling is critical for gathering particular data on species composition, nutrient content, and physiological responses, which can ultimately improve management approaches. Machine learning algorithms such as Random Forest have proven useful in

check for updates

Citation: Samadikhoshkho, Z.; Lipsett, M.G. Visual Servoing for Aerial Vegetation Sampling Systems. *Drones* **2024**, *8*, 605. https://doi.org/10.3390/drones8110605

Academic Editor: Yangquan Chen

Received: 30 August 2024
Revised: 17 October 2024
Accepted: 21 October 2024
Published: 22 October 2024

evaluating satellite reflectance data, allowing for better predictions of plant health and productivity. These advancements show the possibility of merging remote sensing with aerial manipulation systems in vegetation sampling, resulting in more precise management and resource usage in forestry and agriculture.

Vegetation sampling using aerial manipulation systems presents several challenges, including relative position sensing, control of the aerial manipulation system, collision avoidance, and physical interactions with the environment [4–6,8,11]. One of the primary challenges in deploying aerial robots for tree branch sampling is the control aspect. Selecting an effective control method requires careful consideration to maintain a stable interaction between the aerial robot and its surrounding environment. The selected approach has to effectively handle uncertainties, high levels of non-linearity, and complex couplings between the aerial platform and the robotic manipulator payload that interacts with elements in the environment.

The quality of data acquired by UAVs is significantly impacted by a number of parameters, such as ambient variables, air conditions, and sensor specifications. These uncertainties have the potential to cause measurement errors and compromise the validity of data interpretation. For example, recent research [12] has shown that variables like light variability, camera noise, and topography differences can have a major effect on the reflectance values that UAV sensors are able to measure. Reducing these uncertainties is crucial to raising the overall dependability of remote sensing applications for vegetation monitoring as well as the accuracy of measurements conducted by drones.

In our earlier work [13], we proposed Nonlinear Disturbance Observer-Based Control (NDOBC) and Adaptive Sliding Mode Control (ASMC) as robust control strategies for aerial manipulation systems designed for vegetation sampling. These methods employed a decoupled architecture, treating the uncrewed aerial vehicle and the manipulator arm of the sampler payload as separate units, estimating the coupling between the robotic arm and the aerial platform. The coupling was treated as an unmodeled dynamic with unknown disturbances affecting the system's behavior. While this decoupled structure may reduce control accuracy, it is robust against some level of external disturbance. Its performance in highly uncertain environments such as forests requires careful tuning of the controller parameters. To address this issue, the current research proposes using a human-like perception system and visual servoing to enhance control accuracy and adaptability in complex environments.

Visual servoing has a wide range of applications for aerial robots, including object tracking [14–18], object manipulation [19,20], and inspection tasks [21]. In the context of agriculture, computer vision and visual servoing have been widely used to improve the precision and effectiveness of different processes involving aerial manipulation systems or in general aerial robots, such as automated fruit inspection, harvesting, plant tracking, pruning, pollinating, accurate pesticide application, and autonomous navigation in crop fields [22–25]. As an example, a modular software framework was developed in [26] to accommodate an extensive range of agricultural applications and hardware. This framework implements motion control and visual sensing to operate agricultural robotics in dense vegetation.

Tree branch sampling, in particular, requires active branch tracking using on-board sensing, such as vision, to accurately track and grasp the desired branch with the end effector, allowing the manipulator to move accordingly [6]. A vision sensor was employed in [8] to detect and locate canopy tree branches, and [27] addressed the challenge of branch localization and tracking of the canopy in cluttered environments using stereo vision, enhancing the capability of aerial manipulation systems in complex settings.

Visual servoing refers to a closed-loop robot motion control technique that uses visual feedback to minimize the error between the vision feedback observed at the current position of the robot/camera and the desired position [28]. Visual servoing can be classified into three main categories: position-based visual servoing (PBVS), image-based visual servoing (IBVS), and hybrid visual servoing methods.

PBVS defines the control input in 3D Cartesian space using position estimation techniques and a priori knowledge of the geometric model of the observed object [29]. Due to the calculation of error in 3D space, PBVS results in a simpler control law and minimal length trajectory [30], but this also makes the method highly sensitive to calibration noise from camera parameter biases and estimation errors from inaccuracies in the target 3D kinematic model [29,31].

IBVS defines the error in the image plane to enhance the tolerance to camera calibration errors. Additionally, IBVS does not necessitate a geometric representation of the objective [32]. This approach may encounter challenges such as local minima and singularities in the image Jacobian matrix when the number of features is small [32,33]. IBVS is frequently employed in mobile robots [29] and UAVs [34] due to its robustness and well-defined framework [31].

Hybrid visual servoing is a control technique that integrates or switches between PBVS and IBVS to improve control performance [35]. Different environments may require different switching rules, leading to poor generality [32].

Compared with the other two methods, IBVS has been increasingly researched in recent years due to its high steady-state control accuracy [33]. Due to the benefits associated with IBVS, this approach is chosen for the current study. The use of IBVS has notable advantages for agricultural and forestry sampling applications, specifically its ability to handle camera calibration errors well and its lack of requirements for a geometric representation of the target. Aerial vegetation sampling requires maintaining high accuracy and adapting under various environmental conditions.The well-defined framework of IBVS allows for effective control even in the presence of uncertainties and varying conditions. For these reasons, the present work focuses on leveraging IBVS to enhance the performance and reliability of aerial vegetation sampling systems.

Learning-based visual servoing employs data-driven techniques to understand different components of the visual servoing process in unstructured and unknown environments [30]. This approach enables robots to make decisions based on visual feedback, facilitating robust operation in dynamic settings. Key components include visual feature representation [36], adjustment of servoing gains [31,37], and the perception and control process [38].

The precision and efficiency of visual servoing systems are each significantly influenced by the selection of visual features extracted from the image as visual feedback [39]. Traditional visual servoing methods use handcrafted visual features [36], but recent advances in computer vision have shown that data-driven methods are capable of effectively learning and understanding complex patterns from images, particularly when provided with a large number of examples [40]. Data-driven methods have encouraged the use of learning-based feature representations [36]. A common issue during visual servoing is the loss of features from the camera's field of view. In this scenario, it is essential to implement a control approach to resolve the feature loss issue by either handling the feature loss [41,42]—which involves maintaining control regardless of lost features—or by preventing feature loss through long-horizon trajectory planning [43,44], optimal planing and control (like Model Predictive Control (MPC)) [45–47], applying visibility constraints [31,37], or the use of autonomously learned strategies to complete the servo [37]. To tackle the challenges of target loss and low control efficiency, refs. [31,37] introduce a visual servoing control system for UAVs and mobile robots, respectively. These systems use deep reinforcement learning (DRL) approaches, enabling real-time dynamic adjustment of the servo gain.

Unlike traditional visual servoing approaches, some data-driven frameworks (such as the one proposed by [30]) do not require feature extraction and tracking. This method, known as direct visual servoing, bypasses these steps by utilizing pixel intensities directly [48]. In [30], a convolutional neural network (CNN) was trained to perform end-to-end visual servoing without prior knowledge of the geometry of the underlying scene and the intrinsic parameters of the camera. Additionally, an image-based visual servoing technique using Extreme Learning Machine (ELM) and Q-learning adjusts the servo gain in [29,33] to improve convergence speed and stability. Furthermore, a multi-perspective visual servoing framework introduced in [36] uses reinforcement learning to learn an optimal control policy from different perspectives, demonstrating its effectiveness in complex environments.

Besides feature extraction, selecting a robust control approach that can handle both visual error and estimation error is another significant challenge. Numerous learning approaches have been investigated in the field of visual servoing, with reinforcement learning (RL) [29,31,36,37,49,50] and end-to-end visual servoing [38] being particularly prominent. RL-based approaches for visual servoing face several significant challenges. One major drawback is that the parameters learned by RL are often highly specific to the training environment and task, making it difficult to generalize RL policies to new or unseen environments [30]. This lack of generalization limits the applicability of RL in diverse and dynamic settings. RL-based visual servoing can lead to unsmooth actions, especially when the RL agent relies on the current state, resulting in suboptimal performance [36]. RL also requires large amounts of training data to learn effective policies, which can be time-consuming and computationally expensive to gather [38]. Designing appropriate reward functions is another challenge, as poorly designed rewards can hinder the learning process or lead to unintended behaviors [38]. RL algorithms are also prone to becoming trapped in local minima during training, which can prevent the discovery of optimal policies [38]. Although RL has potential for visual servoing, it faces challenges relating to the specificity of the environment, the smoothness of actions, the data needed, the design of reward functions, and the stability of training. Tackling these problems is essential for enhancing the efficiency and practicality of learning-based visual servoing systems. Despite these challenges, integrating RL with traditional control methods presents a promising direction for future work. Hybrid approaches that combine RL with classical visual servoing techniques could leverage the strengths of both methodologies, providing a more flexible solution for complex environments. Such an approach would allow for the robustness of conventional methods while also incorporating the adaptability of RL, thus enhancing the overall performance of aerial manipulation systems. Table 1 provides a concise comparison of the above visual servoing methods.

Table 1. Comparison of different visual servoing methods.

Method	Key Features	Strengths	Limitations
Learning-Based Visual Servoing	Data-driven techniques for decision-making in unstructured environments.	Robust operation in dynamic settings; learns complex patterns from data.	Requires large training datasets; may face issues with generalization across environments.
End-to-End Visual Servoing	Directly maps input images to control commands without feature extraction.	Bypasses feature extraction; potentially faster processing.	May struggle with complex environments; relies on high-quality training data.
Hybrid Approaches	Combines IBVS and PBVS.	Leverages strengths of both methodologies; adaptable to varying conditions.	Complexity in integration; poor generality; may require extensive tuning.
Position-Based Visual Servoing (PBVS)	Uses 3D pose information.	Effective for tasks needing precise positioning.	Sensitive to camera calibration.
Image-Based Visual Servoing (IBVS)	Operates on image features to minimize error in the image plane.	Quick adjustments based on visual feedback; can work with varying object appearances.	Local minima and singularities; can require more features for reliable control.
Reinforcement Learning Approaches	Utilizes RL to dynamically map visual inputs.	Adaptable to changing environments; learns from interaction with the environment.	Large training data requirement; can lead to unsmooth actions and local minima issues.

Some end-to-end methods train both perception and control systems simultaneously [51], while others separate these functions into distinct modules for perception and robot control [52]. While end-to-end training can be effective, it often requires a substantial number of samples and may encounter issues if there is a discrepancy between the training and testing environments [53]. To address these challenges and manage uncertainties effectively in a clustered forest environment while minimizing sampling costs, the latter approach is adopted here, where perception and control are handled through separate modules. This strategy provides greater flexibility and robustness, enabling more efficient adaptation to varied conditions and reducing the overall expense associated with reinforcement learning-based techniques.

2. Detection and Feature Extraction

In the development of robotic systems for the forestry sector, computer vision is crucial for understanding scenes and gathering essential information. This technology not only facilitates tasks such as mapping and navigation, but also plays a significant role in collecting physical samples from the tree canopy [54]. Researchers rely on these samples for conducting breeding experiments, carrying out genetic analysis, and monitoring disease occurrences [27]. To conduct autonomous physical sampling using unmanned aerial vehicles, precise and reliable real-time computer vision algorithms capable of running onboard are required to ensure efficient data collection.

The process of autonomous tree branch sampling includes the identification of appropriate trees, the detection of the desired branch, and the accurate extraction of the cutting or grasping point. These extracted points are then employed as visual features for the visual servoing, providing essential vision feedback for the system. To achieve this, the developed model needs to be trained in a complex, cluttered, and dynamic environment where there is no plain image background. A geometric feature could be extracted to indicate the suitable point for gripping and cutting the branch. In our scenario, we identify the lines of the tree branches and determine the grabbing point (Figure 1). The sway motion of the branch is modeled in [27], where a tracking algorithm is proposed. However, in this work, the sway motion is ignored for simplification purposes in the assumptions. It is acknowledged that environmental factors such as wind or movement in the branches can affect system performance in real-world scenarios. To enhance the robustness and applicability of the system in practical settings, adaptive control strategies or predictive models that account for dynamic environments could be explored in future work. This will help address the complexities introduced by branch motion.

The tree branch line detection algorithm using CNN proposed in [55] is applied here to specify the direction and position of the tree branch, and to determine the desired grasp point required for the visual servo loop. The suggested method employs a CNN to predict straight lines that represent tree branch extensions, followed by a Hough transform to estimate the direction and position of the line. The grip point is then determined as the pixel point having the greatest likelihood of being on the line. To achieve robust evaluation, we used tenfold cross-validation, with 90% of the data for training and 10% for testing. Furthermore, in [55], the approach was tested against multiple corruptions (Gaussian and shot noise) with varying severity levels, indicating its robustness in difficult situations. The experimental study in that paper yielded an F1-score of 96.78%, demonstrating the method's accuracy and precision. The PReLU activation function was employed in all convolution layers except the final layer, which used ReLU to improve feature representation.

The dataset was created by [55] for images containing relevant phrases and keywords linked to tree branches. After collecting the original collection of images, a data-cleaning procedure was performed to remove duplicate or irrelevant images that did not match the topic of the study. To standardize the input size for the CNN model, all remaining photos were shrunk to 128 × 128 pixels. The final collection included 1868 images, each with at least one notable tree branch. The goal was to identify the main tree branch in each image, while smaller branches were ignored. All images were manually annotated with a straight line to indicate the direction of the primary tree branch. For complex or curved branches, the longest visible segment of the branch was identified. This method ensured labeling uniformity and gave explicit targets for the CNN to train and evaluate.

In terms of performance, the CNN model achieved outstanding results [55]. The F1-score, which considers both precision and recall, was 0.9678. Precision, defined as the fraction of correctly identified main branches among all positive predictions, was 0.9698. Recall, which assesses how effectively the model identified all important main branches, was 0.9674.

It is important to address the computational cost of deploying deep learning models in real time on UAVs, which often have limited processing power. To optimize performance, the NVIDIA Jetson Orin NX module can be utilized on the aerial vehicle. This advanced platform provides impressive AI performance of up to 100 Tera Operations Per Second (TOPS), making it capable of handling complex neural network tasks efficiently in real time.

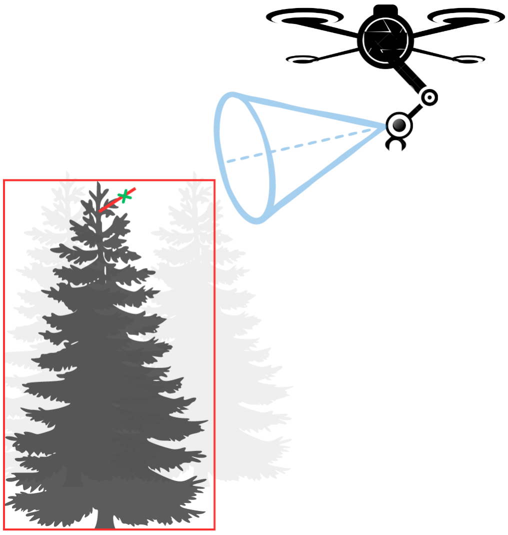

Figure 1. Schematic illustration of aerial tree branch detection and feature extraction.

3. System Modeling

In this study, a UAV is equipped with a two-link rigid manipulator, as shown in Figure 2. In order to provide an accurate representation of this system, it is necessary to determine both an inertial reference frame, $\mathcal{F}^I : \{X_I, Y_I, Z_I\}$, and a body reference frame, $\mathcal{F}^B : \{X_B, Y_B, Z_B\}$, that is fixed to the center of mass of the UAV.

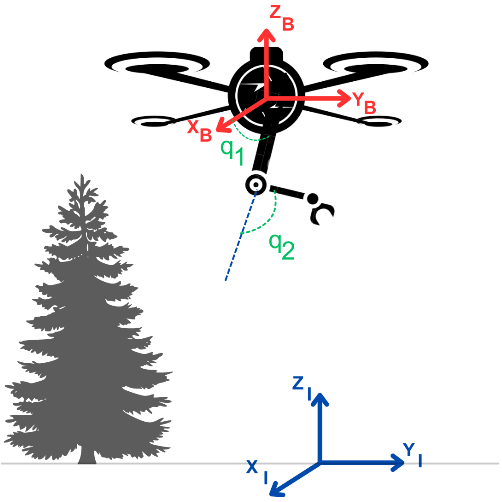

Figure 2. Two-link aerial sampling system of conifer tree branches—inertial reference frame, $\mathcal{F}^I : \{X_I, Y_I, Z_I\}$, and a body reference frame, $\mathcal{F}^B : \{X_B, Y_B, Z_B\}$.

The position and orientation of the UAV in the inertial frame, $\mathcal{F}_I$, are denoted by P_{UAV} and Φ_{UAV}, respectively. Here, $\Phi_{UAV} = [\varphi, \theta, \psi]^T$ represents the Euler angle vector, which includes the roll, pitch, and yaw angles.

Let $\bar{R}$ be the rotation matrix between the inertial and body frames, and $\bar{T}$ be the transformation matrix, deriving the relation between Euler angles and angular velocity. The linear and rotational velocities of the UAV in the inertial frame, $\dot{P}_{UAV}$ and $\dot{\Omega}_{UAV}$, are expressed in terms of the linear velocity of the UAV in the body frame, $\dot{P}^B_{UAV}$, and the time derivatives of the Euler angles, $\dot{\Phi}_{UAV}$. Superscript B denotes the body frame and variables, while the bar indicates matrices.

$$\dot{P}_{UAV} = \bar{R}\dot{P}^B_{UAV}, \tag{1}$$

$$\dot{\Omega}_{UAV} = \bar{T}\dot{\Phi}_{UAV}, \tag{2}$$

The relationship between the rotational velocity of the UAV in the body frame, $\dot{\Omega}^B_{UAV}$, and that in the inertial frame, $\dot{\Omega}_{UAV}$, is expressed as follows:

$$\dot{\Omega}^B_{UAV} = \bar{R}^T\dot{\Phi}_{UAV} = \bar{R}^T\bar{T}\dot{\Phi}_{UAV} = \bar{Q}\dot{\Phi}_{UAV}, \tag{3}$$

where superscript T denotes the transpose of a matrix.

Arm joint angles are represented by the vector $q = [q_1, q_2]^T$. The position of the ith joint in the inertial frame, P_i, can be determined by knowing the position of the UAV and the position of the joint in the body reference frame, P^B_i.

$$P_i = P_{UAV} + \bar{R}P^B_i. \tag{4}$$

Next, the linear velocity, $\dot{P}^B_i$, and angular velocity, ω^B_i, of the ith joint in the body frame are calculated as follows:

$$\dot{P}^B_i = J_{p1}\dot{q}_1 + ... + J_{pi}\dot{q}_i = \bar{J}^i_p\dot{q}, \tag{5}$$

$$\omega^B_i = J_{o1}\dot{q}_1 + ... + J_{oi}\dot{q}_i = \bar{J}^i_o\dot{q}, \tag{6}$$

where $\bar{J}^i_p$ and $\bar{J}^i_o$ are the linear and angular velocity Jacobians for the ith joint, respectively. Finally, these linear and angular velocities are expressed in the inertial reference frame as

$$\dot{P}_i = \dot{P}_{UAV} - Skew(\bar{R}P^B_i) + \bar{R}\bar{J}^i_p\dot{q}, \tag{7}$$

$$\omega_i = \Omega_{UAV} + \bar{R}\bar{J}^i_o\dot{q}. \tag{8}$$

The function $Skew()$ produces the skew-symmetric matrix of any input vector.

Defining the state vector of the system as $\eta = [P_{UAV}, \Omega_{UAV}, q]^T$, the dynamics of the system can be calculated using the Euler–Lagrange theory as follows:

$$\bar{M}(\eta)\ddot{\eta} + \bar{C}(\eta,\dot{\eta})\dot{\eta} + G(\eta) = u, \tag{9}$$

where $\bar{M}$ is a symmetric positive definite inertia matrix, $\bar{C}$ represents the centrifugal and Coriolis forces, and G is the gravitational vector, derived from reference [56]. Also, u denotes the control input.

4. Visual Servoing

4.1. Image-Based Control

By determining the desired grip point from [27], four features around the target point can be selected to design an image-based, visual servo control system. These four features ensure that the IBVS can accurately guide the end-effector to the desired point. Four features are required because they provide sufficient information for accurately guiding the end-effector by determining both the position and orientation of the target relative to the camera. This reduces ambiguity, ensures stability, and enables full six-degree-of-freedom control, including translation in three dimensions and rotation about three axes. These factors collectively enhance the performance and reliability of the IBVS system.

Assume that the desired image vector of the grip from the CNN method is s_d, and the current image vector observed by the camera is $s(t)$. The visual error $e_s(t)$ can be defined as

$$e_s(t) = s(t) - s_d, \tag{10}$$

In visual servoing, the derivative of the vector $s(t)$ is related to the relative velocity between the camera and the object v_{c_o} through the interaction matrix L_s as

$$\dot{s}(t) = L_s v_{c_o} + \frac{\partial s}{\partial t}, \tag{11}$$

where $\frac{\partial s}{\partial t}$ denotes the derivative of features due to the object motion. In the forestry application, it is assumed that the sway motion of a branch is ignored (that is, the object remains stationary), making $\frac{\partial s}{\partial t} = 0$. The time derivative of the visual error can thus be found to be

$$\dot{e_s}(t) = \dot{s}(t) = L_s v_{c_o}, \tag{12}$$

A desired first order control law, $\dot{e_s}(t) = -Ke_s(t)$, is assumed for the visual servo control to ensure that the error converges to zero, where K is a constant control gain. The desired velocity between camera and object v_{c_o} can be found as

$$v_{c_o} = L_s^+\{-K(s(t) - s_d)\}, \tag{13}$$

where

$$L_s^+ = (L_s^T L_s)^{-1} L_s^T, \tag{14}$$

In the present work, the camera is mounted on the gripper as an eye-in-hand configuration, and so v_{c_o} is obtained in the camera frame (which is the same as the gripper frame) and must be related to the gripper's desired velocity in the inertial frame $\dot{x}_d$ to calculate the control signal and guide the gripper to the grip point. $\dot{x}_d$ can be calculated as

$$\dot{x}_d = \bar{R}\bar{R}_{g-UAV} v_{c_o}, \tag{15}$$

where $\bar{R}_{g-UAV}$ denotes the relative rotation matrix between the gripper and the UAV platform.

4.2. Dynamic Control

In order to calculate the control signal in the gripper inertia frame, the system dynamics presented in (9) should be re-written in the gripper inertia frame using the gripper Jacobian matrix $\bar{J}_x$ by

$$\bar{M}_x(x)\ddot{x} + \bar{C}_x(x, \dot{x})\dot{x} + G_x(x) = u_x, \tag{16}$$

where x is the gripper pose in the inertia frame and

$$\bar{M}_x(x) = J_x(\eta)^{-T} M(\eta) J_x(\eta)^{-1}, \tag{17}$$

$$\bar{C}_x(x, \dot{x}) = J_x(\eta)^{-T}(\bar{C}(\eta, \dot{\eta}) - M(\eta) J_x(\eta)^{-1} \dot{J}_x(\eta)) J_x(\eta)^{-1}, \tag{18}$$

$$G_x(x) = J_x(\eta)^{-T} G(\eta), \tag{19}$$

$$u_x = J_x(\eta)^{-T} u. \tag{20}$$

Now, with the gripper equation and desired velocity defined, a control law is needed to ensure that the gripper reaches the target point. In our previous work, ref. [13,56], the adaptive sliding mode controller proved to be both robust and efficient for controlling aerial manipulation systems, and this approach was chosen for this study. To determine the control signal, the error e_x, sliding surface S_x, and reference state x_r are defined as

$$e_x = x - x_d, \tag{21}$$

$$\dot{x}_r = \dot{x}_d - \Lambda e_x, \tag{22}$$

$$S_x = \dot{e}_x + \Lambda e_x. \tag{23}$$

where Λ is a positive diagonal matrix. The control signal, u_x, is obtained from

$$u_x = \bar{M}_x(x)\ddot{x}_r + \bar{C}_x(x,\dot{x})\dot{x}_r + G_x(x) - K_v\dot{e}_x - K_p e_x + \hat{\Delta}, \tag{24}$$

where K_p and K_v are positive definite matrices and $\hat{\Delta}$ is the estimated uncertainty, which can be calculated as

$$\dot{\hat{\Delta}} = -K_\Delta S_x. \tag{25}$$

The stability of this control approach is thoroughly discussed in [57].

4.3. Learning Based Control

This study presents a novel end-to-end learning-based approach for vision-based control of an aerial vegetation sampling system, which is separated into distinct perception and control modules. The control method involves training neural networks to fully model the entire control process and then applying knowledge distillation (KD) to refine smaller networks based on the larger, pre-trained ones. KD enables the development of more compact networks with fewer parameters by transferring insights from larger models. This process not only reduces network size but also introduces a regularization effect that enhances the generalizability and robustness of the smaller networks.

A supervised learning framework addresses the vision-based control challenge. Supervised learning focuses on estimating the relationship between input–output pairs, which can then be used to predict outputs for new inputs. In sampler control, the system control signal is estimated and generated based on the system state variables, current and desired visual features, and errors as

$$x_{train} = [s_d^T, s^T, \eta^T, e_x^T, \dot{e}_x^T]^T. \tag{26}$$

To train the network, inputs are randomly generated within a specified range, and the controller calculates the corresponding outputs. The system's input vector is unordered, with no spatial correlation among its elements. Additionally, due to the lack of temporal trends, all samples are independent and identically distributed, as the controller bases the control signal solely on the current system state, ignoring previous states. This work employs multilayer perceptron (MLP) neural networks for the task [58].

In knowledge distillation, when a large network is trained for a classification task, the layer immediately preceding the softmax function generates a valuable embedding of the input data. Work by [59] suggests that training a smaller network to predict the embedding and classify the sample should be conducted simultaneously, and this method is applied for visual control in this study.

KD starts with training a large neural network, the teacher network, to address a regression problem by minimizing the mean squared error (MSE) between its outputs and the target outputs. As illustrated in Figure 3, a smaller network, the student or distilled network, is then trained to ensure that its output layer accurately predicts the target outputs, while one of its hidden layers learns to predict the embedding, generated by a corresponding hidden layer in the teacher network. The loss function for training the student network is defined as the sum of two MSE loss functions by

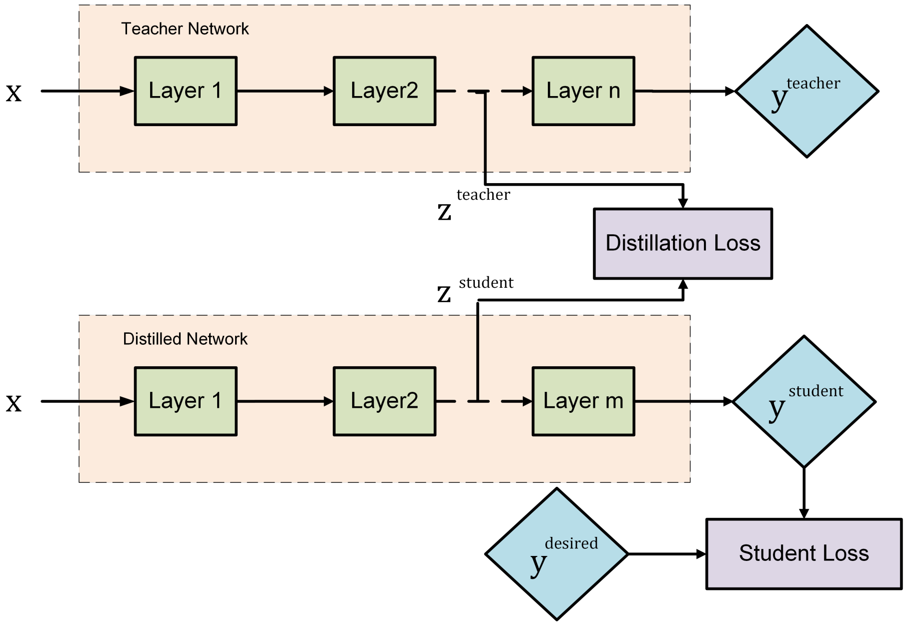

Figure 3. Suggested knowledge distillation block diagram.

$$Loss = \alpha L_{student} + (1-\alpha)L_{distillation}, \tag{27}$$

$$L_{student} = MSE(y^{student}, y^{desired}), \tag{28}$$

$$L_{distillation} = MSE(z^{student}, z^{teacher}). \tag{29}$$

Significantly, the teacher network remains unchanged while the student network is trained, with backpropagation exclusively applied to the student network. In our previous work on KD control [60], we found that training individual networks for each output and distilling these networks separately produced the most effective results, and so the same strategy is used for KD in this research as well. It is critical to take into account the performance trade-offs when employing knowledge distillation (KD) to simplify neural networks. While KD can drastically reduce model complexity, it may also result in decreased control accuracy, especially if the distilled model is unable to capture the nuances of the original model. This drop in precision can have an impact on the system's capacity to perform optimally, particularly in dynamic circumstances where precise control is required. Alternative strategies for balancing performance and computing simplicity include model pruning and architectural optimization. These strategies have demonstrated potential in improving model robustness while maintaining high accuracy, making them viable candidates for future application in our system.

Figure 4 illustrates the proposed end-to-end visual servo control system for aerial vegetation sampling. This system employs a CNN to extract the desired features, which are then provided to the image-based vision control unit to determine the required gripper motion. This motion is subsequently processed by the adaptive sliding mode (ASM) to compute the control signals for both the UAV and the robotic arms. The entire control process is replaced using a KD neural network, which directly computes the control signal by only observing the tree branches.

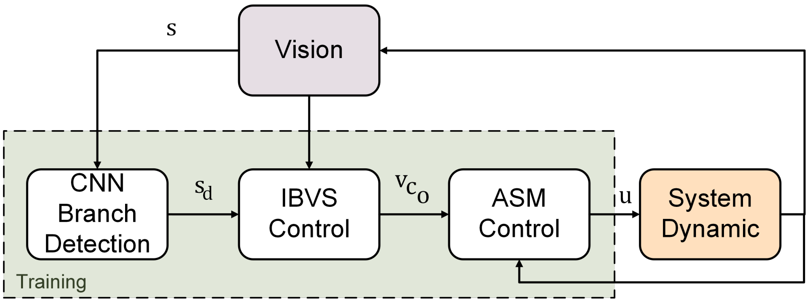

Figure 4. Proposed visual servoing block diagram.

5. Simulation Results

This section evaluates the performance of the proposed controllers for the aerial platform developed at the University of Alberta (see Figure 5). The platform, a modified DJI S1000, features a custom three-degree-of-freedom robotic arm with a cutting mechanism at the end. In the simulation results, we used only two active links, with the third link remaining passive and merely connected to the gripper. A perspective camera with a focal length of 1024 pixels and a principal point at 512 pixels, mounted on the gripper, guides the system to the desired grip point.

To create the teacher network, we concatenate control signal networks to produce an output with eight components. This teacher network is then distilled into the student network with the architecture shown in Table 2. Each teacher network is distilled individually, and these distilled networks are combined to form the final controller. The training uses the PReLU activation function and mean squared error (MSE) with five hidden layers.

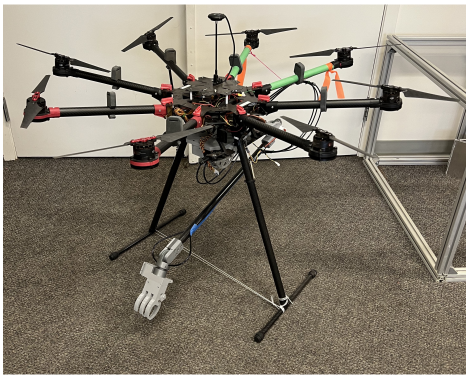

Figure 5. Aerial vegetation sampler experimental platform.

Table 2. Architecture of distilled networks.

	Number of Neurons	Activation
Hidden 1	25	PReLU
Hidden 2	25	PReLU
Hidden 3 ($z^{student}$)	50	—
Hidden 4	25	PReLU
Hidden 5	25	PReLU
Output	1	—

Two scenarios are considered to demonstrate the efficiency of the proposed vision-based control approach for gripping the sample: one from the top and the other based on the initial position relative to the sample.

5.1. Gripping from the Top

In this simulation, the desired grip point is positioned 10 m above the origin. The UAV, a fully actuated drone, starts at [5, 5, 15] meters with initial Euler angles of 10 degrees. Achieving the target point becomes easier when the starting Euler angles are zero. However, reaching the goal becomes more difficult when the initial Euler angles are not equal to zero. That is why, in this simulation scenario, non-zero Euler angles are selected as initial conditions. The first link is set at a 30-degree angle, and the second link has zero initial deflection. The initial velocity of the UAV is set to zero.

Figure 6 displays the feature trajectory, showing the initial feature positions marked with circle symbols (o) and their desired points marked with cross symbols (×). Figure 7 presents the feature trajectory error, demonstrating that the error converges to zero around 30 s. Figure 8 plots both the transnational and rotational velocities of the gripper. The UAV and its gripper 3D trajectory are depicted in Figure 9. Figure 10 shows the gripper position and its position error, confirming that the gripper successfully reached the desired grip point.

Figures 11 and 12 present the overall system states. These figures illustrate an initial oscillation in the Euler angles within the first 10 s. The UAV achieves the grip point with final roll and pitch angles of zero, while its heading is approximately -40 degrees. The angles of the links adjust to 80 and 50 degrees, respectively, to reach the grip point.

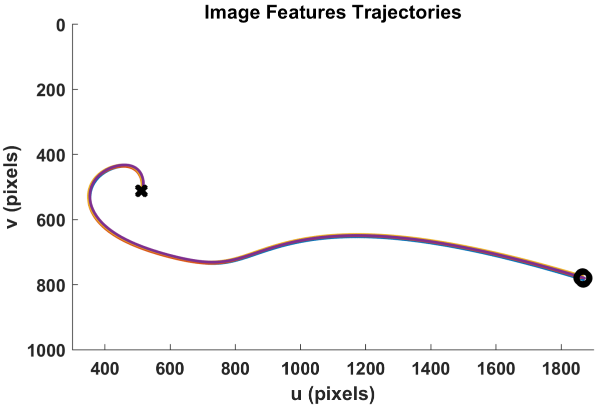

Figure 6. Feature trajectory—top approach.

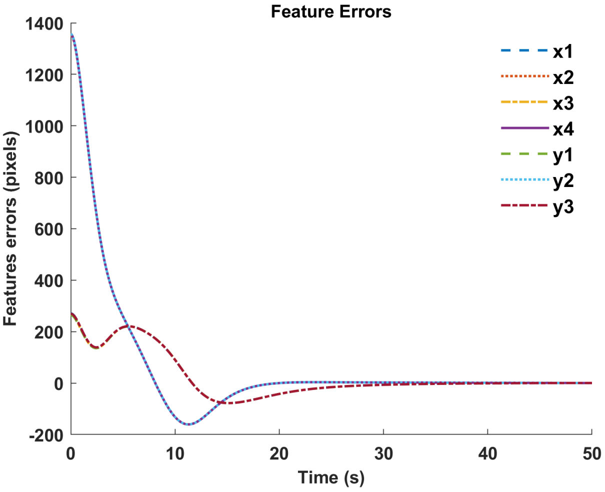

Figure 7. Feature Error—top approach.

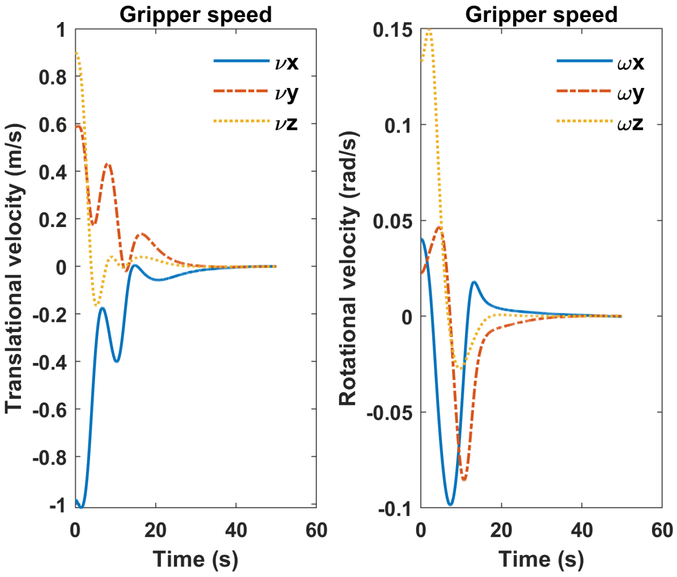

Figure 8. Gripper transnational and rotational velocities—top approach.

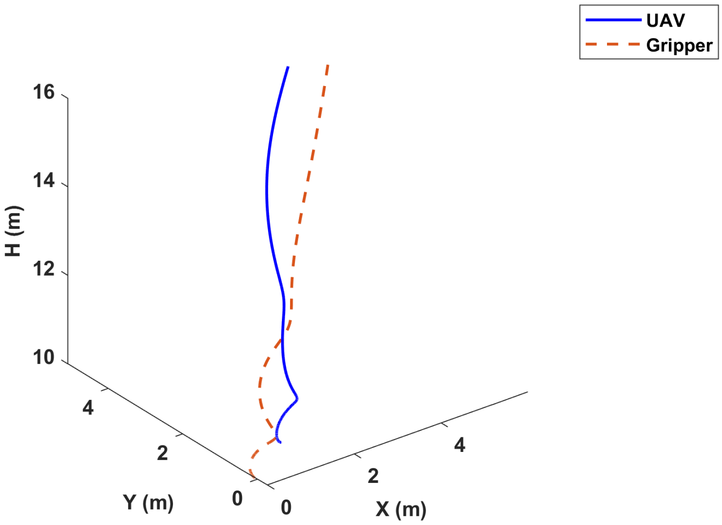

Figure 9. UAV and gripper trajectory—top approach.

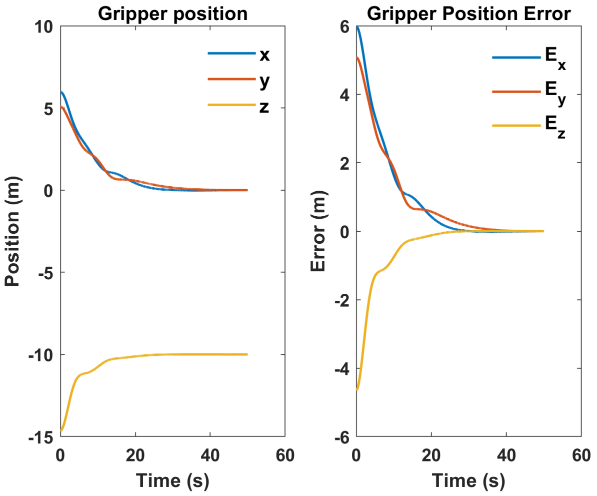

Figure 10. Gripper position and position error—top approach.

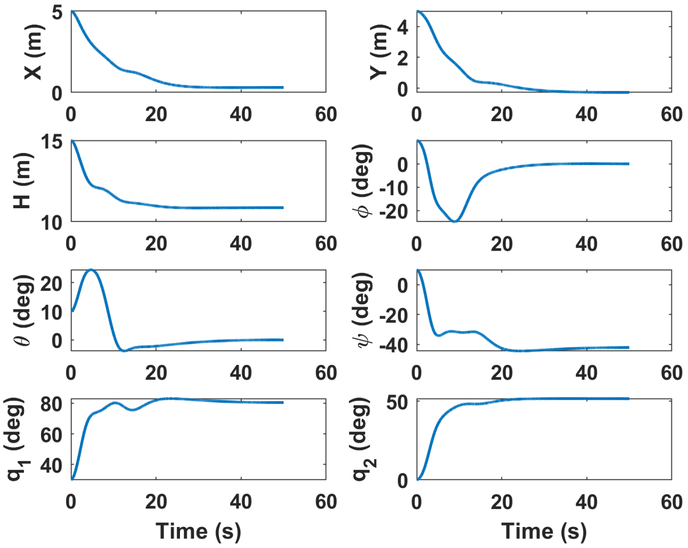

Figure 11. UAV and arm states—top approach.

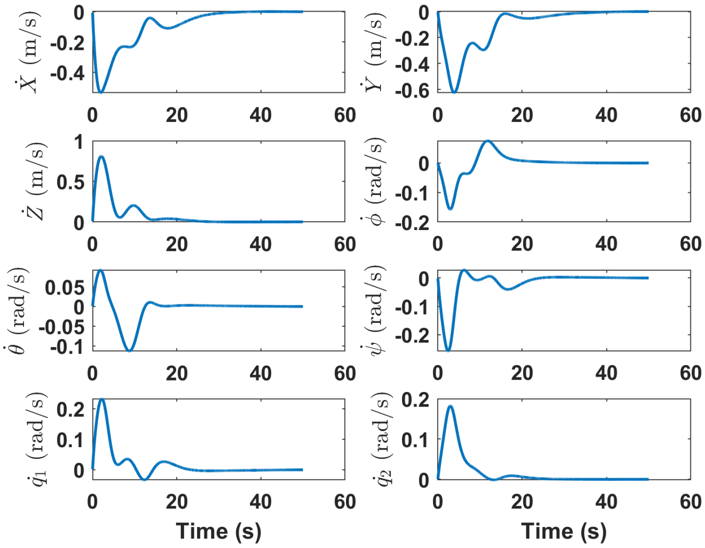

Figure 12. UAV and arms states' derivative—top approach.

5.2. Gripping from Below

In this simulation, the desired grip point remains the same as in the previous case, with similar UAV and arm states. The only difference is the drone's initial height, which is set to 7 m instead of 15 m as in the previous scenario.

Figure 13 indicates that the feature trajectory in the "Gripping from the Top" approach was smoother, while in the "Gripping from Below" approach, there is a noticeable curve that changes the feature direction in the middle of the trajectory. Figure 14 reveals that the convergence time for feature error is similar in both scenarios, although the error dynamics differ slightly.

Figure 15 shows that the magnitudes of both translational and rotational speeds are smaller in the "Gripping from Below" scenario, with the UAV approaching the target more slowly. This slower approach speed could be attributed to the UAV's initial position being closer to the grip point. In Figure 16, the UAV and arm experience different trajectories. Initially, they ascend to a higher altitude relative to the target point before approaching the grip point. In contrast, the "Gripping from the Top" approach involved both the UAV and arm descending in altitude as they neared the target. Figure 17 illustrates the gripper's position, confirming that it successfully reached the desired grip point.

Figure 18 depicts the system states, where the pitch angle initially increased before decreasing, while the roll and heading angles first decreased and then increased, ultimately converging to zero. This pattern is similar to what was observed in the “Gripping from the Top” approach. Figure 19 presents the system states’ derivatives, where a similar rate of change in states is observed across both approaches. This indicates consistent behavior in the dynamics of the UAV and its control system during the gripping maneuvers, regardless of the approach direction.

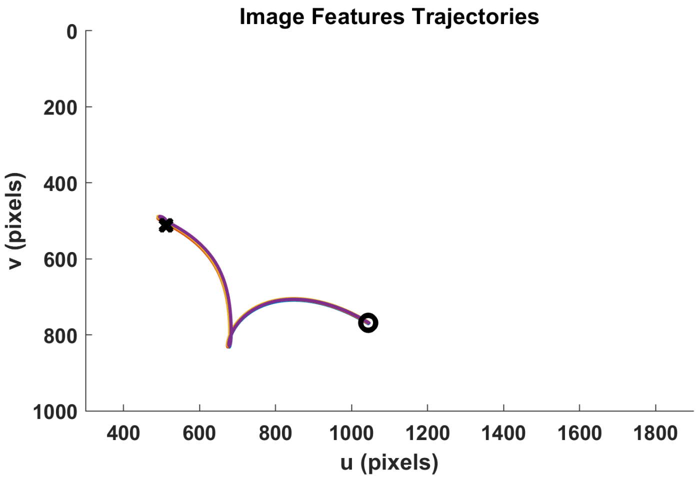

Figure 13. Feature trajectory—down approach.

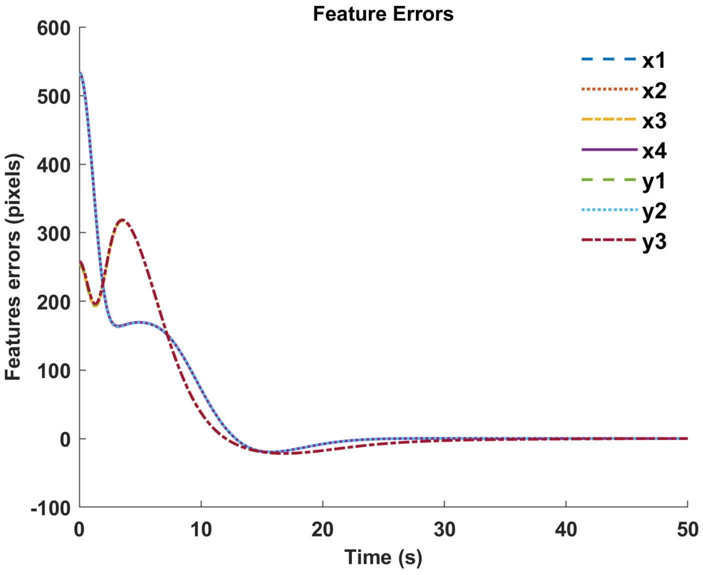

Figure 14. Feature error—down approach.

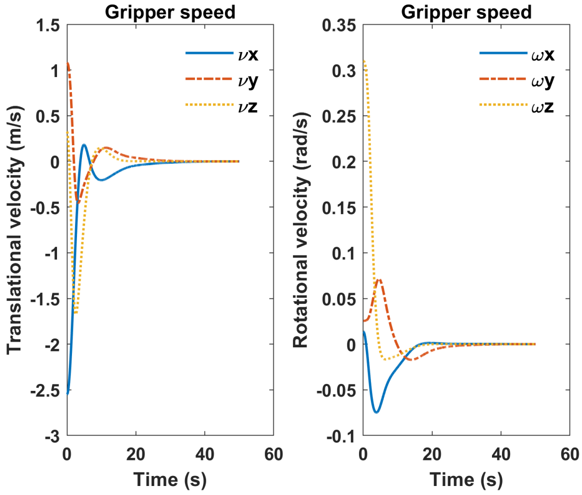

Figure 15. Gripper transnational and rotational velocities—down approach.

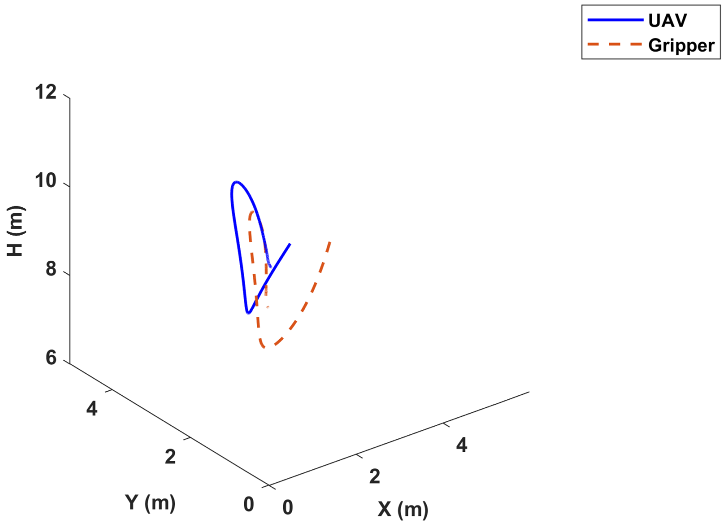

Figure 16. UAV and gripper trajectory—down approach.

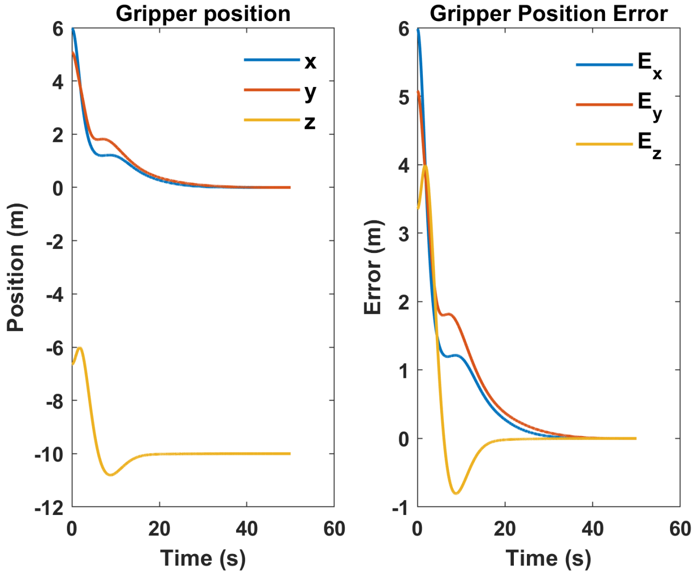

Figure 17. Gripper position and position error—down approach.

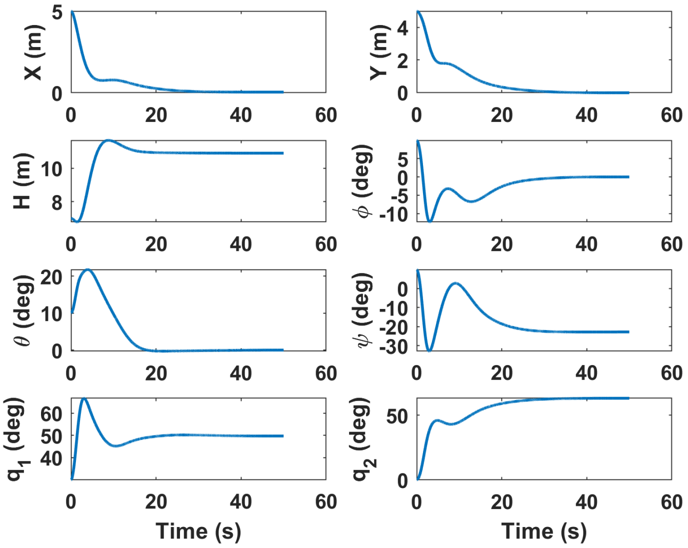

Figure 18. UAV and arm states—down approach.

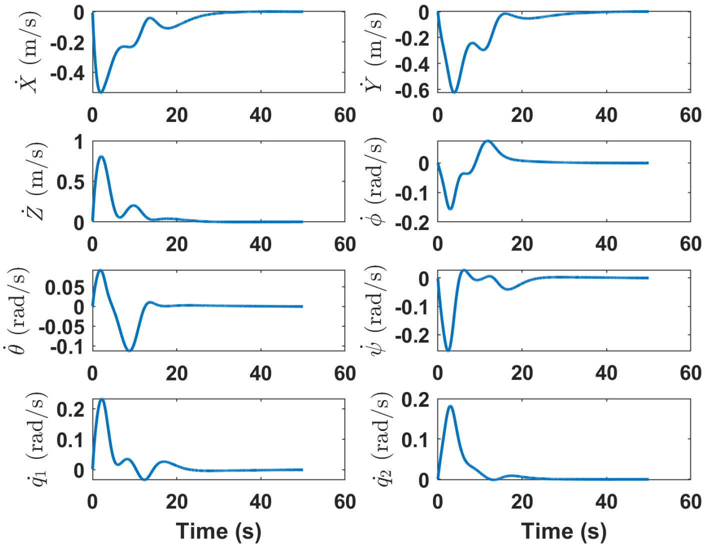

Figure 19. UAV and arms states' derivative—down approach.

6. Conclusions

An innovative visual servoing framework intended for aerial vegetation sampling was presented in this paper. The framework incorporates a number of cutting-edge methods, including a distilled neural network, adaptive sliding mode control, image-based visual servoing, and neural network-based grip point recognition. The method uses data from an external neural network that is devoted to grip point recognition to create control signals through the use of a model-based training architecture. These components work together to give the framework precise control over the aerial system. Simulation results were used to comprehensively assess the effectiveness of the suggested controller, showing that it can guide the gripper to the intended grasp position with nearly no error. Further research endeavors will investigate techniques to tackle feature loss and integrate additional sensors, such LiDAR, into the system to augment its resilience in practical situations. To increase the accuracy of drone-based measurements in vegetation monitoring applications, additional research will be carried out on the effects of data quality and sources of uncertainty. Additionally, the technique can be implemented in real time on the NVIDIA Jetson Orin NX, making use of its sophisticated processing powers to guarantee effective performance in real-world scenarios.

Author Contributions: Conceptualization, Z.S. and M.G.L.; methodology, Z.S. and M.G.L.; software, Z.S.; validation, Z.S.; formal analysis, Z.S.; investigation, Z.S.; resources, Z.S. and M.G.L.; data curation, Z.S.; writing—original draft preparation, Z.S.; writing—review and editing, M.G.L.; visualization, Z.S.; supervision, M.G.L.; project administration, M.G.L. All authors have read and agreed to the published version of the manuscript.

Funding: Funding support is gratefully acknowledged from the Natural Sciences and Engineering Research Council of Canada and from the University of Alberta.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Training data are available upon request by contacting the first author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ASM	Adaptive Sliding Mode
ASMC	Adaptive Sliding Mode Control
CNN	Convolutional Neural Network
DOF	Degrees of Freedom
DRL	Deep Reinforcement Learning
ELM	Extreme Learning Machine
IBVS	Image-based Visual Servoing
KD	Knowledge Distillation
MLP	Multilayer Perceptron
MPC	Model Predictive Control
MSE	Mean Squared Error
NDOBC	Disturbance Observer-Based Control
PBVS	Position-based Visual Servoing
RL	Reinforcement Learning
UAV	Unpiloted Aerial Vehicles

References

1. Charron, G.; Robichaud-Courteau, T.; La Vigne, H.; Weintraub, S.; Hill, A.; Justice, D.; Bélanger, N.; Lussier Desbiens, A. The DeLeaves: A UAV device for efficient tree canopy sampling. *J. Unmanned Veh. Syst.* **2020**, *8*, 245–264. [CrossRef]
2. Liu, Z.; Yan, Y.; Pang, J.; Guo, Q.; Guan, J.; Gu, J. Mini-drone assisted tree canopy sampling: A low-cost and high-precision solution. *Front. Plant Sci.* **2023**, *14*, 1272418. [CrossRef] [PubMed]
3. Li, Z.; Deng, X.; Lan, Y.; Liu, C.; Qing, J. Fruit tree canopy segmentation from UAV orthophoto maps based on a lightweight improved U-Net. *Comput. Electron. Agric.* **2024**, *217*, 108538. [CrossRef]
4. Kutia, J.R.; Stol, K.A.; Xu, W. Canopy sampling using an aerial manipulator: A preliminary study. In Proceedings of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, Denver, CO, USA, 9–12 June 2015; pp. 477–484.
5. Kutia, J.R.; Stol, K.A.; Xu, W. Initial flight experiments of a canopy sampling aerial manipulator. In Proceedings of the 2016 International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 7–10 June 2016; pp. 1359–1365. [CrossRef]
6. Kutia, J.R.; Stol, K.A.; Xu, W. Aerial manipulator interactions with trees for canopy sampling. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 1740–1749. [CrossRef]
7. Schweiger, A.K.; Lussier Desbiens, A.; Charron, G.; La Vigne, H.; Laliberté, E. Foliar sampling with an unmanned aerial system (UAS) reveals spectral and functional trait differences within tree crowns. *Can. J. For. Res.* **2020**, *50*, 966–974. [CrossRef]
8. La Vigne, H.; Charron, G.; Hovington, S.; Desbiens, A.L. Assisted canopy sampling using unmanned aerial vehicles (UAVs). In Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, Athens, Greece, 15–18 June 2021; pp. 1642–1647.
9. Giannico, V.; Lafortezza, R.; John, R.; Sanesi, G.; Pesola, L.; Chen, J. Estimating Stand Volume and Above-Ground Biomass of Urban Forests Using LiDAR. *Remote Sens.* **2016**, *8*, 339. [CrossRef]
10. Garofalo, S.P.; Giannico, V.; Lorente, B.; García, A.J.G.; Vivaldi, G.A.; Thameur, A.; Salcedo, F.P. Predicting carob tree physiological parameters under different irrigation systems using Random Forest and Planet satellite images. *Front. Plant Sci.* **2024**, *15*, 1302435. [CrossRef]
11. Zhang, Y.; Xu, B.; Xiang, C.; Fan, W.; Ai, T. Flight and interaction control of an innovative ducted fan aerial manipulator. *Sensors* **2020**, *20*, 3019. [CrossRef]

12. Khalesi, F.; Ahmed, I.; Daponte, P.; Picariello, F.; De Vito, L.; Tudosa, I. The Uncertainty Assessment by the Monte Carlo Analysis of NDVI Measurements Based on Multispectral UAV Imagery. *Sensors* **2024**, *24*, 2696. [CrossRef]
13. Samadikhoshkho, Z.; Lipsett, M. Decoupled control design of aerial manipulation systems for vegetation sampling application. *Drones* **2023**, *7*, 110. [CrossRef]
14. Liu, Y.; Meng, Z.; Zou, Y.; Cao, M. Visual Object Tracking and Servoing Control of a Nano-Scale Quadrotor: System, Algorithms, and Experiments. *IEEE CAA J. Autom. Sin.* **2021**, *8*, 344–360. [CrossRef]
15. Shi, L.; Li, B.; Shi, W. Vision-based UAV adaptive tracking control for moving targets with velocity observation. *Trans. Inst. Meas. Control.* **2024**, *46*, 01423312241228886. [CrossRef]
16. Chen, H.; Xia, K. Robust Image-Based Visual Servo Target Tracking of UAV with Depth Camera. In Proceedings of the 2024 IEEE International Conference on Industrial Technology (ICIT), IEEE, Bristol, UK, 25–27 March 2024; pp. 1–6.
17. Wang, G.; Qin, J.; Liu, Q.; Ma, Q.; Zhang, C. Image-based visual servoing of quadrotors to arbitrary flight targets. *IEEE Robot. Autom. Lett.* **2023**, *8*, 2022–2029. [CrossRef]
18. Yang, J.; Huo, X.; Xiao, B.; Fu, Z.; Wu, C.; Wei, Y. Visual servo control of unmanned aerial vehicles: An object tracking-based approach. In Proceedings of the 2017 29th Chinese Control And Decision Conference (CCDC), IEEE, Chongqing, China, 28–30 May 2017; pp. 3524–3528.
19. Luo, B.; Chen, H.; Quan, F.; Zhang, S.; Liu, Y. Natural feature-based visual servoing for grasping target with an aerial manipulator. *J. Bionic Eng.* **2020**, *17*, 215–228. [CrossRef]
20. Samadikhoshkho, Z.; Ghorbani, S.; Janabi-Sharifi, F. Vision-based reduced-order adaptive control of aerial continuum manipulation systems. *Aerosp. Sci. Technol.* **2022**, *121*, 107322. [CrossRef]
21. Molina, M.; Frau, P.; Maravall, D. A collaborative approach for surface inspection using aerial robots and computer vision. *Sensors* **2018**, *18*, 893. [CrossRef]
22. Shamshiri, R.R.; Dworak, V.; ShokrianZeini, M.; Navas, E.; Käthner, J.; Höfner, N.; Weltzien, C. *An Overview of Visual Servoing for Robotic Manipulators in Digital Agriculture*; Gesellschaft für Informatik e.V.: Hamburg, Germany, 2023.
23. Häni, N.; Isler, V. Visual servoing in orchard settings. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Daejeon, Republic of Korea, 9–14 October 2016; pp. 2946–2953.
24. Wang, C.L.; Lu, C.Y.; Li, H.W.; Wei, Z.C.; Cheng, X.P.; Mao, Y.J.; Hu, H.N.; Wang, C. Research progress on visual navigation technology of agricultural machinery. *Int. Agric. Eng. J.* **2019**, *28*.
25. Ahmadi, A.; Halstead, M.; McCool, C. Towards autonomous visual navigation in arable fields. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Kyoto, Japan, 23–27 October 2022; pp. 6585–6592.
26. Barth, R.; Hemming, J.; van Henten, E.J. Design of an eye-in-hand sensing and servo control framework for harvesting robotics in dense vegetation. *Biosyst. Eng.* **2016**, *146*, 71–84.
27. Busch, C.A.M.; Stol, K.A.; van der Mark, W. Dynamic tree branch tracking for aerial canopy sampling using stereo vision. *Comput. Electron. Agric.* **2021**, *182*, 106007. [CrossRef]
28. Cong, V.D.; Hanh, L.D. A review and performance comparison of visual servoing controls. *Int. J. Intell. Robot. Appl.* **2023**, *7*, 65–90.
29. Shi, H.; Wu, H.; Xu, C.; Zhu, J.; Hwang, M.; Hwang, K.S. Adaptive image-based visual servoing using reinforcement learning with fuzzy state coding. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 3244–3255. [CrossRef]
30. Saxena, A.; Pandya, H.; Kumar, G.; Gaud, A.; Krishna, K.M. Exploring convolutional networks for end-to-end visual servoing. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, Singapore, 29 May–3 June 2017; pp. 3817–3823.
31. Fu, G.; Chu, H.; Liu, L.; Fang, L.; Zhu, X. Deep reinforcement learning for the visual servoing control of uavs with fov constraint. *Drones* **2023**, *7*, 375. [CrossRef]
32. Shi, H.; Li, X.; Hwang, K.S.; Pan, W.; Xu, G. Decoupled visual servoing with fuzzy Q-learning. *IEEE Trans. Ind. Inform.* **2016**, *14*, 241–252. [CrossRef]
33. Kang, M.; Chen, H.; Dong, J. Adaptive visual servoing with an uncalibrated camera using extreme learning machine and Q-leaning. *Neurocomputing* **2020**, *402*, 384–394. [CrossRef]
34. Chen, J.; Hua, C.; Guan, X. Image based fixed time visual servoing control for the quadrotor UAV. *IET Control. Theory Appl.* **2019**, *13*, 3117–3123. [CrossRef]
35. Jo, K.; Chwa, D. Robust Hybrid Visual Servoing of Omnidirectional Mobile Manipulator With Kinematic Uncertainties Using a Single Camera. *IEEE Trans. Cybern.* **2023**, *54*, 2824–2837. [CrossRef]
36. Zhang, L.; Pei, J.; Bai, K.; Chen, Z.; Zhang, J. A Closed-Loop Multi-perspective Visual Servoing Approach with Reinforcement Learning. In Proceedings of the 2023 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE, Samui, Thailand, 4–9 December 2023; pp. 1–7.
37. Jin, Z.; Wu, J.; Liu, A.; Zhang, W.A.; Yu, L. Policy-based deep reinforcement learning for visual servoing control of mobile robots with visibility constraints. *IEEE Trans. Ind. Electron.* **2021**, *69*, 1898–1908. [CrossRef]
38. He, Y.; Gao, J.; Li, H.; Chen, Y.; Li, Y. Hierarchical Reinforcement Learning-Based End-to-End Visual Servoing With Smooth Subgoals. *IEEE Trans. Ind. Electron.* **2023**, *71*, 11009–11018. [CrossRef]

39. Copot, C.; Ionescu, C.M.; Muresan, C.I.; Copot, C.; Ionescu, C.M.; Muresan, C.I. Image Feature Extraction and Evaluation. *Image-Based and Fractional-Order Control for Mechatronic Systems: Theory and Applications with MATLAB®*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 29–61.
40. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 84–90. [CrossRef]
41. García-Aracil, N.; Malis, E.; Aracil-Santonja, R.; Pérez-Vidal, C. Continuous visual servoing despite the changes of visibility in image features. *IEEE Trans. Robot.* **2005**, *21*, 1214–1220. [CrossRef]
42. Ghasemi, A.; Li, P.; Xie, W.F.; Tian, W. Enhanced switch image-based visual servoing dealing with featuresloss. *Electronics* **2019**, *8*, 903. [CrossRef]
43. Chesi, G. Visual servoing path planning via homogeneous forms and LMI optimizations. *IEEE Trans. Robot.* **2009**, *25*, 281–291. [CrossRef]
44. Kazemi, M.; Gupta, K.; Mehrandezh, M. Global path planning for robust visual servoing in complex environments. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, IEEE, Kobe, Japan, 12–17 May 2009; pp. 326–332.
45. Allibert, G.; Courtial, E.; Chaumette, F. Predictive control for constrained image-based visual servoing. *IEEE Trans. Robot.* **2010**, *26*, 933–939. [CrossRef]
46. Heshmati-Alamdari, S.; Karavas, G.K.; Eqtami, A.; Drossakis, M.; Kyriakopoulos, K.J. Robustness analysis of model predictive control for constrained image-based visual servoing. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), IEEE, Hong Kong, China, 31 May-7 June 2014; pp. 4469–4474.
47. Hajiloo, A.; Keshmiri, M.; Xie, W.F.; Wang, T.T. Robust online model predictive control for a constrained image-based visual servoing. *IEEE Trans. Ind. Electron.* **2015**, *63*, 2242–2250.
48. Bateux, Q.; Marchand, E. Direct visual servoing based on multiple intensity histograms. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), IEEE, Seattle, WA, USA, 26–30 May 2015; pp. 6019–6024.
49. Sampedro, C.; Rodriguez-Ramos, A.; Gil, I.; Mejias, L.; Campoy, P. Image-based visual servoing controller for multirotor aerial robots using deep reinforcement learning. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Madrid, Spain, 1–5 October 2018; pp. 979–986.
50. Zhang, W.; Song, K.; Rong, X.; Li, Y. Coarse-to-fine UAV target tracking with deep reinforcement learning. *IEEE Trans. Autom. Sci. Eng.* **2018**, *16*, 1522–1530. [CrossRef]
51. Florence, P.; Manuelli, L.; Tedrake, R. Self-supervised correspondence in visuomotor policy learning. *IEEE Robot. Autom. Lett.* **2019**, *5*, 492–499. [CrossRef]
52. Fei, H.; Wang, Z.; Kennedy, A. Robust Reinforcement Learning Based Visual Servoing with Convolutional Features. *IFAC-PapersOnLine* **2023**, *56*, 9781–9786. [CrossRef]
53. Chen, B.; Sax, A.; Lewis, G.; Armeni, I.; Savarese, S.; Zamir, A.; Malik, J.; Pinto, L. Robust policies via mid-level visual representations: An experimental study in manipulation and navigation. *arXiv* **2020**, arXiv:2011.06698.
54. Condat, R.; Vasseur, P.; Allibert, G. Focusing on Object Extremities for Tree Instance Segmentation in Forest Environments. *IEEE Robot. Autom. Lett.* **2024**, *9*, 5480–5487. [CrossRef]
55. Silva, R.; Junior, J.M.; Almeida, L.; Gonçalves, D.; Zamboni, P.; Fernandes, V.; Silva, J.; Matsubara, E.; Batista, E.; Ma, L.; et al. Line-based deep learning method for tree branch detection from digital images. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *110*, 102759. [CrossRef]
56. Samadikhoshkho, Z.; Ghorbani, S.; Janabi-Sharifi, F.; Zareinia, K. Nonlinear control of aerial manipulation systems. *Aerosp. Sci. Technol.* **2020**, *104*, 105945. [CrossRef]
57. Samadikhoshkho, Z.; Ghorbani, S.; Janabi-Sharifi, F. Coupled dynamic modeling and control of aerial continuum manipulation systems. *Appl. Sci.* **2021**, *11*, 9108. [CrossRef]
58. Ghorbani, S.; Samadikhoshkho, Z.; Janabi-Sharifi, F. Dual-arm aerial continuum manipulation systems: Modeling, pre-grasp planning, and control. *Nonlinear Dyn.* **2023**, *111*, 7339–7355. [CrossRef]
59. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
60. Samadi Khoshkho, M.; Samadikhoshkho, Z.; Lipsett, M.G. Distilled neural state-dependent Riccati equation feedback controller for dynamic control of a cable-driven continuum robot. *Int. J. Adv. Robot. Syst.* **2023**, *20*, 17298806231174737. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.