Article

Development and Control of a Pneumatic-Actuator 3-DOF Translational Parallel Manipulator with Robot Vision [†]

Lian-Wang Lee [1], Hsin-Han Chiang [2] and I-Hsum Li [3],*

[1] Department of Mechanical Engineering, National Chung Hsing University, No. 145, Xingda Road, South District, Taichung City 40227, Taiwan; leelw@dragon.nchu.edu.tw
[2] Department of Electrical Engineering, National Taiwan Normal University, No 162, Section 1, Heping East Road, Taipeu City 106, Taiwan; hsinhan@ntnu.edu.tw
[3] Department of Mechanical and Electro-Mechanical Engineering, Tamkang University, No. 151, Yingzhuan Road, Tamsui District, New Taipei City 25137, Taiwan
* Correspondence: ihsumlee@gmail.com; Tel.: +886-2-2620-9745 (ext. 2061); Fax: +886-2-2620-9745
† This paper is an extended version of Yeh, C.-P.; Lee, L.-W.; Li, I-H.; Chiang, H.-H. A Translational Parallel Manipulator with Three Horizontal-Axial Pneumatic Actuators for 3-D Path Tracking, In proceedings of 2018 International Conference on System Science and Engineering (ICSSE), New Taipei, Taiwan, 28–30 June 2018.

check for updates

Abstract: A vision-based three degree-of-freedom translational parallel manipulator (TPM) was developed. The developed TPM has the following characteristics. First, the TPM is driven by three rodless pneumatic actuators and is designed as a horizontal structure to enlarge its horizontal working space to cover a conveyor. Then, a robot-vision system (including a webcam mounted on the TPM) collects images of objects on the conveyor and transfers them through the LabVIEW application programming interface for image processing. Since it is very difficult to achieve precise position control of the TPM due to the nonlinear couplings among the robot axes, feedback linearization is utilized to design an adaptive interval type-2 fuzzy controller with self-tuning fuzzy sliding-mode compensation (AIT2FC-STFSMC) for each rodless pneumatic actuator to attenuate nonlinearities, function approximation errors, and external disturbances. Finally, experiments proved that the vision-based three degree-of-freedom TPM was capable of accurately tracking desired trajectories and precisely executing pick-and-place movement in real time.

Keywords: interval type-2 fuzzy systems; path tracking control; parallel manipulator; rodless pneumatic actuator; robotic manipulators; robot vision

1. Introduction

Robotic manipulators are efficient at picking, placing, and assembling objects and at tracking movements. With respect to their kinematic structures, robotic manipulators are generally divided into two types, namely a serial and a parallel type. A serial manipulator is designed as a series of links, which are sequentially connected by actuated joints, from a base to an end-effector. This arm-like structure is highly flexible in large-scale operations. However, the serially linked mechanism provides low positioning accuracy because the error for each joint and link is superimposed and the stiffness is insufficient for heavy loads. For a parallel manipulator (PM), several independent kinematic chains are connected by a moving platform, termed an end-effector, and a fixed base platform, while the actuators are located on or near the fixed base. PMs have many advantages, mainly because the load is shared by the several links connecting the moving platform to the base. Compared to the serial type of manipulators, the advantages of PMs also include higher stiffness, higher load-carrying capacity,

and being able to achieve superior precision with higher speed and acceleration [1]. In addition, for PMs, the error in the position for each single manipulator is the average of errors for all manipulators and is not cumulative. The disadvantages of parallel-type robotic manipulators are that they allow only a limited workspace, require complex kinematic analysis, and are extremely difficult to control [2]. However, the increasing demand for high-speed operation, high precision, and increased stiffness means that PMs are increasingly important to industrial automation.

The first PM was designed as an amusement device by James E. Gwinnett in 1928. Further, a six degree-of-freedom (6-DOF) platform was developed by Stewart [3] and was called the Stewart platform. It has numerous applications in flight simulators and medical applications [1,4]. However, a 6-DOF PM has problems in complex kinematic analysis and motion coupling between axes and, thus, is not ideal for many applications. A PM with a relatively simple structure, such as a three degree-of-freedom (3-DOF) PM, has the inherent advantages of parallel mechanisms but offers additional cost reductions in manufacturing and operations. Hence, development of simple-structure PMs has been the focus of many researchers. In 1988, the DELTA robot with 3-DOF was designed by the research team of Clavel [5], and it was demonstrated to be suitable for industrial applications [6]. Since then, researchers have been involved in developing 3-DOF PMs with different structures and configurations, such as spherical 3-DOF mechanisms, 3-RPS PMs, and orthoglide parallel robots [6,7]. In addition, parallel manipulators have advantages of carrying a high payload with high-performance output motion. In 2017, Zi et al. [8] developed a winding hybrid-driven 3-cable PM, in which the three cables are applied to suspend an end-effector. By controlling the cables and utilizing gravitational force, the end-effector is able to move within a large workspace. Zi et al. used an adaptive fuzzy sliding mode controller for dynamic control compensation and verified the trajectory tracking through experimental studies. A kinematic calibration method was proposed by Qian et al. in [9] for cable-driven PM to improve the efficiency of calibration measurement. Qian et al. verified its accuracy and efficiency according to cable length through numerical simulation. However, the manufacturing of 3-DOF PMs encounters difficulties in the trajectory tracking control due to complicated dynamic and kinematic modeling [10]. To the present, joint space control and task space control are the most used strategies in controlling PMs. In the task space control, the forward kinematic problem can be effectively solved by using the measured lengths of the PM links. As compared to the joint speed control, the more precise position of the PM end-effector can be performed by using vision and position sensors [11,12]. In 2011, Chiang et al. [13] developed a 3-prismatic-universal-universal (3-PUU) PM that is driven by three vertical-axial pneumatic actuators (PAs) and used servo control for path tracking. For robotic arms, the robot vision provides extensive information for controlling the positioning of the end-effector for object handling, thereby, improving the system's accuracy, flexibility, and self-sufficiency [14]. Therefore, vision-based robotic arms have recently attracted considerable attention for industrial applications. Papanikolopoulos et al. [15] and Oh et al. [16] et al. presented visual servoing control, which uses visual information to control the position of the robot's end-effector relative to a target object. Peurla et al. [17] presented a low-cost vision-guided robot arm to place grids on films; that robot arm can reach a 90% success rate in pick-and-place experiments. In [18], joint space control for a three-axial pneumatic PM was developed with a stereo vision 3D position measurement system; the desired trajectory in task space was obtained by solving the inverse kinematic problem. In [19], vision-based control was investigated for the path tracking control of a 3-DOF translational PM (TPM) in which the pose of the end-effector was measured by a stereo vision system.

Recently, three types of actuators have been widely utilized for driving PMs, namely, electric, hydraulic, and pneumatic. Among them, PAs have many advantages over other types of driving actuators, such as a cost-effective actuation, the cleanliness of the working environment, ease of maintenance, and rapid movement and reactions. However, the complexity of the trajectory tracking design in PMs easily raises while using PAs, which mainly originate from the compressibility of air, different regimes of air flow through valves, and the dynamic behavior of the friction forces of pneumatic cylinders. In this regard, various control strategies have been developed by researchers

for the trajectory tracking control of 3-DOF TPMs. In [18], the performance levels of different control algorithms based on proportional-integral-derivative (PID) and adaptive control design were experimentally evaluated on a 3-DOF planar PM. In addition, an adaptive dynamic control scheme was proposed in [19] and implemented for a 3-DOF PM to conduct the experimental study. A convex synchronized control method was used in the trajectory tracking control of a 3-DOF planar PM and evaluated under several experimental conditions [20]. In [21], a dynamic control scheme of parallel kinematic mechanism based on a model-based approach was proposed and experimentally validated on a 3-DOF PM. So far, the rodless PA (RPA) is an important type of PM, which has been widely used in robotics and industrial automation. Rodless pneumatic actuators (RPAs) have numerous advantages, such as low cost, low maintenance requirements, safe operation, ease of operation, simple structure, and high power-to-weight ratio [22,23]. It is well known that the servo position control technique plays a key role in the application of pneumatic servo systems [24]. The smooth and accurate positioning of RPAs for complex control tasks remains challenging due to the nonlinearities of RPAs, which majorly come from low stiffness and compressibility of air, large friction forces, dead zones, and varying rates of air flow through servo valves [25]. These nonlinear characteristics of RPAs also cause a large number of difficulties for modeling and control. To deal with this problem, researchers [26–29] have recently developed advanced models and nonlinear control strategies to improve the performance of RPAs. Tsai [30] presented a sliding-mode controller (SMC) for an RPA and demonstrated that the RPA had favorable robustness and a high tolerance for uncertainties [30]. Lee et al. [31] also proposed an SMC that used functional approximation with H∞ tracking performance for RPA trajectory tracking. Zhao et al. [32,33] utilized an SMC scheme with an active disturbance rejection controller to deal with strong static friction and unknown disturbances in pneumatic servo position systems. However, the detailed dynamic model for RPAs is too difficult to be identified due to complex nonlinearities and time-varying characteristics of RPAs. Motivated by the previous discussions, this paper presents an intelligent model-free control system for RPAs. To improve RPA control performance, the proposed control system employs an adaptive interval type-2 fuzzy approximation technique to design an equivalent controller for a conventional SMC and combines a self-tuning scheme using the fuzzy SMC approach to compensate for function approximation errors, unmodeled dynamics, and disturbances. The proposed control scheme does not require any system dynamic model for controller design, is free from chattering, has been proven to be stable for tracking control, and is robust against uncertainties.

The three purposes of this paper are as follows. The first purpose is to develop, from scratch, a complete mechatronic design of a low-cost 3-DOF PM to effectively control translational motion. The second purpose is to develop a test-rig of a vision-based PM-driven TPM, which allows the implementation and testing of dynamic control schemes for the 3-DOF TPM. The third purpose is to develop robot vision and implement pick-and-place operations for the 3-DOF PM. For these purposes, a 3-DOF TPM, driven by three horizontal-axial RPAs with the associated proportional directional control valve (PDCV), was developed for three-dimensional (3D) path tracking control. The assembly configuration of the RPAs arranges the PM as a horizontal structure, which offers a large horizontal working space. In terms of the design of the controller, the RPA model in the three horizontal motion axes is first derived. A typical RPA suffers from bounded unknown parameters and external disturbances; to allow 3D path tracking control for the moving platform, this study used feedback linearization techniques, an interval type-2 fuzzy system, sliding-mode control theory, and an adaptive control scheme to control the positions for the three horizontal RPAs. To locate an object, images of the object are collected by a webcam and are transferred to LabVIEW through an application programming interface (API) with image processing toolkits to locate the object's coordinates for the system.

In this paper, there are main superiority and innovations in the developed vision-based 3-DOF TPM, which are:

- To develop a low-cost and high-reliability TPM, we use a rodless pneumatic actuator (RPA), which has many superiorities in its low-cost, low maintenance requirements, safe operation and simple structure, to drive the TPM.

- The presented adaptive interval type-2 fuzzy controller with self-tuning fuzzy sliding-mode compensation (AIT2FC-STFSMC) effectively attenuates the nonlinearities in the TPM, which come from two parts: (1) the pneumatic cylinder leads to its low stiffness and compressibility of air and large friction forces; and (2) the use of the valve leads to dead zones and varying rates of air flow through servo valves.
- The assembly configuration of the RPAs arranges the PM as a horizontal structure, which offers a large horizontal working space, and a soft pneumatic gripper installed on the end-effector of the TPM is able to easily grip an object.
- The develop vision-based 3-DOF TPM has the ability to accurately execute pick-and-place movement in real time.

The remainder of this paper is organized as follows. In Section 2, the mechanical parts of the RPA-driven 3-DOF TPM and its experimental setup are described. Further, mathematical models for representations of the used RPA with cylinders and pressure dynamics are developed in Section 3. The control system design with the stability analysis is given in Section 4. Section 5 introduces the robot-vision technique with the LabVIEW API, and three experiments are presented in Section 6 to investigate the trajectory tracking performance and pick-and-place operation. Finally, Section 7 concludes this paper.

2. Test Rig Layout of 3-DOF TPM

In this section, the geometrical configuration of the constructed RPA-driven 3-DOF TPM is first introduced. The layout of the 3-DOF TPM is shown in Figure 1, and its photo is shown in Figure 2. A PC-based control unit labeled 12 in Figure 1, which is installed inside the control box under the fixed platform, controls the system. This RPA-driven 3-DOF TPM communicates through analog-to-digital and digital-to-analog converter (AD/DA) interface cards. Three RPAs with 25-mm-diameter pistons are horizontally installed under the parallel links at 120 degrees to each other to move the platform in three dimensions. Three Festo-mode DGC-25-500 RPAs produce horizontal motion in this design; each RPA has a 500-mm stroke. For each horizontal axis, a Festo 5/3 MPYE-5-1/8-HF-010-B PDCV regulates the flow of air into the cylinder. This device is crucial to controlling the PM. The voltage supplied to each PDCV is at 24 V, and the control voltage is in the range of 0 to 10 V. The maximum recommended nominal flow is at 700 L/min. In this system, the pressure source is set at 6 bar. The two pressure levels in the two cylinder chambers are measured by two Festo model SDE1 pressure sensors, which are installed on the two ports of the cylinder. A PC-based controller provides a control signal that regulates the PDCVs to drive the horizontal RPAs through a D/A interface that operates at a sampling frequency of 200 Hz. To accurately measure the position of each horizontal RPA, three linear encoders with a resolution of 1 μm are installed on the RPAs. The position signals are sent to the PC-based controller through a counter card. For robot vision, a Logitech C920 webcam mounted on the support frame is used to capture images of each object.
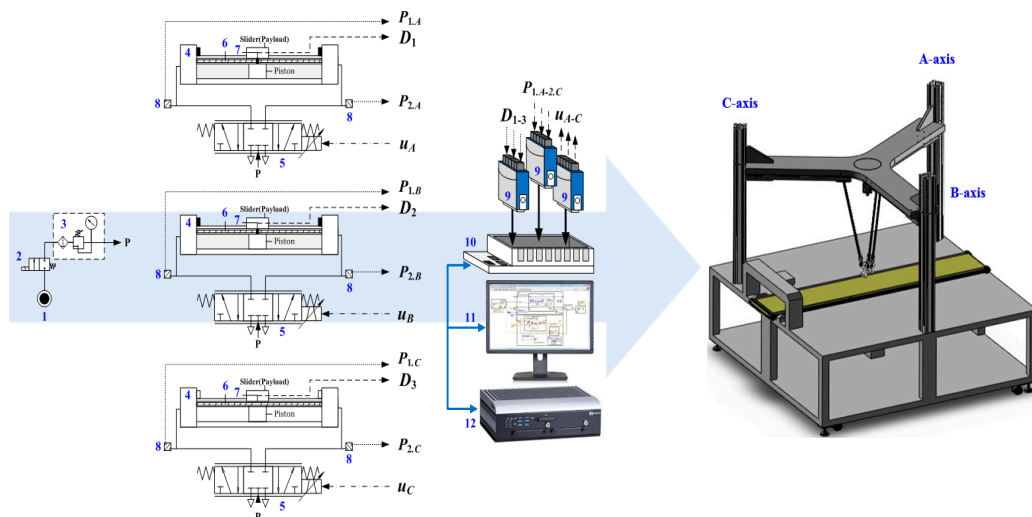
Figure 1. Test rig layout of the rodless pneumatic actuator (RPA)-driven three degree-of-freedom (3-DOF) translational parallel manipulator (TPM). 1. Pressure source; 2. Solenoid valve; 3. Air preparation unit; 4. Rodless pneumatic actuator; 5. Proportional directional control valve; 6. Linear encoder; 7. Scanning head; 8. Pressure sensor; 9. Interface card; 10. Compact-RIO embedded system; 11. LabVIEW software; 12. Industrial PC.
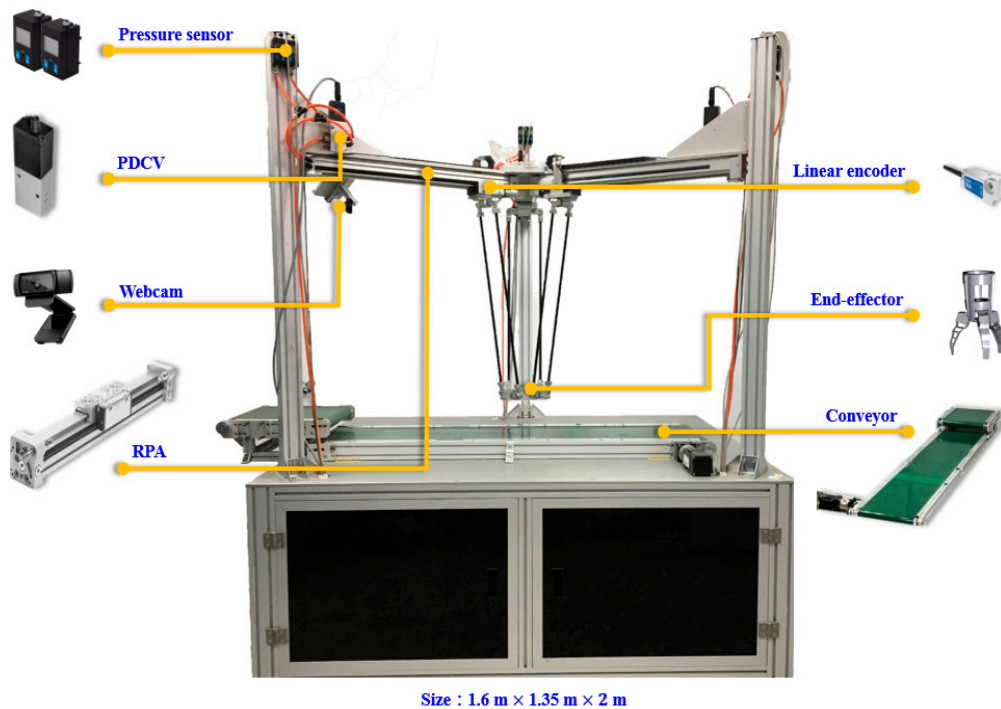

Figure 2. Photo of the RPA-driven 3-DOF TPM.

3. Analysis of Kinematics

Typically, there are two parts for manipulator kinematics analysis, namely inverse kinematics and forward kinematics. From the analysis of inverse kinematics, a set of actuated joint variables can be identified to achieve a targeted position and orientation for the end-effector of TPM. In the forward kinematics, the position and orientation of the end-effector can be determined from the given actuated joint coordinates of TPM. This study employs the vector-loop closure equation [33,34] to derive both of the inverse and forward kinematics in this section. In the beginning, the desired 3D path profiles can be converted to the trajectory profiles for each of the three RPAs by using inverse kinematics. With the

connection of three RPAs in a parallel mechanism with a link and RPA-driven joints, the end-effector mounted on a movable platform can, thus, be driven to perform 3-D motions while controlling each RPA to track the associated trajectory profile.

3.1. Inverse Kinematic Analysis

Initially, two coordinate frames are investigated for the RPA-driven 3-DOF TPM to analyze the kinematic model, as shown in Figure 3. The first frame is a static Cartesian coordinate frame, $O(x_0, y_0, z_0)$, which is fixed at the center of the base, whereas a mobile Cartesian coordinate frame, $P(x_p, y_p, z_p)$, is assigned to the center of the mobile platform. In Figure 3, A_i indicates the joints located at the center of the base and B_i represents the passive joints of the moveable platform, $(i = 1, 2, 3)$. Since three links are installed between the mobile platform and the fixed platform, the lengths of the link R_1 and R_2 in Figure 3 can be expressed as:

$$R_1 = \overline{A_1 B_1} = \overline{A_2 B_2} = \overline{A_3 B_3}, \quad R_2 = \overline{B_1 P} = \overline{B_2 P} = \overline{B_3 P} \tag{1}$$

Let C_{A_i} be the position of the point A_i, and let C_{B_i} be the position of the point B_i; then, it follows that

$$C_{Ai} = \left\{ \begin{array}{c} D_i \times \cos \beta_i \\ D_i \times \sin \beta_i \\ 0 \end{array} \right\} \tag{2}$$

and

$$C_{B_i} = C_P + C_{PB_i} = \left\{ \begin{array}{c} x_p + R_2 \times \cos \gamma_i \\ y_p + R_2 \times \sin \gamma_i \\ z_p \end{array} \right\} \tag{3}$$

where D_i represents the linear displacement of ith RPA that is also the cylinder piston position of ith limb; γ_i and β_i are computed as $\beta_i = \gamma_i = (i - 1) \times 120°$, as displayed in Figure 4. According to Equations (1)–(3), one can calculate

$$\overline{C_{Ai} C_{Bi}}^2 = R_1^2 \tag{4}$$

Equation (4) can be further expressed as:

$$D_i^2 - D_i \times 2[(x_p + R_2 \times \cos \gamma_i) \times \cos \beta_i + (y_p + R_2 \times \sin \gamma_i) \times \sin \beta_i] \\ + (x_p + R_2 \times \cos \gamma_i)^2 + (y_p + R_2 \times \sin \gamma_i)^2 + z_p^2 - R_1^2 = 0. \tag{5}$$

Let U_i and V_i be defined as:

$$U_i = (x_p + R_2 \times \cos \gamma_i) \times \cos \beta_i + (y_p + R_2 \times \sin \gamma_i) \times \sin \beta_i \tag{6}$$

and

$$V_i = (x_p + R_2 \times \cos \gamma_i)^2 + (y_p + R_2 \times \sin \gamma_i)^2 + z_p^2 - R_1^2 = 0 \tag{7}$$

Then, substituting Equations (6) and (7) into Equation (5), the following kinematics of each RPA can be yielded.

$$D_i = U_i \pm \sqrt{U_i^2 - V_i} \tag{8}$$

Notably, Equation (8) has two solutions for each actuator. According to the current assembly of the mechanism, only the positive square root solution is the feasible condition while the actuator is translational from outside to inside. Consequently, the inverse kinematics equation for each RPA of the 3-DOF TPM can be found as:
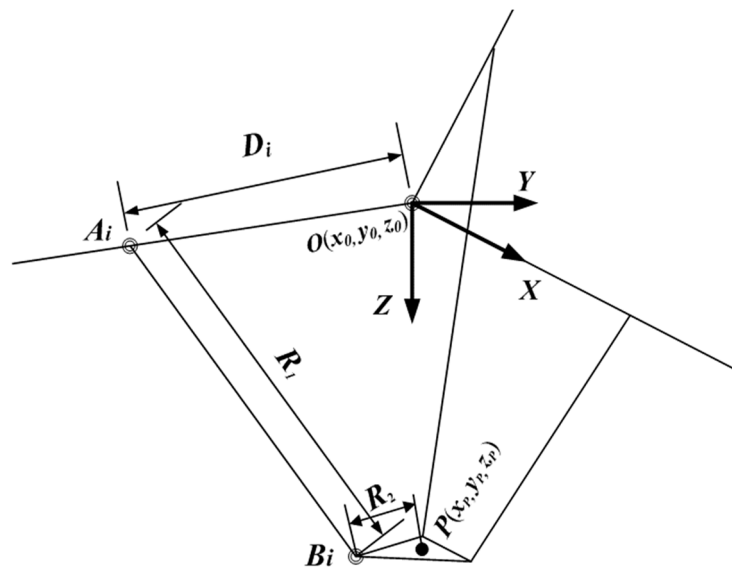
$$D_i = U_i + \sqrt{U_i^2 - V_i} \tag{9}$$
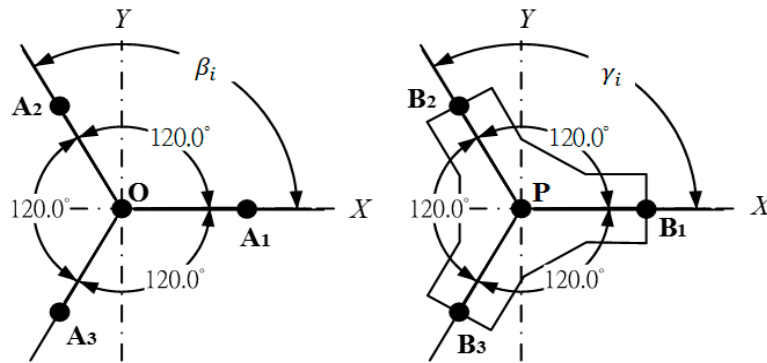
Figure 3. Schematic of the 3-DOF TPM.

Figure 4. Schematic of mobile and fixed platforms for the 3-DOF TPM.

3.2. Forward Kinematic Analysis

The objective of forward kinematics is to acquire the 3-D pose information for the end-effector. If given D_i, then C_{A_i} and C_{B_i} can be calculated by

$$C_{Ai} = \left\{ \begin{array}{c} (D_i - R_2) \times \cos \beta_i \\ (D_i - R_2) \times \sin \beta_i \\ 0 \end{array} \right\} \tag{10}$$

and

$$C_{Bi} = \left\{ \begin{array}{c} x_p + (R_2 - R_2) \times \cos \gamma_i \\ y_p + (R_2 - R_2) \times \sin \gamma_i \\ z_p \end{array} \right\} = \left\{ \begin{array}{c} x_p \\ y_p \\ z_p \end{array} \right\} \tag{11}$$

Substituting Equations (10) and (11) into Equation (4) yields

$$\left\{ \begin{array}{c} x_p - (D_i - R_2) \times \cos \beta_i \\ y_p - (D_i - R_2) \times \sin \beta_i \\ z_p \end{array} \right\}^2 - R_1^2 = 0 \tag{12}$$

Expressing the vectors in Equation (12) as:

$$x_p^2 + (D_i - R_2)^2 \times \cos^2 \beta_i - 2x_p(D_i - R_2) \cos \beta_i + y_p^2 + (D_i - R_2)^2 \sin^2 \beta_i - 2y_p(D_i - R_2) \sin \beta_i + z_p^2 - R_1^2 = 0 \tag{13}$$

Let $(D_i - R_i) = Q_i$, $i = 1, 2$, and 3, Equation (13) can then be rewritten as:

$$x_p^2 + Q_1^2 - 2x_p Q_1 + y_P^2 + z_P^2 - R_1^2 = 0 \tag{14}$$

$$x_p^2 + \frac{1}{4}Q_2^2 + x_p Q_2 + y_P^2 + \frac{3}{4}Q_2{}^2 - \sqrt{3}y_p Q_2 + z_P^2 - R_1^2 = 0 \tag{15}$$

and

$$x_p^2 + \frac{1}{4}Q_3^2 + x_p Q_3 + y_P^2 + \frac{3}{4}Q_3{}^2 + \sqrt{3}y_p Q_3 + z_P^2 - R_1^2 = 0 \tag{16}$$

Therefore, from Equations (14)–(16) the forward kinematics is described as follows:

$$x_p = \frac{y_p \sqrt{3}Q_2 + Q_1^2 - Q_2^2}{2Q_1 + Q_2} \tag{17}$$

$$y_p = \frac{-\left[(Q_3^2 - Q_1^2)(2Q_1 + Q_2) + (Q_1^2 - Q_2^2)(2Q_1 + Q_3)\right]}{2\sqrt{3}(Q_1 Q_2 + Q_2 Q_3 + Q_1 Q_3)} \tag{18}$$

and

$$z_p = -\sqrt{R_1^2 - (x_p - Q_1)^2 - y_p^2} \tag{19}$$

4. Analysis of the Design and Stability of the Controller

4.1. Dynamic Model of the RPA

In the RPA, the orifice area of the PDCVs can be controlled by the input air flow. With the pressure difference resulted from two chambers of the cylinder, the RPA can be moved in the desired position. According to the analysis in [35], the dynamic model of RPA can be derived while considering the dynamics and the mass flow rate of PDCV, the continuity equation, and the motion equation. Thus, the nonlinear model of the RPA can be presented in the form of a fourth-order nonlinear system as [35]:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{1}{M}\left[-K_f x_2 - K_{s-c}(x_1)S(x_2, x_3, x_4) + A(x_3 - x_4)\right] \\
\dot{x}_3 &= \frac{-k\left[x_2 x_3 - \frac{RT_S}{A}C_d C_0 w \hat{f}(x_3, P_s, P_e)u\right]}{x_1 + \Delta} \\
\dot{x}_4 &= \frac{k\left[x_2 x_4 - \frac{RT_S}{A}C_d C_0 w \hat{f}(x_4, P_s, P_e)u\right]}{l - x_1 + \Delta}
\end{aligned} \tag{20}$$

where $x_1 = x$, which is the piston position, $x_2 = \dot{x}$; $x_3 = P_a$, which is the pressure in chamber a; $x_4 = P_b$, which is the pressure in chamber b; the control input u indicates the spool displacement of PDCV; $K_{s-c}(x_1)$ denotes a combination of the static and dynamic frictions; w denotes a port width of the PDCV; l denotes a stroke of the RPA; Δ is the general residual chamber volume; R is the universal gas constant; $T_s = 293$ K is the supply temperature; C_0 is the flow constant, and $C_d = 0.8$ is the discharge coefficient. For the convenience of the analysis, the following function is introduced:

$$\widetilde{f}(p_r) = \begin{cases} 1, & \frac{P_{atm}}{P_u} < p_r \leq C_r \\ C_k[p_r^{\frac{2}{k}} - p_r^{\frac{(k+1)}{k}}]^{\frac{1}{2}}, & C_r < p_r < 1 \end{cases} \tag{21}$$

where P_{atm} is the atmospheric pressure; $p_r = P_d/P_u$ is the ratio between the downstream and the upstream pressures at the orifice; $k = 1.4$ is the specific heat constant; $C_r = (2/k + 1)^{k/(k+1)} = 0.528$, $C_k = \sqrt{2/(k-1) \times (k+1/2)^{(k+1)/(k-1)}} = 3.864$. It can be shown that the function $\widetilde{f}(\cdot)$ and its

derivative are continuous with respect to p_r. According to Equation (21) the functions $\hat{f}(x_3, P_s, P_e)$ and $\hat{f}(x_4, P_s, P_e)$ in Equation (20) are defined as:

$$\hat{f}(x_3, P_s, P_e) = \begin{cases} P_s \widetilde{f}\left(\frac{x_3}{P_s}\right) / \sqrt{T_s}, \ chamber = a \\ x_3 \widetilde{f}\left(\frac{P_e}{x_3}\right) / \sqrt{T_a}, \ chamber = b \end{cases} \tag{22}$$

and

$$\hat{f}(x_4, P_s, P_e) = \begin{cases} x_4 \widetilde{f}\left(\frac{P_e}{x_4}\right) / \sqrt{T_b}, chamber \ = \ a, \\ P_s \widetilde{f}\left(\frac{x_4}{P_s}\right) / \sqrt{T_s}, chamber \ = \ b, \end{cases} \tag{23}$$

where $P_s = 6 \times 10^5$ N/m^2 is the supply pressure; $P_e = 1 \times 10^5$ N/m^2 is the exhaust pressure; T_a and T_b are the cylinder air temperature of chamber A and B, respectively. In Equation (20), $-K_f x_2 - K_{s-c}(x_1)S(x_2, x_3, x_4)$, denotes the sum of the effects of the system static and dynamic friction forces, and

$$K_{s-c}(x_1)S(x_2, x_3, x_4) := \begin{cases} A(x_3 - x_4) \\ \quad as \ x_2 = 0 \, and \ |A(x_3 - x_4)| \leq K_s(x_1) \\ K_c(x_1)sign(x_2) \\ \quad as \ x_2 \neq 0 \, or \ |A(x_3 - x_4)| > K_s(x_1) \end{cases} \tag{24}$$

where A denotes the piston area of the cylinder; $K_s(x_1)$ indicates the position-dependent static friction forces, and $K_c(x_1)$ is the variable position-dependent load due to friction.

4.2. Input–Output Feedback Linearization

According to [36], an arbitrary nonlinear single-input single-output system can be linearized by differentiating its output. By applying feedback linearization theory to an RPA and neglecting the static frictional forces in Equation (20), the system can be expressed as:

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}) + g(\mathbf{x})u \\ y &= h(\mathbf{x}) = x_1 \end{aligned} \tag{25}$$

where the state vector $\mathbf{x} \equiv [x_1 \, x_2 \, x_3 \, x_4]^T$ and u are, respectively, defined as the spool displacement of PDCV and the control input, and the corresponding vector fields $f(\mathbf{x})$ and $g(\mathbf{x})$ are described as:

$$f(\mathbf{x}) = \begin{bmatrix} x_2 \\ \frac{1}{M}\left[-K_f x_2 + A(x_3 - x_4)\right] \\ \frac{-k(x_2 x_3)}{x_1 + \Delta} \\ \frac{k(x_2 x_4)}{l - x_1 + \Delta} \end{bmatrix} \tag{26}$$

and

$$g(\mathbf{x}) = \begin{bmatrix} 0 \\ 0 \\ \frac{kRT_S C_d C_0 w f(x_3, P_s, P_e)}{A(x_1 + \Delta)} \\ \frac{-kRT_S C_d C_0 w f(x_4, P_s, P_e)}{A(l - x_1 + \Delta)} \end{bmatrix} \tag{27}$$

where $f(\mathbf{x})$ and $g(\mathbf{x})$ are partially unknown and smooth vector functions. After linearization, Equation (20) becomes:

$$y^{(3)}(t) = F(\mathbf{x}) + G(\mathbf{x})u + d(\mathbf{x}) \tag{28}$$

where

$$F(\mathbf{x}) = \frac{K_f^2 x_2 - K_f A(x_3 - x_4)}{M^2} - \frac{Akx_2[x_3(l - x_1 + \Delta) + x_4(x_1 + \Delta)]}{M(x_1 + \Delta)(l - x_1 + \Delta)} \tag{29}$$

$$G(\mathbf{x}) = \frac{kRT_s C_d C_0 w[\hat{f}(x_3, P_s, P_e)(l - x_1 + \Delta) + \hat{f}(x_4, P_s, P_e)(x_1 + \Delta)]}{M(x_1 + \Delta)(l - x_1 + \Delta)} \tag{30}$$

$\mathbf{x}(t) = \begin{bmatrix} y(t) & \dot{y}(t) & \ddot{y}(t) \end{bmatrix}^T \in R^3$ is the state vector; $u \in R$ and $y \in R$ are the control input and the system output, respectively; $d(\mathbf{x})$ denotes the external disturbance and the unmodeled friction force of the piston. It is assumed that $|d(\mathbf{x})| \leq D$ for all states $\mathbf{x}(t)$, and $F(\mathbf{x})$ and $G(\mathbf{x})$ are partially unknown functions with uncertain time-varying parameters. Without loss of generality, $G(\mathbf{x})$ can be assumed to be strictly positive. Evidently, an additional disturbance compensator is necessary to take account of the lump of disturbances.

4.3. Development of Control Strategy AIT2FC-STFSMC

As to the control strategy, the proposed AIT2FC-STFSMC is to attenuate disturbances and track trajectories for the RPA-driven 3-DOF TPM with high nonlinearity and time variation. Figure 5 shows the relationship between the AIT2FC and the STFSMC. In the AIT2FC-STFSMC, the adaptive interval type-2 fuzzy controller (AIT2FC) is designed as a trajectories tracking controller, where the interval type-2 fuzzy system is used to mimic an ideal controller. However, an approximation error may happen when using the AIT2FC. Hence, the self-tuning fuzzy sliding-mode compensator (STFSMC) is derived to compensate for the difference between the ideal controller and AIT2FC, as well as external disturbances. The AIT2FC is able to automatically adjust the fuzzy rules and reduce the number of fuzzy rules. Nevertheless, the boundary of approximation error is very difficult to measure in industrial applications. A large pre-set boundary will lead to a large chattering phenomenon in control output and will wear the bearing mechanism and excite unstable dynamics. Otherwise, a small boundary may make the controlled system unstable. To overcome the above-mentioned problem, a simple estimation algorithm is investigated to observe the boundary of approximation error in real-time. According to the on-line adjustment for the boundary, the chattering in control output can be much reduced.

The reference signals are defined as $\mathbf{x}_d(t) = \begin{bmatrix} y_d(t) & \dot{y}_d(t) & \cdots & y_d^{(n-1)}(t) \end{bmatrix}^T$, so the tracking error vector is expressed as:

$$\mathbf{e}(t) = \mathbf{x}_d(t) - \mathbf{x}(t) = [e(t) \ \dot{e}(t) \ldots e^{(n-1)}(t)]^T \tag{31}$$

The sliding surface is:

$$S(t) = c_1 e(t) + c_2 \dot{e}(t) + \ldots + e^{(n-1)}(t) \tag{32}$$

where c_i is specified such that $\sum_{i=1}^{n} c_i \lambda^{i-1}$ is a Hurwitz polynomial and λ is a Laplace operator. If the functions $F(\mathbf{x})$ and $G(\mathbf{x})$ in Equation (28) are known and the external disturbance $d(\mathbf{x})$ is measurable, the ideal control law is derived as:

$$u^* = \frac{1}{G(\mathbf{x})}[\eta S_\Delta(t) + \Lambda_s^T \mathbf{e}(t) - F(\mathbf{x}) - d(\mathbf{x}) + y_d^{(n)}] \tag{33}$$

where $\eta > 0$ is a constant; $\Lambda_s = [0, c_1, c_2, \cdots, c_{n-2}]^T$ is the constant vector, and $S_\Delta(t) = S - \Phi sat(S/\Phi)$, for which $\Phi \geq 0$ is the width of the boundary layer of the sliding surface S. The properties of the function S_Δ are described as follows:

Property 1: As $|S(t)| > \Phi$ then $|S_\Delta(t)| = |S(t)| - \Phi$ and $\dot{S}_\Delta(t) = \dot{S}(t)$.

Property 2: As $|S(t)| \leq \Phi$ then $S_\Delta(t) = \dot{S}_\Delta(t) = 0$.

These properties of the boundary layer are applied in the design of the controller, such that the adaptation terminates as soon as the boundary layer is reached to avoid the possibility of unbounded growth. Differentiating Equation (32) gives:

$$\dot{S}(t) = \mathbf{\Lambda}_s^T \mathbf{e}(t) + x_d^{(n)} - F(\mathbf{x}) - G(\mathbf{x})u - d(\mathbf{x}) \tag{34}$$

Substituting Equation (33) into Equation (34) gives:

$$\dot{S}(t) + \eta S_\Delta(t) = 0 \tag{35}$$

The convergence for $\mathbf{e}(t) = [e(t) \; \dot{e}(t) \ldots e^{(n-1)}(t)]^T$ in Equation (35) can be achieved for $\eta > 0$. However, some variables in Equations (29) and (30) may be unknown or perturbed, and $d(\mathbf{x})$ may not be measurable. Thus, it is difficult to obtain a precise model for the functions $F(\mathbf{x})$ and $G(\mathbf{x})$, and the implementation of the ideal control law u^* is also impossible for the RPA. In this regard, the AIT2FC yields $\hat{u}_{fz}$ to approximate the ideal control law and combines the STFSMC with $u_{comp}(S)$ to compensate for the disturbance and modeling error. The proposed control law is:

$$u = \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}}) + u_{comp}(S) \tag{36}$$

4.4. Design of the AIT2FC

In this section, a single-input AIT2FC is used to formulate the control law $\hat{u}_{fz}$. For the AIT2FC, the ith fuzzy rule is:

$$R^i : if \;\; S \;\; is \;\; \widetilde{F}_{T_2S}^i \;\; then \;\; \hat{u}_{fz} \;\; is \;\; \hat{\alpha}_{T_2fz}^i, \; (i = 1, \ldots, M) \tag{37}$$

where S is the input variable; $\widetilde{F}_{T_2S}^i$ is an interval type-2 fuzzy set; $\hat{\alpha}_{T_2fz}^i$ is an interval type-2 singleton fuzzy set, and M is the number of rules. Using singleton fuzzification, product inference, and center-average defuzzification, the output of the AIT2FC is:

$$\hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}}) = \frac{y_l + y_r}{2} = \frac{1}{2} \begin{bmatrix} \hat{\boldsymbol{\alpha}}_l^T & \hat{\boldsymbol{\alpha}}_r^T \end{bmatrix} \begin{bmatrix} \xi_l \\ \xi_r \end{bmatrix} = \hat{\boldsymbol{\alpha}}^T \xi, \tag{38}$$

where y_l and y_r, respectively, represent the farthest left and the farthest right points of the interval type-2 set. In (38), the weight vector $\hat{\boldsymbol{\alpha}}^T = [\hat{\alpha}_1, \hat{\alpha}_2, \cdots, \hat{\alpha}_M]$ is used to estimate the optimal weight vector $\boldsymbol{\alpha}^{*T} = [\alpha_1^*, \alpha_2^*, \cdots, \alpha_M^*]$ and the parameter $\hat{\boldsymbol{\alpha}}^{*T}$ is reasonably assumed to be bounded. The farthest left point of the interval type-2 set is defined as:

$$y_l = \frac{\sum\limits_{i=1}^{L} \overline{\mu}_{F_{T_2S}^i}(S)\hat{\alpha}_l^i + \sum\limits_{i=L+1}^{M} \underline{\mu}_{F_{T_2S}^i}(S)\hat{\alpha}_l^i}{\sum\limits_{i=1}^{L} \overline{\mu}_{F_{T_2S}^i}(S) + \sum\limits_{i=L+1}^{M} \underline{\mu}_{F_{T_2S}^i}(S)} = \sum\limits_{i=1}^{L} \overline{p}_l^i \hat{\alpha}_l^i + \sum\limits_{i=L+1}^{M} \underline{p}_l^i \hat{\alpha}_l^i$$

$$= \begin{bmatrix} \overline{\hat{\boldsymbol{\alpha}}}_l^T & \underline{\hat{\boldsymbol{\alpha}}}_l^T \end{bmatrix} \begin{bmatrix} \overline{p}_l \\ \underline{p}_l \end{bmatrix} = \hat{\boldsymbol{\alpha}}_l^T \xi_l \tag{39}$$

where $\overline{\mu}_{F_{T_2S}^i}$ and $\underline{\mu}_{F_{T_2S}^i}$, respectively, represent the upper and lower degrees of the membership function.

$\overline{p}_l^i = \overline{\mu}_{F_{T_2S}^i}(S)/W_l$, and $\underline{p}_l^i = \underline{\mu}_{F_{T_2S}^i}(S)/W_l$, in which $W_l = \sum\limits_{i=1}^{L} \overline{\mu}_{F_{T_2S}^i}(s) + \sum\limits_{i=L+1}^{M} \underline{\mu}_{F_{T_2S}^i}(s)$. The farthest right point of the interval type-2 set is defined as:

$$y_r = \frac{\sum\limits_{i=1}^{R} \overline{\mu}_{F^i_{T_2 S}}(S)\hat{\alpha}^i_r + \sum\limits_{i=R+1}^{M} \underline{\mu}_{F^i_{T_2 S}}(S)\hat{\alpha}^i_r}{\sum\limits_{i=1}^{R} \overline{\mu}_{F^i_{T_2 S}}(S) + \sum\limits_{i=R+1}^{M} \underline{\mu}_{F^i_{T_2 S}}(S)} = \sum_{i=1}^{R} \overline{p}^i_r \hat{\alpha}^i_r + \sum_{i=R+1}^{M} \underline{p}^i_r \hat{\alpha}^i_r$$

$$= \begin{bmatrix} \overline{\alpha}^T_r & \underline{\hat{\alpha}}^T_r \end{bmatrix} \begin{bmatrix} \overline{p}_r \\ \underline{p}_r \end{bmatrix} = \hat{\alpha}^T_r \xi_r, \tag{40}$$

where $\hat{\alpha}^i_r$ is farthest right point of $\hat{\alpha}^i_{T_2 f_z}$, $\overline{p}^i_r = \overline{\mu}_{F^i_{T_2 S}}(S)/W_r$, and $\underline{p}^i_r = \underline{\mu}_{F^i_{T_2 S}}(S)/W_r$, in which

$W_r = \sum\limits_{i=1}^{R} \overline{\mu}_{F^i_{T_2 S}}(s) + \sum\limits_{i=R+1}^{M} \underline{\mu}_{F^i_{T_2 S}}(s)$. The parameters L and R in (39) and (40), respectively, are calculated by using type reduction [37]. The adaptive law for the AIT2FC is given as:

$$\dot{\hat{\alpha}} = \eta_1 \cdot S_\Delta \cdot \xi, \tag{41}$$

where $\eta_1 > 0$ is the adaptive learning rate.

4.5. Design of the STFSMC

The design objective of STFSMC is to compensate for the disturbances [38]. The sliding surface S is again specified as the input and the sliding control law u_{fs} is the output. Figure 6 shows the membership functions and the associated linguistic variables as follows:

$$T(S) = \{NB, NM, NS, ZO, PS, PM, PB\} = \{f^1_s, f^2_s, f^3_s, f^4_s, f^5_s, f^6_s, f^7_s\},$$

$$T(u_{fs}) = \{NB, NM, NS, ZO, PS, PM, PB\} = \{f^1_u, f^2_u, f^3_u, f^4_u, f^5_u, f^6_u, f^7_u\}, \tag{42}$$

The fuzzy rules are simply expressed as:

$$R^l : if \ S \ is \ f^l_s \ then \ u_{fs} \ is \ f^{8-l}_u, \ l = 1, \dots, 7 \tag{43}$$

By using singleton fuzzification, max-min inference, and center-average defuzzification, the output of STFSMC is:

$$u_{fs} = \frac{\int_{-1}^{1} u_{fs} F(u_{fs}) du_{fs}}{\int_{-1}^{1} F(u_{fs}) du_{fs}} \tag{44}$$

where $F(u_{fs}) = \bigvee\limits_{l=1}^{7} \left[f^l_s(S) \wedge f^{8-l}_u(u_{fs}) \right]$. To avoid substantial computational cost for the general fuzzy control algorithm, u_{fs} can be calculated for two cases:

Case 1: $k = S/\Phi < 0$:

$$u_{fs} = \begin{cases} 1 & , if \quad k < -1 \\ \frac{7.5k^2 + 13.5k + 5}{9k^2 + 15k + 5} & , if \quad -1 \le k < -\frac{2}{3} \\ \frac{9k^2 + 11k + 2}{18k^2 + 18k + 2} & , if \quad -\frac{2}{3} \le k < -\frac{1}{3} \\ \frac{1.5k^2 + 1.5k}{9k^2 + 3k - 1} & , if \quad -\frac{1}{3} \le k < 0 \end{cases} \tag{45}$$

Case 2: $k = S/\Phi \ge 0$:

$$u_{fs} = \begin{cases} \frac{-1.5k^2 + 1.5k}{9k^2 - 3k - 1} & , if \quad 0 \le k < \frac{1}{3} \\ \frac{-9k^2 + 11k - 2}{18k^2 - 18k + 2} & , if \quad \frac{1}{3} \le k < \frac{2}{3} \\ \frac{-7.5k^2 + 13.5k - 5}{9k^2 - 15k + 5} & , if \quad \frac{2}{3} \le k < 1 \\ -1 & , if \quad k \ge 1 \end{cases} \tag{46}$$

where $\Phi > 0$ is given to denote the width of the boundary layer. It can be verified that $u_{fs} = -\text{sgn}(S)$ as $|S| \geq \Phi$, and, thus, the STFSMC is designed as:

$$u_{comp}(S) = -(k_c + \hat{\rho})u_{fs} \tag{47}$$

where k_c is a compensation gain, which is:

$$k_c = \frac{M_2(\mathbf{x})|S_\Delta(t)|}{2M_0^2(\mathbf{x})} \tag{48}$$

where $M_0(\mathbf{x})$ and $M_2(\mathbf{x})$ are specified variables. $\hat{\rho}$ is an adaptive compensation gain [38] that is represented as:

$$\dot{\hat{\rho}} = \eta_2 \cdot |S_\Delta| \tag{49}$$

where $\eta_2 > 0$ is a learning rate that is greater than zero. Figure 5 illustrates the overall system for the proposed AIT2FC-STFSMC. For the AIT2FC, the parameter g_s is used to ensure that the sliding surface S is within the range of the fuzzy input and the gain factor g_u is used to regulate the fuzzy output $\hat{u}_{fz}$.
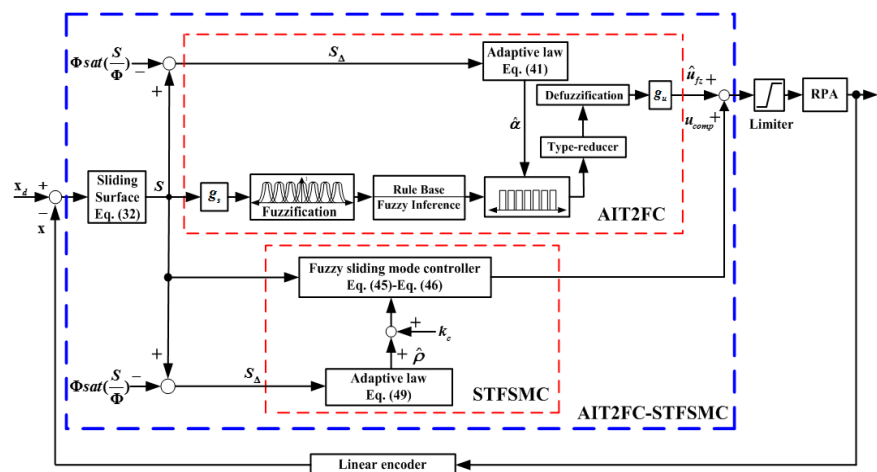

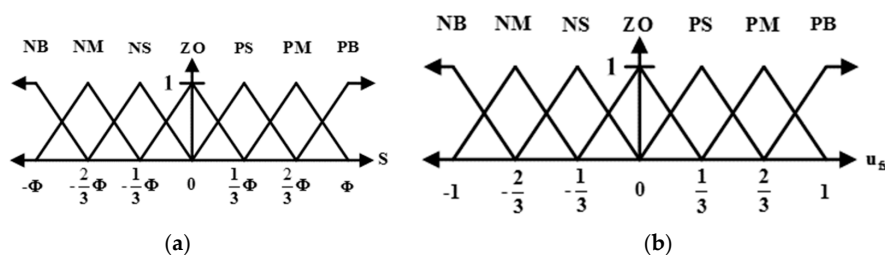**Figure 5.** Overall system for the RPA-driven 3-DOF TPM.

Figure 6. Membership functions for the fuzzy input S and the fuzzy output. (**a**) membership functions for the fuzzy input S; (**b**) membership functions for The fuzzy output u_{fs}.

Assumption 1. *The control signal u is linear with respect to $y^{(n)}$ and $G(\mathbf{x}) \neq 0$ for x in the controllable region U_c.*

Assumption 2. *The internal dynamics of each RPA system are stable under the influence of the AIT2FC-STFSMC.*

Assumption 3. *The boundary conditions, $0 < M_0(\mathbf{x}) \leq G(\mathbf{x}) \leq M_1(\mathbf{x})$ and $\left|\dot{G}(\mathbf{x})\right| \leq M_2(\mathbf{x})$, hold, where $M_0(\mathbf{x})$, $M_1(\mathbf{x})$, and $M_2(\mathbf{x})$ are known functions.*

It is noted that Assumption 1 is needed for the design of the AIT2FC, Assumption 2 is needed for the both of the AIT2FC and the STFSMC, and Assumption 3 is needed for the design of the STFSMC.

Theorem 1. *If the RPA is represented as the form of an affine system (28) that satisfies Assumptions 1, 2, and 3. By using the control law (36), where $\hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}})$ represents the AIT2FC (see Equation (38)) and $u_{comp}(S)$ represents the STFSMC (See Equation (47)), and the adaptive laws (41) and (49), it is observed that (i) the system state* **x** *and the control law u are bounded and (ii) the tracking errors converge to 0 as $t \to \infty$.*

Proof. See Appendix A. □

5. Robot Vision for 3-DOF TPM

5.1. Image Capturing

In this study, images of an object were captured by a webcam and then transferred to LabVIEW through an API with image processing toolkits. The API *IMAQdx Open Camera* is used to open a video source in the RGB color model, in which the *Property Node* is a function applied to set the image resolution and the frames per second. *IMAQ Create* is an API applied to create a buffer for temporarily storing images. The images in the buffer are grabbed by the API *IMAQdx Grab* as requested.

5.2. Define a Template Pattern

To define a template pattern, the captured image for an object in RGB is first converted to grayscale by color transformation, which is represented as:

$$Y = 0.299R + 0.587G + 0.114B, \tag{50}$$

A mask for the object is obtained by manually drawing the minimum rectangular region that contains the object. The region is termed the region of interest (ROI). The API *IMAQ ConstructROI VI* extracts image features within the ROI and creates the template pattern. The template pattern and its features for the targeted object are stored in a database.

5.3. Pattern Recognition

In this paper, pattern recognition is applied to determine the location of an object that has the same image features as the template pattern. Once activated, the pattern recognition system captures an image and uses the *NI IMAQ Vision* API to search for patterns similar to the template over the input image by using block matching. Figure 7 indicates the processes of block matching, in which $f(x, y)$ is the greyscale of the image f with the dimension $M \times N$ and $w(x, y)$ is the greyscale of the image $w(x, y)$ with the dimension $K \times L$. It is noted that $M \geq K$ and $N \geq L$. Block matching searches similar patterns from the pixel coordinate $(1, 1)$ to $(M - K, N - L)$ with a $K \times L$ dimension block, $w(x, y)$, and calculates the relationship $C(i, j)$ between $f(i, j)$ and $w(i, j)$ by Equation (51), which is:

$$C(i, j) = \sum_{x=0}^{L-1} \sum_{y=1}^{K-1} w(x, y) f(x + i, y + j) \tag{51}$$

After searching is conducted, the pattern with the biggest value C is considered a candidate pattern. After that, its correlation is defined as:

$$R(i, j) = \frac{\sum\limits_{x=0}^{L-1} \sum\limits_{y=0}^{K-1} [w(x, y) - \overline{w}][f(x + i, y + j) - f(i, j)]}{\sqrt{\sum\limits_{x=0}^{L-1} \sum\limits_{y=0}^{K-1} [w(x, y) - \overline{w}]^2 \sum\limits_{x=0}^{L-1} \sum\limits_{y=0}^{K-1} [f(x + i, y + j) - f(i, j)]^2}} \tag{52}$$

The pattern is defined as the best matching pattern, especially if $-1 < R(i, j) < 1$. Image recognition is implemented by LabVIEW with three APIs as follows: (1) the *IMAQ Read Image and Vision Info VI* API calls the built template pattern; (2) The *IMAQ Match Pattern* API searches

for the matches with respect to the preset matching parameters; (3) The *Unbundle by Name Function* API calculates the centroid of the optimal matching pattern, which represents the entirety of the optimal matching pattern with respect to the image.
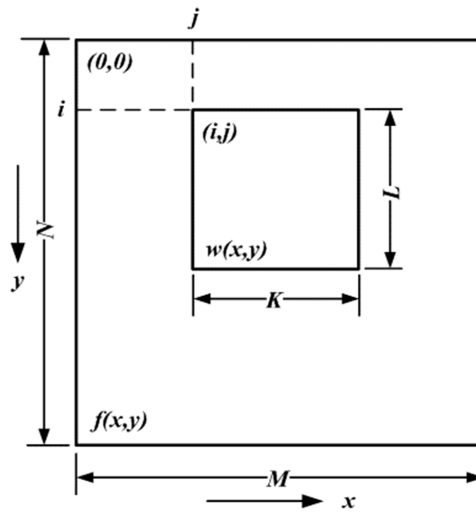

Figure 7. Block matching process.

5.4. Spatial Calibration

Spatial calibration converts a pixel coordinate to a real-world coordinate. In the meantime, two APIs can be applied to compensate for the potential perspective errors and nonlinear distortions in the image, as introduced as follows:

(1)   The *IMAQ Calibration Target to Points—Circular Dots* API detects circular dots in a binary image and returns pixels and real-world points for calibration;
(2)   The *IMAQ Get Calibration Info* API returns calibration information associated with the image.

After spatial calibration is performed, one can identify the coordinate of the target object in the real world.

6. Experiments and Discussion

The RPA is regulated by the PDCV. Instead of using acceleration sensors, in this study the velocity and acceleration are estimated by numerically differentiating the position and the velocity with respect to time. To reduce the signal disturbances during the numerical difference operation, a digital filter is used as follows:

$$y_{out}(i) = 0.7y_{out}(i-1) + 0.15[y_{in}(i) + y_{in}(i-1)] \tag{53}$$

where $y_{out}(t)$ represents the filtered signal and $y_{in}(t)$ is the input data from the position measurement of the piston. The input voltage of the PDCV is applied as the control signal. The membership functions for S and u_{f_s} are displayed in Figure 5.

6.1. Experiment 1(Trajectory Tracking—Square Trajectory)

The square trajectory is often used in many practical applications, such as the pick-and-place operation. It can be segmented by five segments and has four positioning points at the vertex of the square. The profile and the moving direction of the square trajectory are illustrated in Figure 8. Segment 1 is modeled as a fifth-order trajectory [27,33], and segments 2 to 5 are modeled as a line. In the experiment, at the beginning, the end-effector moves from the initial position $(0, 0, -66.6\,\text{cm})$ to $(-15\,\text{cm}, 15\,\text{cm}, -60\,\text{cm})$ in 3 s. Then, the end-effector moves along the square loop path with the edge length of 30 cm and back to the positioning point $(-15\,\text{cm}, 15\,\text{cm}, -60\,\text{cm})$ in 6 s. To effectively control

the RPAs, the trajectories for each RPA are also modeled as a fifth-order polynomial function [27,33] with zero initial and final velocities as well as zero initial and final accelerations.

The experimental results of the RPA response for each axis are shown in Figures 9–11. The estimated end-effector position calculated from the forward kinematics is shown in Figure 12a,b. The estimated position error of the end-effector is calculated from the position error of the actuated joint, as shown in Figure 12c. The root-mean-square error (RMSE) of the path tracking control for the cylinder A, B, and C axes are 0.214, 0.203, and 0.173 cm, respectively. The maximum estimated error of the end-effector position is approximately 0.6097 at 3.37 s.
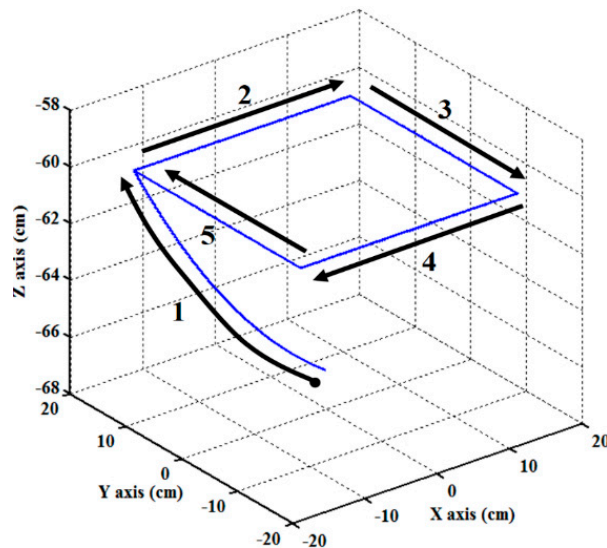

Figure 8. Designed square loop trajectory for the end-effector path tracking control experiment.
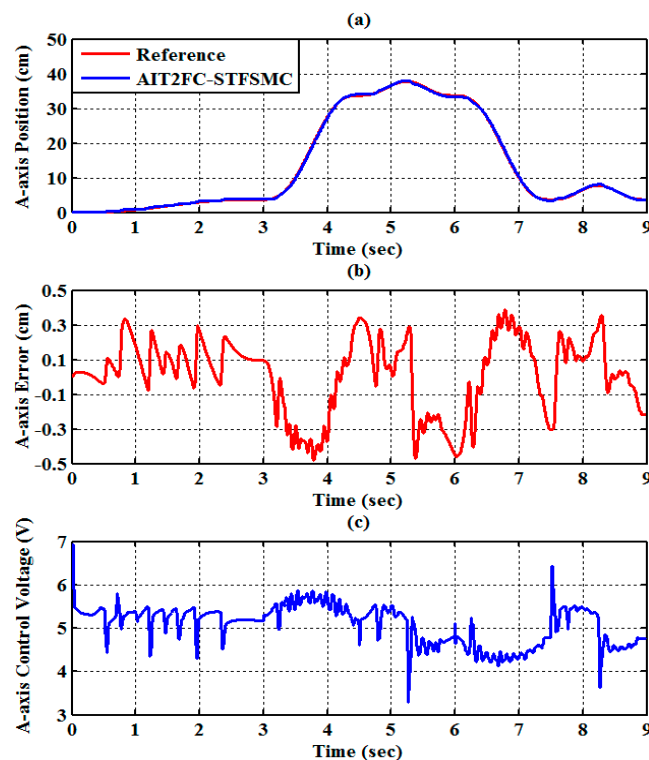

Figure 9. Experimental results of end-effector path tracking for an A-axis RPA via a square loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
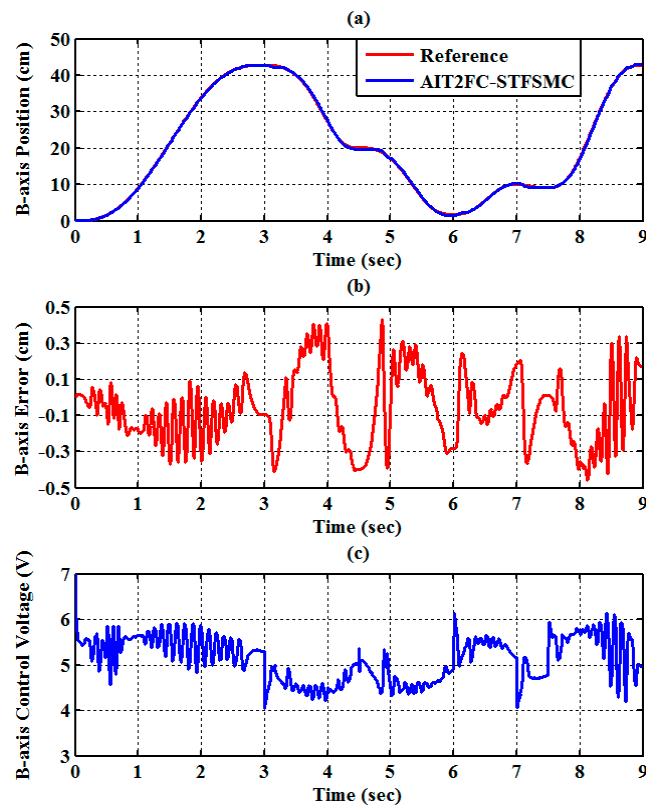
Figure 10. Experimental results of end-effector path tracking for a B-axis RPA via a square loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
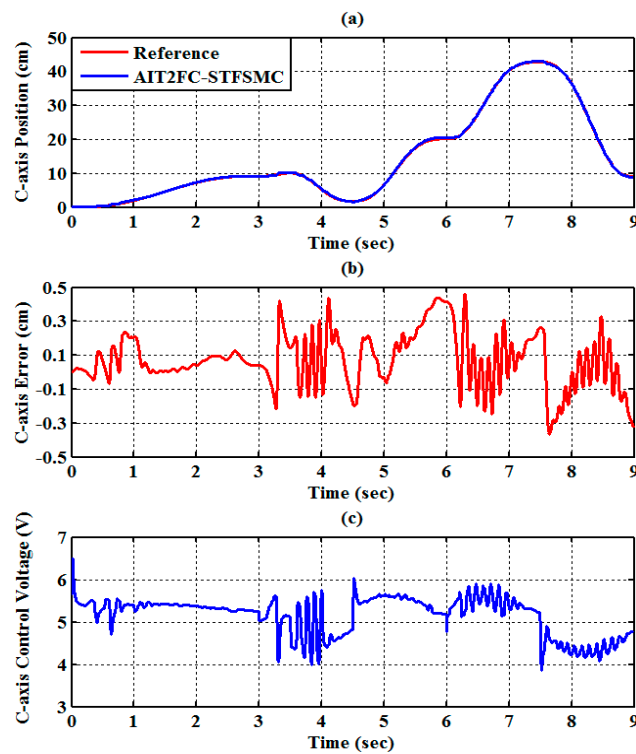

Figure 11. Experimental results of end-effector path tracking for a C-axis RPA via a square loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
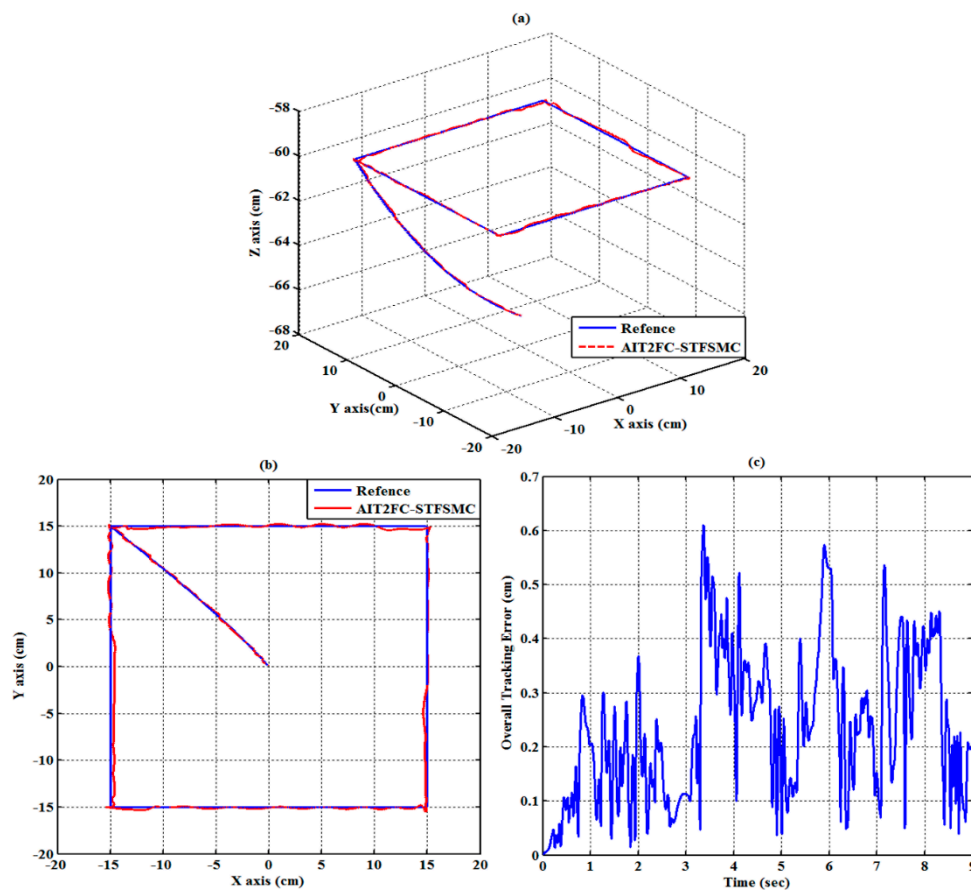
Figure 12. Estimated end-effector position tracking response for the square loop trajectory (**a**) calculated end-effector position (front view), (**b**) calculated end-effector position (top view), and (**c**) calculated end-effector position error.

6.2. Experiment 2 (Trajectory Tracking-Star Trajectory)

In Experiment 2, a star motion is set as a reference 3D motion trajectory. The star trajectory is composed of six segments. The profile and moving direction of the designed trajectory are illustrated in Figure 13. Segment 1 is modeled as a fifth-order trajectory [27,33] from the initial position $(0, 0, -66.6\,\text{cm})$ to $(10\,\text{cm}, 0\,\text{cm}, -60\,\text{cm})$, and segments 2 to 6 are modeled as a straight line. At the beginning, the end-effector moves along segment 1 in 3 s. Then, the end-effector moves sequentially along segment 2 to segment 6, where segment 2 is a straight line from P_1 $(10\,\text{cm}, 0\,\text{cm}, -60\,\text{cm})$ to P_2 $(-8.1\,\text{cm}, 5.9\,\text{cm}, -60\,\text{cm})$, segment 3 is a straight line from P_2 $(-8.1\,\text{cm}, 5.9\,\text{cm}, -60\,\text{cm})$ to P_3 $(3.1\,\text{cm}, -9.5\,\text{cm}, -60\,\text{cm})$, segment 4 is a straight line from P_3 $(3.1\,\text{cm}, -9.5\,\text{cm}, -60\,\text{cm})$ to P_4 $(3.1\,\text{cm}, 9.5\,\text{cm}, -60\,\text{cm})$, segment 5 is a straight line from P_4 $(3.1\,\text{cm}, 9.5\,\text{cm}, -60\,\text{cm})$ to P_5 $(-8.1\,\text{cm}, -5.9\,\text{cm}, -60\,\text{cm})$, and segment 6 is a straight line from P_5 $(-8.1\,\text{cm}, -5.9\,\text{cm}, -60\,\text{cm})$ to P_1 $(10\,\text{cm}, 0\,\text{cm}, -60\,\text{cm})$. The traveling duration for each of segments 2 to 6 is set to 2 s. To effectively control the RPAs, the trajectories for each RPA are also modeled as a fifth-order polynomial function [27,33] with zero initial and final velocities as well as zero initial and final accelerations. The experimental results of the RPA response for each axis are presented in Figures 14–16. The estimated end-effector position calculated from the forward kinematics is depicted in Figure 17a,b The estimated position error of the end-effector is calculated from the position error of the actuated joint and illustrated in Figure 17c. The RMSE values on the path tracking control of cylinders of A, B, and C axes are 0.176 cm, 0.189 cm, and 0.142 cm, respectively. The maximum estimated error of the end-effector position is approximately 0.4833 cm at 3.87 s.
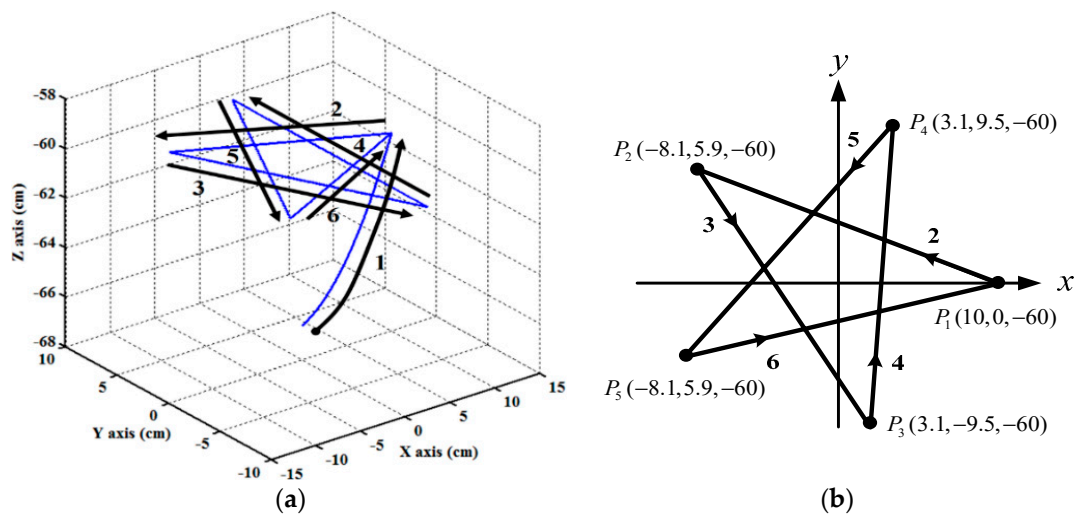
Figure 13. Designed star loop trajectory for the end-effector path tracking control experiment. (**a**) 3D star-motion trajectory; (**b**) top-view of the 3D star-motion trajectory.
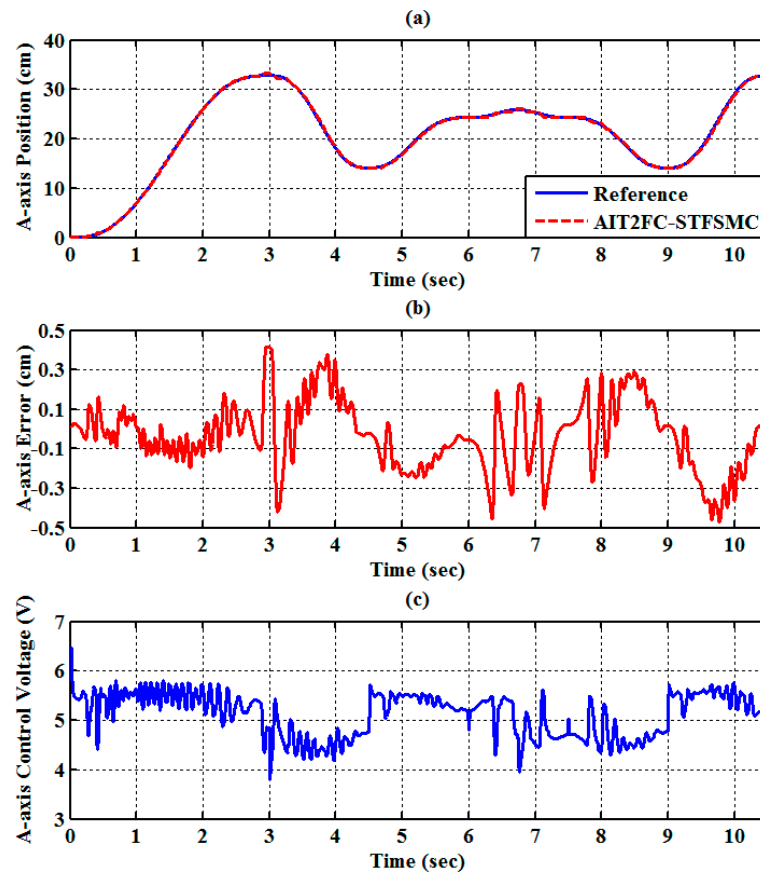

Figure 14. Experimental results of end-effector path tracking for the A-axis RPA via a star loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
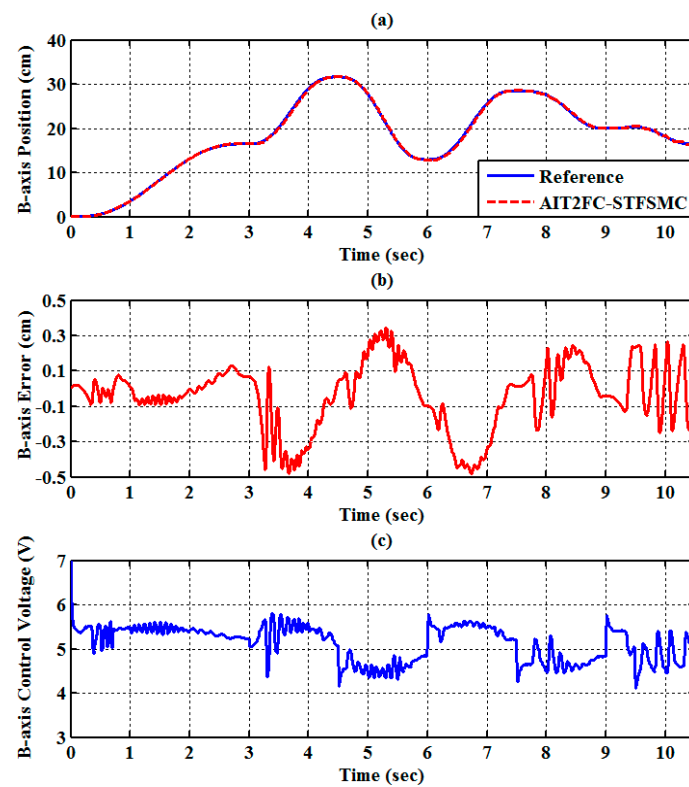
Figure 15. Experimental results of end-effector path tracking for the B-axis RPA via a star loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
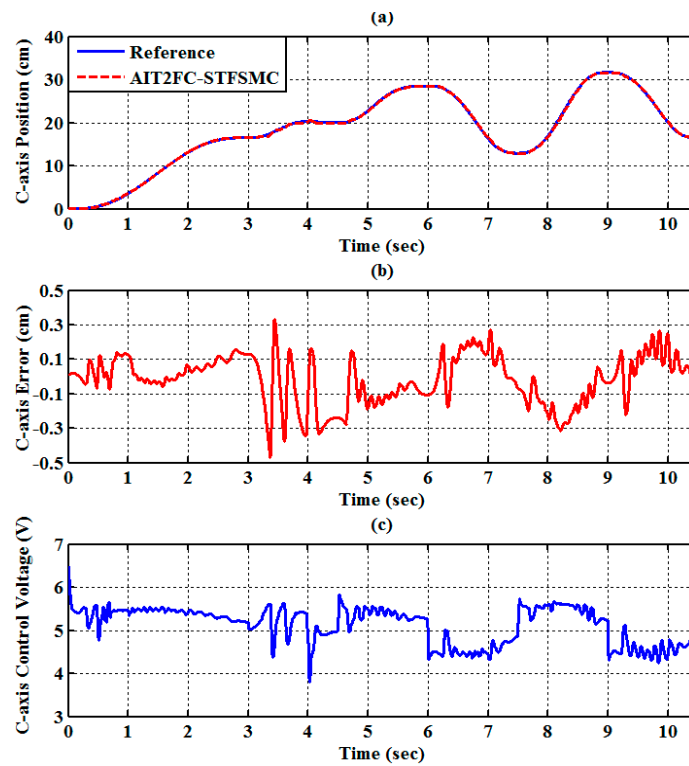
Figure 16. Experimental results of end-effector path tracking for the C-axis RPA via a star loop trajectory (**a**) tracking response, (**b**) tracking error, and (**c**) control voltage.
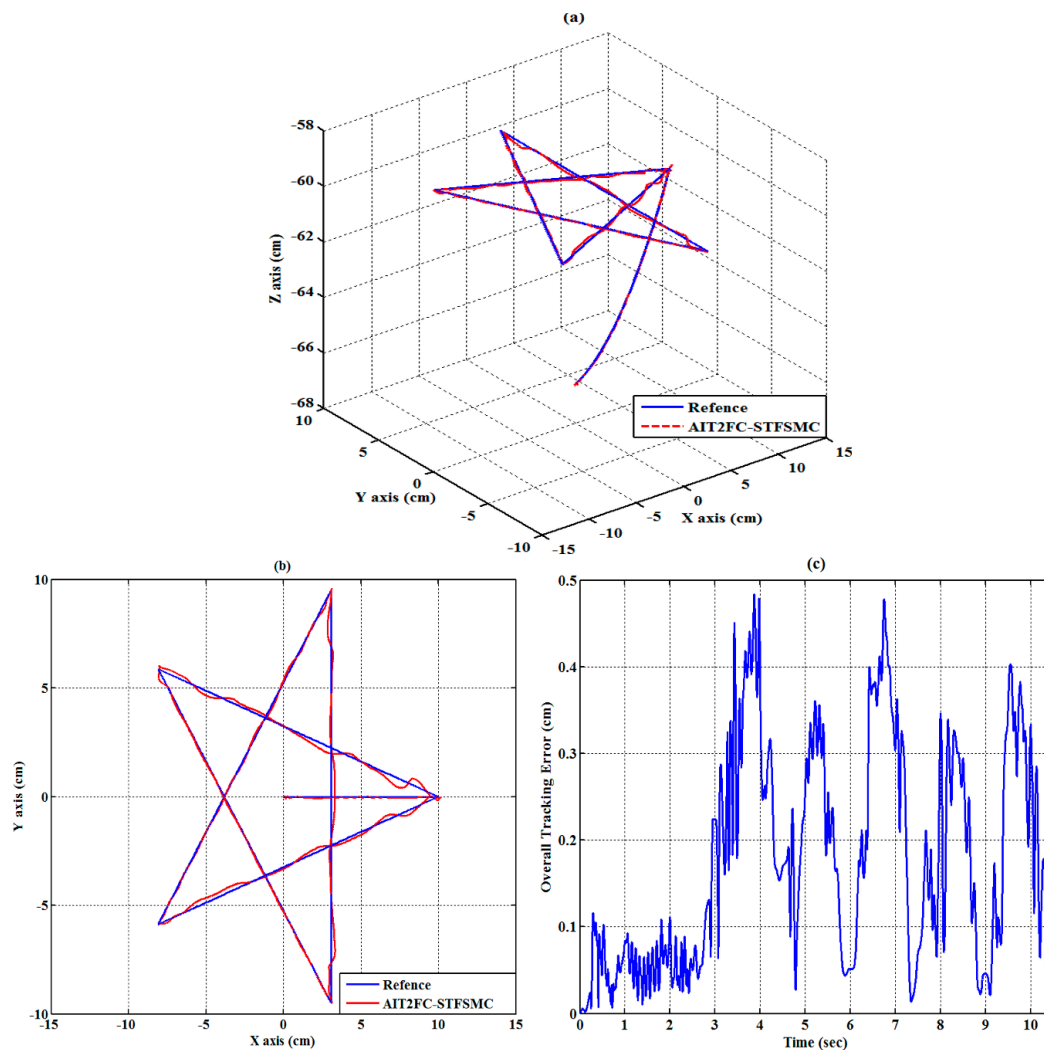
Figure 17. Estimated end-effector position tracking response for the star loop trajectory (**a**) calculated end-effector position (front view), (**b**) calculated end-effector position (top view), and (**c**) calculated end-effector position error.

6.3. Experiment 3 (Robot Vision)

6.3.1. Stationary Object Localization

This experiment measured the accuracy of the object-located technique for a stationary object. In the first experiment, a hexagonal object was placed on the conveyor belt, and the power was turned off. By using the presented robot vision after repeating this experiment 10 times, the average, variation, and maximum of the error for the x-coordinate, y-coordinate, and angle were discovered and are shown in Table 1. The variations of the x-coordinate and y-coordinate were less than 0.03 cm, and the variation of the angle was less than $0.25°$, which are within the position tolerance error for the robot arm. Table 1 shows the measuring accuracy of the visual camera. The maximum error of the x-coordinate and y-coordinate are, respectively, around 0.05 cm and 0.08 cm, as well as the maximum error of the angle which was around $0.09°$. In this paper, the spatial calibration is applied to convert a pixel coordinate to a real-world coordinate. Clearly, the measuring error in the pixel coordinate will produce an error in the real-world coordinate after coordinate transformation. Furthermore, the error in the real-world coordinate will transfer to the motion control for the RPAs. In this paper, a soft pneumatic gripper installed on the end-effector is soft and triangular with hard crossbeams, so that it can buckle and deform in to conform around an object. It is able to easily grip an object with the radius between the

interval of [35 mm 65 mm]. That is, if the motion-error, which may come from the measuring error, happens in trajectory tracking control for the RPAs, the soft pneumatic gripper can still successfully pick the object up. In our design, Experiment 3 shows the vision-based RPA-driven TPM accurately locates the object and successfully executes the pick-and-place operation, as shown in Figure 18.

Table 1. Results of the robot vision test for the stationary object.

Error Terms / Items	*x*-Coordinate (cm)	*y*-Coordinate (cm)	Angle (degree)
Max. of Error	0.051186	0.078865	0.085908
Var. of Error	0.014741	0.022723	0.024787

6.3.2. Pick-and-Place Operation

In this experiment, the conveyor belt moved at a fixed speed of 2.1 cm/s along the y-axis of the image. The conveyor belt conveyed a targeted object at a speed of 2.1 cm/s. The proposed robot-vision system estimated the location of the moving object by calculating the velocity formula. In the physical test, the estimated error of the location of the moving object was approximately 0.1 cm when the conveyor belt moved the object 20 cm along the y-axis of the image. Figure 18 illustrates a pick-and-place experiment. Figure 18a illustrates an object on the conveyor belt. After the power was connected to the conveyor belt, the conveyor belt conveyed the object along the *y*-axis of the image. The image of the moving object was captured by the webcam. Figure 18b depicts the robot vision system locating the object and the PM picking it up. The PM moved the object to the desired location, as illustrated in Figure 18c,d.
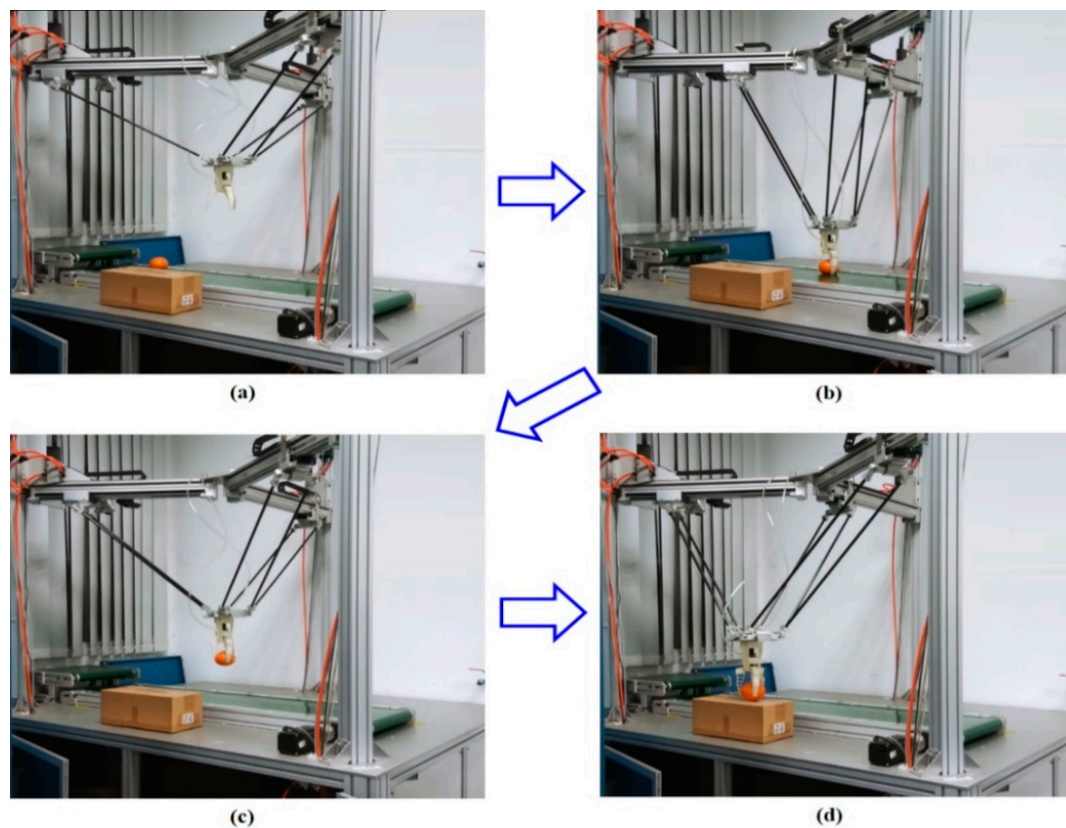

Figure 18. Pick-and-place experiment. (**a**) an object is on the conveyor belt; (**b**) the robot vision system locates the object and the PM picks it up; (**c**) the PM moves the object to the desired location (**d**) the PM puts the object down to the desired location.

7. Conclusions

This study developed and implemented a vision-based RPA-driven 3-DOF TPM, which not only allows 3D path tracking control in a full-scale test rig but also provides robot vision to locate an object for vision-based operations. The AIT2FC-STFSMC was developed to allow path tracking control for the RPAs, in which the AIT2FC approximates the ideal control law and the STFSMC attenuates disturbances and uncertainties. The system's webcam collected images of objects and transferred them to LabVIEW for image processing. Two types of experiments were conducted to confirm the feasibility of the proposed system. First, the evidence demonstrated that the end-effector of the manipulator successfully tracks the two given complex 3D trajectories under the RMSE of 0.22 cm. Second, two experiments were conducted on the vision-based RPA-driven 3-DOF TPM. The first experiment proved that the robot-vision system accurately located a stationary object, and the second experiment confirmed that the TPM successfully completed the pick-and-place operation for a moving object.

8. Patents

There are two Taiwan utility model patents resulting from the developed PM, which are (1) Pneumatic-Actuator 3-DOF Translational Parallel Manipulator with the patent number of M463182; (2) Pneumatic-Actuator 3-DOF Translational Parallel Manipulator with Computer Vision with the patent number of M470742.

Author Contributions: Conceptualization, L.-W.L. and H.-H.C.; methodology, L.-W.L. and I-H.L.; software, H.-H.C. and I-H.L.; validation, L.-W.L., H.-H.C. and I-H.L.; formal analysis, L.-W.L. and H.-H.C.; investigation, L.-W.L. and H.-H.C.; resources, L.-W.L.; data curation, L.-W.L. and H.-H.C.; writing—original draft preparation, L.-W.L.; writing—review and editing, H.-H.C. and I-H.L.; supervision, I-H.L. and L.-W.L.; project administration, L.-W.L.; funding acquisition, L.-W.L.

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

Appendix A

This Theorem can be proven using two steps. In Step 1, the system state **x** and the adaptive control *u* are bounded when the AIT2FC-STFSMC is used for the RPA. Next, one can calculate that the tracking error will converge to zero asymptotically in Step 2.

Step 1: Substituting Equations (33) and (36) into (34) yields

$$\dot{S}(t) + \eta S_\Delta(t) = G\left[u^* - \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}}) - u_{comp}(S)\right]. \tag{A1}$$

If Assumption 1 holds for Equation (A1), a candidate Lyapunov function is specified as:

$$V(S_\Delta(t), \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) = \frac{1}{2}\frac{S_\Delta^2}{G(\mathbf{x})} + \frac{1}{2\eta_1}\widetilde{\boldsymbol{\alpha}}^T\widetilde{\boldsymbol{\alpha}} + \frac{1}{2\eta_2}\widetilde{\rho}^2, \tag{A2}$$

where $\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}} = \boldsymbol{\alpha}^{*\mathbf{T}} - \hat{\boldsymbol{\alpha}}^{\mathbf{T}}$ and $\widetilde{\rho} = \rho^* - \hat{\rho}$ are, respectively, represented as the approximation errors of the parameter vectors $\widetilde{\boldsymbol{\alpha}}^{*\mathbf{T}}$ and ρ^*. η_1 and η_2 are positive constants. Differentiating Equation (A2) with respect to time yields:

$$\dot{V}(S_\Delta, \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) = \frac{S_\Delta \dot{S}_\Delta}{G(\mathbf{x})} - \frac{\dot{G}(\mathbf{x})S_\Delta^2}{2G^2(\mathbf{x})} + \frac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^T\dot{\widetilde{\boldsymbol{\alpha}}} + \frac{1}{\eta_2}\widetilde{\rho}\dot{\widetilde{\rho}}, \tag{A3}$$

　　　If $|S| \leq \Phi$, then $S_\Delta(t) = \dot{S}_\Delta(t) = 0$. It follows $\dot{V}(S_\Delta, \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) = 0$. If $|S| > \Phi$, then $\dot{S}_\Delta = \dot{S}$. Substituting Equation (A3) with Equation (A1) yields:

$$
\begin{aligned}
\dot{V}(S_\Delta, \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) &= -\eta \frac{S_\Delta^2}{G(\mathbf{x})} - \frac{\dot{G}(\mathbf{x})S_\Delta^2}{2G^2(\mathbf{x})} + S_\Delta[u^* - \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}})] - \tfrac{1}{2}S_\Delta u_{comp}(S) + \tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\widetilde{\boldsymbol{\alpha}}} + \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\widetilde{\rho}} \\
&= -\eta \frac{S_\Delta^2}{G(\mathbf{x})} - \frac{\dot{G}(\mathbf{x})S_\Delta^2}{2G^2(\mathbf{x})} + S_\Delta[u^* - \hat{u}_{fz}(S, \boldsymbol{\alpha}^*) + \hat{u}_{fz}(S, \boldsymbol{\alpha}^*) - \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}})] + S_\Delta k_c u_{fs} + S_\Delta \hat{\rho} u_{fs} - \\
&\tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\hat{\boldsymbol{\alpha}}} - \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\hat{\rho}} \leq -\eta \frac{S_\Delta^2}{M_1(\mathbf{x})} + \frac{M_2(\mathbf{x})S_\Delta^2}{2M_0^2(\mathbf{x})} + |S_\Delta|\left|u^* - \hat{u}_{fz}(S, \boldsymbol{\alpha}^*)\right| + S_\Delta[\hat{u}_{fz}(S, \boldsymbol{\alpha}^*) - \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}})] + \\
&S_\Delta k_c u_{fs} + S_\Delta \hat{\rho} u_{fs} - \tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\hat{\boldsymbol{\alpha}}} - \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\hat{\rho}}.
\end{aligned}
\tag{A4}
$$

Because $S_\Delta u_{fs} = -|S_\Delta|$, Equation (A4) is derived as:

$$
\begin{aligned}
&-\eta \frac{S_\Delta^2}{M_1(\mathbf{x})} + |S_\Delta|[\frac{M_2(\mathbf{x})|S_\Delta|}{2M_0^2(\mathbf{x})} - k_c] + |S_\Delta|\left|u^* - \hat{u}_{fz}(S, \boldsymbol{\alpha}^*)\right| + S_\Delta[\hat{u}_{fz}(S, \boldsymbol{\alpha}^*) - \hat{u}_{fz}(S, \hat{\boldsymbol{\alpha}})] - |S_\Delta|\hat{\rho} - \tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\hat{\boldsymbol{\alpha}}} - \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\hat{\rho}} \\
&= -\eta \frac{S_\Delta^2}{M_1(\mathbf{x})} + |S_\Delta|[\frac{M_2(\mathbf{x})|S_\Delta|}{2M_0^2(\mathbf{x})} - k_c] + |S_\Delta|(\rho^* - \hat{\rho}) + S_\Delta(\boldsymbol{\alpha}^{*\mathbf{T}} - \hat{\boldsymbol{\alpha}}^{\mathbf{T}})\xi - \tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\hat{\boldsymbol{\alpha}}} - \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\hat{\rho}}.
\end{aligned}
\tag{A5}
$$

Then, because $\dot{\widetilde{\boldsymbol{\alpha}}}^{\mathbf{T}} = -\dot{\hat{\boldsymbol{\alpha}}}^{\mathbf{T}}$ and $\dot{\widetilde{\rho}} = -\dot{\hat{\rho}}$, Equation (A5) becomes:

$$
\begin{aligned}
&-\eta \frac{S_\Delta^2}{M_1(\mathbf{x})} + |S_\Delta|[\frac{M_2(\mathbf{x})|S_\Delta|}{2M_0^2(\mathbf{x})} - k_c] + |S_\Delta|\widetilde{\rho} + S_\Delta\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\xi - \tfrac{1}{\eta_1}\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}\dot{\hat{\boldsymbol{\alpha}}} - \tfrac{1}{\eta_2}\widetilde{\rho}\dot{\hat{\rho}} = -\eta \frac{S_\Delta^2}{M_1(\mathbf{x})} + |S_\Delta|[\frac{M_2(\mathbf{x})|S_\Delta|}{2M_0^2(\mathbf{x})} - k_c] + \\
&\widetilde{\boldsymbol{\alpha}}^{\mathbf{T}}(S_\Delta\xi - \tfrac{1}{\eta_1}\dot{\hat{\boldsymbol{\alpha}}}) + \widetilde{\rho}(|S_\Delta| - \tfrac{1}{\eta_2}\dot{\hat{\rho}}).
\end{aligned}
\tag{A6}
$$

Substituting Equation (41) and (47)–(49) into Equation (A6) yields:

$$
\dot{V}(S_\Delta, \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) \leq -(\eta / M_1(\mathbf{x})) \cdot S_\Delta^2.
\tag{A7}
$$

　　　From Equation (A7), $S_\Delta(t)$, $\widetilde{\boldsymbol{\alpha}}(t)$ and $\widetilde{\rho}(t)$ are bounded if the initial values of $S_\Delta(0)$, $\widetilde{\boldsymbol{\alpha}}(0)$ and $\widetilde{\rho}(0)$ are bounded. If $S_\Delta(t)$, $\widetilde{\boldsymbol{\alpha}}(t)$, and $\widetilde{\rho}(t)$ are bounded, it implies that $\mathbf{e}(t)$ is bounded. Consequently, all of the system states $\mathbf{x}(t)$ are bounded, and the adaptive control $u(t)$ is also bounded.

Step 2: From Equation (A7), the designed control law, including Equations (38), (41), (49), and (50) can guarantee that $S_\Delta \in L_\infty$ and the error converge. Integrating both sides of Equation (A7) yields:

$$
\int_0^\infty S_\Delta^2 dt \leq \frac{V(S_\Delta(\infty), \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho}) - V(S_\Delta(0), \widetilde{\boldsymbol{\alpha}}, \widetilde{\rho})}{\eta / M_1(\mathbf{x})}
\tag{A8}
$$

　　　The right side of Equation (A8) is bounded, (i.e., $S_\Delta \in L_\infty$). Using Barbalat's lemma [36], it is proven that $S_\Delta = 0$ as $t \to \infty$, which means that the inequality $|S| \leq \Phi$ holds and the tracking error $\mathbf{e}(t)$ will asymptotically converge to zero as $t \to \infty$.

References

1.　Merlet, J.P. *Parallel Robots*; Kluwer: London, UK, 2000.
2.　Lafmejani, A.S.; Masouleh, M.T.; Kalhor, A. Gough-Stewart parallel robot using Backstepping-Sliding Mode controller and geometry-based quasi forward kinematic method. *Robot. Comput. Integr. Manuf.* **2018**, *54*, 96–114. [CrossRef]
3.　Stewart, D. A platform with six degrees of freedom. *Proc. Inst. Mech. Eng.* **1965**, *180*, 371–386. [CrossRef]
4.　Tsai, L.W. *Robot Analysis: The Mechanics of Serial and Parallel Manipulator*; Wiley Interscience: New York, NY, USA, 1999.
5.　Clavel, R. DELTA, a fast robot with parallel geometry. In Proceedings of the 18th International Symposium on Industrial Robots, Lausanne, Switzerland, 26–28 April 1988; pp. 91–100.
6.　Chablat, D.; Wenger, P. Architecture optimization of a 3-DOF translational parallel mechanism for machining applications, the orthoglide. *IEEE Trans. Robot. Autom.* **2003**, *19*, 403–410. [CrossRef]

7.   Huda, S.; Takeda, Y.; Hanagasaki, S. Kinematic design of 3-URU pure rotational parallel mechanism to perform precise motion within a large workspace. *Meccanica* **2011**, *46*, 89–100. [CrossRef]

8.   Zi, B.; Sun, H.; Zhang, D. Design, analysis and control of a winding hybrid-driven cable parallel manipulator. *Robot. Comput. Integr. Manuf.* **2017**, *48*, 196–208. [CrossRef]

9.   Qian, S.; Bao, K.; Zi, B.; Wang, N. Kinematic Calibration of a Cable-Driven Parallel Robot for 3D Printing. *Sensors* **2018**, *18*, 2898. [CrossRef]

10. Andrzej, L.P.; Emanuel, T.J.; Slawomir, B. Design of a 3-DOF tripod electro-pneumatic parallel manipulator. *Robot. Auton. Syst.* **2015**, *72*, 59–70.

11. Cazalilla, J.; Vallés, M.; Valera, Á.; Mata, V.; Díaz-Rodríguez, M. Hybrid force/ position control for a 3-dof 1t2r parallel robot: Implementation, simulations and experiments. *Mech. Based Des. Struct. Mach.* **2016**, *44*, 16–31. [CrossRef]

12. Song, S.K.; Kwon, D.S. Efficient formulation approach for the forward kinematics of 3-6 parallel mechanisms. *Adv. Robot.* **2002**, *16*, 191–215. [CrossRef]

13. Chiang, M.H.; Lin, H.T.; Hou, C.L. Development of a Stereo Vision Measurement System for a 3D Three-Axial Pneumatic Parallel Mechanism Robot Arm. *Sensors* **2011**, *11*, 2257–2281. [CrossRef] [PubMed]

14. Kouskouridas, R.; Amanatiadis, A.; Gasteratos, A. Guiding a robotic gripper by visual feedback for object manipulation tasks. In Proceedings of the 2011 IEEE International Conference on Mechatronics, Istanbul, Turkey, 13–15 April 2011; pp. 433–438.

15. Papanikolopoulos, N.P.; Khosla, P.K.; Kanade, T. Visual tracking of a moving target by a camera mounted on a robot: A combination of control and vision. *IEEE Trans. Robot. Autom.* **1993**, *9*, 14–35. [CrossRef]

16. Oh, P.Y.; Allen, K. Visual servoing by partitioning degrees of freedom. *IEEE Trans. Robot. Autom.* **2001**, *17*, 1–17. [CrossRef]

17. Peurla, M.; Hänninen, P.E.; Eskelinen, E.L. A computer-vision-guided robot arm for automatically placing grids in pioloform film preparation. *Methods Protoc.* **2019**, *2*, 9. [CrossRef]

18. Chiang, M.H.; Lin, H.T. Development of a 3D Parallel Mechanism Robot Arm with Three Vertical-Axial Pneumatic Actuators Combined with a Stereo Vision System. *Sensors* **2011**, *11*, 11476–11494. [CrossRef] [PubMed]

19. Lin, H.T.; Chiang, M.H. The Integration of the Image Sensor with a 3-DOF Pneumatic Parallel Manipulator. *Sensors* **2016**, *16*, 1026. [CrossRef] [PubMed]

20. Ren, L.; Mills, J.K.; Sun, D. Experimental comparison of control approaches on trajectory tracking control of a 3-DOF parallel robot. *IEEE Trans. Control Syst. Technol.* **2007**, *15*, 982–988. [CrossRef]

21. Cazalilla, J.; Valles, M.; Mata, V.; Rodriguez, M.D.; Valera, A. Adaptive control of a 3-DOF parallel manipulator considering payload handling and relevant parameter models. *Robot. Comput. Integr. Manuf.* **2014**, *30*, 468–477. [CrossRef]

22. Rad, C.R.; Hancu, O. An improved nonlinear modelling and identification methodology of a servo-pneumatic actuating system with complex internal design for high-accuracy motion control applications. *Simul. Model. Pract. Theory* **2017**, *75*, 29–47. [CrossRef]

23. Riachy, S.; Ghanes, M. Anonlinear controller for pneumatic servo systems: Design and experimental tests. *IEEE/ASME Trans. Mechatron.* **2014**, *19*, 1–11. [CrossRef]

24. Zhao, L.; Xia, Y.; Yang, Y.; Liu, Z. Multicontroller Positioning Strategy for a Pneumatic Servo System Via Pressure Feedback. *IEEE Trans. Ind. Electron.* **2017**, *64*, 4800–4809. [CrossRef]

25. Lee, L.W; Li, I.H. Wavelet-Based Adaptive Sliding-Mode Control with H$_\infty$ Tracking Performance for Pneumatic Servo System Position Tracking Control. *IET Control Theory Appl.* **2012**, *6*, 1–16. [CrossRef]

26. Sobczyk, M.R.; Gervini, V.I.; Perondi, E.A.; Cunha, M.A.B. A continuous version of the LuGre friction model applied to the adaptive control of a pneumatic servo system. *J. Frankl. Inst.* **2016**, *353*, 3021–3039. [CrossRef]

27. Qiu, Z.C.; Wang, B.; Zhang, X.M.; Han, J.D. Direct adaptive fuzzy control of a translating piezoelectric flexible manipulator driven by a pneumatic rodless cylinder. *Mech. Syst. Signal Process.* **2013**, *36*, 290–316. [CrossRef]

28. Kaitwanidvilai, S.; Olranthichachat, P. Robust loop shaping–fuzzy gain scheduling control of a servo-pneumatic system using particle swarm optimization approach. *Mechatronics* **2011**, *21*, 11–21. [CrossRef]

29. Bone, G.M.; Ning, S. Experimental comparison of position tracking control algorithms for pneumatic cylinder actuators. *IEEE/ASME Trans. Mechatron.* **2007**, *12*, 557–561. [CrossRef]

30. Tsai, Y.C.; Huang, A.C. FAT-based adaptive control for pneumatic servo systems with mismatched uncertainties. *Mech. Syst. Signal Process.* **2008**, *22*, 1263–1273. [CrossRef]

31. Lee, L.W.; Li, I.H. Design and Implementation of a Robust FNN-based Adaptive Sliding-Mode Controller for Pneumatic Actuator Systems. *J. Mech. Sci. Technol.* **2016**, *30*, 381–396. [CrossRef]

32. Zhao, L.; Yang, Y.; Xia, Y.; Liu, Z. Active Disturbance Rejection Position Control for a Magnetic Rodless Pneumatic Cylinder. *IEEE Trans. Ind. Electron.* **2015**, *62*, 5838–5846. [CrossRef]

33. Yang, H.; Sun, J.; Xia, Y.; Zhao, L. Position Control for Magnetic Rodless Cylinders with Strong Static Friction. *IEEE Trans. Ind. Electron.* **2018**, *65*, 5806–5815. [CrossRef]

34. Yeh, C.P.; Lee, L.W.; Li, I.H.; Chiang, H.H. A Translational Parallel Manipulator with Three Horizontal-Axial Pneumatic Actuators for 3-D Path Tracking. In Proceedings of the 2018 International Conference on System Science and Engineering (ICSSE), New Taipei City, Taiwan, 28–30 June 2018; pp. 1–6.

35. Wang, J.; Pu, J.; Moore, P.R. A practicable control strategy for servo pneumatic actuator systems. *Control Eng. Pract.* **1999**, *7*, 1483–1488. [CrossRef]

36. Slotine, E.J.; Li, W. *Applied Nonlinear Control*; Prentice Hall: Englewood Cliffs, NJ, USA, 1991.

37. Li, I.H.; Lee, L.W. Interval type 2 hierarchical FNN with the H-infinity condition for MIMO non-affine systems. *Appl. Soft Comput.* **2012**, *12*, 1996–2011. [CrossRef]

38. Chiang, M.H.; Lee, L.W.; Liu, H.H. Adaptive fuzzy controller with self-tuning fuzzy sliding-mode compensation for position control of an electro-hydraulic displacement-controlled system. *Int. J. Intell. Fuzzy Syst.* **2014**, *26*, 815–830.