IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON INDUSTRY 4.0

Received October 19, 2016, accepted November 1, 2016, date of publication November 29, 2016, date of current version January 4, 2017.

Digital Object Identifier 10.1109/ACCESS.2016.2633378

Pose Determination of a Robot Manipulator Based on Monocular Vision

YONG-LIN KUO, BO-HAN LIU, AND CHUN-YU WU
Graduate Institute of Automation and Control, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
Corresponding author: Y.-L. Kuo (yl_kuo@yahoo.com)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 105-2221-E-011-053.

ABSTRACT This paper presents an approach to determine the pose of a robot manipulator by using a single fixed camera. Conventionally, the pose determination is usually achieved by using the encoders to sense the joint angles, and then the pose of the end effector is obtained by using the direct kinematics of the manipulator. However, when the encoders or the manipulators are malfunctioning, the pose may not be accurately determined. This paper presents an approach based on machine vision, where a single camera is fixed away from the base of the manipulator. Besides, based on the kinematics of the manipulator and a calibrated camera, the pose of the manipulator can be determined. Furthermore, a graphical user interface is developed, which is convenient for users to operate the entire system. Two examples are demonstrated, and the estimated results are compared with those from the encoders. The proposed approach does not compete with the encoders. Instead, the approach can be treated as a backup method, which can provide a reference solution.

INDEX TERMS Pose determination, robot manipulator, monocular vision.

I. INTRODUCTION

Motion control of the robot manipulators mostly relies on encoders to sense the joint angles and to fulfill the feedback control. However, the encoders may not provide accurate joint angles of a robot manipulator due to the effects of elastic joints, joint frictions, flexible links, gearbox backlashes, manipulator wears, etc. Therefore, this paper proposes a monocular vision based approach, which does not compete with the encoders but can be treated as a backup or reference solution if the encoders or the manipulator are malfunctioning.

Most of researches addressing pose determination or estimation investigate the kinematics calibrations of the robot manipulators due to the errors caused from manufacturing or assembling. The conventional methods use laser trackers, coordinate measuring machines, etc. Driels *et al.* [1] completed the kinematics calibrations of a robot manipulator by using a coordinate measuring machine, which can determine the full pose of the end-effector. Nguyen *et al.* [2] proposed a method to measure the pose of robot manipulators using a laser tracker to track a set of target points on the end-effector, and the purpose of the method is to identify the kinematics errors of the manipulators. Nubiola *et al.* [3] applied two calibration methods, an optical CMM and a laser tracker, to a small industrial robot, and their accuracies are compared, where an error model was taken into account in the calibrations.

An alternative method to calibrate the robot manipulators uses machine vision. Wang *et al.* [4] proposed a calibration method for a robot manipulator by using a camera, which can obtain the position errors of multiple points, and a neural network is used to generate an error model. Zhang *et al.* [5] presented a solution for the pose determination of the end-effector of a parallel manipulator by identifying a parallelogram formed by four points in an image, and an error matrix based on seven error functions is established to check the rigidity of the end-effector of the manipulator. Larouche and Zhu [6] investigated an eye-in-hand robot manipulator to autonomously capture a mobile object in 3D space using predictive control, and the online target estimation is achieved by using the Kalman filter. Huang *et al.* [7] presented a monocular camera calibration method by using a circular array and applied the method to a robot system. Boby and Saha [8] proposed a method to measure the pose of an industrial robot by using a single image from an un-calibrated camera and to determine the camera internal parameters, and this method is based on an eye-in-hand technique and uses a calibration grid. Recently, some researches applied multiple

cameras or stereo cameras to calibrate the robot manipulators. Assa and Janabi-Sharifi [9] exploited multiple cameras to achieve the accurate and robust relative pose estimation of a manipulator toward an object based on nonlinear optimizations and the virtual visual servoing. Li *et al.* [10] performed the kinematics calibrations using a stereo camera mounted on one link of the robot manipulator, where a joint angle is constrained and the end-effector performs a circular arc motion. Miseikis *et al.* [11] presented a calibration method using a variable number of stereo cameras and a checkerboard, and this method allows a quick recalibration after any setup changes.

Perspective-*n*-point is one of the methods to estimate the poses of objects using their projection images obtained from calibrated cameras. Quan and Lan [12] proposed a method to have a unique solution of pose determination using four and five points, where the five-point method can be extended to deal with more than five points. Ansar and Daniilidis [13] presented a set of linear solutions of pose estimation for multiple points and multiple lines, and the sensitivity of solutions to noise is conducted, which can be used to be a predictor of the proposed method. Wu and Hu [14] presented a systematic study on the perspective-*n*-point problems and provided an elimination method to solve the perspective-*n*-point problems. Zhou *et al.* [15] proposed an algorithm to simultaneously estimate pose and correspondences based on a perspective-*n*-point method and the singular value decomposition.

Regarding to monocular vision, there are some applications about pose estimation. Katsuki *et al.* [16]proposed the measurement of a 3D object by identifying an artificial mark attached on the object, where image processing completes edge extraction, erosion, and color extraction to extract the feature points of the mark. Zhang *et al.* [17] proposed a two-stage iterative algorithm using inverse projection ray based on monocular vision, and the algorithm can estimate the pose of an object relative to a single camera using corresponding feature points. Chen *et al.* [18] proposed a pose determination method using monocular vision, where a quaternion formulation of four feature points is used to develop six nonlinear equations on the image coordinates. Liu *et al.* [19] presented a monocular vision detection algorithm, where the different poses of the object are projected onto a virtual camera and the local-mean Hausdorff Distance is used to accelerate the image matching. Wu *et al.* [20] developed a method to measure the position and attitude parameters of a weld stud using monocular vision, where the mathematical model related to the position and attitude of the weld stud is included. Guo *et al.* [21] presented a 3D head pose estimation method based on monocular vision, and the method uses a linear combination of the head poses corresponding to some training feature sampling points. Sharma and D'Amico [22] assessed the performances of three initial pose estimation techniques for spacecraft to complete the formation-flying and on-orbit missions based on monocular vision, where the techniques uses a minimum number of features to estimate the spacecraft pose with respect to the camera. Andriluka *et al.* [23] proposed a three-stage monocular human pose estimation, where the first stage obtains an initial 2D articulation estimate, the second stage extracts people tracklets from consecutive frames, and the third stage uses the image evidence accumulated to recover 3D pose. Agarwal and Triggs [24] presented a learning-based 3D human pose recovery method using monocular image sequences and direct nonlinear regression. Andreff and Martinet [25] presented the vision-based modeling an control of a Gough-Stewart parallel robot, where the image of the robot are obtained by a fixed camera and the legs of the robot in the image are identified by line geometry. Agarwal *et al.* [26] presented a dynamically equivalent modeling for articulated multibody systems using uncalibrated monocular videos, where the dynamical model structure is given and the unscented Kalman filter is used to estimate the model parameters.

In review of literature, there are a limited number of papers addressing the pose determination of robot manipulator by using a single fixed camera. This paper considers a scenario, a robot manipulator moving in a three dimensional workspace, and the pose needs to be determined on-line. Conventionally, the pose of a robot manipulator is determined through direct kinematics of the manipulator after instantaneously sensing all joint angles by using encoders. However, if the encoders and/or the structure of the manipulators are malfunctioning due to the effects of elastic joints, joint frictions, flexible links, gearbox backlashes, manipulator wears, etc., the pose may not be accurately determined. Therefore, this paper proposes a simple approach by using a single instant image, which is obtained by a fixed digital camera away from the base of the manipulator. Since it is impossible to identify a point coordinate in a three dimensional space through a two dimensional projection image, the proposed approach applies the perspective-*n*-point technique, where there are four points specified on the surface of the robot manipulator. Besides, one needs some a priori information, including the dimensions and type of the robot manipulator and the intrinsic parameters of the digital camera, which will be estimated through a calibration experiment. The information will be included in the proposed algorithm.

This paper is organized as follows. Section II states the problem and the proposed approach. Section III introduces the kinematics of the manipulator studied in this paper. Section IV presents the proposed approach to determine the pose of the manipulator. Section V demonstrates two experiments and shows their results. Section VI concludes the contributions of this study.

II. PROBLEM STATEMENT AND PROPOSED APPROACH

A robot manipulator is considered to be moving in a three dimensional workspace. When the encoders mounted on the joints of the manipulator may not provide accurate joint angles, a proposed approach can provide a reference solution

of the joint angles by using a single image obtained by a single fixed camera away from the base of the manipulator, where the camera can capture a full image of the manipulator.

The perspective-n-point technique will be applied to identify four feature points on the surface of the robot manipulator. Besides, the type and dimensions of the manipulator, the distance between the camera and the base of the manipulator, and the intrinsic parameters of the camera should be given. Based on the type and dimensions of the manipulator, the kinematics equations of the manipulator can be derived. Also, based on the intrinsic parameters of the camera, the perspective projection equations can be obtained. The proposed approach is to solve the two sets of equations for the joint angles. Therefore, there are two steps to apply the proposed approach. The first step is to obtain the information about the manipulator and the camera through a simple calibration experiment. The second step is to solve the two sets of equations for the joint angles through the multivariate nonlinear regression. Furthermore, a graphical user interface is developed to show the instantaneous values of the joint angles, their history data, and the dynamical image of the manipulator. It is convenient for users to operate the entire pose determination system.

In order to demonstrate the proposed approach, a modified Scorbot robot is used in this study. The following sections will introduce all tasks in the proposed approach.

III. KINEMATICS OF SCORBOT ROBOT

A. DENAVIT-HARTENBERG TRANSFORMATION MATRICES

The Denavit-Hartenberg (D-H) transformation matrices can be conveniently applied to the derivations of manipulator kinematics. The matrix defines the transformation from a Cartesian coordinate system to another. To illustrate the coordinate transformation, three rotation matrices and three translation matrices are defined as [27]

Rotation matrices:

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\theta & -s_\theta & 0 \\ 0 & s_\theta & c_\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$R_y(\theta) = \begin{bmatrix} c_\theta & 0 & s_\theta & 0 \\ 0 & 1 & 1 & 0 \\ -s_\theta & 0 & c_\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$R_z(\theta) = \begin{bmatrix} c_\theta & -s_\theta & 0 & 0 \\ s_\theta & c_\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where the subscripts x, y and z refer to the rotating axes; θ is the rotating angle;c_θ and s_θ represent $\cos\theta$ and $\sin\theta$, respectively.

Translation matrices:

$$T_x(r) = \begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_y(r) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T_z(r) = \begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where r is the translating displacement.

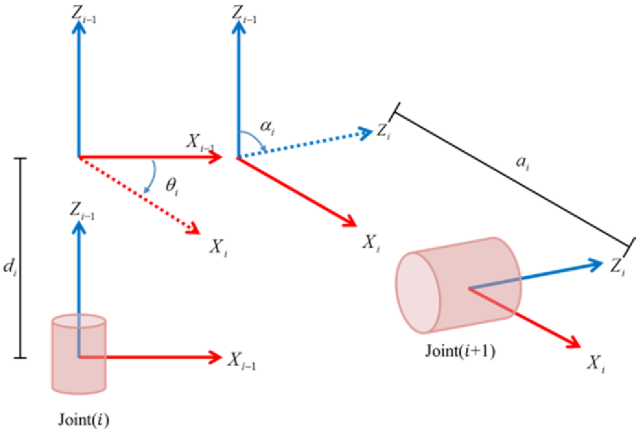

FIGURE 1. D-H transformation.

Based on the D-H convention, each coordinate system is attached on a link of a manipulator, and the transformation procedures from coordinate system $i-1$ to i are shown in Fig. 1. First, the coordinate system $i-1$ translates d_i along the Z_{i-1} axis. Secondly, the transformed coordinate system rotates θ_i along the Z_{i-1} axis. Thirdly, the transformed coordinate system translates a_i along the X_i. Finally, the transformed coordinate system rotates α_i along the X_i axis. Therefore, the coordinate transformation matrix is expressed as

$$\begin{aligned} {}^{i-1}A_i &= T_{Z_{i-1}}(d_i) \cdot R_{Z_{i-1}}(\theta_i) \cdot T_{X_i}(a_i) \cdot R_{X_i}(\alpha_i) \\ &= \begin{bmatrix} {}^{i-1}\bar{R}_i & {}^{i-1}\bar{P}_i \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (3)$$

where $\bar{R}$ and $\bar{P}$ refer to the rotation and translation with respect to the origin of coordinate system $i-1$, and they are expressed as

$${}^{i-1}\bar{R}_i = \begin{bmatrix} c_{\theta_i} & -c_{\alpha_i}s_{\theta_i} & s_{\alpha_i}s_{\theta_i} \\ s_{\theta_i} & c_{\alpha_i}c_{\theta_i} & -s_{\alpha_i}c_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} \end{bmatrix}, \quad {}^{i-1}\bar{P}_i = \begin{bmatrix} a_i c_{\theta_i} \\ a_i s_{\theta_i} \\ d_i \end{bmatrix} \quad (4)$$

B. SCORBOT ROBOT

Fig. 2 shows the schematic diagram of the Scorbot Robot [28], and it has five degree degrees of freedom (or axes), where the second, third, and fourth joint axes are perpendicular to the paper plane and are parallel to one

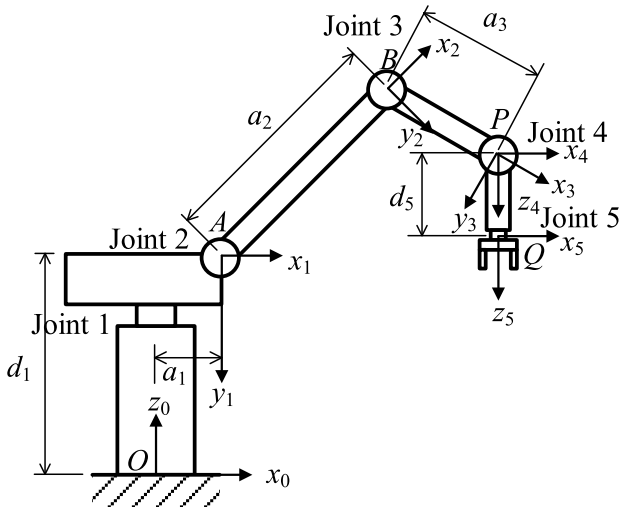

FIGURE 2. Schematic diagram of the Scorbot robot.

TABLE 1. D-H Parameters of the scorbot robot.

Joint i	α_i	a_i	d_i	θ_i
1	$-\pi/2$	0	d_1	θ_1
2	0	a_2	0	θ_2
3	0	a_3	0	θ_3
4	$-\pi/2$	0	0	θ_4
5	0	0	d_5	θ_5

another at points A, B, and P, respectively; the first joint axis is vertically upward, and the fifth joint axis is intersected with the fourth joint axis at point P; there are five coordinate systems (x_i, y_i, z_i) defined based on each joint, in which i refer to the joint number; a_i and d_i are the distances between two coordinate systems. Based on the aforementioned D-H transformation, the D-H parameters of the Scorbot robot are shown in Table I, and the transformation matrix of the fifth coordinate system with respect to the zeroth coordinate system is expressed as

$$ {}^0A_5 = \begin{bmatrix} {}^0\bar{R}_5 & {}^0\bar{P}_5 \\ 0 & 1 \end{bmatrix} \tag{5} $$

where

$$ {}^0\bar{R}_5 = \begin{bmatrix} c_1c_{234}c_5 + s_1s_5 & -c_1c_{234}s_5 + s_1c_5 & -c_1s_{234} \\ s_1c_{234}c_5 - c_1s_5 & -s_1c_{234}s_5 - c_1c_5 & -s_1s_{234} \\ -s_{234}c_5 & s_{234}s_5 & -c_{234} \end{bmatrix} \tag{6} $$

$$ {}^0\bar{P}_5 = \begin{bmatrix} c_1(a_1 + a_2c_2 + a_3c_{23} - d_5s_{234}) \\ s_1(a_1 + a_2c_2 + a_3c_{23} - d_5s_{234}) \\ d_1 - a_2s_2 - a_3s_{23} - d_5c_{234} \end{bmatrix} \tag{7} $$

where the subscripts i, ij or ijk of c and s represent θ_i, $\theta_i + \theta_j$ or $\theta_i + \theta_j + \theta_k$, respectively.

C. EXPERIMENTAL 6-DOF ROBOT MANIPULATOR

To demonstrate the proposed approach, this study adapts an experimental 6-DOF robot manipulator shown in Fig. 3, which is designed and manufactured by the TeraSoft Inc.

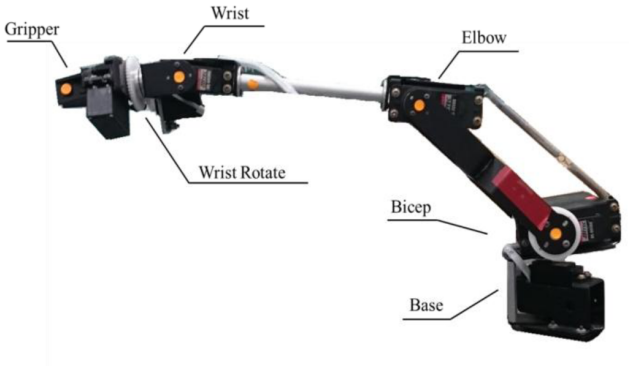

FIGURE 3. An experimental 6-DOF robot manipulator.

As a matter of fact, the robot is a modified Scorbot robot, where the parameter a_1 is zero and an additional axis is used to operate the gripper.

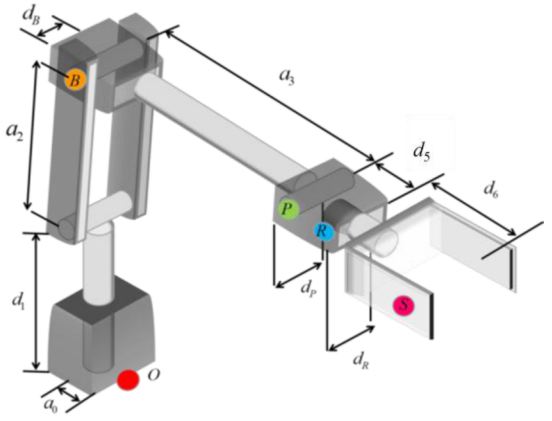

FIGURE 4. Feature points on the experimental 6-DOF robot manipulator.

To identify some specific points on the manipulator, one defines four feature points on the surface of the manipulator. It is worth to note that the original points O, A, B, P, and Q shown in Fig. 2 locate at the centers of the joint axes, and they cannot be identified by a digital camera. Thus, the points O, B, and P are re-defined, and they locate at the surfaces of the joint axes as shown in Fig. 4. Besides, two additional points R and S are defined, which locate at the surfaces of the fifth joint and the gripper, respectively. These five points are attached by five round shape stickers with different colors so as to be identified by a digital camera and some image processing techniques. There are some offset distances to the joint axes, which are given as a_O, d_B, d_P, and d_R for points O, B, P, and R. Furthermore, the distance between axis x_5 and point S is defined as d_6, and the distance between axis z_5 and point S is defined as a_s, which is used to describe that the gripper is open or closed.

By using the D-H transformation, the coordinates of points B, P, R, and S with respect to point O can be obtained as follows.

$$ x_B = a_1c_1 + a_2c_1c_2 + d_Bs_1 \tag{8} $$

$$ y_B = a_1s_1 + a_2s_1c_2 + a_O - d_Bc_1 \tag{9} $$

$$z_B = -a_2 s_2 + d_1 \tag{10}$$
$$x_P = a_1 c_1 + a_2 c_1 c_2 + a_3 c_1 c_{23} + d_P s_1 \tag{11}$$
$$y_P = a_1 s_1 + a_2 s_1 c_2 + a_3 s_1 c_{23} + a_O - d_P c_1 \tag{12}$$
$$z_P = -a_2 s_2 - a_3 s_{23} + d_1 \tag{13}$$
$$x_R = a_1 c_1 + a_2 c_1 c_2 + a_3 c_1 c_{23} + d_5 c_1 (c_{23} c_4 - s_{23} s_4) + d_R s_1 \tag{14}$$
$$y_R = a_1 s_1 + a_2 s_1 c_2 + a_3 s_1 c_{23} + a_O + d_5 s_1 (c_{23} c_4 - s_{23} s_4) - d_R c_1 \tag{15}$$
$$z_R = -a_2 s_2 - a_3 s_{23} + d_1 - d_5 (c_{23} s_4 + s_{23} c_4) \tag{16}$$
$$x_S = a_1 c_1 + a_2 c_1 c_2 + a_3 c_1 c_{23} - a_S [c_1 (s_{23} c_4 + c_{23} s_4) c_5 - s_1 s_5] + d_5 c_1 (c_{23} c_4 - s_{23} s_4) + d_6 c_1 (c_{23} c_4 - s_{23} s_4) \tag{17}$$
$$y_S = -a_1 s_1 + a_2 s_1 c_2 + a_3 s_1 c_{23} + a_O - a_S [s_1 (s_{23} c_4 + c_{23} s_4) c_5 - c_1 s_5] + d_5 s_1 (c_{23} c_4 - s_{23} s_4) + d_6 s_1 (c_{23} c_4 - s_{23} s_4) \tag{18}$$
$$z_S = -a_2 s_2 - a_3 s_{23} - a_S (c_{23} c_4 - s_{23} s_4) c_5 + d_1 - d_5 (s_{23} c_4 + c_{23} s_4) - d_6 (s_{23} c_4 + c_{23} s_4) \tag{19}$$

Note that there are six variables, θ_1, θ_2, θ_3, θ_4, θ_5, and a_S, where θ_i are the joint angles and a_S is the distance of the two fingers of the gripper. The six variables will be determined through the identification of the four feature points, B, P, R, and S, which will be introduced in the following section, and they can be used to represent the pose of the manipulator.

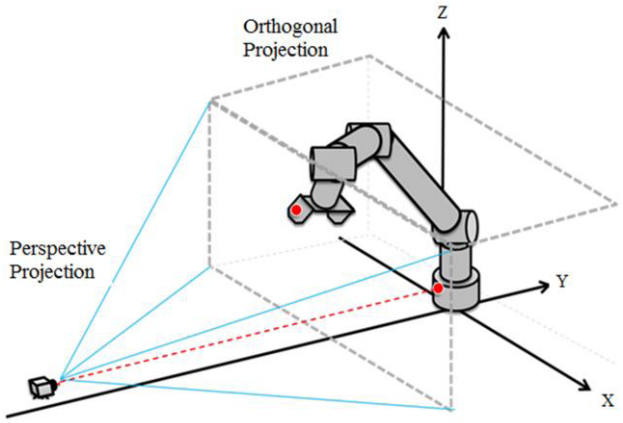

FIGURE 5. Experimental setup of the monocular vision system.

IV. POSE DETERMINATION BASED ON THE MONOCULAR VISION

A. EXPERIMENTAL SETUP AND VALIDATION

Fig. 5 shows the experimental setup of the proposed monocular vision system, which consists of a robot manipulator, a digital color camera, and a personal computer (not shown in this figure). Both the robot manipulator and the camera are fixed, and the optical axis of the camera should be parallel to the Y axis, where a fixed point on the robot manipulator will be selected as an origin so as to define a Cartesian coordinate system as shown in Fig. 5.

The experimental 6-DOF robot manipulator is shown in Fig. 3, and the D-H parameters of the manipulator will be determined later through some experiments and the multivariate nonlinear regression. The manipulator is controlled by a real-time PC-based controller, called the Micro-Box, which is manufactured by MathWorks, Inc. The relevant computer codes are developed by using the software Matalb/Simulink, and the codes include the inverse kinematics of the manipulator and PID controllers. The motion control of the manipulator can be achieved through the software. Besides, one developed the computer codes to complete the path planning and the tracking control based on a planned path.

The type of the digital color camera is CYCLOPS SU1000C-8, which is manufactured by Aisys Vision Company. The effective resolution is 3664×2748 pixels, and maximum frame rate is 7.5 fps.

The personal computer has two tasks: one is to operate the robot manipulator through a control system, and the other one is to estimate the pose of the robot manipulator through the image processing software and the kinematics based computer codes.

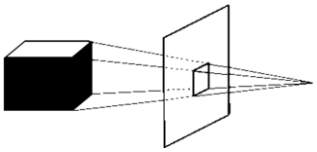

FIGURE 6. Perspective projection.

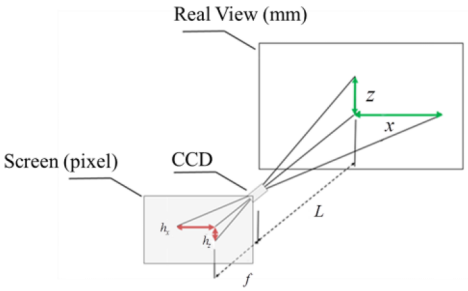

FIGURE 7. Pin-hole imaging.

B. 3D PIN-HOLE IMAGING OF THE FEATURE POINTS

The proposed approach utilizes a single fixed CCD camera, and the distance between an object and the camera is finite, so the 3D projection belongs to the perspective projection as shown in Fig. 6. Besides, the pin-hole imaging technique is used to identify the positions of the feature points as shown in Fig. 7. Therefore, by using the simple trigonometry, the perspective projection leads to the formulas as

$$\frac{x}{L} = \frac{h_x}{f}, \quad \frac{z}{L} = \frac{h_z}{f} \tag{20}$$

where (x, z) are the coordinates of the feature points on a two-dimensional plane; (h_x, h_z) are the pixel positions of the feature points on the camera screen; f is a scaling factor

related to the camera's optical properties; L is the distance between the plane and the camera. Thus, if f and L are given, and (h_x, h_z) can be obtained through a digital image, then (x, z) can be calculated by (20).

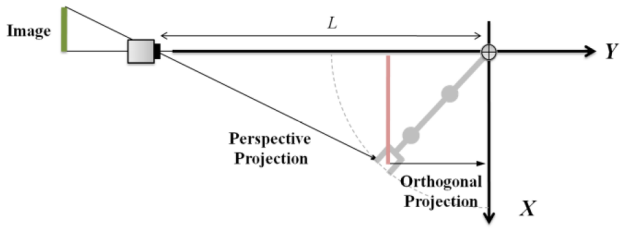

FIGURE 8. Projection of a robot manipulator.

A critical issue is that the feature points on the experimental robot manipulator are not fixed on a plane. Instead, they can move in a three-dimensional space as shown in Fig. 8. In other words, the distance L between the plane and the camera is not constant and unknown. Therefore, (20) will be modified as

$$\frac{x}{(L+y)} = \frac{h_x}{f}, \quad \frac{z}{(L+y)} = \frac{h_z}{f} \tag{21}$$

C. MULTIVARIATE NONLINEAR REGRESSION

Eq. (21) represents two equations in terms of three unknowns (x, y, z), so it is impossible to solve them. If a feature point on the manipulator is selected, then the coordinates (x, y, z) of the point can be expressed as functions of joint angles of the manipulator through its kinematics equations. Thus, in order to solve (21) for three unknowns (x, y, z), the kinematics equations as shown in (8)-(19) will be substituted into (21) for each feature point, so each feature point provides two equations in terms of six unknowns, which are five joint angles and the gripper gap of the manipulator. Furthermore, one defines four feature points to generate eight equations to be solved for the six unknowns, and the multivariate nonlinear regression will be performed to solve this problem. One applies the Matlab function nlinfit in the Matlab Optimization Toolbox to solve the multivariate nonlinear regression problem, where the objective functions are the residuals of the eight equations, which are defined as

$$J_1 = xf - (L+y)h_x, \quad J_2 = zf - (L+y)h_z \tag{22}$$

Note that (22) refers to any feature point, so there are eight objective functions to be minimized so as to find a set of optimal solution. Besides, the function nlinfit uses the Levenberg-Marquardt nonlinear least squares algorithm [29].

D. PARAMETER ESTIMATION OF THE CAMERA

Before performing the multivariate nonlinear regression presented in Subsection C, it is necessary to obtain the scaling factor f shown in (22). An experiment is designed so as to measure the values L, x, and h_x. Note that the experiment is designed for applying (20), so the distance L between a test point and the camera is fixed and the distance can be measured. Besides, the x coordinate of the test point can be measured, and the pixel position (h_x, h_z) can be obtained through an image obtained by the camera. Thus, a test point can determine a scaling factor. However, the measurements of the experiment might have errors, so one repeatedly performs this experiment to collect multiple sets of data. Also, a simple linear regression based on (20) is performed, where the least square method is used to solve the linear regression problem, and the regression function is written as

$$h_x = xf/L, h_z = zf/L \tag{23}$$

In this experiment, the distance between the test point and the camera is fixed as $L = 1355$ mm, and the position of the test point locates at 30 different positions, whose x coordinate ranges from 36 to 1258 mm. After performing the linear regression, the scaling factor f is estimated as $f = 5067.32$ pixels. To show the estimation accuracy, the coefficient of determination R^2 and the root mean square error e_{RMS} for h_x are respectively defined as

$$R^2 = 1 - [\sum_{i=1}^{n} (h_{x_i} - \hat{h}_{x_i})^2]/[\sum_{i=1}^{n} (h_{x_i} - \bar{h}_x)^2] \tag{24}$$

$$e_{RMS} = ([\sum_{i=1}^{n} (h_{x_i} - \hat{h}_{x_i})^2]/n)^{0.5} \tag{25}$$

where h_{x_i}, $\hat{h}_{x_i}$, and $\bar{h}_x$ are the measured values, the estimated values, and the mean measured value, respectively; n is the number of data. Note that the values of R^2 and e_{RMS} are more close to one and zero, respectively, the estimated values is more accurate. Based on the experimental data, the values of R^2 and e_{RMS} are obtained as 0.9994 and 3.0627, respectively.

E. PARAMETER ESTIMATION OF THE ROBOT MANIPULATOR

To obtain more accurate pose estimations, it is necessary to have more accurate D-H parameters. Theoretically, these parameters can be directly measured by a straight ruler. However, it is difficult to locate the coordinate axes, since they are located inside the joints. Therefore, one intends to apply the multivariate nonlinear regression to obtain these parameters instead of directly measuring the parameters by a straight ruler. One arbitrarily assigns 90 static poses of the manipulator and identifies the positions of the four feature points through a digital camera. The identification of the four feature points is based on the aforementioned kinematics of the experimental 6-DOF robot manipulator. The multivariate nonlinear regression is also performed by a Matlab function nlinfit. The objective functions are the residuals of (8) to (19), which are the same as (22) and will be minimized to estimate the eight parameters as

$$\beta = [\, a_2 \;\; d_1 \;\; d_B \;\; d_P \;\; a_3 \;\; d_R \;\; d_5 \;\; d_6 \,]^T \tag{26}$$

The estimated results are listed in Table II, where the measured values are obtained by a straight ruler. Besides, the diagonal elements of the covariance matrix of the estimated parameters and the mean square error are used to evaluate

TABLE 2. Estimated parameters of the experimental 6-DOF robot manipulator.

Estimated parameters	Estimated values (mm)	Measured value (mm)
a_2	143.5879	145
d_1	43.3867	40
d_B	44.2829	40
d_P	58.9250	60
a_3	188.5091	185
d_R	60.0098	60
d_5	20.7383	25
d_6	59.5518	60

the accuracy of the estimated parameters. Smaller values of both represent smaller variations with respect to the estimated values. After calculations, the covariance elements and the mean square error are diag(Cov) = [0.4595 0.2745 0.3601 0.3915 0.5512 0.3915 0.6823 0.8554]^T and MSE = 11.7458, respectively.

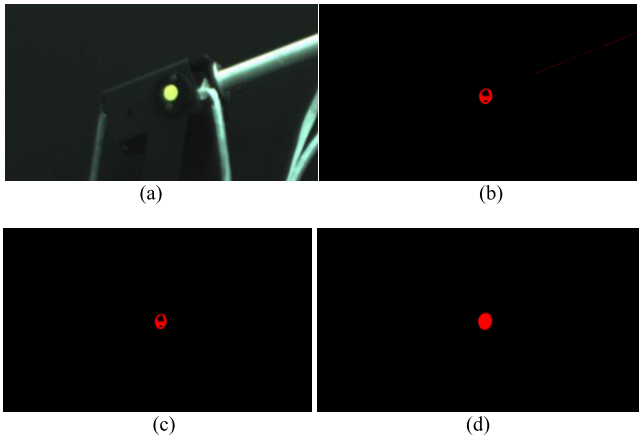

FIGURE 9. Image processing. (a) Original image; (b) image binary; (c) noise elimination; (d) image filling and dilation.

F. GRAPHICAL USER INTERFACE

In order to conveniently operate the entire system, a graphical user interface (GUI) is developed by using the software LabView. There are three tasks in the GUI computer codes. The first task is to initiate the camera and to create a buffer to restore the images. The second task is to perform the image processing. Since the colors of the feature points are different, the color thresholds are defined in Table III. After the areas of the feature points are identified, the central pixel position can be calculated by performing some image processing techniques, including image binary, noise elimination, region filling and dilation. Fig. 9 shows an example of image processing about point B. The results show that the image processing can enhance the feature point. The third task is to estimate the pose of the manipulator. One uses

TABLE 3. Color thresholds for the feature points.

Feature points	R	G	B
B	150-255	120-220	50-100
P	70-160	170-255	47-120
R	25-55	65-130	80-180
S	120-255	43-120	50-140

the software Matlab to complete the pose estimation based on the kinematics of the manipulator. The Matlab script is included in the software LabView. Besides, all estimation results are restored in several files. The graphical user interface is shown in Fig. 10, and the flowchart of the entire system is shown in Fig. 11. The graphical user interface can instantly display the six estimated variables and show their variations. Note that the flowchart is based on the given robot manipulator, so its type and dimension and the parameters of the camera should be given, including the scaling factor f and the fixed distance between the manipulator's base and the camera.

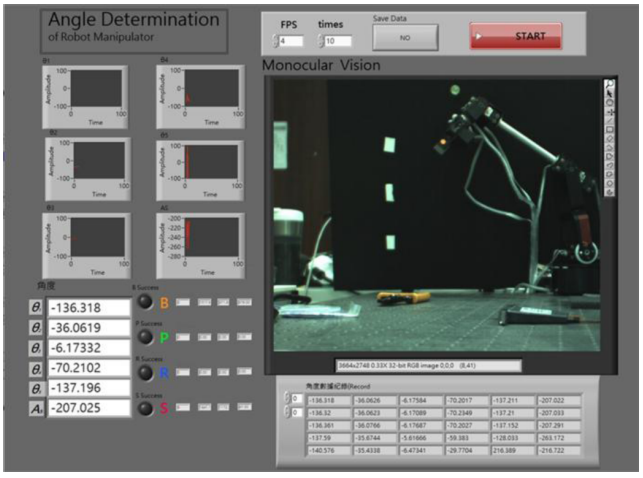

FIGURE 10. A graphical user interface for the robot manipulator.

V. EXPERIMENTAL RESULTS

A. PATH PLANNING

To demonstrate the proposed approach, the path planning for the end-effector of the manipulator is introduced first, so the end-effector will move by tracking the planned path. A simple path planning is used to generate a desired path, which will pass through two prescribed points. And, each coordinate of the path can be expressed as a quintic polynomial, which is a function of time and is written as

$$x(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 \qquad (27)$$

where x is a coordinate, and it can be replaced by y and z for the other two coordinates; t is time; c_i are the coefficients, and they can be determined by the boundary

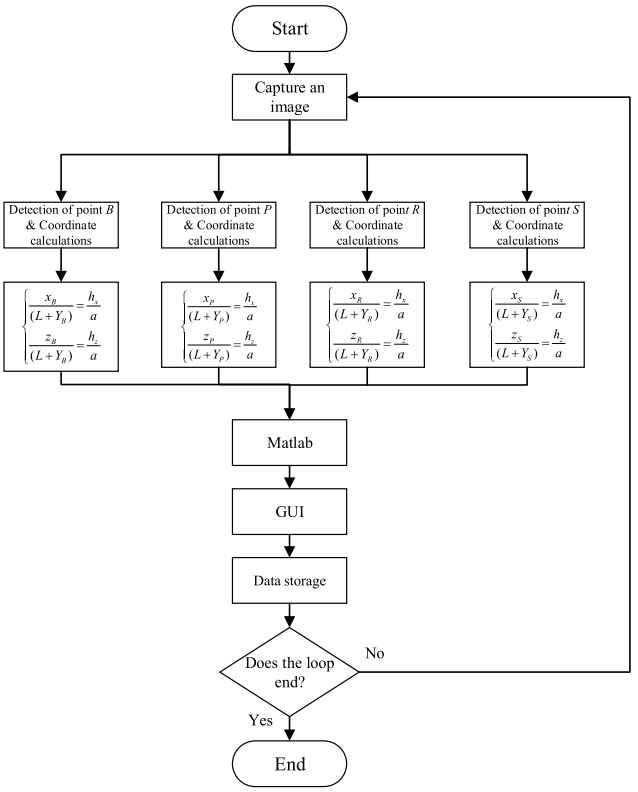

FIGURE 11. A flowchart of the pose estimation system.

conditions as

$$\begin{bmatrix} x_0 \\ \dot{x}_0 \\ \ddot{x}_0 \\ x_f \\ \dot{x}_f \\ \ddot{x}_f \end{bmatrix} = \begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 & t_0^4 & t_0^5 \\ 0 & 1 & 2t_0 & 3t_0^2 & 4t_0^3 & 5t_0^4 \\ 0 & 0 & 2 & 6t_0 & 12t_0^2 & 20t_0^3 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{bmatrix} \tag{28}$$

where $[\, x_0 \ \dot{x}_0 \ \ddot{x}_0 \ x_f \ \dot{x}_f \ \dot{x}_f \,]^T$ represents the position, velocities and accelerations of the starting and end points; t_0 and t_f are the starting and end time. The equation can be solved for the coefficients if the position, velocities and accelerations of the starting and end points are all specified, and the starting and end time are also given. In the following examples, the velocities and accelerations of both ends are specified as zeros in order to have zero oscillations at both ends.

Based on the inverse kinematics Simulink codes provided by the TeraSoft Inc., the joint angles can be determined. Note that the fifth joint and the gripper do not affect the position of the end-effector, so their motions will be assigned separately.

B. MOTION CONTROL OF THE ROBOT MANIPULATOR

The motion control of the robot manipulator is achieved by a semi-closed-loop control system, and Fig. 12 shows the software flowchart of operating the manipulator. The path planning block generates a desired coordinate of the end-effector. The inverse kinematics block converts the coordinate

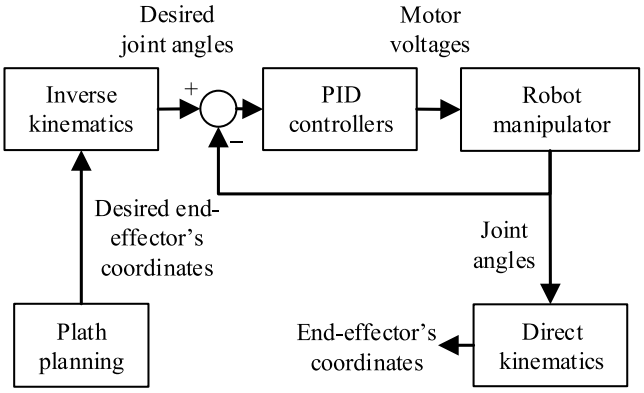

FIGURE 12. A software flowchart to operate the robot manipulator.

to the desired joint angles. The control law is a PID controller, which computes the voltages applied to the motors based on the errors of joint angles, where the actual joint angles are determined by the robot manipulator block and are sensed by the encoders. The direct kinematics block converts the joint angles to the coordinates of the end-effector, which can be compared with the desired coordinates. The entire Simulink codes are also provided by the TeraSoft Inc. except for the path planning block.

C. ERROR EVALUATIONS

In order to evaluate the errors by using the proposed approach, the errors are defined as the differences between the command values and the estimated values. Three error formulas are defined as

$$\text{MAD} = (\sum_{i=1}^{n} |e_i|)/n \tag{29}$$

$$\text{RMSE} = [(\sum_{i=1}^{n} e_i^2)/n]^{0.5} \tag{30}$$

$$\text{MAPE} = (\sum_{i=1}^{n} |e_i| / y_i)/n \times 100\% \tag{31}$$

where y_i is the command value, and e_i is the difference between the command value and the estimated or encoder value.

D. EXAMPLE 1

The desired path is assigned as starting at the position coordinate (6, 10, 6) cm, passing through the position coordinate (8, 13, 3) cm, and ending at the position coordinate (9, 14, 3) cm, so the path consists of two segments, where each segment can be expressed as a quintic polynomial as (25). Besides, the fifth joint is assigned to rotate from 30 to 10 degrees counterclockwise after 14 seconds from the starting time. The gap of the gripper is operated from 0 to 4 cm at 10 seconds from the starting time.

Fig. 13 shows the estimated results by the proposed approach. The figure includes six graphs, and each graph shows the estimated values θ_1, θ_2, θ_3, θ_4, θ_5, and a_S, which are the functions of time. Each graph also shows the encoder

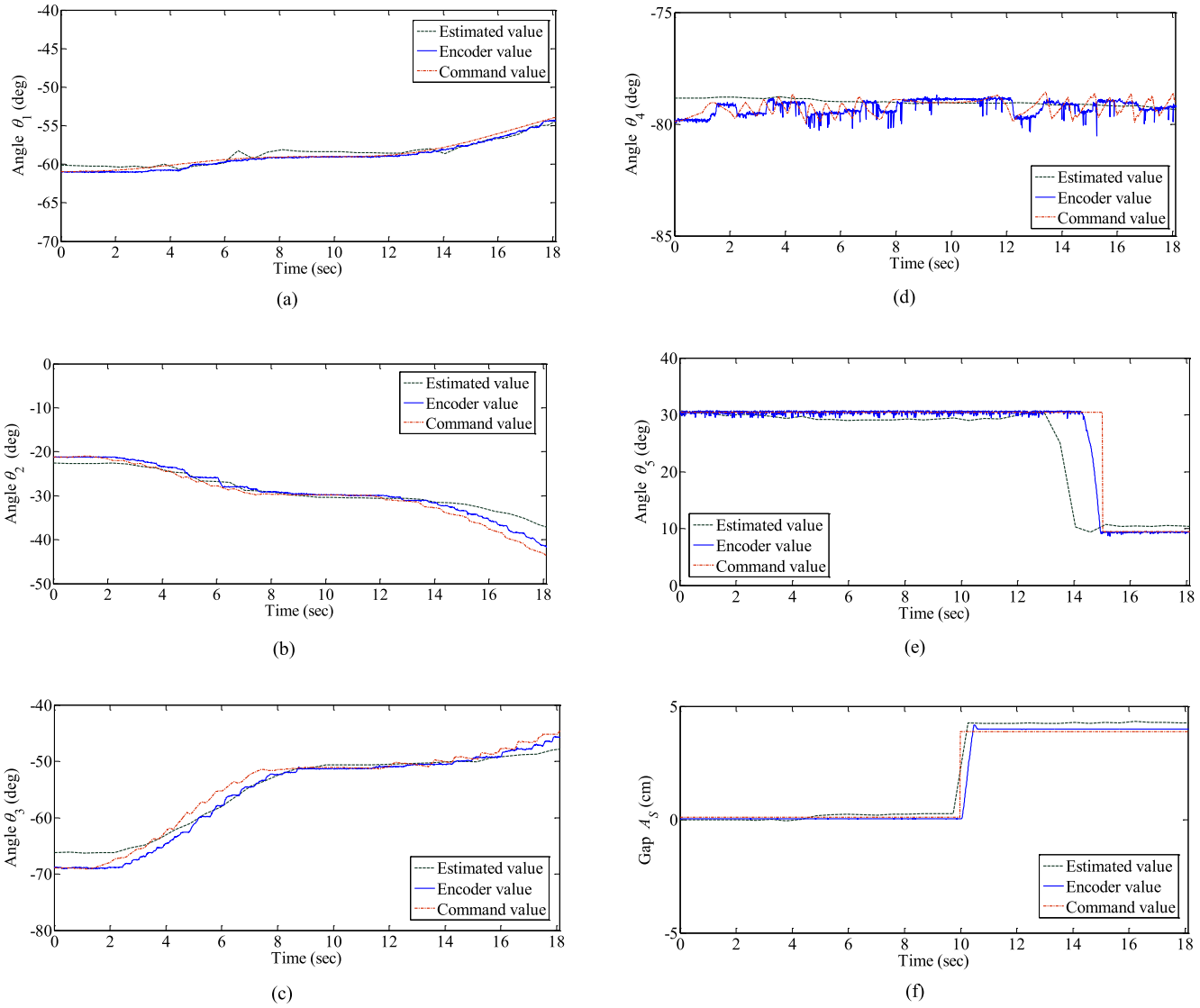

FIGURE 13. Experimental results compared with the command value and the encoder value for example 1. (a) Angle θ_1; (b) angle θ_2; (c) Angle θ_3; (d) angle θ_4; (e) angle θ_5; (f) Gap A_S.

values and the command values compared with the estimated values. The results show that the estimated values are close to the encoder values. By examining Fig. 13, there are some differences between the command signals and the encoder signals. There are two reasons. One is that the dynamics of the motors and the manipulators are not included in the control system, so the control precision is not high. And, the other one is that the manufacturer of the manipulator uses low-cost encoders, so the accuracy of the encoders is also not high.

Table IV shows the error comparisons, and the results show that the errors of the estimated values are close to the errors of the encoder values. This indicates that the proposed approach provide similar results with the encoders. As mentioned before, the proposed approach does not compete with the encoders. Instead, the proposed approach can be a backup if the encoders are malfunctioning, the gearboxes wear, or the base of the manipulator is loose.

TABLE 4. Errors of the estimated values for example 1.

Errors		θ_1	θ_2	θ_3	θ_4	θ_5	A_S
MAD	Estimate	1.83	2.19	2.07	0.50	0.26	0.36
	Encoder	1.37	2.28	2.31	0.52	0.35	0.17
RMSE	Estimate	2.54	2.63	2.41	0.58	3.20	0.43
	Encoder	1.89	2.84	2.86	0.83	1.46	0.33
MAPE (%)	Estimate	2.43	3.48	3.13	1.02	0.53	0.76
	Encoder	1.83	3.60	4.29	1.11	0.73	0.47

E. EXAMPLE 2

Similar to example 1, the desired path is planned as starting at (7, 7, 7) and ending at (8, 10, 5) cm, and the path is generated by using the quintic polynomials as (25). Besides, the fifth axis rotates clockwise from 10 to 30 degrees after 5 seconds from the starting time, and the gap of the gripper

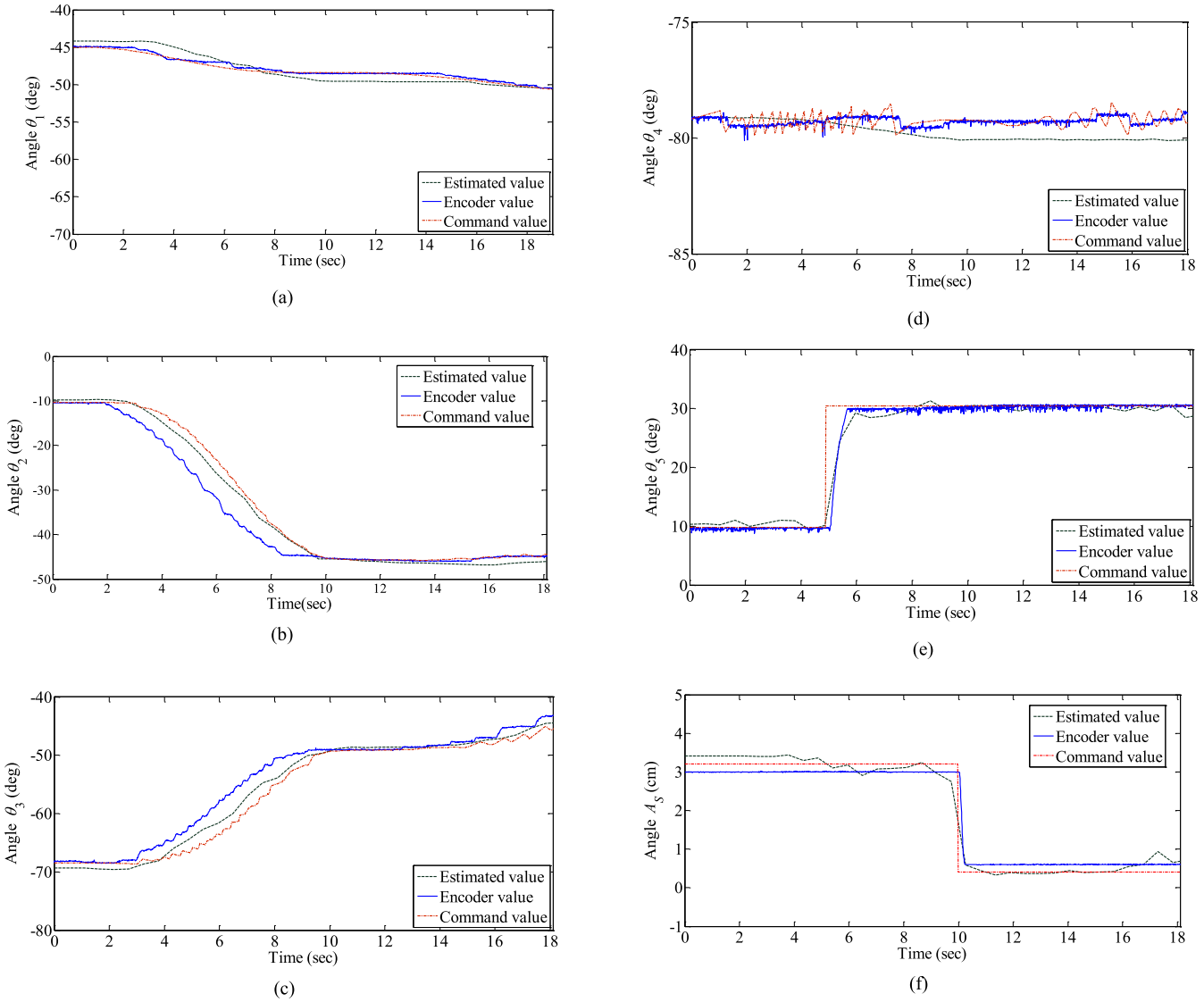

FIGURE 14. Experimental results compared with the command value and the encoder value for example 2. (a) Angle θ_1; (b) angle θ_2; (c) Angle θ_3; (d) angle θ_4; (e) angle θ_5; (f) Gap A_S.

TABLE 5. Errors of the estimated values for example 2.

Errors		θ_1	θ_2	θ_3	θ_4	θ_5	A_S
MAD	Estimate	0.83	1.02	0.92	1.14	0.81	0.09
	Encoder	0.21	2.32	1.67	0.22	0.49	0.11
RMSE	Estimate	0.93	1.28	1.15	1.36	0.95	0.12
	Encoder	0.26	3.86	2.43	0.27	0.52	0.15
MAPE (%)	Estimate	1.75	4.39	1.59	1.68	2.98	3.46
	Encoder	0.44	7.45	2.91	0.28	2.03	5.49

is closed from 4 to 0 cm after 10 seconds from the starting time.

Fig. 14 shows the estimated results by the proposed approach. The results show that the command values, the encoder values, and the estimated values are close to each other. Table V shows the error comparisons, and the results show that the errors of the estimated values are close to the errors of the encoder values. This indicates that the proposed approach provide similar results with the encoders.

VI. CONCLUSION

This paper proposes an approach to determine the pose of a robot manipulator. The encoders are conventionally used to sense the joint angles, and then the direct kinematics is used to calculate the pose of the end-effector. When the encoders are malfunctioning, the pose may not be accurately determined. The proposed approach utilizes machine vision, where a single camera is fixed away from the base of the manipulator. Based on a single instant image, the kinematics of the manipulator, and the calibrated camera, the pose of the manipulator can be determined. Two examples demonstrate the proposed approach and show the feasibility. It is worth

to note that the proposed approach can be applied to any robot manipulator. This study selects a Scorbot robot as an illustrative example. This approach does not compete with the encoders. Instead, it will provide a reference or backup solution. This study is just to demonstrate the feasibility of the proposed approach, and there is more room to enhance its accuracy. For instant, it takes time to complete the image processing and to calculate the joint angles using the proposed approach. Besides, the estimation accuracy depends on the camera resolution. However, both problems can be overcome by upgrading the hardware.

REFERENCES

[1] M. R. Driels, W. Swayze, and S. Potter, "Full-pose calibration of a robot manipulator using a coordinate-measuring machine," *Int. J. Adv. Manuf. Technol.*, vol. 8, no. 1, pp. 34–41, 1993.
[2] H. N. Nguyen, J. Zhou, and H.-J. Kang, "A new full pose measurement method for robot calibration," *Sensors*, vol. 13, no. 7, pp. 9132–9147, 2013.
[3] A. Nubiola, M. Slamani, A. Joubair, and I. A. Bonev, "Comparison of two calibration methods for a small industrial robot based on an optical CMM and a laser tracker," *Robotica*, vol. 32, no. 3, pp. 447–466, 2014.
[4] D. Wang, Y. Bai, and J. Zhao, "Robot manipulator calibration using neural network and a camera-based measurement system," *Trans. Inst. Meas. Control*, vol. 34, no. 1, pp. 105–121, 2012.
[5] S. Zhang, Y. Ding, K. Hao, and D. Zhang, "An efficient two-step solution for vision-based pose determination of a parallel manipulator," *Robot. Comput.-Integr. Manuf.*, vol. 28, no. 2, pp. 182–189, 2012.
[6] B. P. Larouche and Z. H. Zhu, "Autonomous robotic capture of non-cooperative target using visual servoing and motion predictive control," *Auto. Robots*, vol. 37, no. 2, pp. 157–167, 2014.
[7] L. L. Huang, W. G. Li, Q. L. Yang, and Y. C. Chen, "Monocular camera calibration method for robot system," *Appl. Mech. Mater.*, vol. 741, pp. 697–700, Mar. 2015.
[8] R. A. Boby and S. K. Saha, "Single image based camera calibration and pose estimation of the end-effector of a robot," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2435–2440.
[9] A. Assa and F. Janabi-Sharifi, "Virtual visual servoing for multicamera pose estimation," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 2, pp. 789–798, Apr. 2015.
[10] J. Li, A. M. Kaneko, G. Endo, and E. F. Fukushima, "In-field self-calibration of robotic manipulator using stereo camera: Application to humanitarian demining robot," *Adv. Robot.*, vol. 29, no. 16, pp. 1045–1059, 2015.
[11] J. Miseikis, K. Glette, O. J. Elle, and J. Torresen. (2016). "Automatic calibration of a robot manipulator and multi 3D camera system." [Online]. Available: https://arxiv.org/abs/1601.01566
[12] L. Quan and Z. Lan, "Linear N-point camera pose determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 774–780, Aug. 1999.
[13] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 578–589, May 2003.
[14] Y. Wu and Z. Hu, "PnP problem revisited," *J. Math. Imag. Vis.*, vol. 24, no. 1, pp. 131–141, 2006.
[15] H. Zhou, T. Zhang, and W. Lu, "Vision-based pose estimation from points with unknown correspondences," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3468–3477, Aug. 2014.
[16] R. Katsuki, J. Ota, T. Arai, and T. Ueyama, "Proposal of artificial mark to measure 3D pose by monocular vision," *J. Adv. Mech. Design, Syst., Manuf.*, vol. 1, no. 1, pp. 155–169, 2007.
[17] S. Zhang, X. Cao, F. Zhang, and L. He, "Monocular vision-based iterative pose estimation algorithm from corresponding feature points," *Sci. China Inf. Sci.*, vol. 53, no. 8, pp. 1682–1696, 2010.
[18] J. Chen, Q. Zhang, and B. Zhang, "A quaternion pose determination solution based on monocular vision model," *Proc. SPIE*, vol. 8196, p. 819605, 2011.
[19] L. Y. Liu, M. Luo, and Y. M. Wu, "Study of pose detection algorithm for specific object based on monocular vision," *Appl. Mech. Mater.*, vols. 651–653, pp. 517–523, Sep. 2014.
[20] B. Wu, F. Zhang, and T. Xue, "Monocular-vision-based method for online measurement of pose parameters of weld stud," *Measurement*, vol. 61, pp. 263–269, Feb. 2015.
[21] Z. Guo, Q. Zhou, Z. Liu, and C. Liu, "Training-based head pose estimation under monocular vision," *IET Comput. Vis.*, May 2016, to be published, doi: 10.1049/iet-cvi.2015.0457.
[22] S. Sharma and S. D'Amico, "Comparative assessment of techniques for initial pose estimation using monocular vision," *Acta Astron.*, vol. 123, pp. 435–445, Jun./Jul. 2016.
[23] M. Andriluka, S. Roth, and B. Schiele, "Monocular 3D pose estimation and tracking by detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 623–630.
[24] A. Agarwal and B. Triggs, "Recovering 3D human pose from monocular images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 44–58, Jan. 2006.
[25] N. Andreff and P. Martinet, "Unifying kinematic modeling, identification, and control of a Gough–Stewart parallel robot into a vision-based framework," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1077–1086, Dec. 2006.
[26] P. Agarwal, S. Kumar, J. Ryde, J. J. Corso, and V. N. Krovi, "Estimating dynamics on-the-fly using monocular video for vision-based robotics," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 4, pp. 1412–1423, Aug. 2014.
[27] J. Denavit and R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," *Trans. ASME J. Appl. Mech.*, vol. 22, pp. 215–221, Jun. 1955.
[28] L. W. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*. Hoboken, NJ, USA: Wiley, 1999.
[29] G. A. F. Seber and C. J. Wild, *Nonlinear Regression*. Hoboken, NJ, USA: Wiley, 2003.

YONG-LIN KUO received the Ph.D. degree in mechanical engineering from the University of Toronto, Canada, in 2005. He is currently an Associate Professor with the National Taiwan University of Science and Technology, Taiwan. His research interests include system dynamics and control, robot manipulators, spacecraft dynamics and control, computational solid mechanics, kineto-elasto-dynamics, and enhanced finite element method.

BO-HAN LIU received the B.S. degree in electrical engineering from National Ilan University, Taiwan, in 2014, and the M.S. degree in automation and control from the National Taiwan University of Science and Technology, Taiwan, in 2016. His research interests include image processing, automatic control, robotics, and 3D printing.

CHUN-YU WU received the B.S. degree in mechanical engineering from Tamkang University, Taiwan, in 2012, and the M.S. degree in automation and control from the National Taiwan University of Science and Technology, Taiwan, in 2014. Since 2014, he has been a Mechanical Design Engineer with Pegatron Corp., Taiwan. His research interests include machine design, robotics, and machine vision.