Depth Estimation of Objects with Known Geometric Model for IBVS Using an Eye-in-hand Camera

Ju-Feng Wu
Department of Electrical Engineering
National Cheng Kung University, Tainan 701, Taiwan

Ming-Yang Cheng*
Department of Electrical Engineering
National Cheng Kung University, Tainan 701, Taiwan
E-mail: mycheng@mail.ncku.edu.tw

Abstract—**One of the most crucial and challenging issues in Image-Based Visual Servoing (IBVS) is the derivation of accurate depth information of feature points, which is indispensable in the calculation of the image Jacobian. For an object without a known geometric model, it is difficult to retrieve the depth information of feature points using a single camera. However, in many application scenarios such as industrial manufacturing processes, geometric models of artificial objects such as polyhedrons are known in advance. As a result, this paper aims at developing an estimation algorithm that can be used to retrieve the depth information of an object with a known geometric model. Several computer simulations and real experiments have been conducted to verify the effectiveness of the proposed depth estimation algorithm. In addition, a 6-DOF industrial robot is used to perform an eye-in-hand IBVS task with the aid of the proposed depth estimation algorithm.**

*Keywords— **Eye-in-hand, Image-based visual servoing (IBVS), Image Jacobian, Depth estimation***

I. INTRODUCTION

Since it was introduced in the late 1970s [1], visual servoing has gradually gained popularity in robotic applications. According to the renowned tutorial paper by Hutchinson *et. al.* [2], visual servoing can be divided into two structures — Position-Based Visual Servoing (PBVS) and Image-Based Visual Servoing (IBVS). In addition, there are two different camera configurations used in visual servoing — eye-in-hand and eye-to-hand. Later on, Chaumette *et. al.* [4,5] revisited the topic of “visual servo control” and made several amendments to the original tutorial paper on visual servo control. In addition to PBVS and IBVS, several other visual servoing structures such as hybrid visual servoing, partitioned visual servoing, and switching visual servoing have been proposed [4-6]. Furthermore, Janabi-Sharifi *et. al.* conducted an in-depth study on the comparison between PBVS and IBVS [7]. As the computing power of CPU keeps growing, so does the application of visual servoing. Recently, visual servoing has been applied to a variety of fields such as drones [8], industrial assembly lines [9], and agriculture harvesting [10].

This paper focuses on the problem of IBVS executed by a 6-DOF industrial robot equipped with a single eye-in-hand camera. One of the keys to the IBVS approach is the calculation of image Jacobian. In particular, the depth information of feature points of the object is essential in the calculation of image Jacobian, i.e. interaction matrix [11-16]. Many researchers were devoted to tackling this problem. For instance, Chaumette *et. al.* [11] used a mobile camera to retrieve the depth information of a 3D scene point. Luca *et. al* employ a nonlinear observer so as to estimate the depth of a feature point [12]. Another popular approach for estimating the depth of a feature point from an image sequence is the Kalman filter-based method [13]. Although these methods have been shown to be effective in practice, they all share a common drawback. That is, they rely on a sequence of images taken by a moving camera. In contrast, if the geometric model of an object is known, then it is possible to retrieve the depth information of feature points of an object using a single 2D image [14,15]. Recently, Zhao *et. al.* proposed a neural network based method to estimate the depth of an object which is essential when calculating the interaction matrix on line [16], while Chang *et. al.* proposed a parametric curve based approach to retrieve depth estimation [17].

In this paper, the geometric model of an artificial object such as a cuboid is assumed to be known. In particular, with the assumption that the dimensions of the six faces (i.e. rectangles) of a cuboid are known in advance, this paper proposes an algorithm to estimate the depth of any face of a cuboid. The main idea of the proposed algorithm is to use any three vertices (i.e. feature points) located on the same face of a cuboid. One can exploit the known dimension of the cuboid to perform depth estimation. In order to assess the performance of the proposed approach, a 6-DOF industrial robot is used to perform an eye-in-hand IBVS task. Experimental results verify the effectiveness of the proposed approach.

The remainder of the paper is organized as follows. Section II provides a brief review on the camera model and IBVS. Section III introduces the proposed depth estimation algorithm. Experimental results and conclusions are given in Section IV and V, respectively.

II. THE CAMERA MODEL AND IMAGE-BASED VISUAL SERVOING

A brief review on camera modeling and image-based visual servoing is provided in the following.

A. *Perspective projection and intrinsic parameters*

Suppose that a 3D scene point $P_c(x,y,z)$ is represented in the camera coordinate system (i.e. camera frame) and $P_f(u,v)$ is the

IV. Computer Simulation and Experimental Results

Computer simulation and several visual servoing experiments are performed to verify the effectiveness of the proposed approach.

A. Computer Simulation of Depth Estimation

In the computer simulation, this paper chooses any three known feature scene points, say $P_1(x_1,y_1,z_1)$, $P_2(x_2,y_2,z_2)$ and $P_3(x_3,y_3,z_3)$. Equations (17)~(19) are used to provide an iterative solution of depths, which are $\hat{z}_1$, $\hat{z}_2$, and $\hat{z}_3$.

In the image plane, the unit is "pixel". Therefore, the coordinates (u,v) in the image plane calculated using perspective projection should be integers. A total of three computer simulations are performed and simulation results are listed in Tables 1~3. The simulation results listed in Tables 1~3 indicate that the error in depth estimation is less than 2 %. Nevertheless, if the coordinates (u,v) in the image plane are allowed to be real numbers rather than restricted to integers, then the estimation accuracy can be further improved. Three computer simulations for the case of image coordinates represented in real numbers are listed in Tables 4~6. The simulation results listed in Tables 4~6 indicate that the estimation error is zero, suggesting the estimation error in Tables 1~3 comes from the fact that the image coordinates are restricted to integers.

TABLE I. 1st computer simulation of depth estimation (integer image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.8	0.4	0.1	0.0999	0.1%
2	–0.3	–0.5	0.2	0.2004	0.2%
3	0.5	–0.2	0.4	0.3993	0.175%

TABLE II. 2nd computer simulation of depth estimation (integer image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.3	0.08	0.8	0.8018	0.225%
2	0.4	–0.1	0.7	0.7063	0.9%
3	–0.2	0.3	0.6	0.5887	1.883%

TABLE III. 3rd computer simulation of depth estimation (integer image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.03	0.04	0.5	0.5002	0.04%
2	0.04	–0.1	0.6	0.6003	0.05%
3	0	0	0.3	0.3003	0.1%

TABLE IV. 1st computer simulation of depth estimation (real image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.8	0.4	0.1	0.1000	0
2	–0.3	–0.5	0.2	0.2000	0
3	0.5	–0.2	0.4	0.4000	0

TABLE V. 2nd computer simulation of depth estimation (real image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.3	0.08	0.8	0.80	0
2	0.4	–0.1	0.7	0.70	0
3	–0.2	0.3	0.6	0.60	0

TABLE VI. 3rd computer simulation of depth estimation (real image coordinates)

Point	*x*	*y*	*z*	Estimated depth	Error
1	0.03	0.04	0.5	0.50	0
2	0.04	–0.1	0.6	0.60	0
3	0	0	0.3	0.30	0

B. Feature Depth Estimation Experiment

In the feature depth estimation experiment, an FLEA3 FL3-U3-13E4C-C camera by Point Gray is mounted on the end-effector of an 6-DOF robot to capture the image of an object as shown in Fig. 2. The planar object used in the experiment is a 2cm×2cm black square as shown in Fig. 3. Suppose that the four vertices of the square have been detected and labeled (vertex 1, 2, 3 and 4 in Fig.4) using computer vision techniques. In each experimental run, at first the coordinates of vertex 1, 2, 3 are calculated using (17)~(19). Then the coordinates of vertex 1, 2, 4 are calculated using (17)~(19). The depth feature estimation experiments are performed three times and estimation results are listed in Table 7. As is apparent, in each experimental run, vertices 1 and 2 are calculated twice. The average values of vertices 1 and 2 in each experimental run are listed in Table 7. In addition, Table 8 lists the difference between the first depth estimation and the second depth estimation for vertices 1 and 2 in each experimental run. Table 8 shows that the estimation difference is less than 1.22%. It suggests that the proposed depth estimation approach is able to provide reliable results.

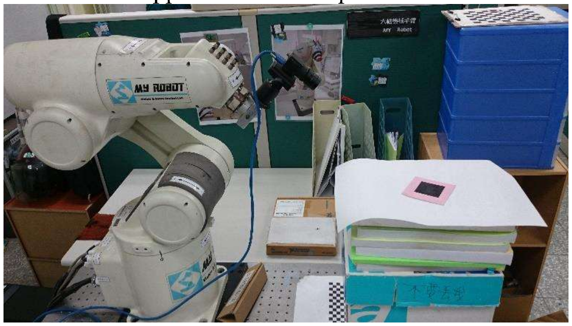

Fig. 2. The experimental system includes a 6-DOF industrial manipulator equipped with an eye-in-hand camera

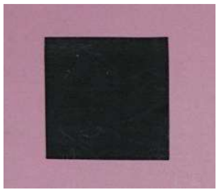

Fig. 3. The planar object used in the feature depth estimation experiment is a 2cm×2cm square.

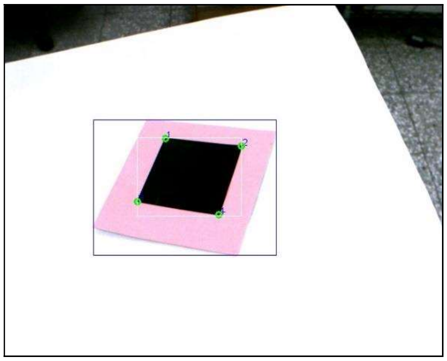

Fig. 4. Detection of vertices of the square

TABLE VII. Depth estimation results for vertex 1,2,3 and 4

Experiment	depth estimation of vertex 1 (m)	depth estimation of vertex 2 (m)	depth estimation of vertex 3 (m)	depth estimation of vertex 4 (m)
1	0.378398	0.391463	0.343311	0.355166
2	0.273016	0.289799	0.240187	0.25645
3	0.619527	0.641810	0.593967	0.608483

TABLE VIII. Difference between the first depth estimation and the second depth estimation for vertex 1 and 2

Experiment	depth estimation difference for vertex 1 (m)	depth estimation difference for vertex 2 (m)	depth estimation difference for vertex 1 (%)	depth estimation difference for vertex 2 (%)
1	0.001975	0.001353	0.52%	0.35%
2	0.001755	0.001166	0.64%	0.4%
3	0.007561	0.006559	1.22%	1.02%

C. Image-based Visual Servoing Experiment

Image–based visual servoing experiments are conducted on an experimental system consisting of a 6-DOF industrial robot manufactured by ITRI (Type AR 03; total weight 43 kg; maximum payload of end-effector: 7 kg) and an FLEA3 FL3-U3-13E4C-C camera by Point Gray is mounted on the end-effector of the robot (Fig. 3). In addition, an industrial PC (with Intel 2.8 GHz CPU) equipped with a CPU-based motion control card Imp-2 by ITRI is responsible for implementing the proposed visual servoing algorithm. Moreover, all the servo drives of the servomotors of the 6-DOF robot are set to torque mode since the direct IBVS is used in all visual servoing experiments. Note that the depth value used in the calculation of image Jacobian in all visual servoing experiments is set to the average depth value of the desired depth and the estimated depth value of the current position [4].

The control block diagram of the IBVS scheme used in the experiment is illustrated in Fig. 5. Note that the gravity compensation part in Fig. 5 is based on the approach developed in [18] and the velocity loop controller is of PI type. In addition, the vision loop controller is of P-type. The hand-eye calibration is performed using the approach proposed in [19]. Since the IBVS scheme is adopted in this paper, the accuracy of hand-eye calibration is not as crucial as that for PBVS. Results of the direct IBVS experiment are shown in Figs. 6~9. In particular, Fig. 6 shows the trajectories of four feature points on the image plane, while Fig. 7 shows the trajectory of the end-effector in the robot frame. In addition, the translational velocity and the angular velocity of the end-effector are shown in Fig. 8 and Fig. 9, respectively. The results shown in Fig. 6 indicate that the images of all four feature scene points converge to their corresponding desired positions on the image plane. Fig. 8 and Fig. 9 show that, after about 15 seconds, both the translation velocity and angular velocity of the end-effector are reduced to almost zero. Namely, the images of these four feature scene points have reached their desired positions on the image plane as shown in Fig. 6.

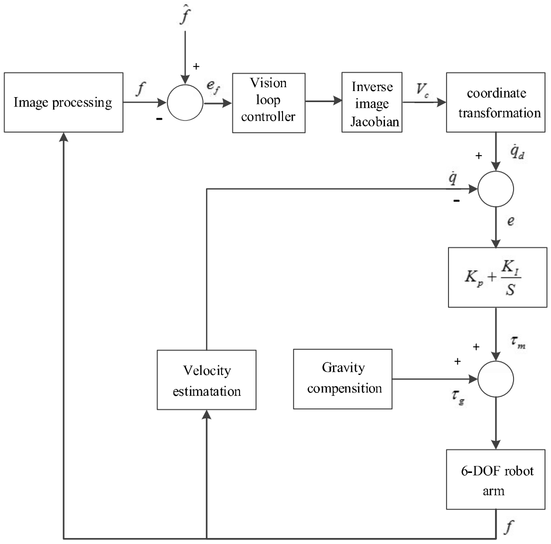

Fig. 5. Control block diagram of the IBVS scheme used in the experiment.

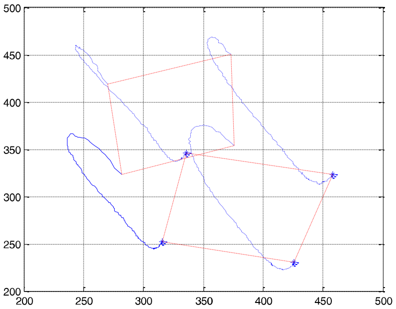

Fig. 6. Trajectories of four feature points on the image plane.

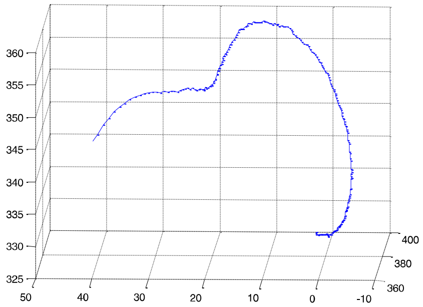

Fig. 7. Trajectory of the end-effector in the robot frame