Proceedings of the 2016 IEEE
International Conference on Robotics and Biomimetics
Qingdao, China, December 3-7, 2016

A New Image-Based Visual Servoing Method with Rotational Compensation

De Xu, Jinyan Lu, Peng Wang, Zhengtao Zhang, Dapeng Zhang, Zize Liang

Abstract—**A new image-based visual servoing method based on point features is presented to partially decouple the position and orientation controls. The control laws are designed for rotation and translation, respectively. Rotation takes priority over translation, but the movement of features caused by rotation is compensated in the translation control law. A monitor is designed to manage the adjustments for orientation and position. It limits the rotational amount for each step in the given range to ensure the features be in the camera's field of view. It also limits the translational amount for each step in the given range to ensure the reachability. Experimental results verify the effectiveness of the proposed method.**

Index Terms—**Interaction matrix, sensitive features, depth estimation, visual servoing, visual control**

I. Introduction

Visual servoing has been one of hot topics in the community of robotics and automation for 20 years at least [1][2]. There are three types of visual servoing systems, which are named to image-based visual servoing, position-based visual servoing, and hybrid visual servoing, according to the features used in the systems belonging to image space or Cartesian space. A position-based visual servoing system uses position and orientation errors to control the robot's motion. Current positions and orientations of the tracked objects are computed from their image features combining with the cameras' parameters including intrinsic and extrinsic parameters. Generally, stereo-vision is necessary for a position-based visual servoing system since 3-dimensional (3D) positions are required. Calibration errors in camara parameters, robot model and hand-eye relation have important influence on the control accuracy. The objects' poses are estimated in Cartesian space. The objects can be kept in the cameras' fields of view in the servoing process and the robot's trajectory is more reasonable and steady. An image-based visual servoing system uses image feature errors to control the robot's motion. It has the advantages of high accuracy over the position-based visual servoing system although its features do not directly describe the objects' positions and orientations. A hybrid visual servoing system combines the characteristics of image-based and position-based visual servoing systems [3]. Malis *et al* proposed a 2.5D (2.5-dimensional) hybrid visual servoing system. It controls the position with the image-based visual servoing method to achieve high accuracy and controls the orientation with the position-based visual servoing method in order to avoid servoing failure in the procedure of orientation adjustment. It can acheive the stability in the whole task space [4]. Many researchers developed a lot of visual servoing approaches and applied them in practice [2, 5-8]. For example, an adaptive visual servoing method was presented in [2] to realize automatically tracking with the depth-independent interaction matrix using common image features. The features' depths are estimated online. An image-based visual servoing approach for a fixed wing unmanned aerial vehicle to track the parallel power grid was developed by Mills *et al* [6]. The image-based visual servoing methods are regarded as the more potential methods in automation field because of their high accuracy, needless of 3D reconstruction and robustness with respect to the calibration errors of the cameras and robot. The stability of image-based visual servoing systems can be achieved if the objects can be viewed in whole visual servoing procedure. The main problem for image-based visual servoing systems is to avoid servoing failures since the objects are out of the cameras' field of view in the camera's rotation.

The features such as points, lines, distances, angles, and areas can be found in visual servoing systems. But it is very common for points to be used as image features in visual servoing systems [1, 3, 9-13]. For example, point features were used in the visual servoing for a manipulator with eye-in-hand vision system in [10] and eye-to-hand vision system in [11]. In this work, point features are employed too.

The motivation of this work is to develop a new visual servoing method to decouple the rotation and translation. The position and orientation control laws are separately designed. The features' changes resulting from rotational motions are compensated in the position control. A monitor is designed to limit the linear and angular velocities in the given ranges in order to keep the objects in the camera's field of view and ensure the robot's reachability.

The rest of this paper is organized as follows. The interaction matrix for points between the features' variation and the camera's motion is given in section II. The depth estimation for feature points is also provided. In section III, the image-based visual servoing system with rotational compensation is proposed. The control laws and the monitor design are presented. In section IV, experiments and results are provided. Finally, the paper is concluded in section V.

This work was supported in part by the National Natural Science Foundation of China under Grant 61227804, 61379097 and 61303177, and the special fund of Jiangsu Province for the transformation of scientific and technological achievements under Grant BA2015144.

D. Xu, J. Lu, P. Wang, Z. Zhang, D. Zhang are with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing 100190, China (e-mail: de.xu@ia.ac.cn).

Z. Liang is with the State Key Laboratory of Management and Control for Complex Systems, IACAS, Beijing 100190, China.

II. Interaction Matrixes for Point Features

A. Interaction matrixes

The intrinsic parameters of a camera can be described using pinhole model if the lens distortion is negligible or corrected in advance. For a point (x_c, y_c, z_c) in the camera's frame, its position projecting on the imaging plane with the normalized focal length is denoted as $(x_{1c}, y_{1c}, 1)$. Generally, (x_c, y_c, z_c) is unknown for monocular vision, while x_{1c} and y_{1c} are calculated from the image coordinates and the camera's intrinsic parameters.

$$z_c\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}k_x & 0 & u_0\\0 & k_y & v_0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_c\\y_c\\z_c\end{bmatrix}\Rightarrow\begin{bmatrix}x_{1c}\\y_{1c}\end{bmatrix}=\begin{bmatrix}x_c/z_c\\y_c/z_c\end{bmatrix}=\begin{bmatrix}(u-u_0)/k_x\\(v-v_0)/k_y\end{bmatrix} \tag{1}$$

where (u, v) is the image coordinates, (x_c, y_c, z_c) is the point's Cartesian coordinates in the camera's frame, $(x_{1c}, y_{1c}, 1)$ is the point's position projecting on the imaging plane with the normalized focal length 1. k_x, k_y, u_0 and v_0 are the camera's intrinsic parameters.

The variation of a point feature on the imaging plan is given in (2) when the camera moves [9].

$$\begin{bmatrix}\dot{x}_{1c}\\\dot{y}_{1c}\end{bmatrix}=L_{p1}\begin{bmatrix}v_{ca}\\\omega_{ca}\end{bmatrix} \tag{2}$$

where $v_{ca}=[v_{cax}, v_{cay}, v_{caz}]^{\mathrm{T}}$ and $\omega_{ca}=[\omega_{cax}, \omega_{cay}, \omega_{caz}]^{\mathrm{T}}$ are the translational and rotational velocities of the camera, L_{p1} is the interaction matrix for a point.

$$L_{p1}=\begin{bmatrix}-\frac{1}{z_c} & 0 & \frac{x_{1c}}{z_c} & x_{1c}y_{1c} & -(1+x_{1c}^2) & y_{1c}\\0 & -\frac{1}{z_c} & \frac{y_{1c}}{z_c} & 1+y_{1c}^2 & -x_{1c}y_{1c} & -x_{1c}\end{bmatrix} \tag{3}$$

B. Depth Estimation

It can be found from (3) that the depth z_c of a point in the camera's frame must be estimated in order to compute the interaction matrix L_{p1}. In fact, the depth of feature point can be estimated with the point's variations on the imaging plane and the camera's velocity after the camera moves. From (2) and (3), we have

$$\begin{aligned}z_c=&\frac{1}{2}\frac{x_{1c}v_{caz}-v_{cax}}{\dot{x}_{1c}-x_{1c}y_{1c}\omega_{cax}+(1+x_{1c}^2)\omega_{cay}-y_{1c}\omega_{caz}}\\&+\frac{1}{2}\frac{y_{1c}v_{caz}-v_{cay}}{\dot{y}_{1c}-(1+y_{1c}^2)\omega_{cax}+x_{1c}y_{1c}\omega_{cay}+x_{1c}\omega_{caz}}\end{aligned} \tag{4}$$

III. Image-based Visual Servoing

A. Traditional Control Law

For n points, we have the relations between the features' variations and the camera's motions as given in (5).

$$\begin{aligned}\begin{bmatrix}\dot{x}_{1c1}\\\dot{y}_{1c1}\\\vdots\\\dot{x}_{1cn}\\\dot{y}_{1cn}\end{bmatrix}&=\begin{bmatrix}-\frac{1}{z_{c1}} & 0 & \frac{x_{1c1}}{z_{c1}} & x_{1c1}y_{1c1} & -(1+x_{1c1}^2) & y_{1c1}\\0 & -\frac{1}{z_{c1}} & \frac{y_{1c1}}{z_{c1}} & 1+y_{1c1}^2 & -x_{1c1}y_{1c1} & -x_{1c1}\\\vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\-\frac{1}{z_{cn}} & 0 & \frac{x_{1cn}}{z_{cn}} & x_{1cn}y_{1cn} & -(1+x_{1cn}^2) & y_{1cn}\\0 & -\frac{1}{z_{cn}} & \frac{y_{1cn}}{z_{cn}} & 1+y_{1cn}^2 & -x_{1cn}y_{1cn} & -x_{1cn}\end{bmatrix}\begin{bmatrix}v_{cax}\\v_{cay}\\v_{caz}\\\omega_{cax}\\\omega_{cay}\\\omega_{caz}\end{bmatrix}\\&=L_{pn}\begin{bmatrix}v_{ca}\\\omega_{ca}\end{bmatrix}\end{aligned} \tag{5}$$

where L_{pn} is the interaction matrix for n points.

The traditional control law is

$$\begin{bmatrix}v_{cap}\\\omega_{cap}\end{bmatrix}=-\lambda_p L_{pn}^{+}\begin{bmatrix}\Delta x_{1c1}\\\Delta y_{1c1}\\\vdots\\\Delta x_{1cn}\\\Delta y_{1cn}\end{bmatrix} \tag{6}$$

where λ_p is the adjustment factor. L_{pn}^{+} is the pseudo-inverse matrix of L_{pn}. Δx_{1ci} and Δy_{1ci} are the position variation of i-th point projecting on the imaging plane with the normalized focal length 1.

The control law makes the errors reduce in exponent curve if the errors are small. However, the ill-conditioned interaction matrix L_{pn} will result in the large amount of adjustment ω_{cap} in rotation, which makes the point features be out of the camera's field of view. The visual servoing process would be failure once the point features were loss. In other words, the control system with the control law (6) is local stable in the case the errors are small. Its stability can not be ensured for all error spaces since the low rank or ill-condition of L_{pn} brings servoing failure.

B. Proposed Control Law

The control law (6) can make the errors reduce with exponent curve if the point features are kept in the camera's field of view. The camera's rotation will result in the point features change in a large range. Obviously, the point features can be kept in the camera's field of view if the rotation adjustment is limited in a reasonable range and the changes of point features formed by rotation are compensated. In the proposed control law, ω_{cap} computed from (6) is limited to the given range to control the rotation, that is, $\omega_{ca}=\omega_{cap}$. The sum of v_{cap} computed from (6) and v_{caw} computed from the rotational motions is employed to control the translation, that is, $v_{ca}=v_{cap}+v_{caw}$. The variations of the point features projecting on the imaging plane resulting from the camera's rotation can be calculated as

$$[\dot{x}_{1c1w} \quad \dot{y}_{1c1w} \quad \cdots \quad \dot{x}_{1cnw} \quad \dot{y}_{1cnw}]^T = L_{wn}\begin{bmatrix}\omega_{cax}\\ \omega_{cay}\\ \omega_{caz}\end{bmatrix} \tag{7}$$

where L_{wn} is the interaction matrix between point features and the camera's rotations for n points, $(\dot{x}_{1ciw}, \dot{y}_{1ciw})$ are the variations of the i-th point feature on the imaging plane resulting from the camera's rotation.

$$L_{wn} = \begin{bmatrix} x_{1c1}y_{1c1} & -(1+x_{1c1}^2) & y_{1c1} \\ 1+y_{1c1}^2 & -x_{1c1}y_{1c1} & -x_{1c1} \\ \vdots & \vdots & \vdots \\ x_{1cn}y_{1cn} & -(1+x_{1cn}^2) & y_{1cn} \\ 1+y_{1cn}^2 & -x_{1cn}y_{1cn} & -x_{1cn} \end{bmatrix} \tag{8}$$

The compensation of rotational motions is computed from the variations of point features on the imaging plane and their interaction matrix.

$$\begin{bmatrix} v_{cawx} \\ v_{cawy} \\ v_{cawz} \end{bmatrix} = -L_{vn}^{+}\begin{bmatrix} \dot{x}_{1c1w} \\ \dot{y}_{1c1w} \\ \vdots \\ \dot{x}_{1cnw} \\ \dot{y}_{1cnw} \end{bmatrix} = -L_{vn}^{+}L_{wn}\begin{bmatrix}\omega_{cax}\\ \omega_{cay}\\ \omega_{caz}\end{bmatrix} \tag{9}$$

where L_{vn}^{+} is the pseudo-inverse matrix of interaction matrix L_{vn} between point features and the camera's translations, v_{caw}= $[v_{cawx}, v_{cawy}, v_{cawz}]^T$ is the compensation velocity to compensate the variations of point features caused by rotation motions.

$$L_{vn} = \begin{bmatrix} -1/z_{c1} & 0 & x_{1c1}/z_{c1} \\ 0 & -1/z_{c1} & y_{1c1}/z_{c1} \\ \vdots & \vdots & \vdots \\ -1/z_{c1} & 0 & x_{1cn}/z_{cn} \\ 0 & -1/z_{c1} & y_{1cn}/z_{cn} \end{bmatrix} \tag{10}$$

In fact, $v_{ca} = L_{vn}^{+}[\dot{x}_{1c1} \quad \dot{y}_{1c1} \quad \cdots \quad \dot{x}_{1cn} \quad \dot{y}_{1cn}]^T - L_{vn}^{+}L_{wn}\omega_{ca}$ can be derived from (5). Its first term is replaced by v_{cap}, its second term is the compensation term as given in (9).

C. Monitor Design

A monitor is designed to check and adjust the computed values of linear and angular velocities. The adjustment factor λ_p is adjusted according to the maximum velocities, which makes the maximum velocities be in the given range. The factor k indicating whether the velocities are out of range is calculated in (11). The adjustment factor λ_p is modified as given in (12).

$$k = \max\left(\|v_{ca}\|/v_{caT}, \|w_{ca}\|/w_{caT}\right) \tag{11}$$

where v_{caT} is the given linear velocity threshold, w_{caT} is the given angular velocity threshold, k is a factor.

$$\lambda_p = \begin{cases} \lambda_{pd}/k, & k>1 \\ \lambda_{pd}, & k \le 1 \end{cases} \tag{12}$$

where λ_{pd} is the initial set value of λ_p.

The linear and angular velocities v_{ca} and w_{ca} are recalculated after λ_p is modified. The new values of v_{ca} and w_{ca} are in the given ranges $[-v_{caT}, v_{caT}]$ and $[-w_{caT}, w_{caT}]$, respectively.

D. Control System Design

A new image-based visual servoing system is designed [14-16]. It consists of 6 blocks such as orientation control, position control, visual sensing, depth estimation, monitor and robot. Its block diagram is as given in Fig. 1. The desired point features are denoted as F_{pd}=(x_{1cd1}, y_{1cd1}, ..., x_{1cdn}, y_{1cdn}) and current ones F_{pc}=(x_{1c1}, y_{1c1}, ..., x_{1cn}, y_{1cn}). The angular velocity ω_{ca} is computed using (6) according to the errors between F_{pd} and F_{pc} and the interaction matrix L_{pn} for point features. It is used to control the robot's rotation to adjust the camera's orientation. The compensated translational velocity v_{caw} resulting from the camera's rotation is computed using (9) according to the angular velocity ω_{ca} and the interaction matrixes L_{vn} and L_{wn}. The translational velocity v_{cap} is calculated in (6). The sum of v_{caw} and v_{cap} is considered as the translational velocity to be sent to the robot to adjust the camera's position. The depth estimation block is used to estimate the depths of feature points using (4) with known increments instead of moving velocities. The visual sensing block captures images and extracts the features of points. The monitor checks whether the translational and rational velocities are out of given ranges or not. The factor λ_p is changed according to (11) and (12). The robot is the plant of the control system, which executes the motion commands from the orientation and position control blocks.

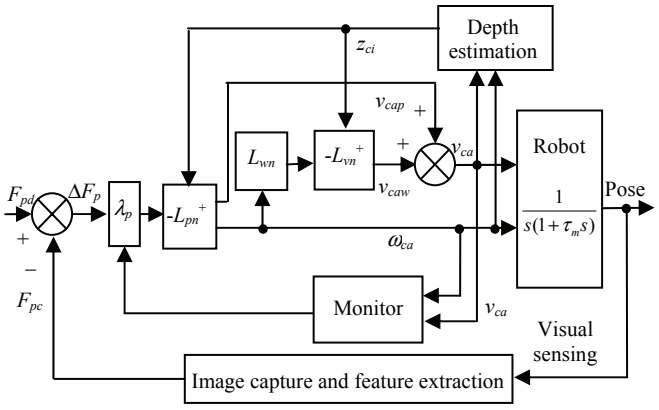


Fig. 1 Block diagram of the proposed visual control system

It is necessary to select enough points to ensure the interaction matrix L_{pn} is full rank. It seems from (5) that three points are enough to let L_{pn} be full rank. In fact, the problem to ensure the full rank L_{pn} can be considered to PnP (perspective n points) problem. The PnP problem has multiple solutions, which will cause control problem. Four coplanar points in which any three points are not collinear can ensure the solution is unique [17]. Therefore, four feature points at least are

necessary for the proposed visual servoing system. Of course, more feature points are helpful to increase the computation accuracy.

IV. Experiment and Results

A. Experiment System and Calibration

The experiment system consisted of a manipulator, a camera, a computer and a whiteboard, as shown in Fig. 2. The manipulator was UP6, a 6-DOF robot produced by Yaskawa Company. The camera was fixed on the end-effector of the manipulator, which formed an eye-in-hand vision system. The camera was MER-200-14GX produced by Daheng Company, its image size was 1626×1238 in pixel. The object to be tracked is a rectangle pattern attached on the whiteboard. The computer was used to deal with image processing and visual servoing algorithms.

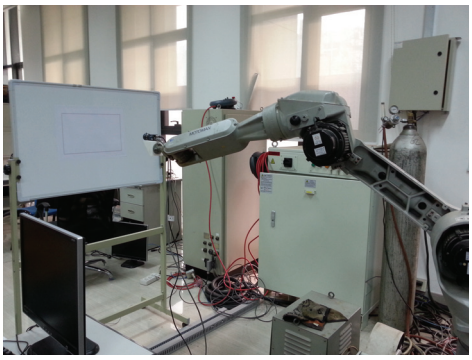


Fig. 2 The experimental system

The camera and the hand-eye relation were calibrated before experiments. The 4-order Brown distortion model was adopted for the lens distortion. The hand-eye parameters were denoted as the transformation matrix from the end-effector coordinates of the manipulator to the camera coordinates. The calibration results were as follows.

$$M_{in}=\begin{bmatrix}1896.56 & 0 & 803.91\\ 0 & 1905.84 & 610.19\\ 0 & 0 & 1\end{bmatrix} \tag{13}$$

where M_{in} was the intrinsic parameters matrix.

$$(k_{c1},k_{c2},k_{c3},k_{c4})= \\ (-0.118454,\ 0.249453,\ -0.000444,\ -0.002901) \tag{14}$$

where k_{c1} to k_{c4} were the distortion factors of the lens.

$$M_{ex}=\begin{bmatrix}-0.0230 & -0.9722 & 0.0132 & -17.93\\ 0.9963 & -0.0016 & 0.0311 & -72.55\\ -0.0259 & -0.0385 & 0.9886 & -149.64\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{15}$$

where M_{ex} was the hand-eye parameters matrix, the unit of position vector was mm.

Before the experiment, the end-effector of the manipulator was firstly moved to the rectangle pattern to capture the desired image. The 4 corners were selected as the image features. The desired point features were computed using (1) with the intrinsic parameters and the image coordinates of the corners, as listed in table I.

TABLE I The Desired Point Features

Point	Point 1	Point 2	Point 3	Point 4
u, v (pixel)	151.1 261.2	1435.7 245.0	1441.0 1020.7	167.4 1055.2
x_{1c}, y_{1c} (mm)	-0.3442, -0.1831	0.3331, -0.1916	0.3359, 0.2154	-0.3356, 0.2335

In experiments, the manipulator could only accept pose commands. The computed camera velocity was converted to increament via mutiple the cycle time T_c. Then it was transformed to the end-effector's motion of the manipulator with (16).

$$T_e = M_{ex}T_{ca}M_{ex}^{-1} \tag{16}$$

where T_e is the matrix formed with the end-effector's motion, T_{ca} is the matrix formed with the camera's motion.

B. Approaching Experiment with the Proposed Method

In experiments, the error threshold of point features was 0.02mm on the imaging plane with the normalized focal length 1. The orientation adjustment of one step for each direction was limited to 5°. The translation adjustment of one step for each direction was limited to 200mm. The adjustment factor including the sampling cycle time, $\lambda_p T_c$, was set to 0.6. The adjustments finished once the errors were smaller than the thresholds above.

The experiments were well conducted with the proposed control laws as described above. At first, the end-effector was actively moved in order to estimate the feature points' depths. Once the depths of feature points were estimated, the visual servoing could be conducted. In the following steps, the current features were extracted and their errors to desired ones were computed. Then the angular and linear velocities were computed as described in section III with (6) and (9). The end- effector was well approached to the desired position with desired orientation relative to the pattern in experiments. The results in one experiment were shown in Fig. 3 to 6. Fig. 3 showed the feature points' errors on the imaging plane with the normalized focal length 1. Fig. 4 showed the feature points' coordinates on the image plane. The solid symbols indicated the desired positions. Fig. 5 displayed the position and orientation errors of the end-effector. Fig. 6 was the end-effector's trajectory. The approaching process finished after 15 steps. The errors of the end-effector after the approaching were (4.11mm, -3.64mm, 2.73mm, -0.99°, -0.14°, -0.72°).

C. Approaching Experiment with Traditional Method

The block diagram of the traditional control system was designed as shown in Fig. 7. The experiment conditions including the initial pose of the end-effector, the threshold, the

adjustment factor were as same as ones in the experiments with the proposed method.

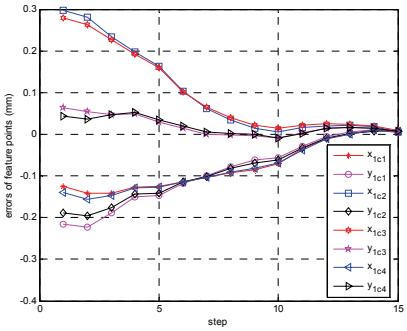


Fig. 3 Feature points' errors on the imaging plane with the normalized focal length.

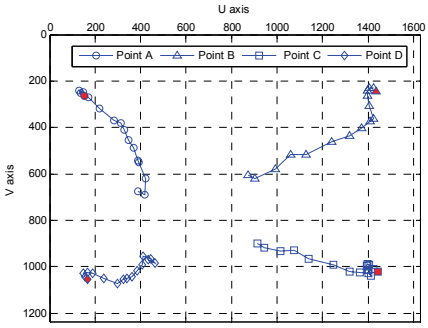
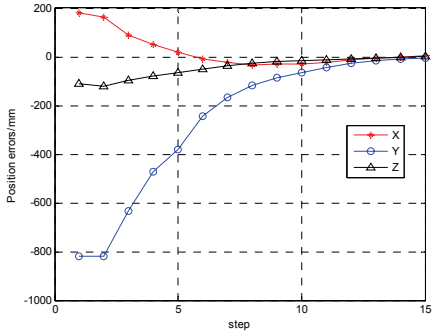
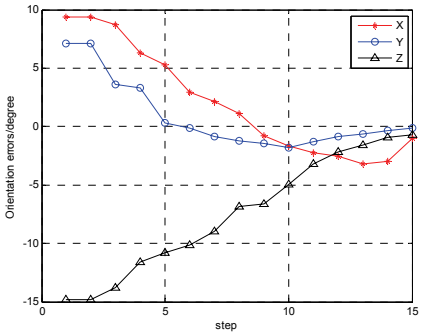


Fig. 4 Feature points' trajectories on the image plane. The solid symbols indicated the desired positions.



(a)



(b)

Fig. 5 Position and orientation errors of the end-effector, (a) position errors, (b) orientation errors.

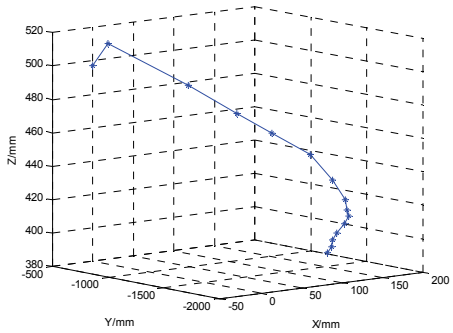


Fig. 6 The end-effector's trajectory.

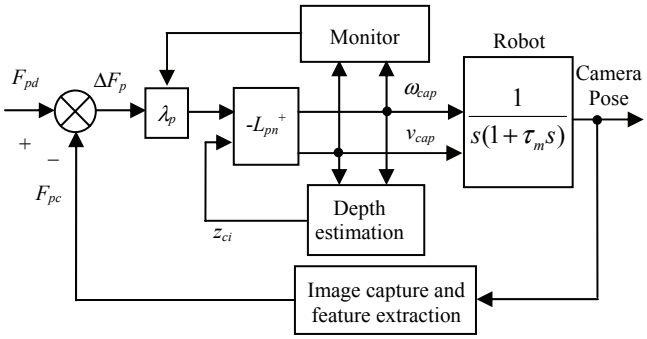


Fig. 7 Block diagram of traditional visual control system

The angular and linear velocities were computed with the traditional method as described in (6). The end-effector was also approached to the desired position with desired orientation relative to the pattern in experiments. The results in one experiment were shown in Fig. 8 to 11. Fig. 8 showed the feature points' errors on the imaging plane with the normalized focal length 1. Fig. 9 showed the feature points' coordinates on the image plane. The solid symbols indicated the desired positions. Fig. 10 displayed the position and orientation errors of the end-effector. Fig. 11 was the end-effector's trajectory. The approaching process finished after 19 steps. The errors of the end-effector after the approaching were (3.73mm, 7.02mm, -14.11mm, -1.38°, 0.55°, -0.84°).

It can be found from the experimental results shown in Fig. 3 ~ 6 and Fig. 8 ~ 11 that the trajectories of the end-effector and the point features in the experiment with the proposed method were more steady than ones in the experiment with the traditional method. The former trajectories were without fluctuations and converged quickly.

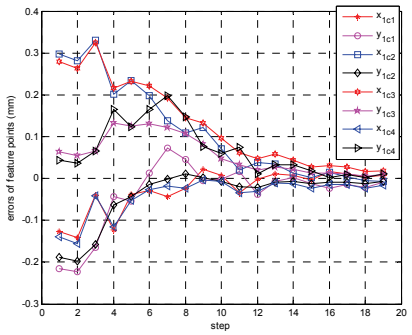


Fig. 8 Feature points' errors on the imaging plane with the normalized focal length in an experiment with the traditional method.

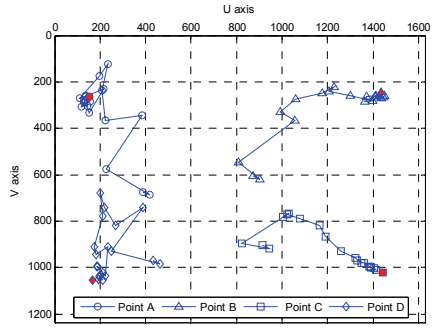
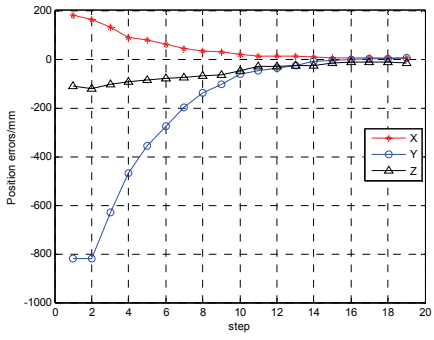
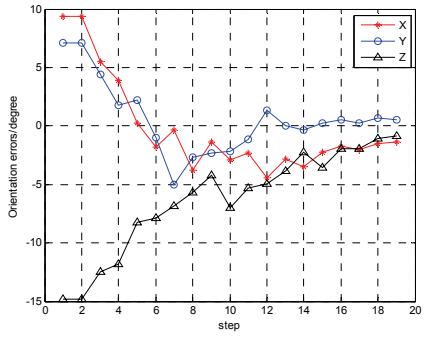


Fig. 9 Feature points' trajectories on the image plane in an experiment with the traditional method. The solid symbols indicated the desired positions.



(a)



(b)

Fig. 10 Position and orientation errors of the end-effector in an experiment with the traditional method, (a) position errors, (b) orientation errors.

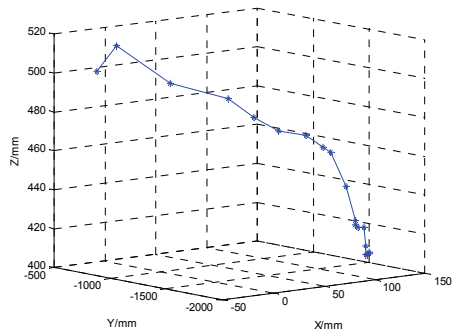


Fig. 11 The end-effector's trajectory in an experiment with the traditional method.

V. Conclusion

A new image-based visual servoing method with partially decoupled position and orientation controls is developed. The orientation control is directly realized with point features and interaction matrix. The corresponding translations resulting from rotational motions are introduced into the position control as compensation. The linear and angular velocities are limited to the given ranges in order to ensure the objects being kept in the camera's field of view and the camera being moved to the desired pose. The advantages of the proposed method are as follows. The orientation control is independent of the position control, which has the merit that the position controller and orientation controller can be separately designed. The translation compensation of rotation is very helpful to keep the tracked objects in the camera's field of view.

References

[1] G. D. Hager, S. Hutchinson, P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, 1996.

[2] Y.-H. Liu, H. Wang, W. Chen, D. Zhou, "Adaptive visual servoing using common image features with unknown geometric parameters," *Automatica*, vol. 49, no. 8, pp. 2453-2460, 2013.

[3] E. Malis, F. Chaumette, S. Boudet, "2-1/2-D visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238-250, 1999.

[4] D. Kragic, H. Christensen, "Survey on visual servoing for manipulation," Computational Vision and Active Perception Laboratory Tech. Rep., 2002.

[5] F. Alkhalil, C. Doignon, "Stereo visual servoing with decoupling control," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1671-1676.

[6] S. Mills, N. Aouf, L. Mejias, "Image based visual servo control for fixed wing UAVs tracking linear infrastructure in wind," *Proceedings of IEEE International Conference on Robotics and Automation*, 2013, pp. 5769 - 5774.

[7] R. Pieters, Z. Ye, P. Jonker, H. Nijmeijer, "Direct motion planning for vision-based control," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 4, pp. 1282-1288, 2014.

[8] D. Kruse, J. T. Wen, R. J. Radke, "A sensor-based dual-arm tele-robotic system," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 4-18, 2015.

[9] B. Espiau, F. Chaumette, P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313-326, 1992.

[10] Y. Liu, H. Wang, "An adaptive controller for image-based visual servoing of robot manipulators," *Proceedings of the World Congress on Intelligent Control and Automation*, 2010, pp. 988-993.

[11] H. Wang, Y.-H. Liu, W. Chen, "Visual tracking of robots in uncalibrated environments," *Mechatronics*, vol. 22, no. 4, pp. 390- 397, 2012.

[12] T. Drummond, R. Cipolla, "Real-time visual tracking of complex structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 932-946, 2002.

[13] A. I. Comport, E. Marchand, M. Pressigout, F. Chaumette, "Real-time markerless tracking for augmented reality the virtual visual servoing framework," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615-628, 2006.

[14] S. Liu, D. Xu, D. Zhang, Z. Zhang, "High precision automatic assembly based on microscopic vision and force information," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 382-393, 2016

[15] D. Xu, M. Tan, Y. Liu, J. Wang, "Control strategy for a low cost manipulator to transport and align IC mask-plates," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1018-1027, 2009.

[16] Y. H. Liu, H. Wang, C. Wang, and K. K. Lam, "Uncalibrated visual servoing of robots using a depth-independent interaction matrix," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 804–817, Aug. 2006.

[17] D. Xu, Y. F. Li, M. Tan, "A general recursive linear method and unique solution pattern design for the perspective-*n*-point problem," *Image and Vision Computing*, vol. 26, no. 6, pp. 740-750, 2008.