Image Based Visual Servo Application on Video Tracking with Monocular Camera Based on Phase Correlation Method

Yoshi Ri*a) Student Member, Hiroshi Fujimoto*b) Senior Member

Video tracking control can be applied to teaching motion in a robot. It contains two issues: reference extraction process from video and tracking control process. In this paper, image based visual servo control is used in reference tracking, and feature extraction based on robust phase correlation method is utilized in reference extraction process. Under 4-DOF motion constraints and for a planer object, proposed control scheme uses feedforward control with the time-invariant image jacobian derived in this paper, and evaluated experimentally.

Keywords: Visual Servoing, Phase Only Correlation, Distance Estimation

1. Introduction

Image sensor became widely used in a robot along with its miniaturization and improvement of computer performance. Since image contains rich information on environment, there are many researches on guide or control robot from images.

Visual navigation or path following is one of the attractive theme which contains both recognition process and control process. Image based navigation often uses some geometric feature such as lines [1] or points marker [2] to help recognizing environment. Therefore, robust and accurate geometric feature extraction method such as SIFT [3] or ORB [4] is important for robots to control their motion from image.

On the other hand, some interesting researches use direct approach, which utilize information of image pixels, to determine robot move and achieve path following problem [5] [6]. However, these methods consider the reference path only and do not care timing information.

In this paper, we assume a plane object and a robot has 4 DOF including 3D translation and rotation around optical axis. This assumption simulates for UAV guidance or robot teaching such as a precise positioning stage.

The goal is to reproduce motion from video; in other words, desired task is to match current image with reference video frame in same time scale. To accomplish this task, control method based on image based visual servoing [7] was applied.

Proposed solution has two process: feature extraction process from reference video and tracking control process using visual servoing. First, robust feature extraction algorithms based on phase correlation [8] [9] is proposed. Then, the image reference are extracted from video using the proposed algorithms with certain key frame. Finally, tracking control is achieved on proposed effective feedforward visual servoing scheme.

a) Correspondence to: ri14@hflab.k.u-tokyo.ac.jp
b) Correspondence to: fujimoto@k.u-tokyo.ac.jp
* The University of Tokyo
5-1-5, Kashiwanoha, Kashiwa, Chiba, 227-8561 Japan
Phone: +81-4-7136-3881
Fax: +81-4-7136-3881

2. Image Based Visual Servoing

Visual servoing is a control technique that take some information from a camera image as a feedback and then decide a robot move iteratively. There are roughly two approaches, one is position based visual servoing and the other is image based visual servoing [7].

In a position based approach, relative camera pose detected from visual measures is used as a control input. Although an adequate 3D trajectory can be obtained from this approach, vision based pose estimation often has uncertainty and the error in 3D reconstruction directly affect performance. On the other hand, image based approach can suppress this 3D reconstruction error by using an image feature as a control input. Instead, 3D trajectory with image based approach sometimes became undesirable one, this problem can be avoided using combined advanced approach [10]. The control scheme in this paper is based on image based method.

In image based visual servoing, we need to define what kind of image feature $\boldsymbol{\xi}$ to be used as control input. Once an image feature $\boldsymbol{\xi}$ is chosen, reference velocity of robot $\boldsymbol{V}_{ref}$ can be calculated as below:

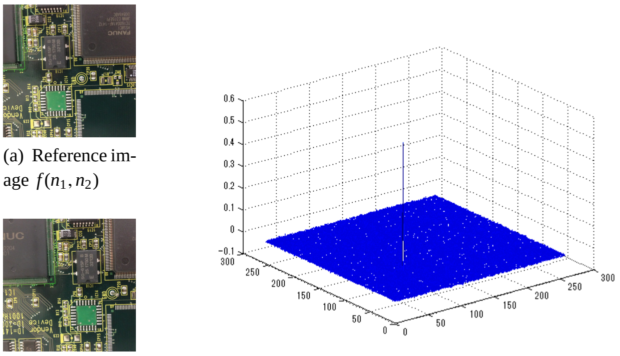
$$\boldsymbol{V}_{ref} = -\lambda \boldsymbol{J}^{+} \left(\boldsymbol{\xi}_{ref} - \boldsymbol{\xi}_{c}\right) \tag{1}$$

while λ means a feedback gain, $\boldsymbol{\xi}_{ref}$ and $\boldsymbol{\xi}_{c}$ are desired image features and current one. $\boldsymbol{J}^{+}$ means pseudo inverse of $\boldsymbol{J}$, which is called image jacobian matrix representing for relationships between infinitesimal displacement of image feature $\dot{\boldsymbol{\xi}}$ and camera velocity $\boldsymbol{V}$ in (2).

$$\boldsymbol{J}\boldsymbol{V} = \dot{\boldsymbol{\xi}} \tag{2}$$

generally, $\boldsymbol{J}$ can be expressed as function of relative pose $\boldsymbol{X}$ and image feature $\boldsymbol{\xi}$. Since the relative camera pose is time-variant and often hard to estimate from visual measures, it is also difficult to estimate this image jacobian $\boldsymbol{J}$ in (1).

For example, when using a set of feature points coordinates $\boldsymbol{\xi}_i = [x_i, y_i]^T$ $(i = 1, 2, 3..., n)$ as an image feature, $\boldsymbol{\xi}$ and camera velocity $\boldsymbol{V} = [v_x\ v_y\ v_z\ \omega_x\ \omega_y\ \omega_z]^T$ have a following



(a) Reference image $f(n_1, n_2)$

(b) Compared image $g(n_1, n_2)$

(c) POC function $r(k_1, k_2)$

Figure 1: The appearnce of POC function

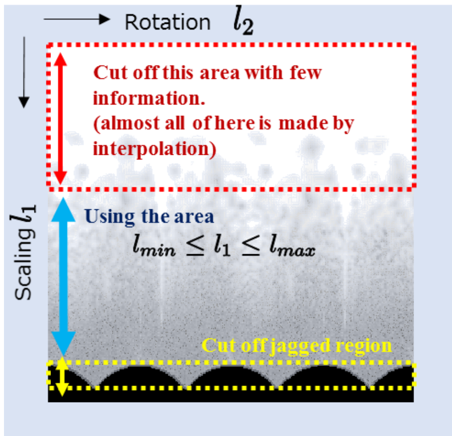


Figure 2: The appearance of $F_{LP}(l_1, l_2)$ and the way of filtering.

relationship:

$$\dot{\xi}_i = J_i V$$

$$J_i = \begin{bmatrix} \frac{-1}{Z_i} & 0 & \frac{x_i}{Z_i} & x_i y_i & -(1+x_i{}^2) & y_i \\ 0 & \frac{-1}{Z_i} & \frac{y_i}{Z_i} & 1+y_i{}^2 & -x_i y_i & -x_i \end{bmatrix} \quad (3)$$

where Z_i is the distance between a camera and each point, that is often unavailable from monocular camera measures. So, in many cases, this distance Z_i is often approximated as a suitable constant value Z_0, which means the reference distance in reference frames [7].

3. Reference Extraction from Video Frames

In a video tracking task, it is very important to decide how to descrive the reference path as some parameters. Although a set of feature points coordinates is a manageable geometric feature, there are measurement problem and matching problem. For example, we will suffer from feature mismatching and losing caused by occlusion or large move.

In our problem setting, camera move is restricted to 4-DOF, containing three degree of translational move and rotational move around depth axis, image transformation parameters including image translation, rotation and scaling are used as a video reference. Under this condition, a technique based on the frequency domain of the image sometime called as phase correlation is effective. A practical displacement estimation algorithm is shown in this chapter.

3.1 Image transformation parameter detect using phase correlation Phase-Only-Correlation (POC) is one of the image registration techniques using spatial frequency phase information of image with 2D Discrete Fourier Transformation (DFT) [9]. With this technique, we can obtain translational error between two images in subpixel order. It also can be expanded to detect rotation and scaling error using. This method does not need any feature extraction and matching process, so it is suitable for real-time processing. The principle of the algorithm is shown at [9], so we show only algorithm procedure.

3.1.1 Translational Displacement Detection Consider two $N \times N$ images $f(n_1, n_2)$, $g(n_1, n_2)$. Then the frequency domains of these images transformed by 2D DFT are written as $F(k_1, k_2)$, $G(k_1, k_2)$. The cross spectrum $R(k_1, k_2)$ between $F(k_1, k_2)$ and $G(k_1, k_2)$ can be written as below

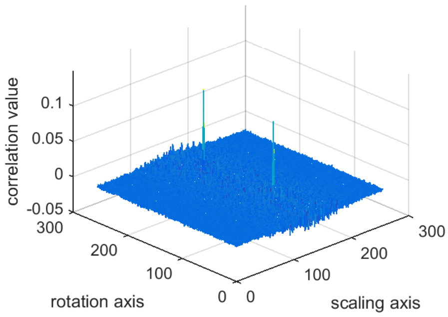


Figure 3: An example of POC function between $F_{LP}(l_1, l_2)$ and $G_{LP}(l_1, l_2)$. Two peaks shows there are an ambiguity in rotation detection.

$$R(k_1, k_2) = \frac{F(k_1, k_2)\overline{G(k_1, k_2)}}{|F(k_1, k_2)\overline{G(k_1, k_2)}|} \quad (4)$$

where $\overline{G(k_1, k_2)}$ is a complex conjugate of $G(k_1, k_2)$. POC function $r(n_1, n_2)$ can be calculated by applying 2D IDFT to $R(k_1, k_2)$.

If the reference image and current image is similar, then POC function $r(n_1, n_2)$ has a sharp peak as shown in Fig. 1. The coordinate of the peak represents for the image displacement between the reference and the current image. Therefore, image displacement parameters can be detected by applying 2D IDFT to the multiples of phase information of two images. There are some additional work such as filtering and function fitting in order to estimate more accurate translation.

3.1.2 Rotation and Scaling Detection Rotational and scaling displacements can be detected from the amplitudes of frequency domains $|F(k_1, k_2)|$ and $|G(k_1, k_2)|$ [9]. By using log-poler transformation shown in the algorithm1, the rotational and scaling displacements θ and κ can be converted to transitional displacements δ_x and δ_y.

The relationship between θ, κ and δ_x, δ_y is shown in (5).

$$(\delta_x, \delta_y) = (N\theta/\pi, -N \log_N \kappa) \quad (5)$$

3.1.3 Affine Parameter Detection Flow Algorithm1 shows the detection flow of proposed algorithm. There are two problems in practical use of conventional known method [9]. One is that the method shown in 3.1.2 cannot specify rotation because it detects 2 peaks in Fig. 3 that means 180 degree

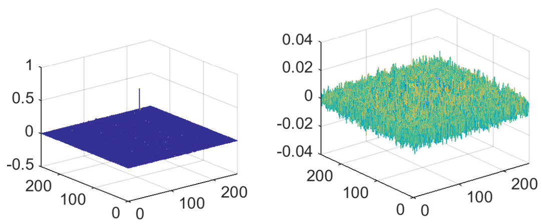
Algorithm 1 Practical image transformation $(\xi_x, \xi_y, \theta, \kappa)$ detection algorithm

Step1
Calculate 2D fourier spectrum of $f(n_1, n_2)$ and $g(n_1, n_2)$, and get amplitudes $F(k_1, k_2)$ and $G(k_1, k_2)$. Then, take the logarithm of spectrum amplitudes $\log|F(k_1, k_2)| + 1$ and $\log|G(k_1, k_2)| + 1$.

Step2
Apply log-poler transformation to spectrum amplitudes and get $|F_{LP}(l_1, l_2)|$ and $|G_{LP}(l_1, l_2)|$. This transformation can be written as $F_{LP}(l_1, l_2) = F_p(r\cos\phi + c_1, r\sin\phi + c_2)$, while $r = N^{l_1/N}$ and $\phi = 2\pi l_2/N$ means radius and angle from the center of F_p and c_1, c_2 is the center coordinates of F_p. Fig. 2 is an example of this transformation and shows filtering method for robust and accurate estimation. The low frequency region is almost made of interpolation and the high frequency region has noise and unnecessary area, so we use cut off filter shown in Fig. 2 and each cut off frequency $l_{min} = N\log_N N/2\pi$ and $l_{max} = N\log_N N/2$.

Step3
Rotation and scaling between $f(n_1, n_2)$ and $g(n_1, n_2)$ can be converted as translational error between $F_{LP}(l_1, l_2)$ and $G_{LP}(l_1, l_2)$. By using translational detection in 3.1.1 and (5),rotation and scaling can be detected. However, as shown in Fig. 3 there is an ambiguity in rotation detection that 180 degrees rotated value is also detected.

Step4
Finally, correct $g(n_1, n_2)$ with detected scaling and one of rotation and get get corrected image $g'(n_1, n_2)$. If the rotation estimation is proper, then $g'(n_1, n_2)$ has only translational error from $f(n_1, n_2)$ and detect it. Otherwise, detect translational error between $g'(n_1, n_2)$ and $f_{inv}(n_1, n_2)$, which is 180 degree rotated image of $g'(n_1, n_2)$, and then reverse positive and negative. Fig. 4 shows the false detection can be sorted by looking at peak value of POC function r.



(a) When detection is succeeded. (b) When detection is failed.

Figure 4: Appearance of POC function r when detection is (a) succeeded or (b) failed. Existence of the peak shows the detection succeeded

rotated rotation is also detected. And the other problem is that the robustness and accuracy of rotation and scaling detection in 3.1.2 highly depend on image filtering in the [Step3] in algorithm1.

Former problem can be solved by matching with a normal image and a 180 degrees rotated image and an practical adequate filtering method are shown in algorithm1. The image transformation parameters detected in this method is sometimes called affine parameters, but remember the translational displacement parameter detected in this method is described on the desired image frame, which is later become important in a derivation of image jacobian

3.2 Reference Trajectory Description Based on Key Frame Proposed method uses image displacement parameters between a particular reference image and current video frames.

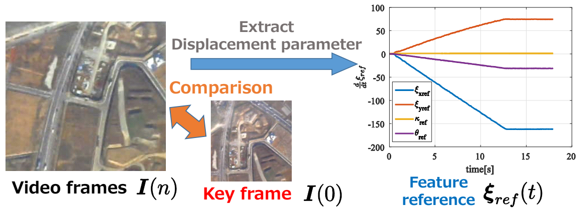


Figure 5: The process to extract image feature reference $\xi_{ref}(t)$ from video frames and feature reference got from this process

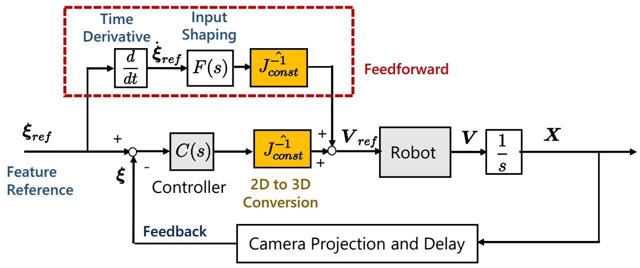


Figure 6: Block diagram of the proposed method.

More specifically, when n_{th} frame of video is defined as $\boldsymbol{I}(n)$, the reference feature at n_{th} frame $\xi_{ref}(t)$ is described as the image displacement parameter from a specific frame called key frame $\boldsymbol{I}(0)$. In proposed method, the first frame of the video $\boldsymbol{I}(1)$ is chosen as a key frame. Fig. 5 shows the reference extraction flow.

3.3 Reference Concussion Using Relay Images When robot moves further from initial pose, it often happens that current frame cannot be directly compared with the key frame. In that case, simple reference concussion theory using relay images is used.

The reference feature at n_{th} frame can be calculated from reference at a relay image and a relationship between the relay image and n_{th} frame. Therefore, by choosing adequate relay images, all frames in video can be transferred to image feature reference ξ. In proposed approach, the peak of POC function shown in Fig. 4(a) is used to switch relay images.

4. Video Tracking Control Using Image Based Visual Servo Scheme

Video tracking control scheme is proposed in this section. With a feature reference ξ_{ref} extracted in 3.2, we can apply image based visual servo control scheme shown in (1).

In this section, feedforward approach using constant image jacobian are proposed and its effectiveness is shown in a simulation.

4.1 Video Tracking with Feedforward Method Image based control scheme shown in (1) is a feedback approach and it causes some delay in a video tracking. So, the feedforward approach shown in Fig. 6 is proposed to improve tracking performance.

The video feature references ξ_{ref} is calculated in advance to make a robot velocity reference $\boldsymbol{V}_{ref}$ and finally integrated to robot pose $\boldsymbol{X}$.

In this paper, we assume that the robot response is quicker enough than sensing delay including camera imaging and image processing.

Therefore, proposed control scheme utilizing video information for feedforward is written as (6).

$$\boldsymbol{V}_{ref}(t) = -\lambda \boldsymbol{J}^{+}\left(\boldsymbol{\xi}_{ref}(t) - \boldsymbol{\xi}_{c}(t)\right) + \boldsymbol{J}^{+}\frac{d}{dt}\boldsymbol{\xi}_{ref}(t) \qquad (6)$$

$\boldsymbol{J}^{+}\frac{d}{dt}\boldsymbol{\xi}_{ref}(t)$ is feedforward term that compensate the change of feature reference, which needs accuracy for making proper move.

Generally, $\boldsymbol{J}$ is time-variant and difficult to estimate like shown in (3), our approach solve this problem by using time-invariant $\boldsymbol{J}_{const}$ in (12) instead. Therefore, (6) can be revised as (7):

$$\boldsymbol{V}_{ref}(t) = -\lambda \boldsymbol{J}_{const}^{-1}\left(\boldsymbol{\xi}_{ref}(t) - \boldsymbol{\xi}_{c}(t)\right) + \boldsymbol{J}_{const}^{-1}\frac{d}{dt}\boldsymbol{\xi}_{ref}(t) \qquad (7)$$

4.2 Derivation of Time-Invariant Image Jacobian [11]

Surprisingly, $\boldsymbol{J}$ can be time-invariant matrix with a cirtain coordinate transformation in our situaltion: 4 DOF and planer objects.

This section shows that image jacobian matrix can be described as time-invariant matrix in a particular case that the camera move has only 3D translational and rotation around depth axis. In this section, the relationships between relative camera pose from object $\boldsymbol{X} = (X, Y, Z, \Theta)$ and image transformation parameters $\boldsymbol{\xi} = (\xi_x, \xi_y, \kappa, \theta)$ are derived following the image based method. X and Y means parallel axis to the planer object and Z is vertical and depth axis.

A relative camera pose in the key frame used in 3.2 is defined as $\boldsymbol{X}_0 = (X_0, Y_0, Z_0, \Theta_0)$.

Rotation and scaling displacements from the key frame are independent from each other, and correspond to relative camera rotation and depth as following equations:

$$\theta = \Theta_0 - \Theta \qquad (8)$$

$$\kappa = \frac{Z_0}{Z} \qquad (9)$$

The point in this method is that to describe translational displacements on the coordinate systems in the reference frame. The relationship between relative camera pose and 2D translation detected in 3.2 can be written as below:

$$\begin{pmatrix} \xi_x \\ \xi_y \end{pmatrix} = \frac{Z_0}{f}\boldsymbol{R}(-\Theta_0)\begin{pmatrix} X - X_0 \\ Y - Y_0 \end{pmatrix} \qquad (10)$$

where, $\boldsymbol{R}(\phi)$ is a 2×2 rotation matrix representing for ϕ rotate, and f is focal length.

An image jacobian matrix can be derived from time derivative equations of (8), (9) and (10). Using inverse of scaling $1/\kappa$ as a feature, time derivative of (9) can be written in more simpler way. Finally, we choose a image feature $\boldsymbol{\xi}$ as $\boldsymbol{\xi} = (\xi_x, \xi_y, 1/\kappa, \theta)$ and then we have equation in (11).

$$\begin{bmatrix} \dot{\xi}_x \\ \dot{\xi}_y \\ (\dot{\frac{1}{\kappa}}) \\ \dot{\theta} \end{bmatrix} = \boldsymbol{J}_{const}\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\Theta} \end{bmatrix} \qquad (11)$$

Table 1: Variables in a simulation.

Initial Pose $(X_1, Y_1, Z_1, \Theta_1)$	$(0.2, -0.2, 0.5, 1)$ [m,rad]
Position reference $\xi_{ref}(t)$	$(R * \cos\omega t, R * \sin\omega t, R/5 * t, 0)$ [m,rad]
Variable for spiral moves (R, ω)	$(0.25, 2 * \pi/5)$ [m,rad]
Feedback gain λ	0
Sampling Time	1 [ms]

In (11), the image jacobian matrix can be written as time-invariant matrix expressed as follows:

$$\boldsymbol{J}_{const} = \begin{bmatrix} \frac{f}{Z_0}\cos\Theta_0 & \frac{f}{Z_0}\sin\Theta_0 & 0 & 0 \\ -\frac{f}{Z_0}\sin\Theta_0 & \frac{f}{Z_0}\cos\Theta_0 & 0 & 0 \\ 0 & 0 & \frac{1}{Z_0} & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \qquad (12)$$

here, Θ_0 and Z_0 are relative camera pose parameters at the key frame. Θ_0 can be easily detected from (8), while Z_0 need some estimation methods.

4.3 Comparison in Simulation

Then, we compare the proposed method with conventional method shown in (6) using time-variant image jacobian. In this simulation, reference camera move is defined as spiral motion that includes circular movement in X, Y axis and uniform linear motion in Z axis.

Simualtion variables are shown in Table 1. Against the proposed method in (7), feature points based method using (6) are used.

According to (3), the image jacobian $\boldsymbol{J}$ for 4-DOF move used in this simulation is written as (13).

$$\boldsymbol{J} \simeq \begin{bmatrix} \frac{-1}{Z_0} & 0 & \frac{x_1}{Z_0} & y_1 \\ 0 & \frac{-1}{Z_0} & \frac{y_1}{Z_0} & -x_1 \\ \frac{-1}{Z_0} & 0 & \frac{x_2}{Z_0} & y_2 \\ 0 & \frac{-1}{Z_0} & \frac{y_2}{Z_0} & -x_2 \end{bmatrix} \qquad (13)$$

Time-variant depth of objects from camera Z_1, Z_2 is approximated by reference depth Z_0.

In this simulation, we suppose that Z_0 is properly estimated and set feedback gain $\lambda = 0$ to compare the accuracy of feedforward.

Fig. 7 and Fig. 8 show that the approximated image jacobian cannot make proper move, though feedback term can suppress this error.

Thus, this simulation shows proposed feedforward method using constant jacobian is better than conventional feature point based method.

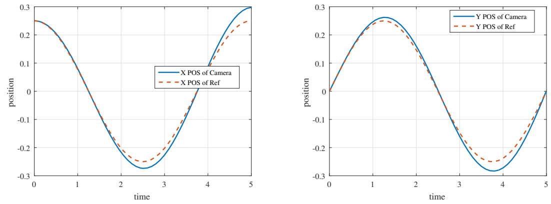
5. Distance Estimation for Image Jacobian Estimation

5.1 Method 1: Scaling Parameter Based

Inverse of scaling parameter $1/\kappa$ corresponds with depth Z like shown in (14).

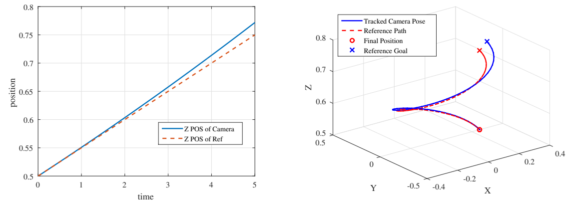
$$Z_0 = \frac{\Delta Z}{\Delta 1/\kappa} \qquad (14)$$

here, Δ means tiny variation and this equation shows that Z_0 can be estimated from Z axis move and scaling variation. Recursive least-squares can be applied to solve this estimation.



(a) X axis reference trajectory and tracked one

(b) Y axis reference trajectory and tracked one

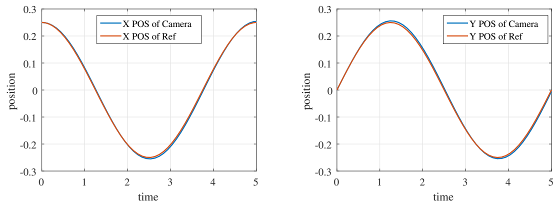


sk

(c) Z axis reference trajectory and tracked one

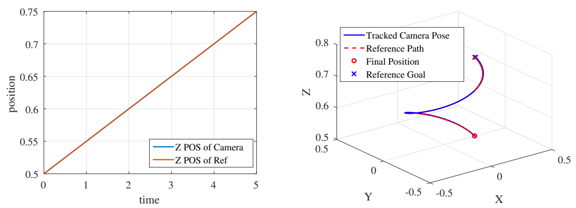
(d) Red broken line: reference trajectory. Blue solid line: tracked one

Figure 7: Tracking result using approximated jacobian in (6)



(a) X axis reference trajectory and tracked one

(b) Y axis reference trajectory and tracked one



(c) Z axis reference trajectory and tracked one

(d) Red broken line: reference trajectory. Blue solid line: tracked one

Figure 8: Tracking result using proposed time-invariant jacobian in (7)

There is a problem that scaling parameter κ is often around 1.0 and has small variation, so this estimation is often sensitive to image noise.

5.2 Method 2: Translation Parameter Based There is the other method using the relationship between image translation and camera translation move. Image translation and 3D translation are related to reference depth and focal length.

$$Z_0 = f \frac{\sqrt{\Delta X^2 + \Delta Y^2}}{\sqrt{\Delta\xi_x^2 + \Delta\xi_y^2}} \tag{15}$$

Recursive least-squares can be also applied to solve this estimation. To solve this equation, the priori information about focal length is needed.

5.3 Distance Estimation Experiment Using experimental data, distance is estimated with practically measured data using 5.1 and 5.2. Both estimation methods using RLS program and forgetting factor is set to 0.95 at first and then

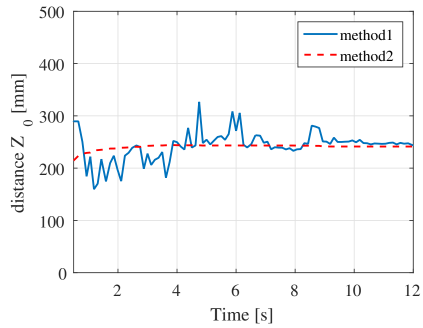


Figure 9: Experimental results of distance estimation.Blue line: method using scaling, red line: method using translation

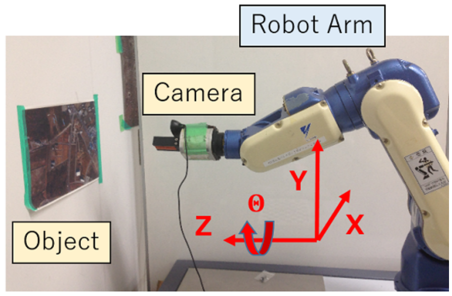


Figure 10: Experimental setup

converged to 1.

As Fig. 9 shows, scaling based method become oscillatory because it is sensitive to scaling changes. On the other hand, translation based method is much more staple but depends on focal length f that is a camera model parameter. So, there is a need of further approach using advanced filtering algorithms such as kalman filter.

6. Experiment

We conducted verification experiments on the contents proposed in each chapter. A part of the experimental setup is shown in Fig. 10. Using 6-DOF robot manipulator are driven by RTLinux, then camera is mounted on the tip of robot and a laptop computer do an image processing and a socket communication with RTLinux PC. Large part of image processing are made of C++.

For saving computational time, only 256 times 256 pixels of image are used as a control input.

Reference video is taken with the camera on robot shown in Fig. 10, and reference move is constant velocity move.

6.1 Reference Shaping and Filtering Another problem in reference shaping and filtering is found in practical reference tracking. The problem is that the time variance of reference $\frac{d}{dt}\xi_{ref}(t)$ often becomes noisy like Fig. 11.

This noise causes oscillatory motion of the robot, that often hinders image acquisition and finally results in a false tracking. At this time, a 10 point moving average filter is applied to the $\frac{d}{dt}\xi_{ref}(t)$ in Fig. 11. The smoothed reference in Fig. 12 is used instead.

There should be some reasonable filtering decision frameworks for more precise tracking.

6.2 Experiment and Evaluation To evaluate the proposed control low in (7), feedback only method in (1) are

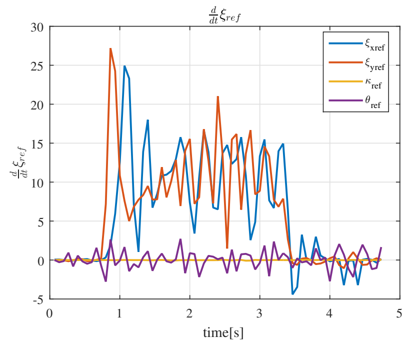


Figure 11: Raw time derivative of renference $\frac{d}{dt}\xi_{ref}(t)$ in Fig. 5.

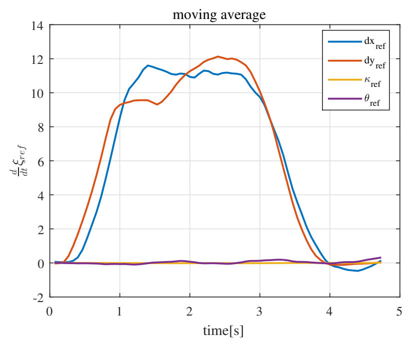


Figure 12: Smoothed reference $\frac{d}{dt}\xi_{ref}(t)$ by filtering Fig. 11

chosen as a comparison method.

The feedback gain λ in (1) and (7) is manually tuned to $\lambda = 1.5$. Considering frame rate of camera, image processing time and communication time, sampling period of camera is set to 125ms.

Fig. 13 shows the results with the comparison method. Blacked lines mean reference and solid lines are actual image feature trajectories. It is obvious that tracking error cannot be suppressed without feedforward control.

On the other hand, Fig. 14 shows that the proposed control scheme can compensate tracking error compared with Fig. 13. However, it still have some overshoot in Fig. 14 and move is still noisy.

There should be some improvement in filtering and reference extraction part and sensing part including image feature extraction.

7. Conclusion

In this paper, the method for video tracking tasks with monocular camera was proposed. There are two main issues in this task: the process for making reference and the control scheme for reference tracking. Since these two processes are closely related each other, an integrated method design is necessary.

Proposed method is mainly aiming to utilize time-invariant image jacobian for image-based feedforward control. In the image based video tracking approach, feedforward control is necessary to catch up the reference move and accurate estimation of image jacobian is needed. The time-invariant image jacobian matrix derived in this paper can solve those problem more easily and simple experiment shows its effectiveness.

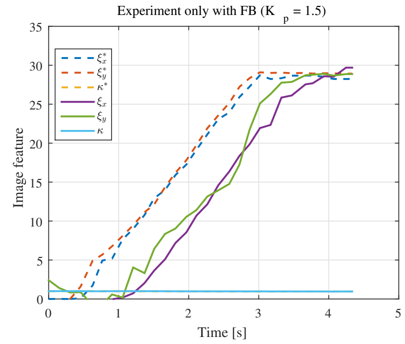


Figure 13: Experimental results without feedforward control. Imgae feature reference: broken line. Actual feature trajectories: solid line.

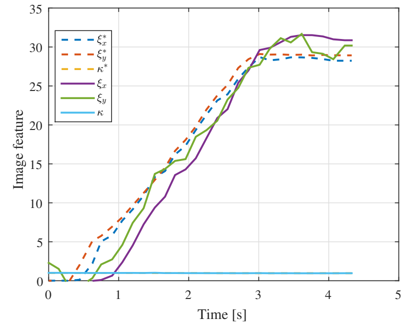


Figure 14: Experimental results with proposed feedforward control. Imgae feature reference: broken line. Actual feature trajectories: solid line.

References

(1) Carrillo, Luis Rodolfo Garca, et al.: "Quad rotorcraft switching control: An application for the task of path following." *IEEE Transactions on Control Systems Technology* 22.4 pp.1255-1267. 2014.

(2) O. Bourquardez, R. Mahony, and N. Guenard,:"Image-Based Visual Servo Control of the Translation Kinematics of a Quadrotor Aerial Vehicle," *IEEE Transactions on Robotics* vol.25, no. 3, pp.833838, 2009.

(3) D. Lowe:"Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vis., vol. 60, no. 2, pp. 91-110, (2004).

(4) E. Rublee, V. Rabaud, K. Konolige, G. Bradski:"ORB: an efficient alternative to SIFT or SURF" *IEEE Intl. Conf. on Computer Vision (ICCV)*, volume 13, (2011).

(5) Z. Chen, S. T. Birchfield, and S. Member,:"Qualitative Vision-Based Path Following,"*IEEE Transactions on Robotics*, vol.25, no. 3, pp.749754, 2009.

(6) A. Dame and E. Marchand. :"A new information theoretic approach for appearance-based navigation of non-holonomic vehicle." Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011.

(7) F. Chaumette and S. Hutchinson. : "Visual servo control. Part 1: Basic Approaches." *IEEE Robot. Autom.Magazine* pp.82-90, 2006.

(8) H. F. Shekarforoush, J. B. Zerubia, S. Member, and M. Berthod, :"Extension of Phase Correlation to Subpixel Registration," *IEEE Transactions on Image Processing*, vol. 11, NO. 3, March, 2002.

(9) K. Takita, T. Aoki, Y. Sasaki, T. Higuchi and K. Kobayashi. :"High-Accuracy Subpixel Image Registration Based on Phase-Only Correlation," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* Vol. E86-A, pp. 1925-1934. 2003.

(10) E. Malis, F. Chaumette and S. Boudet.:"2-1/2 D visual servoing,"*IEEE Transactions on Robotics and Automation.*(1999)

(11) Y, Ri., and H. Fujimoto. :"Proposal of visual servoing using phase-only-correlation (POC)." Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE. IEEE, 2015.