

A Study on a Vision-Based Target Tracking Control System Design

Thinh Huynh*, Hwan-Cheol Park**, Dong-Hun Lee*** and Young-Bok Kim****†

(Received 01 September 2020, Revised 21 January 2021, Accepted 04 February 2021)

Abstract : This paper presents a vision-based tracking method for controlling camera line of sight (LOS) with a gimbaled structure carried on a dynamic platform. Due to intentional maneuvers, inadvertent motions, and additional disturbances effected on the gimbal, the vision tracking performance and gimbal motion response may be significantly reduced. Thus, a combined tracking algorithm is proposed to increase the robustness and effectiveness of image-based tracking process. From the tracked locations, desired motions of the gimbal are obtained. In the closed loop control system, camera and image-based tracking algorithm work as a transportation delay, so a standard image-based visual control displays limited performance. Therefore, a PID controller based on the Smith Predictor is implemented for the gimbal motion control to overcome this problem. Finally, experimental studies validating proposed system are presented.

Key Words : LOS (Line of Sight), Gimbal, Vision-based tracking, KLT (Kanade Lucas Tomasi) algorithm, Blob detection

1. Introduction

In many applications such as surveillance and target tracking, a camera, or any types of optical sensor, carried on mobile vehicle needs to track

continuously a target object or area. Gimbaled structures, mounted on the vehicle, driven by motors and equipped with gyroscopes, are used to steer the camera accurately despite unpredicted motions of both the target and the vehicle. Many studies involved LOS control with the gimbal have been done. Typically, the gimbal motion control system is configured as a high-bandwidth rate loop inside a lower bandwidth pointing and tracking position loop.^{1,2)} Where the inner loop stabilizes the camera, the outer track loop ensures that the LOS remains pointed toward the target. Using visual information in the servo loop to control the tracking motion is getting lots of attention. This type of control system is well known as visual servoing. Two basic approaches can be identified: position-based visual servo, in which vision data are used to reconstruct

****† Young-Bok Kim(ORCID:<https://orcid.org/0000-0001-6035-6744>) : Professor, Department of Smart Robot Convergence & Application Engineering, Pukyong National University.

E-mail : kpjiwoo@pknu.ac.kr, Tel : 051-629-6197

**Hwan-Cheol Park(ORCID:<https://orcid.org/0000-0002-5670-7246>) : 1st Engineer, Training Ship Packyong, Pukyong National University.

***Dong-Hun Lee(ORCID:<https://orcid.org/0000-0001-6157-9763>) : Graduate School Student, Department of Mechanical System Engineering, Pukyong National University.

*Thinh Huynh(ORCID:<https://orcid.org/0000-0001-7329-8432>) : Graduate School Student, Department of Smart Robot Convergence & Application Engineering, the Graduate School, Pukyong National University.

the target in 3D space; and image-based visual servo, in which the error is generated directly from the image plane feature. Their performance, stability and design methods are discussed in the references³⁻⁵⁾ and many others. The scenarios in these studies are usually to control the pose of a camera attached on the end effector of a robot manipulator in 6 DOF such that it observes desired features of an object. In the particular case of gimbal motion control, only rotational motions, corresponding with gimbal channels, can be controlled, while the camera's linear motions act as disturbances. Besides that, the main objective of visual servoing becomes to keep the target in the field of interest despite unknown motions of the camera and the target. For example, simulation and experiment studies of Hurak^{6,7)} applied the image-based visual technique for an inertially stabilized double gimbal airborne camera system.

However, the system performance was limited by long sampling time of the computer vision process. Another one is the study of Bibby,⁸⁾ where a pan-tilt gimballed device carried on a ship tracked a target at sea. Unfortunately, the study focused on designing the visual tracker, so performance of the motion control system was not presented clearly.

Motivated from these studies, we proposed a tracking control system for the pan-tilt gimbal system using information extracted from the camera and system's gyroscope. A visual tracking algorithm, which is based on well known KLT algorithm^{9,10)} and improved by Forward-backward error technique,¹¹⁾ is implemented. In various scenarios, the algorithm may fail to track the target. Thus, image color processing, Blob detection and features matching are combined to ensure to detect the lost object. The 2D information from the tracker is then extracted to estimate the object's location and to compute the desired gimbal motions. Moreover, the standard image-base visual servoing system performance and stability are briefly summarized,

based on the consideration that computer vision is a time-consuming process. Then, a control system encouraged by Smith predictor¹²⁾ and PID controller is proposed and evaluated.

The paper is organized as follows. The detail tracking procedure and its results are presented in Section 2. In Section 3, the design of motion control law is discussed. Then section 4 shows experimental results validating the introduced control law. Finally, conclusions are drawn in Section 5.

2. Visual Tracking Algorithm

2.1 Tracking procedure

The KLT (Kanade-Lucas-Tomasi) algorithm to track the motion of features in an image stream was introduced in 1990s. The algorithm assumes that images taken at near time instants are usually strongly related to each other, then the displacement $d(\xi, \eta)$ of the point at $x = (x, y)$ in image plane between time instants t and $(t+\tau)$ can be chosen to minimize the residue error defined by the following double integral over the given window W .⁹⁾

$$\varepsilon = \int_W [I(x-d) - J(x)]^2 \omega dx. \quad (1)$$

where $I(x) = I(x, y, t)$ is the point at t , $J(x) = I(x, y, t+\tau)$ is that point at $(t+\tau)$, and ω is the weighting function depending on the image intensity pattern. The KLT algorithm finds the displacement d by rewriting the derivative of residual error in the form of

$$Gd = e \quad (2)$$

where the matrix G can be computed from one frame by estimating gradients and computing their second order moments, and vector e can be computed from the difference between the two frames. Based on KLT algorithm, the procedure of tracking target object by the camera attached in the gimbal is proposed as follows :

■ In the first taken frame, a rectangle of interests (ROI) containing the target object is chosen. In grayscale image, corners, parts of ROI contain complete motion information, are detected using minimum eigenvalue algorithm. The bounding box covers all these points, and center of the box is set as the initial location of the target.

■ Next, KLT algorithm is used to track these corners from frame to frame.

■ The tracking results are validated by using forward-backward error technique. In this technique, two tracking trajectories, forward point tracking and backward tracking validation, are compared and if they differ significantly, the tracking is considered as incorrect.

$$\text{distance}(T^{kf}, T^{kb}) = \|x_t - \hat{x}_t\| \leq \zeta \quad (3)$$

where T^{ka} is tracking trajectory (f : forward, b : backward) at k^{th} sequence image, x_t and \hat{x}_t are tracked point and validated point from forward and backward tracking trajectories.

■ In the case of successful tracking, the new location of the object is estimated from its center in previous frame and the displacement obtained by KLT algorithm.

$$I(x, y, t + \tau) = I(x + \xi, y + \eta, t) \quad (4)$$

2.2 Detecting object using Blob analysis and image features

As mentioned, the implementation of KLT algorithm may fail to track adequately in various circumstances of interest. Then, the target bounding box and object location in the image frame need to re-detect. Detecting process is implemented as follows :

■ At first, background rejection and object's region detection are performed by a combination of image thresholding technique, morphological opening technique, and blob detection. In detail :

Background is rejected by using image thresholding techniques.

The image is then converted into a binary image. Morphological opening is performed for removing small objects from an image while preserving the shape and size of larger objects in the image.

Blob detection method is called to detect these regions in the binary image. The object should be in one of these regions, but we cannot conclude where it is exactly.

■ Then, corner features are then extracted in each region, and compared to features in the first frame. If these features are matched together, the object is detected. The bounding box and center are obtained.

■ Finally, KLT algorithm is re-initialized to track the object from these new features.

A flowchart presenting tracking and detecting procedure is shown in Fig. 1. An experimental study is shown in Fig. 2.

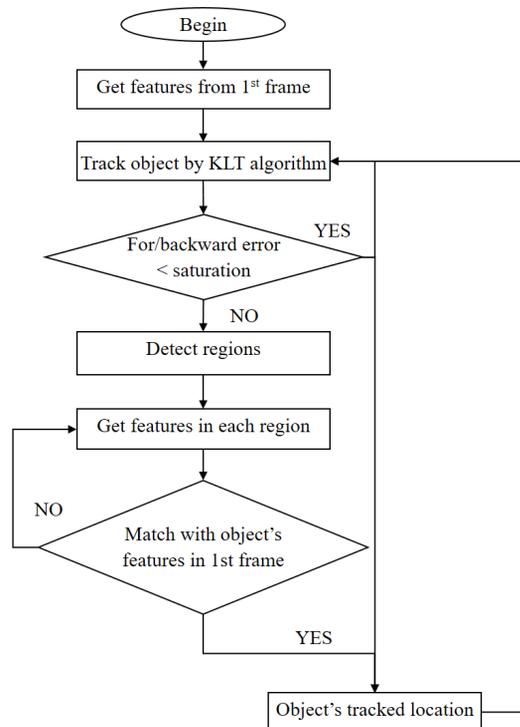


Fig. 1 Flowchart of the proposed tracking procedure

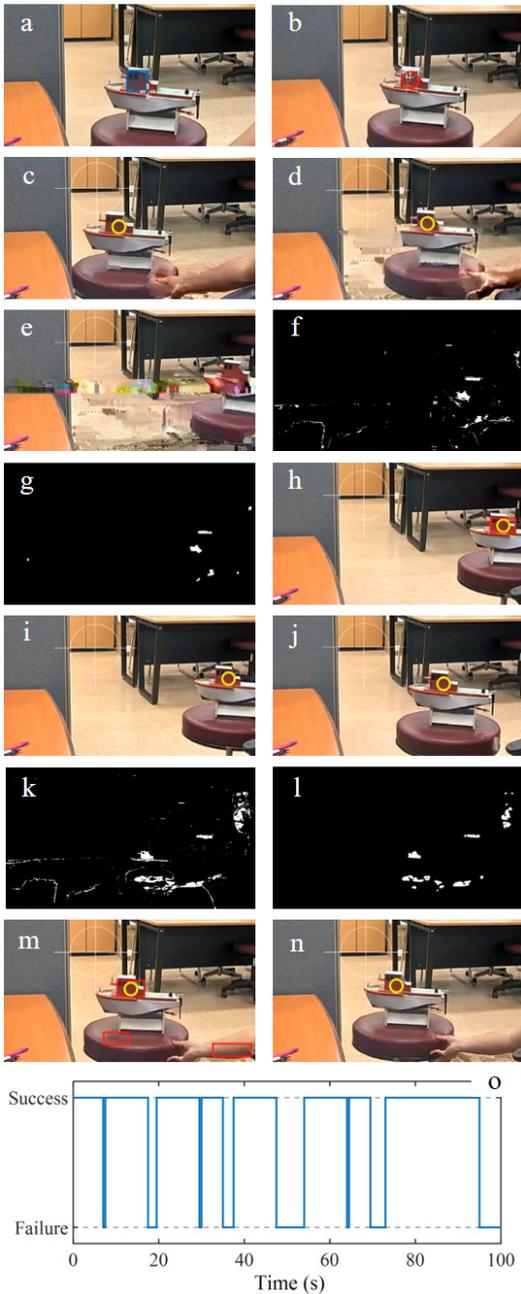


Fig. 2 Tracking and detecting results

- a : Choosing the ROI in the first taken frame
- b : Corner features in ROI and center location
- c, d : Tracking object by KLT algorithm
- e : Tracking failed due to the degraded image
- f, g : Background rejection by image thresholding

technique (f) and morphological opening performance (g)

h : Detecting object by Blob detector and features matching

i, j : Re-tracking by the KLT algorithm

k~n : Object detection process when tracking is failed due to the fast-moving target

o : Success rate

3. Controller design for gimbal motion control

3.1 Pinhole camera model

In this study, the well-known pinhole camera model is used to obtain the relation between locations of the target object in 3D world coordinate and in 2D image plane. In this model, the optical axis runs through the pinhole perpendicular to the image plane. The location of the object expressed in the coordinate attached with the camera is given by $P = [X, Y, Z]^T$. The point of the object in the image plane is $p = [x, y]^T$. The coordinates $[x, y]$ represents the columns and the rows of the image in pixel, respectively. The pinhole camera model yields that :

$$x = \frac{fX}{s_x Z}, y = \frac{fY}{s_y Z} \quad (5)$$

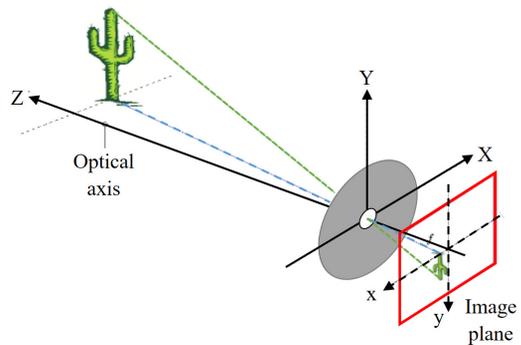


Fig. 3 Pinhole camera model

where f is called the focal distance, s_x and s_y are the pixel dimensions. Then automatically :

$$X = \frac{s_x}{f} xZ, \quad Y = \frac{s_y}{f} yZ \quad (6)$$

3.2 Calculation of desired tracking motions from vision data

Consider a two-axis gimbal carrying the camera attached at the center. The gimbal consists of an outer channel actuating pan motion and an inner channel operating tilt motion. In general, the main goal of tracking control applying for gimbal system is to keep the target object in a pre-defined frame of view around the center of the image frame.

If $\psi_{di}(i=1,2)$ is the angular position of the object in the reference coordinate having its origin at the inner gimbal, then the additional angle that the gimbal needs to steer in order to point to the object is $\psi_{ai} = \psi_{di} - \psi_i$, where ψ_i is the current position of the gimbal, including the tilt and the pan positions.

In order to obtain steering angles needed, let us consider the two-axis gimbal as an universal joint connecting two channels whose axis are inclined to each other by the tilt angle. Fig. 4 shows planes of rotation of the pan axis and the camera's axis. Target point A in horizontal axis in image plane and its corresponding location in the camera plane are illustrated. To track the target, the camera need to steer an angle of ψ_{ai} in camera plane as.

$$\psi_{ai} = \arctan\left(\frac{X}{Z}\right) \quad (7)$$

Then, from the kinematics of the universal joint, the steering angle about pan axis

$$\psi_{a2} = \arctan\left(\frac{X}{Z} \cos\psi_1\right) \quad (8)$$

Although point A initially lies on the horizontal axis, control the pan motion alone can not bring it

to the center of the image, as shown in Fig. 4. Consequently, correction in tilt motion is needed.

$$\psi_{a1} = -\arcsin\left[(1 - \cos\psi_a) \sin\psi_1\right] \quad (9)$$

In general, if the point of the target in the image plane is $p = [x, y]^T$, the steering angles that the gimbal needs to perform are :

$$\begin{cases} \psi_{a2} = \arctan\left(x \frac{s_x}{f} \cos\psi_1\right) \\ \psi_{a1} = \arctan\left(y \frac{s_y}{f}\right) \\ -\arcsin\left[\left[1 - \cos\left(\arctan\frac{x s_x}{f}\right)\right] \sin\psi_1\right] \end{cases} \quad (10)$$

Assume that the focal length f is a constant during tracking, then the steering angles can be computed from the captured image without knowledge of the distance to the object.

3.3 Controller design

A standard image-based visual tracking problem uses the additional angles calculated from the tracker as the control error as in Fig. 5.

The control law can be easily designed based on the Moore-Penrose pseudo-inverse matrix of camera interaction matrix.^{3,4)}

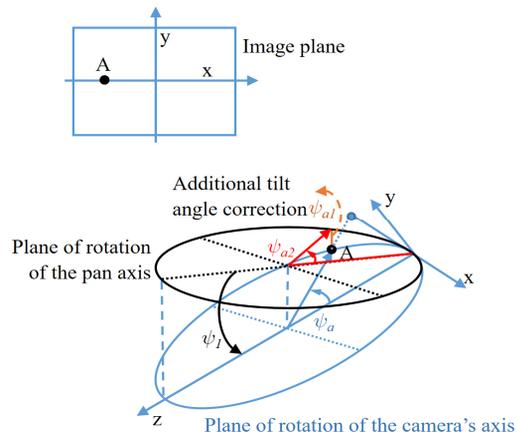


Fig. 4 Relationship between target location in image plane and gimbal's steering angles

Moreover, from the viewpoint of control system, camera and tracker add transportation delay into the system. The delay time consists of time of taking snapshot, transmitting data, and sampling time of tracking algorithm. Therefore, the steering angle obtained from the tracker is actually $\psi_{ai} e^{-sT}$, T is the delay time. Then, the closed-loop system with a standard image-based controller can be described as :

$$H(s) = \frac{K_t(s)G_i(s)e^{-sT}}{1 + K_t(s)G_i(s)e^{-sT}} \quad (11)$$

As seen in Eq. (11), e^{-sT} appears in both numerator and denominator of the transfer function proving that time delay not only reduces system performance but also influent to system stability. Suppose that we can express :

$$K_t(s)G_i(s)e^{-sT} = \frac{N(s)}{D(s)}e^{-sT} \quad (12)$$

where $N(s)$ and $D(s)$ are real polynomials of degree p and q respectively. Then the characteristic equation of the closed-loop system is :

$$\delta(s) = D(s) + N(s)e^{-sT} \quad (13)$$

If $q \geq p$ and $|b_p/a_q| < 1$, where a_q, b_p are the leading coefficients of $D(s)$ and $N(s)$ respectively, the Nyquist criterion is applicable to check the stability of the system.¹⁶⁾ If not, the system is unstable. The complete set S_R of adjustable parameters of controller $K_t(s)$ such that the system

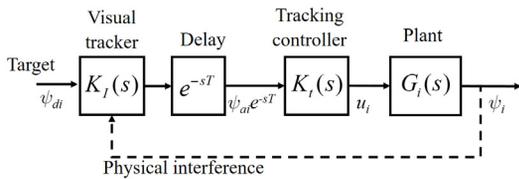


Fig. 5 Configuration of an standard image-based visual tracking control system

stability is preserved for any delay time $T \in [0, T_0]$ is obtained as :

$$S_R = S_I \setminus S_L \quad (14)$$

where $S_I(S_0 \setminus S_N)$, S_0 is the complete set of controller adjustable parameters that stabilize the delay-free plant $G_i(s)$, S_N is the set such that the delay-free open-loop expression $K_t(s)G_i(s)$ is an improper transfer function.

And S_L is the set of controller parameters such that $K_t(s)G_i(s)e^{-sT} = -1$ for a specific delay time $T \in [0, T_0]$. It is easily seen that $S_R \subseteq S_I \subseteq S_0$, means that the stability of the delay-time system is preserved in a stricter condition comparing to the delay-free one. The controller parameters make the system achieves good results may not in the subset S_R , so the control performance can be reduced. The longer delay time, the more limited controller parameters and the more sluggish system response.¹²⁾ Based on Smith predictor, a new control structure is proposed as Fig. 6. A Smith Predictor is implemented as $\psi_i - \psi_i e^{-sT}$. When the amount of delay is known, the position of the target computed from image tracker can be obtained by Eq. (15) such that the system is represented as Eq. (16), respectively.

$$\psi_{di} e^{-sT} = (\psi_{ai} + \psi_i) e^{-sT} \quad (15)$$

$$H_p(s) = \frac{K_t(s)G_i(s)e^{-sT}}{1 + K_t(s)G_i(s)} \quad (16)$$

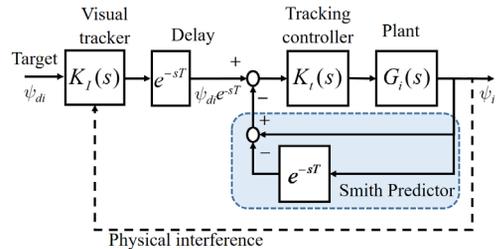


Fig. 6 Configuration of the proposed control system

The delay is brought out of the characteristic equation, so the system stability is preserved in the same condition with the delay-free one. The set of controller adjustable parameters is now S_0 .

However, time delay still gives an upper bound on the achievable bandwidth of the system. It is recommended that the bandwidth of a system with delay does not exceed $\pi/2T$ or $2/T$.^{12,15)}

Now, let us generally describe the plant of tracking motion system as follows :

$$\ddot{\psi}_i + a_i \dot{\psi}_i + b_i \psi_i = c_i u_i \quad (17)$$

The plant contains the closed-loop of gimbal stabilization control, where there are the plant of the gimbal mechanism and a stabilization controller. In Eq. (17), a_i , b_i and c_i are the nominal parameters of the system model corresponding to system dynamics, and u_i is the control signal of proposed tracking controller.

A PID controller is implemented in the form as :

$$K_{ti}(s) = P_i + I_i \frac{1}{s} + D_i \frac{N_i}{1 + N_i \frac{1}{s}} \quad (18)$$

where controller gains are tuned so that system bandwidth is satisfied.

4. Experiment

Experimental studies are performed on a two-axis gimbal used in ocean surveillance applications. Nominal parameters of the system defined in Eq. (17) are experimentally obtained using system identification method. Their values are presented as follows :

$$\begin{aligned} a_1 &= 21.97, b_1 = 0.28, c_1 = 22.25 \\ a_2 &= 30.40, b_2 = 0.03, c_2 = 28.44 \end{aligned}$$

System configuration is shown in Fig. 7. The camera is mounted at the center of tilt gimbal.

Video taken by the camera is streamed using Real Time Streaming Protocol (RTSP) with HD resolution and 30 fps of frame rate.

Total delay time that the vision tracker brings to the system is about 1 sec, including 0.5 sec for streaming and 0.5 sec of sampling time. Moreover, camera's absolute angular position and rate are measured by an Attitude Heading Reference System (AHRS) attached directly.

Two channels of the gimbal are actuated independently by two servo system through driving belts. There is already a P controller for speed control in each servo which is operated as a stabilization controller of the gimbal.

All actuators and sensor communicate through RS232 interface. The controller is implemented in Matlab/Simulink. Sampling time is chosen as 0.1 sec. The hardware system specifications are listed in Table 1, and the controller gains are represented in Table 2.

Fig. 9 shows snapshots cropped from the image sequence during experiment. A white circle marks the center of the image with 2 deg of tolerance. In the beginning, the object location, which is determined by tracking algorithm, is at the lower right corner of the image.

The reference for control system is computed as Eqs. (10) and (15), thus the controller is able to control tilt and pan actuators to steer the LOS from current location to the target. Then, the LOS is controlled to keep the target at the center of the image in this experiments. In the case of failed tracking, the reference is updated after object location is detected by the detection method.

Due to the limited bandwidth, the system response is not really fast and accurate. However, in real-world applications like ocean surveillance, target objects are barely moving as fast.

Practical requirement is just to keep the target in a frame of view but not necessary in the center of

image. Therefore, the responses of experimental system are reasonable. In addition, the performance is smooth, robust, and acceptably precise as shown in Fig. 8.

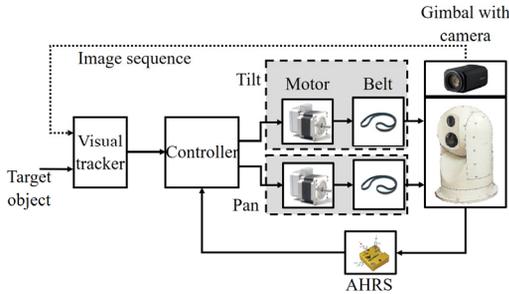


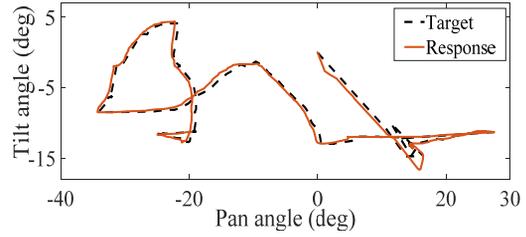
Fig. 7 Configuration of the experiment system

Table 1 Experiment system specifications

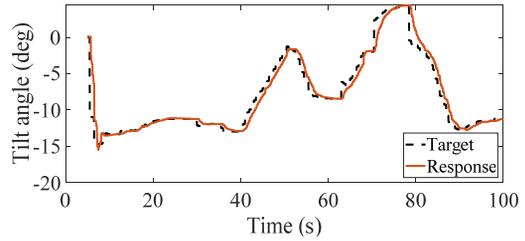
Apparatus	Item	Value	
Camera: Hanwha XNZ-6320	Used resolution	1280×720	
	Used framerate	30 fps	
	Angular field of view	61.8° horizontal 36.2° vertical	
	Video Format	H.264	
	Video streaming:	Real Time Streaming Protocol (RTSP)	
Actuator: CoolMuscle CMI-C23S30	Rated Power	45 W	
	Max. Speed	3000 rpm	
	Rated Torque	0.294 Nm	
	Supply Voltage	24[VDC]±10%	
	Supply Current	3.9 A	
Sensor: NTRexLAB MW-AHRS	Signal Processing:	Fusion of acceleration and gyro sensor with Kalman filter	
	Angle's resolution	0.001°	
	Static Error	0.1°	
	Dynamic Error	2°	
	Response Time	1 ms	

Table 2 Designed controller gains

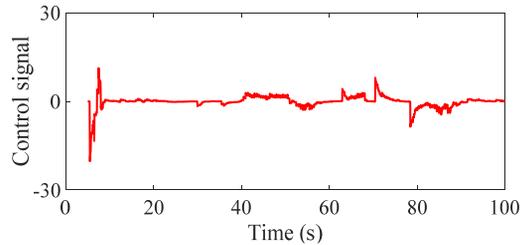
Gimbal Channel	Tilt	Pan
PID Gains	$P_1 = 1.483$	$P_2 = 1.459$
	$I_1 = 0.019$	$I_2 = 0.019$
	$D_1 = 0.036$	$D_2 = 0.082$
	$N_1 = 10$	$N_2 = 10$



(a) Tracking performance



(b) Angle (top) and control signal (bottom) of the tilt



(c) Angle (top) and control signal (bottom) of the pan

Fig. 8 Control performance of the gimbal system

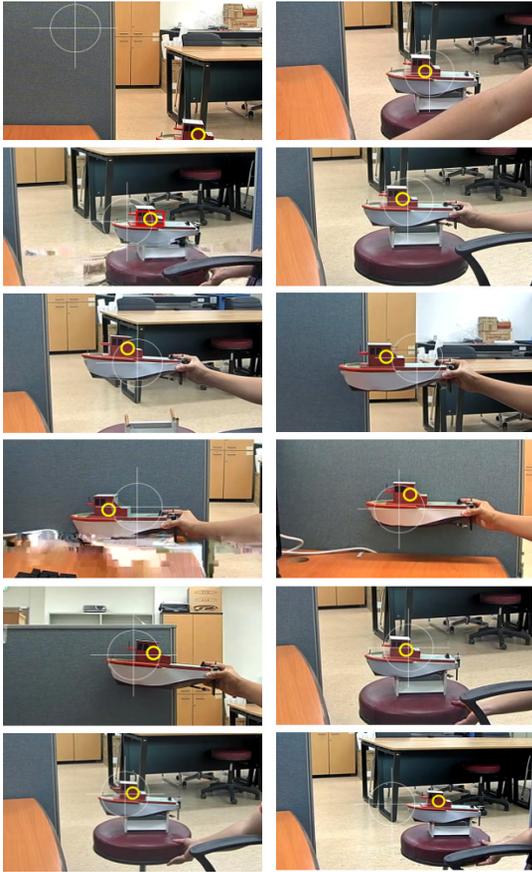


Fig. 9 Performance of the vision based tracking control system

5. Conclusions

In this paper, a visual-based tracking control system has been implemented for the gimbal motion control. Proposed tracking procedure combined KLT tracking algorithm, image color processing, Blob detector and features matching for a robust and reliable performance. The relationship between a location in the image plane and steering angle needed of the gimbal has been derived. In addition, the delay effects that vision tracker brought into the control system has been discussed. A PID controller has been tuned satisfying system limitation and achieving acceptable performance. Moreover, by

experimental studies, the reliability and robustness of the system have been evaluated.

Author contributions

Y. B. Kim; Conceptualization. T. Huynh; Data curation. H. C. Park; Formal analysis. H. C. Park; Funding acquisition. T. Huynh; Investigation. Y. B. Kim; Methodology. Y. B. Kim; Project administration. D. H. Lee; Software. Y. B. Kim; Supervision. D. H. Lee; Validation. T. Huynh; Writing-original draft. H. C. Park; Writing-original draft. D. H. Lee; Writing-review & editing.

References

1. J. M. Hilbert, 2008, "Inertially Stabilized Platform Technology Concepts and Principles", IEEE Control Systems Magazine, Vol. 28, No. 1, pp. 26-46. (DOI:10.1109/MCS.2007.910256)
2. M. K. MASTEN, 2008, "Inertially Stabilized Platforms for Optical Imaging Systems", IEEE Control Systems Magazine, Vol. 28, No. 1, pp. 47-64. (DOI:10.1109/MCS.2007.910201)
3. F. Chaumette and S. Hutchinson, 2006, "Visual Servo Control I : Basic Approaches", IEEE Robotics & Automation Magazine, Vol. 13, No. 4, pp. 82-90. (DOI:10.1109/MRA.2006.250573)
4. F. Chaumette and S. Hutchinson, 2007, "Visual Servo Control II, Advanced Approaches [Tutorial]", IEEE Robotics & Automation Magazine, Vol. 14, No. 1, pp. 109-118. (DOI:10.1109/MRA.2007.339609)
5. G. Palmieri, M. Palpacelli, M. Battistelli and M. Callegari, 2012, "A Comparison Between Position-Based and Image-Based Dynamic Visual Servoings in the Control of a Translating

- Parallel Manipulator", *Journal of Robotics*, Vol. 2012, Article ID 103954. (DOI:10.1155/2012/103954)
6. Z. Hurák and M. Řezáč, 2012, "Image-Based Pointing and Tracking for Inertially Stabilized Airborne Camera Platform", *IEEE Transactions on Control Systems Technology*, Vol. 20, No. 5, pp. 1146-1159. (DOI:10.1109/TCST.2011.2164541)
 7. Z. Hurák and M. Řezáč, 2009, "Combined Line-of-Sight Inertial Stabilization and Visual Tracking: Application to an Airborne Camera Platform", *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, Shanghai, pp. 8458-8463. (DOI:10.1109/CDC.2009.5400793)
 8. C. Bibby and I. Reid, 2005, "Visual Tracking at Sea", *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 1841-1846. (DOI:10.1109/ROBOT.2005.1570381)
 9. T. Carlo and T. Kanade, 1991, "Detection and tracking of point features", *Computer Science Department*, Carnegie Mellon University.
 10. S. Jianbo and C. Tomasi, 1994, "Good Features to Track", *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 593-600.
 11. K. Zdenek, K. Mikolajczyk and J. Matas, 2010, "Forward-Backward Error: Automatic Detection of Tracking Failures", *Proceedings of the 20th International Conference on Pattern Recognition 2010*, pp. 2756-2759.
 12. Q. C. Zhong, 2006, "Robust control of time-delay systems", Springer-Verlag: London Limited.
 13. K. Watanabe and M. Ito, 1981, "A Process-Model Control for Linear Systems with Delay", *IEEE Transactions on Automatic Control*, Vol. 26, No. 6, pp. 1261-1269. (DOI:10.1109/TAC.1981.1102802)
 14. A. Yilmaz, O. Javed, and M. Shah. 2006, "Object Tracking : A Survey", *ACM Computing Surveys*, Vol. 38, No. 4. pp. 2-45. (DOI:10.1145/1177352.1177355)
 15. K. J. Åström, 2000, "Limitations on Control System Performance", *European Journal of Control*, Vol. 6, No. 1, pp. 1-19.
 16. G. J. Silva, A. Datta and S. P. Bhattacharyya, 2005, "PID Controllers for Time-Delay Systems", Birkhuser Basel.